Expectation Maximization Approach to Data-Based Fault Diagnostics

Magdi S. Mahmoud   and   Haris M. Khalid

Abstract—The data-based fault detection and isolation (DBFDI) process becomes more potentially challenging if the faulty component of the system causes partial loss of data. In this paper, we present an iterative approach to DBFDI that is capable of recovering the model and detecting the fault pertaining to that particular cause of the model loss. The developed method is an expectation-maximization (EM) based on forward-backward Kalman filtering. We test the method on a rotational drive-based electro-hydraulic system using various fault scenarios. It is established that the developed method retrieves the critical information about presence or absence of a fault from partial data-model with minimum time-delay and provides accurate unfolding-in-time of the finer details of the fault, thereby completing the picture of fault detection and estimation of the system under test. This in turn is completed by the fault diagnostic model for fault isolation. The obtained experimental results indicate that the developed method is capable to correctly identify various faults, and then estimating the lost information.

Keyword: Fault detection; Fault isolation; Kalman filter; Expectation maximization; Fault diagnostic model; Rotational hydraulic drive-based electro-hydraulic system.

The following variables are used in the paper table I.

TABLE I

NOMENCLATURE

Symbols	Function
$I(t)$	input current
τ_v	servo-valve time constant
$A_v(t)$	servo-valve opening area
K_x	servo-valve area constant
K_v	servo-valve torque motor constant
$Q_1(t)$	flow from the servo-valve
$Q_2(t)$	flow to the servo-valve
$P_L(t)$	load pressure difference
P_s	source pressure
C_d	flow discharge coefficient
ρ	fluid mass density
$P_1(t), P_2(t)$	rotational drive chambers pressure
β	fluid bulk modulus
D_m	actuator: volumetric disp. parameter
C_L	leakage coefficient
$\theta(t)$	angular displacement
$sign$	change in direction of actuator motion
J	moment of inertia
B	viscous damping coefficient
T_L	load torque
C_{0_i}	latent variable

I. INTRODUCTION

Data-based fault detection and isolation (DBFDI) has always been the subject of considerable interest in the process industry where the whole model structure for the plant is usually not available. The intensity of importance for this subject is the ever increasing requirements on the reliable operation of control systems, which are, in most cases, subject to a number of faults either in the internal closed loops or from environmental factors. The data generated from the assumed model are compared with measured data from the physical system to create residuals that relate to specific faults. Faults encountered can be of many types starting from a faulty sensor in the production line to a broken transducer or a burnt out coil not transforming the assigned accurate information. Once system faults have occurred, they can cause unrecoverable losses and result in unacceptable environmental pollution, etc. Occasionally, the occurrence of a minor fault has resulted in disastrous effects. With an accurate process model and under appropriate assumptions, it is possible to accomplish fault detection and isolation (FDI) for specific fault structures.

Generally, fault diagnostic methods can generally be divided into two categories:
- Model-based fault diagnostics, and
- Data-based fault diagnostics including knowledge-based Fault diagnostics.

Model-based Fault diagnostic methods are generally dependent on the mathematical models of the process developed either from first principles or from identification of the system. The data extracted from the model is then compared with measured data from the physical system to create residuals that relate to specific faults. With an accurate process model and under appropriate assumptions, it is possible to accomplish Fault diagnostics for specific fault structures (see, for example, [4], [6], [9], [12]). Data-based methods, on the other hand, rely on process measurements in order to perform fault diagnostics. Analyzing process measurements gives the location and direction of the system trajectory in the state space. The databases contain a great amount of redundant information and must be processed by means of suitable algorithms, most of which belong to the great area of data mining. The strategy presented in this work is an alternative methodology to process large databases and to design appropriate monitoring systems integrated to fault treatment approaches. Tools such as adaptive principle component analysis, fuzzy logic and neural networks have also demonstrated their capability on treating important data-based systems. It is then possible, particularly for linear process systems, to extract information about the fault by comparing the location and/or direction of the system trajectory in the state space with past faulty behavior (for example, [16], [23]). Several methods have been developed that manipulate the measured

data to reduce their dimension and extract information from the data with respect to actuator/sensor faults using principle component analysis (PCA) or partial least squares(PLS)techniques (for example, [11], [20], [15], [14]. These methods reduce the dimensionality of the data by eliminating directions in the state space with low common-cause variance. Other methods have been developed that consider the contribution of particular states to the overall shift from normal operation [11]. Some data-based methods take advantage of PCA to find correlations within the data [5]. Work has also been done to group data-based on process structure or process distinct time scales as in multi-block or multi-scale PCA ([19], [3], [2]). While many of these methods have been successful in achieving fault detection, fault isolation remains a difficult task, particularly for non-linear processes where historical data under faulty operation are in-sufficient to discriminate between faults.For a comprehensive review of model-based and data-based FDI methods,the reader may refer to [18], [17]. Considering other applications, several results involving FDI-FTC based on linearized models of aircraft dynamics have been reported [24], [25], [26]. An application of multi-variable adaptive control techniques to flight control reconfiguration was considered in [24]. The objective was to redesign automatically flight control laws to compensate for actuator failures or surface damage. [27] and [28] focus on the robust control problems and their applications on aerospace engineering.

In [21], fuzzy logic techniques have been applied to classify frequency spectra representing various rolling element bearing faults. The application of basic fuzzy logic techniques has allowed fuzzy numbers to be generated which represent the similarity between frequency spectra. Correct classification of different bearing fault spectra was observed when the correct combination of fuzzy set shapes and range of membership domains were used. In my opinion, handling a critical issue of fault diagnosis with fuzzy logic alone is taking the case very easy, and not understanding the criticality of the fault diagnosis issue, where an overlooked detection might cost a serious happening. Moreover, the classification of the faults are being observed using a combination of fuzzy sets, this might if well done, will work properly, but in case of noise, it will result in false alarms. Also, a data analysis of different faults, pertaining to original fault scenarios and signatures, and training them through neural network with the help of fuzzy logic inference system could have given a much better and reliable approach towards fault detection and classification.

In [22], a fault tree is used to analyze the system by evaluating the basic events (elementary causes), which can lead to a root event (a particular fault). Then, a multiple-model adaptive estimation algorithm is used to detect and identify the model-known faults. Finally, based on the system states of the robot and the results of the estimation, the model-unknown faults are also identified using logical reasoning. In my opinion, data-based methods can diagnose the fault incipiently. Therefore, the fault can be detected rapidly. The main issue in data-based method is to obtain the data. The fault can then be diagnosed from this data accurately, as done in our case of Expectation Maximization-Based FDI where we are also able to recover the loss data. Whereas in case of [22], the technique is relying heav-

ily on the model, for which the model should be strictly accurate to give correct results. But in case of large scale systems and plants, where the model is often not available, we may face difficulties by either capturing the noise and disturbances as faults, or overlooking the faults as noise. The technique proposed in [22] is model dependent and might be restricted to only particular plant models.

In this paper, a data-based fault diagnostics scheme is developed using expectation maximization boiled on data-driven Kalman filter. The salient feature is that it does not rely on prior knowledge and mathematical information about the system under consideration. We construct an iterative approach to data-based fault detection and isolation (DBFDI) that is capable of recovering the model and detecting the fault pertaining to that particular cause of the model loss. It is essentially an expectation-maximization (EM) based on forward-backward Kalman filtering. We test the method on a rotational drive-based electro-hydraulic system using various fault scenarios. It is shown that the developed method retrieves the critical information about presence or absence of a fault from partial data-model with minimum time-delay and provides accurate unfolding-in-time of the finer details of the fault, thereby completing the picture of fault detection and estimation of the system under test. In turn, this is completed by the fault diagnostic model for fault isolation. The obtained experimental results indicate that the developed method is capable to correctly identify various faults, and then estimating the lost information.

The paper is organized as follows: the problem statement pertaining to fault diagnostics and the proposed solution formulation is presented in Section 2. Then, follows the evaluation of the proposed scheme in Section 3. The results of simulating the techniques implemented are presented in Section 4. Finally, some concluding remarks are given in Section 5.

II. DATA-BASED FAULT DIAGNOSTICS SCHEME

To construct an effective fault diagnostics scheme, we have assumed various faults in the system have been successfully monitored, estimated and protected through tolerance by the encapsulation of the expectation maximization algorithm and diagnostic model scheme. Fig. 1 shows the proposed implementation plan. Fault tolerant control systems are designed to achieve high reliability and survivability of the dynamic systems and processes. The fault tolerant scheme can work in various steps. The fault tolerant scheme has the following general possible steps [13], [7]:

• Fault modeling of the system comprising of sensor and actuator faults.

• Fault detection and estimation using forward-backward Kalman filter-based expectation maximization.

• Fault isolation using diagnostics model.

Consider a linear-time discrete model of the system:

$$x(k+1) = A_d x(k) + B_d u(k)$$
$$y(k) = C_d x(k) + D_d u(k) \tag{1}$$

where A_d, B_d, C_d, and D_d are the matrices of the discrete-time system of appropriate dimensions.
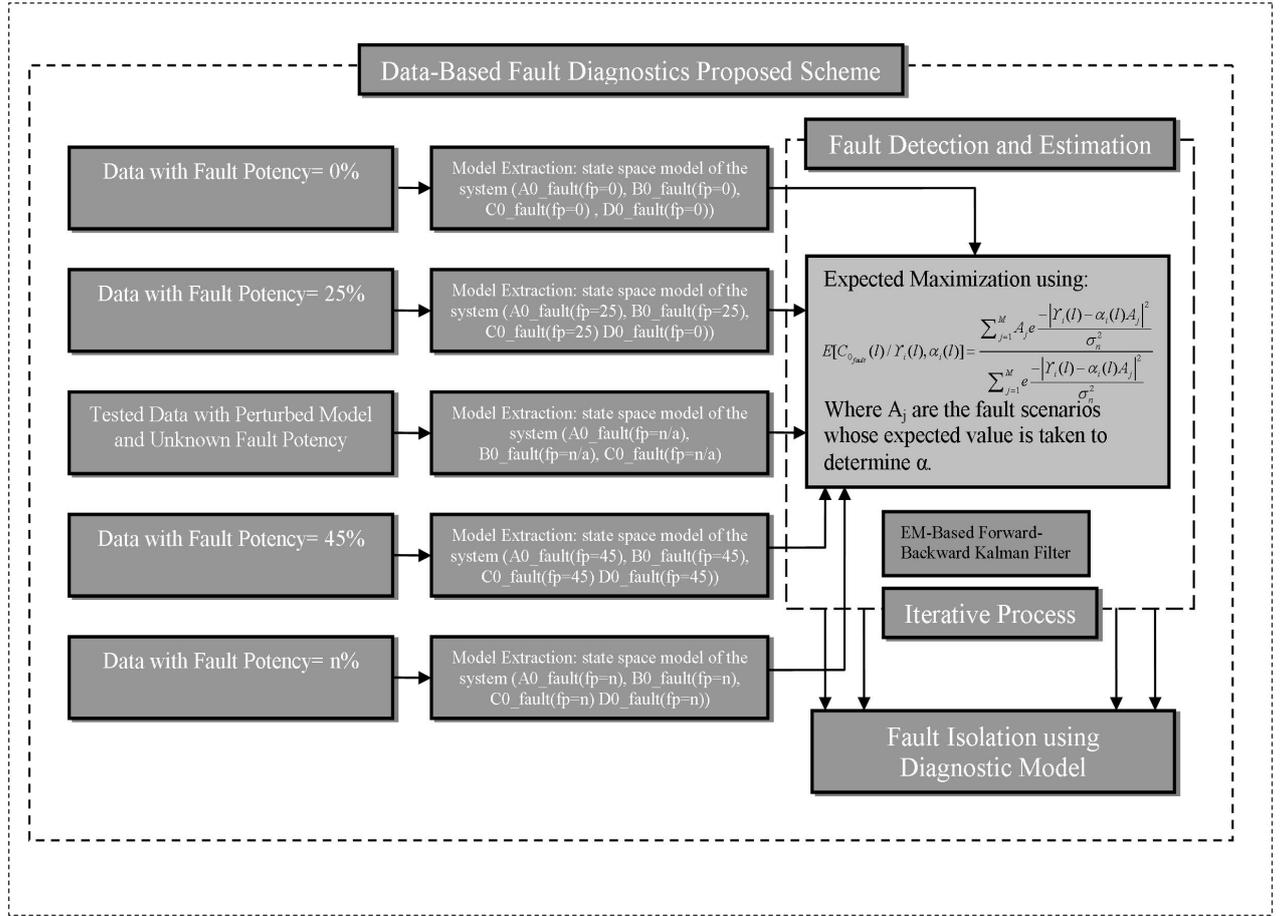
Fig. 1. Implementation plan- This figure has been adjusted and is clear now

A. System modeling with sensor and actuator faults

During the system operation, faults or failures may affect the sensors, the actuators, or the system components. These faults can occur as additive or multiplicative faults due to a malfunction or equipment aging. For fault detection and identification (FDI), a distinction is usually made between additive and multiplicative faults. The faults affecting a system are often represented by a variation of system parameters. Thus, in the presence of a fault, the system model can be written as:

$$x_f(k+1) = A_f x_f(k) + B_f u_f(k)$$
$$y_f(k+1) = C_f x_f(k) \qquad (2)$$

where the new matrices of the faulty system are defined by:

$$A_f = A + \delta A; B_f = B + \delta B; C_f = C + \delta C; \qquad (3)$$

δA, δB, and δC correspond to the deviation of the system parameters with respect to the nominal values. However, when a fault occurs on the system, it is very difficult to get these new matrices on-line. Process monitoring is necessary to ensure effectiveness of process control and consequently a safe and a profitable plant operation. As presented in the next paragraph, the effect of actuator and sensor faults can also be represented as an additional unknown input vector acting on the dynamics of the system or on the measurements. The effect of actuator and sensor faults can also be represented using an unknown input vector $f_j \epsilon \mathbf{R}^l$, $j = a$ (for actuators), s (for sensors) acting on the dynamics of the system or on the measurements.

A.1 Actuator faults

It is important to note that an actuator fault corresponds to the variation of the global control input U applied to the system, and not only to u:

$$U_f = \Gamma U + U_{f0} \qquad (4)$$

where
- U is the global control input applied to the system.
- U_f is the global faulty control input.
- u is the variation of the control input around the operating point U_0, ($u = U - U_0$, $u_f = U_f - U_0$.

- U_{f0} corresponds to the effect of an additive actuator fault.
- ΓU represents the effect of a multiplicative actuator fault with $\Gamma = diag(\alpha)$.

$$\alpha = [\alpha_1....\alpha_i....\alpha_m]^T \tag{5}$$
$$U_{f0} = [U_{f01}....U_{f0i}....U_{f0m}]^T \tag{6}$$

The i_{th} actuator is faulty if $a_i \neq 1$ or $u_f o \neq 0$. In the presence of an actuator fault, the linearized system 1 can be given by:

$$x(k+1) = Ax(k) + B(\Gamma U(k) + U_{f0} - U_0) \tag{7}$$
$$y(k) = Cx(k) \tag{8}$$

The previous equation can be re-written as follows:

$$x(k+1) = Ax(k) + B(\Gamma - 1)U(k) + U_{f0}$$
$$y(k) = Cx(k) \tag{9}$$

By defining $f_a(k)$ as an unknown input vector corresponding to actuator faults 2, equation can be represented as follows:

$$x(k+1) = Ax(k) + Bu(k) + F_a f_a k$$
$$y(k) = Cx(k) \tag{10}$$

where F_a=B, and f_a= $(\Gamma - 1) U + U_{f0}$. If the ith actuator is declared to be faulty, then F_a corresponds to the ith column of matrix B and f_a corresponds to the magnitude of the fault affecting this actuator.

A.2 Sensor faults

In the similar way, considering f_s as an unknown input illustrating the presence of a sensor fault, the linear faulty system will be represented by:

$$x(k+1) = Ax(k) + Bu(k) + F_s f_s k$$
$$y(k) = Cx(k) \tag{11}$$

The state-space representation of a system that may be affected by actuator and/or sensor fault:

$$x(k+1) = Ax(k) + Bu(k) + F_a f_a k$$
$$y(k) = Cx(k) + F_s f_s k \tag{12}$$

where matrices F_a and F_s are assumed to be known and f_a and f_s correspond to the magnitude of the actuator and the sensor faults, respectively. The magnitude and time occurrence of the faults are assumed to be completely unknown. In the presence of sensor and actuator faults, system can also be represented by the unified general formulation:

$$x(k+1) = Ax(k) + Bu(k) + F_x f(k)$$
$$y(k) = Cx(k) + F_y f(k) \tag{13}$$

where $f=[f_a^T \quad f_s^T]^T \epsilon \mathcal{R}^v$ $(v = m + q)$ is a common representation of sensor and actuator faults. $F_x \epsilon \mathcal{R}^{n \times v}$ and $F_y \epsilon \mathcal{R}^{q \times v}$ are respectively the actuator and sensor faults matrices with $F_x = [B \quad 0_{n \times q}]$ and $F_y = [B \quad 0_q]$. The objective is to isolate faults. This is achieved by generating residuals sensitive to certain faults and insensitive to others, commonly called

structured residuals . The fault vector f in 13 can be split into two parts. The first part contains the $'d'$ faults to be isolated f_0 ϵR^d. In the second part, the other $"v - d"$ faults are gathered in a vector $f^* \epsilon \mathcal{R}^{v-d}$. Then, the system can be written by the following equations:

$$x(k+1) = Ax(k) + Bu(k) + F_x^0 f^0(k) + F_x^* f^*(k)$$
$$y(k) = Cx(k) + F_y f(k) + F_y^0 f^0(k) + F_y^* f^*(k) \tag{14}$$

Matrices F_x^0, F_x^*, F_y^0, and F_y^*, assumed to be known, characterize the distribution matrices of f^* and f^0 acting directly on the system dynamics and on the measurements respectively.

In case of an ith actuator fault, the system can be represented according to 14 by:

$$x(k+1) = Ax(k) + Bu(k) + B_i f^0(k) + [\bar{B}_i \quad 0_{n \times q}]f^*(k)$$
$$y(k) = Cx(k) + [0_{q \times (p-1)} \quad I_q]f^*(k) \tag{15}$$

where B_i is the ith column of matrix B and $\bar{B}_i$ is matrix B without the ith column. Similarly, for a jth sensor fault, the system is described as follows:

$$x(k+1) = Ax(k) + Bu(k) + [B \quad 0_{n \times (q-1)}]f^*(k)$$
$$y(k) = Cx(k) + E_j f^0(k) + [0_{q \times p} \quad \bar{E}_j]f^*(k) \tag{16}$$

where $E_j = [0...1...0]^t$ represents the jth sensor fault effect on the output vector and $\bar{E}_j$ is the identity matrix without the jth column.

III. THE EXPECTED MAXIMIZATION ALGORITHM

In what follows, we look at the situation where joint fault detection and data estimation is considered. By and large, the expected maximization (EM) algorithm is a method for finding maximum likelihood estimates of parameters in statistical models, where the model depends upon unobserved latent variables. In our case, the latent variable is C_{0_i}. The EM has two steps. The $E - step$ is obtained with respect to the underlying unknown variables conditioned on the observations and the $M - step$ provides a new estimation of the parameters (or when some of the data are missing).

With reference to [1], the formulation has been applied for the fault diagnosis case, where the EM is being used for fault detection and data estimation. In the ideal case, we will be estimating fault α_i using the measurement equation, for maximizing the corresponding log likelihood function:

$$\Upsilon_{eh_i} = C_{0_i}\alpha_{i-1} + \upsilon_i \tag{17}$$

Where Υ_{eh_i} presents the electro-hydraulic profile, C_{0_i} represents the measurement matrix perturbed by the fault which cause some data loss which is to be recovered/estimated, α_{i-1} represents the leakage profile (a type of fault), and Υ_i represents the noise assumed to be Gaussian. For example, when the system is obeying the input output relation (so that in ℓn $p(\Upsilon_{eh_i}/C_{0_i}, \alpha_i) = -\|\Upsilon_i - C_{0_i}\alpha_i\|_{\sigma_n^{-2}}^2$ up to some additive constant), (so that $\ell n\, p(\alpha_i) = -\|\alpha_i\|_{\Pi^{-1}}^2$ in this case, the maximum a posteriori MAP estimate is given by:

$$\hat{\alpha}_i^{MAP} = argmin_{\alpha_i}[\|\Upsilon_i - C_{0_i}\alpha_i\|_{\sigma_n^{-2}}^2 + \|\alpha_i\|_{\Pi^{-1}}^2] \tag{18}$$

Considering the case of monitoring the fault detection, however, the input C_{0_i} is not observable as we have different scenarios for the profiles of faults depending upon the potency of the fault considered. Thus, we use the expectation-maximization algorithm and maximize instead an average form of the log-likelihood function. Thus, the E-step for the expected maximization Algorithm for the example given above when starting from an initial estimate, $\hat{\alpha}_i^0$ to estimate $\hat{\alpha}_i$ is calculated iteratively, with the estimate at the $j-th$ iteration given by:

$$\hat{\alpha}_i^{MAP} = arg\max_{\alpha_i}[E_{\alpha_i/\Upsilon_{eh_i}^{j-1} ln\, p(\Upsilon_{eh_i}/C_{0_i},\alpha_i)+lnp(\alpha_i)}] \quad (19)$$

Likewise, the M-step for the example given above will be as follows:

$$\hat{\alpha}_i^j = argmin_{\alpha_i}[\|\Upsilon_i - E[C_{0_i}]\alpha_i\|_{\sigma_n^{-2}}^2$$
$$+ \|\alpha_i\|_{cov[i^*]+\|\alpha_i\|_{\Pi^{-1}}^2}^2] \quad (20)$$

Where the two moments of C_{0_i} are taken given the output Υ_i and the most recent flow/height of water estimate $\hat{\alpha}_i^{j-1}$. We now derive the EM algorithm for the time variant case.

A. The EM-based forward-backward Kalman filter

Consider the system expressed in Section IV, essentially described by the state-space model:

$$\underline{\alpha}_{i+1} = F\underline{\alpha}_i + G\underline{u}_i \quad (21)$$

$$\Upsilon_{eh_i} = C_{0_i}\alpha_{i-1} + \nu_i \quad (22)$$

We can obtain the *maximum a posteriori* estimate by maximizing the log-likelihood as:

$$L = \ell n\,(\Upsilon_{eh_0}^t/C_0^t,\underline{\alpha}_0^t) + \ell n\, p(\underline{\alpha}_0^t) \quad (23)$$

Where T is the sampling time. Now, for describing the terms of likelihood, consider the two equations of the state space model (16) and (17). Considering the equation 22, we can express the first term of likelihood as:

$$\ell n\, p(\Upsilon_{eh_0}^t/C_0^t,\underline{\alpha}_0^T) = \sum_{i=0}^{T} \ell n\, p(\Upsilon_{eh_i}/C_i,\underline{\alpha}_i)$$
$$= -\sum_{i=0}^{T}\|\Upsilon_{eh_i} - C_i\underline{\alpha}_i\|_{1/\sigma_n^2}^2 \quad (24)$$

Similarly, considering the equation 21, we can express the second term of likelihood as:

$$\ell n\, p(\underline{\alpha}_0^t) = \sum_{i=1}^{T} \ell n\, p(\underline{\alpha}_i,\underline{\alpha}_{i-1}) + \ell n\, p(\underline{\alpha}_0)$$
$$= -\sum_{k=1}^{T}\|\underline{\alpha}_k - F\underline{\alpha}_{k-1}\|_{1/\sigma_n^2 GG*}^2 - \|\underline{\alpha}_0\|_{\pi_0^{-1}}^2 \quad (25)$$

Considering these two expressions 24 and 25, we get:

$$L = -\sum_{i=0}^{T}\|\Upsilon_{eh_i} - C_i\underline{\alpha}_i\|_{1/\sigma_n^2}^2$$
$$- \sum_{k=1}^{T}\|\underline{\alpha}_k - F\underline{\alpha}_{k-1}\|_{1/\sigma_n^2 GG*}^2 - \|\underline{\alpha}_0\|_{\pi_0^{-1}}^2 \quad (26)$$

Now, the forward-backward Kalman is implemented to get the input and output sequences. *Forward run:* Starting from the initial condition $P_{01-1} = var(\Upsilon_{eh})$ and $\underline{\alpha}_{01-1}$ and for $i = 1,....,T$, calculate

$$R_{e,i} = \sigma_n^2 I_{N+P} + C_{0_{leak}} P_{i/i-1} C_{0_{leak}} \quad (27)$$

$$K_{f,i} = P_{i/i-1} C_{0_{leak}}{}^* R_{e,i}^{-1} \quad (28)$$

$$\hat{\alpha}_i = (I_{N+P} - K_{f,i}, C_{0_{leak}})\hat{\alpha}_{i-1} + K_{f,i} * Y_i \quad (29)$$

$$\hat{\alpha}_{i+1/i} = F\underline{\alpha}_i \quad (30)$$

$$P_{i+1,i} = F_i(P_{i/i-1} - K_{f,i}R_{e,i}, K_{f,i}^*)F^* + \frac{1}{\sigma_n^2}GG^* \quad (31)$$

Backward run: Starting from $\lambda_{T+1/T} = 0$ and for $i = T, T-1,....,0$, calculate

$$\lambda_{i/T} = I_{P+N} - C_{0_{leak}}^* K_{f,i}^* F_i^* \lambda_{T+1/T}$$
$$+ C_{0_{leak}} R_{e,i}^{-1}(y_i - C_{0_{leak}}\hat{\alpha}_{i-1}) \quad (32)$$

$$\hat{\alpha}_{i/T} = \hat{\alpha}_{i/i-1} + P_{i/i-1}\lambda_{i/T} \quad (33)$$

The desired estimate is $\underline{\alpha}_{i/T}$: The forward-backward Kalman derives the MAP estimate of the system impulse response. In the forward step, the filer obtains the MAP estimate of given . Our aim, however, is to obtain the MAP estimate of $\underline{\alpha}_i$ given the whole sequence C_i^0 . The backward step adds the contribution of C_{i+1}^t to the MAP estimate of $\underline{\alpha}_i$.

B. Initial system estimation

The purpose is to get the observation C_0 from α, in order to get Υ.

We can obtain the initial system estimation from the measurement equation of (12). We can do this by implementing the Forward Backward Kalman Filter to the state-space model with substitution of $C_{0_i} \to C_{0_i I_P}$ and $\Upsilon_i \to \Upsilon_{eh_i, I_P}$:

$$\underline{\alpha}_{i+1} = F\underline{\alpha}_i + G\underline{u}_i \quad (34)$$

$$\Upsilon_{height, I_P} = C_{0_i I_P}\alpha_{i-1} + \nu_{i/p} \quad (35)$$

C. Calculating the input moments:

Input moments can be calculated the application of Bayes rules for evaluating the pdf of the function of the system: *Applying Bayes rule*

$$f(C_{0_i}(l)/Y_i(l),\alpha_i(l)) = \frac{f(C_{0_i}(l)/Y_i(l),\alpha_i(l))}{f(Y(l)_i/\alpha_i(l))}$$
$$= \frac{f(C_{0_i}, Yi/\alpha_i)}{\sum_{C_{0I}=A1}^{A_M} f(C_{0_i}, Yi/\alpha_i)}$$
$$= \frac{f(Yi/C_{0_i},\alpha_l)f(C_{0l},\alpha_l)}{\sum_{C_{0I}=A1}^{A_M} f(Yi/C_{0_i},\alpha_l)f(C_{0l},\alpha_l)}$$
$$= \frac{e^{\frac{-|Y_i-\alpha C_{0_i}|^2}{\sigma_n^2}}}{\sum_{j=1}^{M} e^{\frac{-|Y_i-\alpha A_j|^2}{\sigma_n^2}}}$$
$$\quad (36)$$

Where we have dropped the dependence on l. We have used the fact here that C_{0_l} is drawn from the alphabet $A = A_1, A_2, A_3, A_4$. A is not fixed here. There are fours As considered for the expectation of the fault scenarios where each A is showing a particular fault scenario. that is for finding the expected value at each instant, the expectation of four As will be taken. We can use this to show that: *First moment*:

$$E[C_{0_{fault}}(l)/Y_i(l), \alpha_i(l)] = \frac{\sum_{j=1}^{M} A_j e^{\frac{-|Y_i - \alpha C_{0_i}|^2}{\sigma_n^2}}}{\sum_{j=1}^{M} e^{\frac{-|Y_i(l) - \alpha_i(l)A_j|^2}{\sigma_n^2}}} \quad (37)$$

Second moment:

$$E[C_{0_{fault}}(l)/Y_i(l), \alpha_i(l)] = \frac{\sum_{j=1}^{M} |A_j|^2 e^{\frac{-|Y_i - \alpha C_{0_i}|^2}{\sigma_n^2}}}{\sum_{j=1}^{M} e^{\frac{-|Y_i(l) - \alpha_i(l)A_j|^2}{\sigma_n^2}}} \quad (38)$$

Thus, the EM-Based FB Kalman algorithm has been shown for implementation of fault detection and estimation.

IV. Fault Isolation using Diagnostics Model

The approach employed for the fault isolation is based on a diagnostic model, which directly relates the diagnostic parameters to the input and output. The diagnostic parameters are identified off line by performing a number of experiments. The diagnostic model relating the reference input r the diagnostic parameter γ and the residual $e(k)$, is given by:

$$e(k) = y(k) - y^0(k) = \sum_{i=1}^{q} \psi^T(k-1)\theta_i^{(1)} \Delta\gamma_i + \nu(k) \quad (39)$$

where, $\Delta\gamma_i = \gamma - \gamma_i^0$ is the perturbation in γ ; $y^0(k)$ and γ_i^0 are the fault-free (nominal) output and parameter, respectively, $\theta_i^{(1)} = \frac{\delta\theta}{\delta\gamma_i}$, and ψ is the data vector formed of the past outputs and past reference inputs. The gradient $\theta_i^{(1)}$ is estimated by performing a number of off line experiments which consist of perturbing the diagnostic parameters, one at a time. The input-output data from all the perturbed parameter experiments is then used to identify the gradients $\theta_i^{(1)}$. The outcome can be seen in the form of the cross-spectral density between the faulty data and fault-free data.

Remark IV.1: Power spectral density function shows the strength of the variations(energy) as a function of frequency. In other words, it shows at which frequencies variations are strong and at which frequencies variations are weak. It is a very useful tool if you want to identify oscillatory signals in our time series data and want to know their amplitude. For example, we have a chemical plant, and we have hydraulic drives operating to drive the fluid in the pipe network, and some of them have motors inside to pull the fluid with pressure. You detect unwanted vibrations from somewhere. You might be able to get a clue to locate offending machines by looking at power spectral density which would give you frequencies of vibrations. When we have two sets of time series data at hand and we want to know the relationships between them, we compute coherency function and some other functions computed from cross spectral density function of two time series data and power spectral density functions of both time series data. In this paper, we have two time series data of fault and fault-free case respectively. This property of power spectral density helps to treat the isolation case in a better way.

Remark IV.2: Cross-correlation is a measure of similarity of two waveforms as a function of a time-lag applied to one of them, whereas in autocorrelation, there will always be a peak at a lag of zero, unless the signal is a trivial zero signal. This property of cross-correlation helps to capture the fault signatures more coherently.

V. System Description

The electro-hydraulic system for this study is a rotational hydraulic drive at the LITP (Laboratoire dinteǵration des technologies de production) of the University of Québec École de technologie supérieure (ÉTS). The set-up is generic and allows for simple extension of the results herewith to other electro-hydraulic systems, for example, double-acting cylinders. Referring to the functional diagram in Fig. 2, a DC electric motor drives a pump, which delivers oil at a constant supply pressure from the oil tank to each component of the system. The oil is used for the operation of the hydraulic actuator and is returned through the servo-valve to the oil tank at atmospheric pressure. An accumulator and a relief valve are used to maintain a constant supply pressure from the output of the pump. The electro-hydraulic system includes two Moog Series 73 servo-valves which control the movement of the rotary actuator and the load torque of the system. These servo-valves are operated by voltage signals generated by an Opal-RT real-time digital control system. The actuator and load are both hydraulic motors connected by a common shaft. One servo-valve regulates the flow of hydraulic fluid to the actuator and the other regulates the flow to the load. The actuator operates in a closed-loop while the load operates open-loop, with the load torque being proportional to the command voltage to the load servo-valve. While the actuator and load chosen for this study are rotary drives, the exact same set-up could be used with a linear actuator and load, and thus, they are represented as generic components in Fig. 2. The test set-up includes three sensors, two Noshok Series 200 pressure sensors with a 010V output corresponding to a range of 20.7MPa (3000 PSI) that measure the pressure in the two chambers of the rotational drive, as well as a tachometer to measure the angular velocity of the drive. In order to reduce the number of sensors used (a common preference for commercial application), angular displacement is obtained by numerically integrating the angular velocity measurement.

Fig. 3 shows the layout of the system and the Opal-RT RT-LAB digital control system. The RT-LAB system consists of a real-time target and a host PC. The real-time target runs a dedicated commercial real-time operating system (QNX), reads sensor signals using an analog-to-digital (A/D) conversion board and generates output voltage signals for the servo-valves using a digital-to-analog (D/A) conversion board. The host PC is used to generate code for the target using MATLAB/Simulink and Opal-RTs RT-LAB software and also to monitor the system. Controller parameters can also be adjusted on-the-fly from the host in RT-LAB.
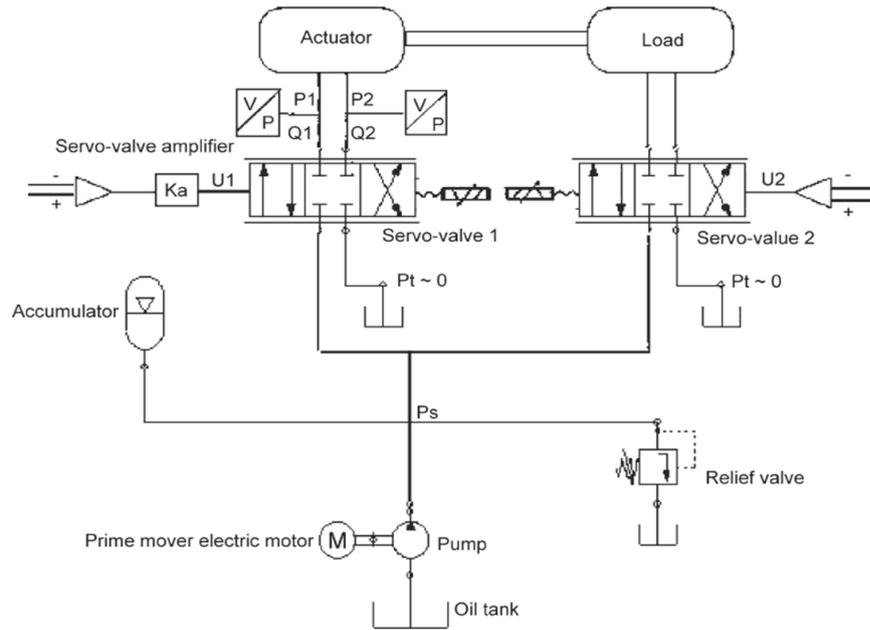
Fig. 2. Functional diagram -This figure has been adjusted and is clear now
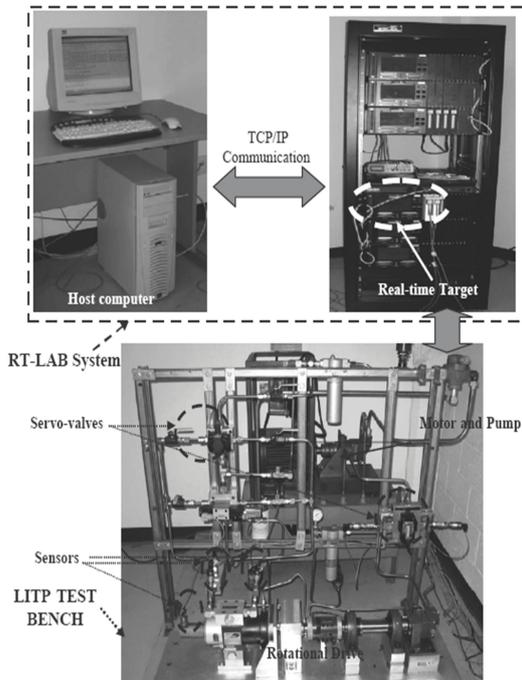

Fig. 3. Physical layout

A. Fault model for rotational hydraulic drive

In general, rotational hydraulic drive system is a drive or transmission system that uses pressurized hydraulic fluid to drive hydraulic machinery. The rotational hydraulic drive may experience various faults that reduce the performance and reliability. These can occur in components such as the pipe system, sensors, actuators, controllers, communication system elements and the actual platform. Since, in the rotational hydraulic drive, achieving good driving control is essential. This also requires the flow system of the hydraulic fluid should be working appropriately. Also, an excessive torque load can result in effecting the control of hydraulic drive. With these factors in mind, the faults considered in this study are those that cause leakage fault and controller fault.

A mathematical model of the system described is now developed based on the approach in [8] and [10]. First, the servo-valves are modeled with following assumptions:

1. The servo-valves are matched and symmetric.

2. The internal leakage inside the servo-valve can be neglected.

The dynamic equation for the servo-valve spool movement is given as [8] and [10].

$$\tau_v(\frac{dA_v(t)}{dt}) + A_v(t) = K_x.K_v.I(t), \qquad (40)$$

where t denotes time, $I(t)$ is the command input current, τ_v is the servo-valve time constant, $A_v(t)$ is the servo-valve opening area with sign dependent on flow direction, K_x is the servo-valve area constant, and K_v is the servo-valve torque motor con-

stant. $A_v(t)$ is said to have a positive sign when the servo-valve directs the flow such that the supply drives $P_1(t)$ and $P_2(t)$ drives the fluid to the tank. The reverse configuration is represented using a negative sign for $A_v(t)$, although the actual servo-valve opening area is always a positive number.

Let $Q_1(t)$ represent the flow from the servo-valve and $Q_2(t)$ represent the flow to the servo-valve. Then,

$$Q_1(t) = Q_2(t) = C_d A_v(t) \sqrt{\frac{P_s - P_L(t)}{\rho}} \qquad (41)$$

where $P_L(t)$ is the load pressure difference, P_s is the source pressure, C_d is the flow discharge coefficient and ρ is the fluid mass density. $P_L(t)$ and P_s are given by $P_L(t) = P_1(t) - P_2(t)$ and $P_s = P_1(t) + P_2(t)$ with $P_1(t)$ and $P_2(t)$ denoting the pressure in the two chambers of the rotational drive.

The fluid dynamic equation of the actuator, considering the compressibility of oil and internal leakage is given by:

$$\frac{V}{2\beta}\dot{P_L}(t) = C_d A_v(t) \sqrt{\frac{P_s - P_L(t) sign(A_v t)}{\rho}}$$
$$- D_m \dot{\theta}(t) - C_L P_L(t), \qquad (42)$$

where V is the oil volume under compression in one chamber of the actuator, β is the fluid bulk modulus, D_m is the volumetric displacement parameter of the actuator, C_L is the leakage coefficient, $\theta(t)$ is the angular displacement, and $sign$ is the sign function, which accounts for the change in direction of motion of the actuator.

Neglecting friction, the torque-acceleration equation of the actuator is given by:

$$J\ddot{\theta}(t) = D_m(P_1(t) - P_2(t)) - B\dot{\theta}(t) - T_L \qquad (43)$$

where J is the moment of inertia, B is the viscous damping coefficient, and T_L is the load torque. The variables $\dot{\theta}(t)$, $P_L(t)$, $A_v(t)$ are now normalized by dividing them by their respective maximum values denoted by ω_{max}; P_s and $A_v^{max} = K_x.K_v$. I_{max} to reduce numerical errors while performing simulation and real-time computations. The $sign$ function is approximated by the sigmoid function defined as:

$$sigm(x) = \frac{1 - e^{-ax}}{1 + e^{-ax}} \qquad (44)$$

where $a > 0$, to address the non-differentiable nature of the sign function.

B. Fault Scenarios

Fault scenarios are created by using the rotational hydraulic drive in the simulation program. In these scenarios leakage fault and controller fault are being considered.

B.0.a Scenario I: Leakage Fault.   In this scenario, while the system is working in real time, leakage faults is being introduced in the hydraulic fluid flow linked to the servo-valve of the system. The leakage fault is considered as $\omega_h C_{L_{leakage}} x_3(t)$ in state 3.

B.0.b Scenario II: Controller Fault.   In this scenario, while the system is working in real time and getting the input for driving the dynamics of the system, a fault has been introduced by increasing the torque load in the hydraulic drive, then effecting the controller, $-\frac{\omega_h}{\alpha} t_{L_{fault}}$ is considered in state 2 of the system.

Using equations (40)-(43) and the fault scenarios, the fault model of the system can be represented in state space form as:

$$\dot{x}_1(t) = \omega_{max} x_2(t) \qquad (45)$$
$$\dot{x}_2(t) = -\gamma \frac{\omega_h}{\alpha} x_2(t) + \frac{\omega_h}{\alpha} x_3(t) -$$
$$\frac{\omega_h}{\alpha} t_L - \frac{\omega_h}{\alpha} t_{L_{fault}} \qquad (46)$$
$$\dot{x}_3(t) = -\alpha \omega_h x_2(t) - \omega_h C_L x_3(t)$$
$$+ \alpha \omega_h x_4(t) \sqrt{1 - x_3(t)} sigm(x_4(t))$$
$$- \omega_h C_{L_{leakage}} x_3(t) \qquad (47)$$
$$\dot{x}_4(t) = -\frac{1}{\tau_v} x_4(t) + \frac{i(t)}{\tau_v} \qquad (48)$$

where

$$x_1(t) = \theta(t), \; x_2(t) = \frac{\dot{\theta}(t)}{\omega_{max}},$$
$$x_3(t) = \frac{P_L(t)}{P_s}, \; x_4(t) = \frac{A_v(t)}{A_v^{max}},$$
$$u_1(t) = i(t) = \frac{I(t)}{I_{max}}, \; u_2 = t_L = \frac{T_L}{P_s D_m},$$
$$\gamma = \frac{B\omega_{max}}{P_s D_m},$$
$$\omega_h = \sqrt{\frac{2\beta D_m^2}{JV}},$$
$$\alpha = \frac{(C_d A_v^{max} \sqrt{P_s \rho}) J \omega_h}{P_s D_m^2},$$
$$c_L = \frac{J C_L \omega_h}{D_m^2}$$

and $C_{L_{leakage}}$ is the leakage fault considered in state 3, $t_{L_{fault}}$ is the controller fault in the form of torque load in state 2.

Using the sign convention for $A_v(t)$ and the definition of $x_3(t)$, it follows that $0 \leq x_3(t) sigm(x_4(t)) \leq 1$. It is also noted here that $0 \leq x_3(t) sigm(x_4(t)) \leq 1$, because $P_1(t)$ and $P_2(t)$ are both positive and the condition $x_3(t) sigm(x_4(t)) = 1$ implies that $P_1(t) = P_s$ and $P_2(t) = 0$ or $P_2(t) = P_s$ and $P_1(t) = 0$, indicating zero pressure drop across the open ports of the servo-valve and thus, no flow to or from the actuator, a situation that would occur if the rotational motion of the drive is impeded.

VI. EVALUATION OF THE PROPOSED SCHEME

The evaluation of the proposed scheme has been made on the the electro-hydraulic system. The following sections show the detailed implementation and simulation of the proposed scheme.

A. Simulation Results

In what follows, we present simulation results for the proposed fault diagnostics scheme covering the fault detection and
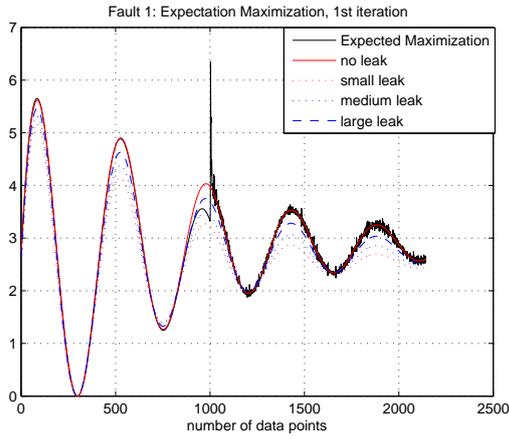
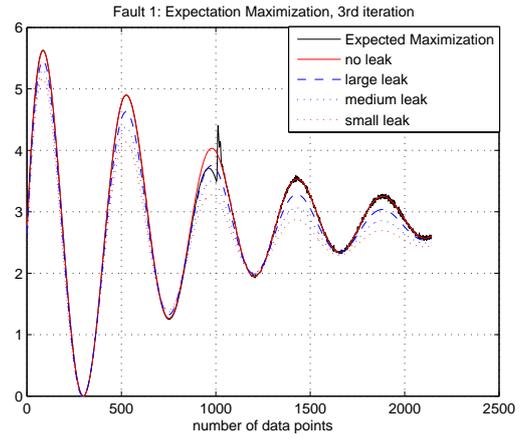Fig. 4. Fault 1: Implementation results on the leakage profile at iteration 1

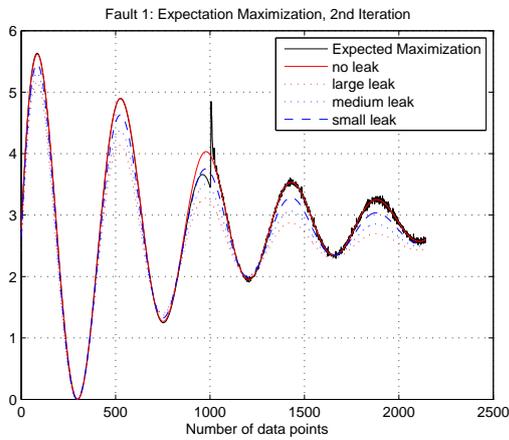Fig. 6. Fault 1: Implementation results on the leakage profile at iteration 3

Fig. 5. Fault 1: Implementation results on the leakage profile at iteration 2
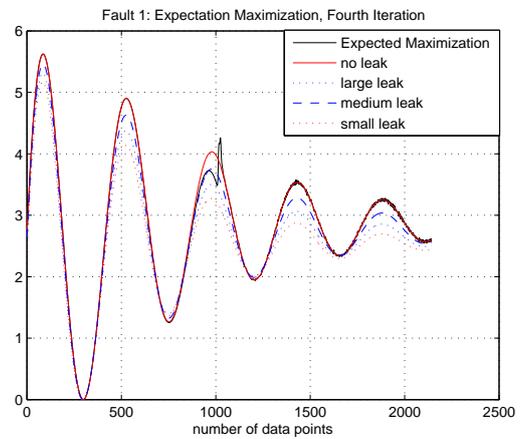

Fig. 7. Fault 1: Implementation results on the leakage profile at iteration 4

estimation. The tasks of our EM-Based Forward Backward Kalman scheme have been executed here with an increasing precision accompanied with a more detailed fault picture by increasing the number of iterations. Two sets of faults have been considered here i.e. the leakage fault in state 3 and controller fault. Firstly, the data collected from the plant has been initialized and the parameters have been being optimized which comprises of the pre-processing and normalization of the data. Then, the EM Based Forward Backward Kalman is implemented with an iterative process giving not only the recovery of the correct data, but also detecting the correct fault profile.

A.1 Fault 1 (Leakage): EM algorithm implementation

The EM Based FB Kalman scheme has been followed and employed here for the leakage fault to get a final profile of the lost data and the fault detection. It has been shown that the estimated profile at iteration 3 (Fig. 6) and Iteration 4(Fig. 7) is performing better in following the Original output as compared to the initial iterations 1 and 2 respectively (See Fig. 4 and Fig. 5), thus pointing clearly to the fault detection and recovery of the correct data.

A.2 Fault 2 (Controller Fault): EM algorithm implementation

The EM Based FB Kalman scheme has been followed and employed here for the controller torque fault to get a final profile of the lost data and the fault detection. It has been shown that the estimated profile at iteration 3 (Fig. 10) and Iteration 4(Fig 11) is performing better in following the Original output as compared to the initial iterations 1 and 2 respectively (See Fig. 8 and Fig. 9), thus pointing clearly to the fault detection and recovery of the correct data.

B. Simulation results: isolation using fault diagnostic model

B.1 Fault 1 (Leakage): isolation using fault diagnostic model

Further, for the leakage type fault 1, the information is fused in the fault-diagnostic model, and the results show that on the scale of number of observations for the leakage fault, we can judge the isolation of the fault by cross spectral density. as can be seen in Fig. 12, Fig. 13 and Fig. 14 for small, medium and large intensity of leakage faults respectively.
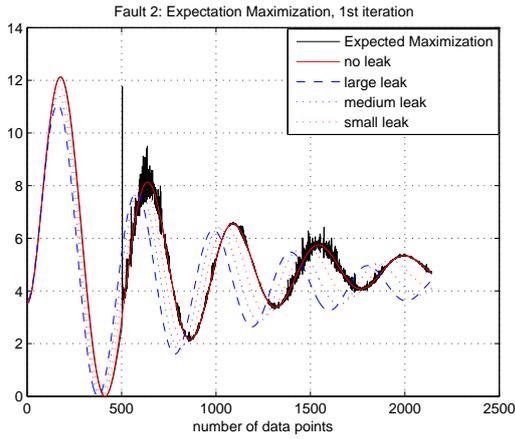
Fig. 8. Fault 2: Implementation results on the controller fault at iteration 1
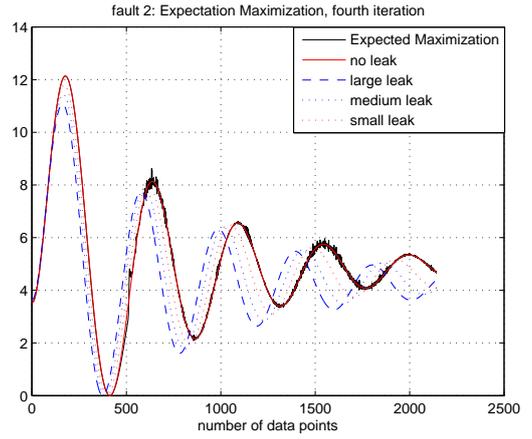

Fig. 9. Fault 2: Implementation results on the controller fault at iteration 2

Fig. 10. Fault 2: Implementation results on the controller fault at iteration 3

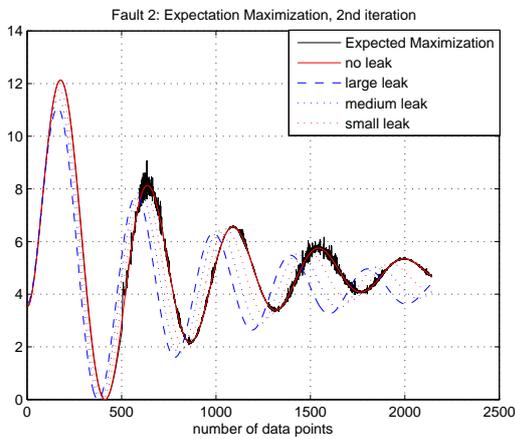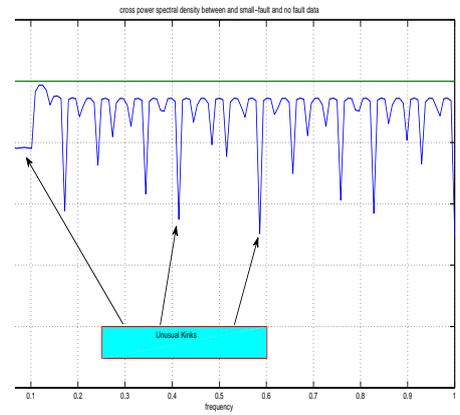Fig. 11. Fault 2: Implementation results on the controller fault at iteration 4

Fig. 12. Fault 1: Cross-power spectral density between small fault and no fault data
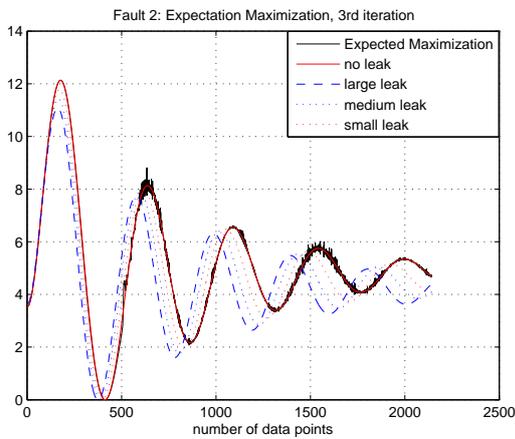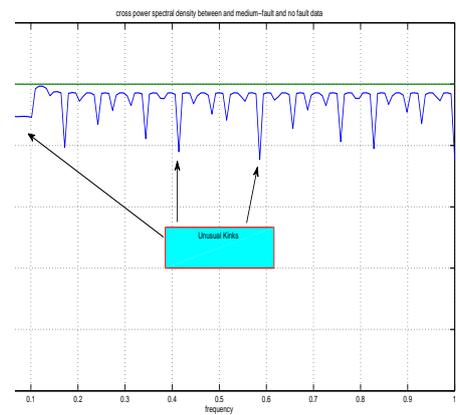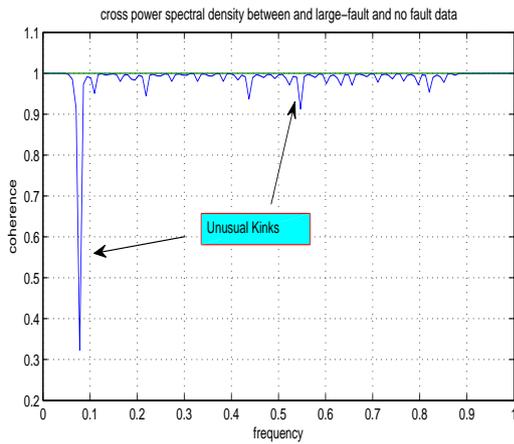

Fig. 13. Fault 1: Cross-power spectral density between medium fault and no fault data

Fig. 14. Fault 1: Cross-power spectral density between large fault and no fault data
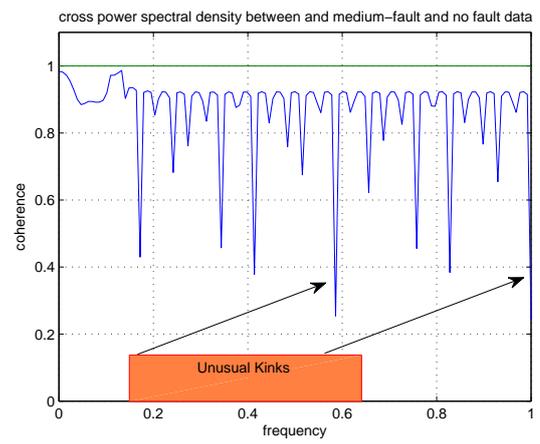

Fig. 16. Fault 2: Cross-power spectral density between medium fault and no fault data
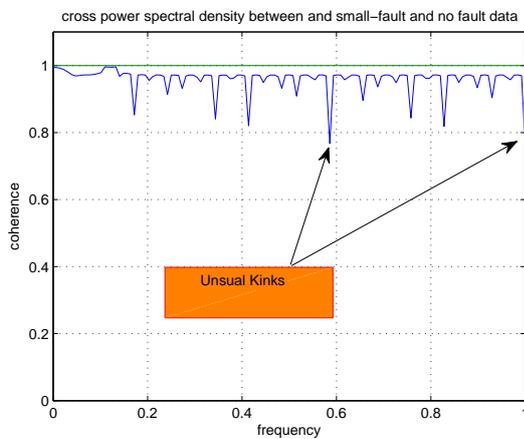

Fig. 15. Fault 2: Cross-power spectral density between small fault and no fault data
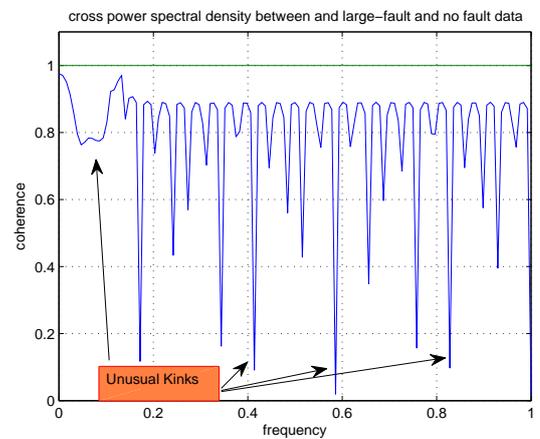

Fig. 17. Fault 2: Cross-power spectral density between large fault and no fault data

B.2 Fault 2 (Controller Fault): isolation using fault diagnostic model

Further, for the controller type fault 2, the information is fused in the fault-diagnostic model, and the results show that on the scale of number of observations for the controller fault, we can judge the isolation of the fault by cross spectral density. as can be seen in Fig. 15, Fig. 16, and Fig. 17 for for small, medium and large intensity of controller faults respectively.

VII. CONCLUSIONS

This work has presented a general approach to integrating data-based fault detection and estimation with fault isolation scheme. The proposed scheme has been developed based on expectation maximization (EM) algorithm and diagnostics model. The proposed scheme can function when information about the system faults, and structure and dynamics of the underlying data generation mechanism is inaccessible, incomplete or partially missing. The EM Approach has been motivated by its articulation on forward-backward Kalman filter thereby initiating the picture of fault detection and estimation. This picture is then completed by fault diagnostic isolation scheme. The proposed scheme has been evaluated on electro-hydraulic system thus ensuring the effectiveness of the approach.The major contribution of the paper is the integration of iterative expectation maximization-based approach boiled on forward-backward kalman filter for loss data-recovery with fault detection and fault diagnostic model for fault isolation to achieve both accuracy and reliability of data-based FDI scheme.

REFERENCES

[1] Al-Naffouri, T. Y. "An EM-based forward-backward Kalman filter for the estimation of time-variant channels in OFDM", *IEEE Trans. Signal Processing*, vol. 1, no.11, 2007, pp. 3924–3930.

[2] Aradhye, H.B., Bakshi, B.R., Strauss,R.A., Davis,J.F.,2003. Multi-scale SPC using wavelets: theoretical analysis and properties. A.I.Ch.E.Journal 49,939958.

[3] Bakshi, B. R., "Multi-scale PCA with application to multi-variate statistical process monitoring", *A. I. Ch. E. Journal*, vol. 44, 1998, pp. 1596-1610.

[4] Frank, P. M., "Fault diagnosis in dynamic systems using analytical and knowledge-based redundancya survey and some new results" , *Automatica*, vol. 26, 1990, pp. 459-474.

[5] Gertler, J., L. Weihua, Y. Huang and T. McAvoy,"Isolation enhanced principal component analysis", *A. I. Ch. E. Journal*, vol. 45, 1999, pp. 323-334.

[6] Hammouri, H., P. Kabore, S. Othman and J. Biston, "Failure diagnosis and non-linear observer application to a hydraulic process", *J. the Franklin Institute*, vol. 339, 2002, pp. 455-478.

[7] Hassan, N., D. Theilliol, J. -C. Ponsart and A. Chamseddine, *Fault-tolerant control systems: design and practical applications*, Springer, London, 2009.

[8] Kaddissi, C., J.-P. Kenne and M. Saad, "Identification and real time control of an electro-hydraulic servo system based on nonlinear backstepping", *IEEE/ASME Transactions on Mechatronics*, vol. 12, 2007, pp. 12-22.

[9] Kabore, P. and H. Wang, "Design of fault diagnosis filters and fault-tolerant control for a class of non-linear systems", *IEEE Trans. Automatic Control*, vol. 46, 2001, pp. 1805-1810.

[10] LeQuoc, S., R. M. H. Cheng,, and K. H. Leung, "Tuning an electro hydraulic servo valve to obtain a high amplitude ratio and a low resonance peak", *J. Fluid Control*, vol. 20, 1990, pp. 30-49.

[11] MacGregor, J. and T. Kourti, "Statistical process control of multi-variate processes", *J. Quality Technology*, vol. 28, 1996, pp. 409-428.

[12] Mhaskar, P., C. McFall, A. Gani, P. Christofides and J. Davis, "Isolation and handling of actuator faults in non-linear systems", *Automatica*, vol. 44, 2008, pp. 53-62.

[13] Mogens B., M. Kinnaert, J. Lunze and M. Staroswiecki, *Diagnosis and fault tolerant control*, 2nd Edition. Springer, 2006.

[14] Negiz, A. and A. Cinar, "Statistical monitoring of multi-variable dynamic processes with state-space models", *A. I. Ch. E. Journal*, vol. 43, 1997, pp. 2002-2020.

[15] Raich, A. and A. Cinar, "Statistical process monitoring and disturbance diagnosis in multi-variable continuous processes", *A. I. Ch. E. Journal*, vol. 42, 1996, pp. 995–1009.

[16] Romagnoli, J. and A. Palazoglu, *Introduction to process control*, CRC Press, Baco Raton, 2006.

[17] Venkatasubramanian, V., R. Rengaswamy, K. Yin and S. Kavuri, "A review of process fault detection and diagnosis part I: quantitative model-based methods", *Computers and Chemical Engineering*, vol. 27, 2003, pp. 293-311.

[18] Venkatasubramanian, V., R. Rengaswamy, S. Kavuri and K. Yin, "A review of process fault detection and diagnosis part III: process history based-methods", *Computers and Chemical Engineering*, vol. 27, 2003, pp. 327-346.

[19] Westerhuis, J. A., T. Kourti and J. F. MacGregor, "Analysis of multi-block and hierarchical PCA and PLS models", *J. Chemometrics*, vol. 12, 1998, pp. 301-321.

[20] Wise, B. M. and N. B. Gallagher, "The process chemometrics approach to monitoring and fault detection", *J. Process Control*, vol. 6, 1996, pp. 329-348.

[21] Mechefske, C., "Objective Machinery fault diagnosis using fuzzy logic", *Mechanical Systems and Signal Processing*, vol. 12, no.6, 1998, pp. 855–862.

[22] Jiang, Y., H. Wang, L. Fang and M. Zhao,"Fault detection and identification based on combining logic and model in a wall-climbing robot", *Journal of Control Theory and Applications*, vol. 7, no. 2, 2009, pp. 157-162.

[23] Yoon, S. and J. MacGregor, "Fault diagnosis with multi-variate statistical models part I: using steady state fault signatures", *J. Process Control*, vol. 11, 2001, pp. 387-400.

[24] Bodson, M., J. Groszkiewicz, Multivariable adaptive algorithms for reconfigurable flight control, *IEEE Trans. Control Systems Technology*, 1997; 5(1): 217-229.

[25] Chandler, P., M. Pachter and M. Mears, "System identification for adaptive and reconfigurable control," *J. Guidance, Control Dynamics*, 1995; 18(3): 516-524.

[26] Monaco, J. D., D. Ward, R. Barron and R. Bird, "Implementation and flight test assessment of an adaptive reconfigurable flight control system," *Proc. AIAA GNC Conf.*, New Orleans, LA, Aug. 1997, AIAA-973738.

[27] Gao, H. and X. Yang and P. Shi, "Multi-Objective Robust Hinf Control of Spacecraft Rendezvous," *IEEE Transactions on Control System Technology*, vol. 17, no. 4, pp.794-802, 2009.

[28] Yang, X. and H. Gao and P. Shi, "Robust Orbital Transfer for Low Earth Orbit Spacecraft with Small-Thrust," *Journals of the Franklin Institute*, vol. 347, no. 10, pp. 1863-1887, 2010.