Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

TuC11.4

Diagnosis of Parametric Faults Based on Identification and Statistical Methods

L. Brito Palma, F. Vieira Coito, R. Neves da Silva

Abstract—This paper proposes a fault diagnosis approach for detection and diagnosis of parametric faults in linear dynamical systems. It is assumed that the system is modeled by an auto-regressive (ARX) model, and a recursive least-squares (RLS) algorithm with variable forgetting factor is used for on-line parameter identification. The fault detection task is done using a statistical principal component analysis (PCA) method applied to ARX parameters. The fault isolation is done using an influence matrix method. The proposed methodology is applied to a simulation model of a DC motor under closed-loop control, and the performance is discussed.

Keywords: fault detection and diagnosis, system identification, statistical analysis.

I. INTRODUCTION

THE operation of technical processes requires increasingly advanced supervision and fault diagnosis methods to increase reliability, safety and economy, [7]. The monitoring of faults in feedback control system components is commonly named Fault Diagnosis, or Fault Detection and Isolation (FDI). Process monitoring and diagnosis techniques can be classified as being associated with one or more of three approaches to FDI, namely, *analytical*, *data-driven*, and *knowledge-based*.

During the last two decades, many investigations have been mostly made using *analytical approaches* to FDI based on quantitative models. The main idea is the generation of signals that reflect inconsistencies between nominal and faulty system operation. Such signals, termed *residuals*, are usually generated using analytical approaches: observers [3], parameter estimation [7], or parity equations [6].

The *data-driven* approaches to FDI are based directly on process data. The strength of data-driven techniques is their ability to transform the high-dimensional data into a lower dimension, in which the important information is captured, especially in large-scale systems that produce a large amount of multivariate data. Some common multivariate data-driven techniques are principal component analysis (PCA), Fisher discriminant analysis (FDA), partial least-squares (PLS), and canonical variate analysis (CVA) [4]. The PCA is the most widely statistical multivariate technique used in industry.

The *knowledge-based* approaches to FDI use qualitative models, and are especially suited for systems in which detailed mathematical models are not available. These kinds of approaches are usually based on causal analysis, expert systems, and/or pattern recognition [4]; fuzzy and neural techniques are commonly used. Like the analytical approach, most applications of the knowledge-based approach have been applied to systems with a relatively small number of inputs, outputs, and states.

Each process monitoring and fault diagnosis technique has its strengths and limitations. The combination of different FDI schemes can result in better process monitoring performance for many applications, [4], [7], [11].

The FDI methodology proposed in this paper follows this research direction of combined approach to FDI, focused on the main goal of increasing the overall performance. Section II presents the theoretical aspects of the proposed fault detection and diagnosis approach. Section III describes the simulation model of the DC motor used for simulations. Simulation results are depicted in section IV, and the conclusions appear in the last section.

The main contributions of the paper are: a) the fault detection approach based on the application of PCA analysis to parameters of ARX models; b) the improvement of fault isolation performance by introducing a variable forgetting factor in the RLS algorithm.

II. FAULT DETECTION AND DIAGNOSIS APPROACH

A. Discrete-time input-output ARX modeling

In this work, it is assumed that a discrete-time dynamical system is modeled by a single-input single-output (SISO) linear and time-invariant (LTI) system. The polynomial representation based on an auto-regressive with exogenous input model, ARX(na,nb,nk), will be used [13]:

$$A(q^{-1})\, y(k) = q^{-nk} B(q^{-1})\, u(k) + e(k) \qquad (1)$$
$$A(q^{-1}) = 1 + a_1\, q^{-1} + \ldots + a_{na}\, q^{-na}$$
$$B(q^{-1}) = b_0 + b_1\, q^{-1} + \ldots + b_{nb-1}\, q^{-(nb-1)}$$

In (1) the input and output signals are, respectively, $u(k)$ and $y(k)$, being k the discrete-time instant. The signal $e(k)$ is assumed to be a white noise signal. The argument q^{-1} denotes the backward shift operator, so $q^{-1}u(k) = u(k-1)$, etc.

The orders of the polynomials are *na* and *nb*, and *nk* is the pure time delay. The ARX parameter vector is expressed by $\theta = [a_1\, a_2\, \ldots\, a_{na}\, b_0\, b_1\, \ldots\, b_{nb-1}]$.

B. Closed-loop control architecture

Typically, most of the dynamical systems in industry work under closed loop control, as depicted in Fig. 1. Assuming a SISO system, the process plant is represented by the block P, and the blocks A and S represent, respectively, the actuator and the sensor. The blocks A/D and D/A are the analog-to-digital and digital-to-analog converters. The digital computer implements the supervision and control algorithms. The digital signals are also represented in the figure: the reference signal, $r(k)$, and the input and output signals, $u(k)$ and $y(k)$, respectively. The other signals are analog signals: the real process input $u_r(t)$, and the real process output $y_r(t)$. The signals $w(t)$ and $v(t)$ are, respectively, the disturbance input to the plant, and the disturbance or noise in the sensor.

C. Recursive least squares algorithm

In this work, a modified recursive least-squares (RLS) algorithm was used to perform the on-line identification of the parameters of the ARX model (1). This means that the parameters will be time-varying signals, and a function of the discrete time *k*:

$$\theta(k) = [a_1(k)\, a_2(k)\, \ldots\, a_{na}(k)\, b_0(k)\, b_1(k)\, \ldots\, b_{nb-1}(k)] \quad (2)$$

The traditional RLS algorithm may be found in [13]. A variable forgetting factor, $\lambda(k) = 1 - 1/\tau(k)$, is proposed in this work. $\lambda(k)$ is a function of the memory horizon $\tau(k)$. $\tau(k)$ is a design parameter, and different values are used to cope with different operating conditions: a) the start-up ($\tau(k) = \tau_s$), the nominal region without faults ($\tau(k) = \tau_n$), and the faulty situations where an exponential variation is assumed ($\tau(k) = \tau_f\, \eta(k) + \tau_s\,(1 - \eta(k))$; $\eta(k) = exp(-(k-k_f)/\beta)$). For the faulty case, k_f is the time instant of fault detection, and β is a design parameter (a decay time constant); k_f is updated after fault detection. The variable forgetting factor allows an increase in the RLS performance, especially in faulty situations where occurs changes in the system dynamics.
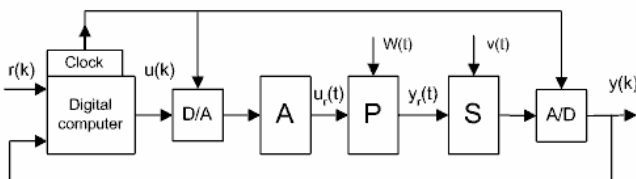

Fig. 1. Closed loop control architecture.

D. Fault detection based on principal component analysis

It is assumed that the system is working in steady-state and it is modeled by an ARX model; the parameter identification is done through a RLS algorithm, and parametric faults occur. A principal component analysis (PCA) model is introduced in this paper to perform fault detection. The main contribution of the paper is the application of PCA to ARX parameters, $\theta(k)$, instead of the traditional way of application to the input-output signals. The PCA technique is used here for the reduction of the parameter space of the ARX model, assuming that correlations exist between the parameters. This space reduction allows, typically, a reduction of dimension from five, or more, to two, or three dimensions; that allows a better visualization and understanding of the system behaviour.

The PCA multivariate analysis has been extensively studied and applied in many fields of science and engineering [4], [8], [14]. First a static PCA technique will be explained, and applied to build a PCA nominal model for the nominal region (without faults). This model will be used as a PCA reference model for fault detection. After that, will be explained how the PCA can be extended to dynamic PCA in order to cope with dynamical systems.

The static PCA technique will be explained in a concise form, applied to ARX parameters. Consider a system modelled by an ARX(*na,nb,nk*) model, with *m* elements in the parameter vector $\theta(k)$. If *n* samples are acquired in the nominal region, then a data matrix $X \in \mathscr{R}^{n \times m}$ will be available to compute the PCA nominal model:

$$X = [x_1\, x_2\, \ldots\, x_m];\ x_i \in \mathscr{R}^{n \times 1} \quad (3)$$

First the original data ($d_i = x_i$) is scaled, in order to each variable (each ARX parameter) has zero mean and unity standard deviation. This is done by:

$$x_i = (d_i - \mu_i)\, /\, \sigma_i \quad (4)$$

In the second step, the correlation matrix, $S \in \mathscr{R}^{m \times m}$, is computed accordingly to (5). The eigenvalue matrix $\Lambda \in \mathscr{R}^{m \times m}$, and the eigenvector matrix $V \in \mathscr{R}^{m \times m}$, are obtained by a singular value decomposition (SVD).

$$S = 1\, /\, (n - 1)\, X^T X = V\, \Lambda\, V^T \quad (5)$$

The third step, consists in choosing the number of principal components, *a*, to retain in the PCA model. A number of techniques are available, but there appears to be no dominant technique. In this work, the percent variance test is used, [4], [8]; this method determines *a* by calculating

the smallest number of loading vectors (*"principal components"*) needed to explain a specific minimum percentage of the total variance (generally, greater than 80 %). The first a eigenvalues, $\lambda_1 \geq \ldots \geq \lambda_a$, explains the greatest variance, and the corresponding loading vectors define the scores space.

In the fourth step, the scaled data is projected onto the scores space. Selecting the columns of the loading matrix $P \in \mathfrak{R}^{m \times a}$ to correspond to the loading vectors $V \in \mathfrak{R}^{m \times m}$ associated with the first a eigenvalues, the projections of the observations in X into the lower-dimensional space are contained in the score matrix, $T \in \mathfrak{R}^{n \times a}$:

$$T = X P \qquad (6)$$

If the number of principal components is two, $a = 2$, then a two-dimensional scores plot can be used to visualize the scores; the points in the two-dimensional scores space: $\{t_1(i), t_2(i)\}; i=1, 2, \ldots, n$.

In the final step it is necessary to compute the threshold for the scores. In this work, was assumed that only the first two principal components ($a = 2$) are retained in the PCA model; for this case, the elliptical confidence region for the scores is limited by an ellipse (threshold) accordingly the T^2 statistics, [4]:

$$t_1^2 / \lambda_1 + t_2^2 / \lambda_2 \leq T_\alpha^2 \qquad (7)$$
$$T_\alpha^2 = \frac{a(n-1)(n+1)}{n(n-a)} F_\alpha(a, n-a)$$

In (7), t_1 and t_2 are the projections along the orthogonal axes defined by the two main loading vectors, and λ_1 and λ_2 are the eigenvalues of greater variance. The T^2 statistics threshold is T_α^2, and $F_\alpha(a, n-a)$ can be obtained from an F-distribution table for a certain level of significance α, [8].

Several techniques exist for the implementation of dynamic PCA methods, [4]. In this work, a method based on a sliding window averaging the observations was implemented. The observations (ARX parameter vector), $\theta(i,k); i=1, 2, \ldots, L$, within a sliding window of length L are projected in the scores space defined accordingly (6); the obtained scores defines a data cluster. The centroid of the scores (cluster) associated to the sliding window is computed based on a *k-means clustering algorithm*, [2]. A fault is detected if the cluster centroid falls outside the elliptical confidence region defined by (7).

E. Fault isolation based on influence matrix method

A method for fault diagnosis based on an influence matrix (IMX) was first developed by Ono and co-workers, [10],

and some developments were done by Doraiswami and co-workers, [5], [12]. This method allows the detection, isolation and estimation of faults, and was developed for systems represented by input-output models. Assuming that the model parameters are multi-linear in the physical parameters (resistance, controller gain, etc), the influence of each physical parameter on the model parameters (called the *influence vector*) can be interpreted as a fault template line associated with that physical parameter. The influence matrix is assumed to have been computed off-line, and stored for later use in on-line fault diagnosis, [12].

Let's consider the parameter (feature) vector of an ARX(na,nb,nk) model as a column vector $\theta = [a_1 \ a_2 \ldots a_{na} \ b_0 \ b_1 \ldots b_{nb-1}]^T$, and the physical parameters denoted by $\gamma = [\gamma_1 \ \gamma_2 \ldots \gamma_p]^T$. The relationship between the model parameters and the physical parameters can be expressed as $\theta = f(\gamma)$. Let the nominal values of the physical parameters be denoted by γ^{nom}, and the vector of their deviations due to a fault by $\Delta\gamma = [\Delta\gamma_1 \ \Delta\gamma_2 \ldots \Delta\gamma_p]^T$. The vector $\theta = f(\gamma^{nom} + \Delta\gamma)$ can be written in the form of a Taylor's expansion, and assuming there is only a single fault and each component of the feature vector is multi-linear in the physical parameters, then the Taylor's expansion will be simplified as [12]:

$$\theta = \theta^{nom} + \frac{\partial\theta}{\partial\gamma_i}\Big|_{\gamma=\gamma^{nom}} \Delta\gamma_i \qquad (8)$$

In (8), the partial derivative of the feature vector, θ, with respect to the ith physical parameter, γ_i, is termed the ith influence vector, denoted by Ω_i. For each physical parameter, γ_j, the associated influence vector is defined as:

$$\Omega_j^{nom} = \frac{\partial\theta}{\partial\gamma_j}\Big|_{\gamma=\gamma^{nom}} \ ; for \ j = 1, \ldots, p \qquad (9)$$

Then the influence matrix, Ω, is the matrix whose columns are the p influence vectors given by (9):

$$\Omega = [\Omega_1 \ldots \Omega_p] = [\frac{\partial\theta}{\partial\gamma_1} \ldots \frac{\partial\theta}{\partial\gamma_p}]_{\gamma=\gamma^{nom}} \qquad (10)$$

Accordingly (8) and (9), the feature vector deviation, $\Delta\theta$, caused by a perturbation or fault, $\Delta\gamma_j$, obeys (11). When a fault occur, the feature vector deviation $\Delta\theta$ will be aligned with the associated influence vector Ω_i.

$$\Delta\theta = \Omega_i \Delta\gamma_i \qquad (11)$$

The presence of noise will cause error in estimation of ARX parameters, and the estimated $\Delta\theta$ will never lie

exactly in the direction of Ω_i, but will be expected to have the largest projection on Ω_i than on the other influence vectors, assuming that the influence vectors are well separated in the Euclidean space of feature vectors.

After the faulty physical parameter, say γ_i, has been determined, then an estimate of the value of its deviation, say $\Delta\gamma_i$, may be obtained by solving a least-squares estimation (LSE) problem. The optimal solution to this LSE problem is:

$$\Delta\gamma_i^* = (\Omega_i^T \Omega_i)^{-1} \Omega_i^T \Delta\theta = (\Omega_i^T \Delta\theta) / \|\Omega_i\|^2 \qquad (12)$$

The magnitude of the projection of $\Delta\theta$ on Ω_i is given by (13), where $\hat{\Omega}_i^T$ is the unitary vector in the direction of Ω_i.

$$P_{\Omega_i}^{\Delta\theta} = \|\Omega_i \Delta\gamma_i^*\| = |\hat{\Omega}_i^T \Delta\theta| \qquad (13)$$

In [12] is proposed the following method for single fault diagnosis. Having estimated θ (hence $\Delta\theta$), the index of the faulty physical parameter, γ_i, is determined by the largest projection of $\Delta\theta$ on the influence vectors, that is:

$$i = argmax_j \{ |\hat{\Omega}_j^T \Delta\theta| \; ; j=1, \ldots, p \} \qquad (14)$$

In this work, a new different method is proposed. Having estimated θ (hence $\Delta\theta$), the index of the faulty physical parameter, γ_i, is determined by the minimum angle (direct or indirect) between the feature vector deviation, $\Delta\theta$, and the influence vectors, Ω_i. With this method can be determined if a fault corresponds to an increase or a decrease of the physical parameter. The method consists in determine the direct angles (angles between $\Delta\theta$ and each Ω_i), and the indirect angles (angles between $\Delta\theta$ and each $-\Omega_i$). The faulty physical parameter is the one with the minimum angle obtain from all (direct and indirect) angles. If $\Delta\theta$ is more aligned with Ω_i than occurs an increase in γ_i, i.e., $\Delta\gamma_i > 0$; if $\Delta\theta$ is more aligned with $-\Omega_i$ than occurs a decrease in γ_i, i.e., $\Delta\gamma_i < 0$. An optimal estimation of the fault magnitude is computed by (12).

III. THE SIMULATION MODEL

In this section is described the simulation model used for the computer simulations, and the operating conditions.

A. The DC motor continuous-time model

To test the proposed FDI methodology a DC motor continuous-time simplified model was used. In armature controlled DC motors, the applied voltage $u_a(t)$ controls the angular velocity $\omega_r(t)$ of the shaft. A simplified continuous-time transfer function of the DC motor can be given by (15), a 2nd order system, [9]. K_m is the torque constant, L is the armature inductance, R is the armature resistance, J is the rotor inertia, K_f is the viscous-friction coefficient, and K_b is the back-emf constant. The values, in S.I. units, used for the simulations are: $K_m = K_b = 0.1$ [Nm/A], $L = 0.5$ [H], $R = 2$ [Ω], $J = 0.8$ [Kg m^2 / s^2], and $K_f = 0.2$ [Nms].

$$G_m(s) = \omega_r(s) / u_a(s) = \qquad (15)$$
$$K_m / ((L J) s^2 + (L K_f + R J) s + (R K_f + K_m K_b))$$

B. The closed-loop ARX model

To model the overall system, under closed-loop control, as depicted in Fig. 1, an ARX($na=3,nb=2,nk=6$) model was used relating the output signal $y(k)$ and the reference signal $r(k)$. In order to guarantee the persistent excitation conditions, a white noise dither signal with variance 1e-4 was added to $r(k)$.

A digital PI Controller, with anti-windup term, was implemented and tuned based on the relay method, [1]; the controller parameters are $K_p = 1.69$, $T_i = 6.42$ [s].

The actuator has unitary gain, and the sensor has a gain of four in order to have a closed-loop unitary static gain. A white noise signal with variance 1e-7 was added to the sensor.

C. The operating conditions

All the simulations were performed on the Matlab/Simulink environment. A hybrid simulation model was used: the DC motor model runs as a continuous-time model, and the algorithms runs on discrete-time; this allows a situation nearest the real environments.

The nominal ARX parameters, θ^{nom}, have been estimated in the nominal region; around 2400 samples have been used. For the RLS algorithm, the variable forgetting factor depends on the memory horizon (a design parameter). The values used for the memory horizon are: a) start-up region, $\tau_s = 20 / T_s$; b) nominal region, $\tau_n = 120 / T_s$; c) faulty situations, $\tau_f = 10 / T_s$, and $\beta = 12 / T_s$.

For the fault detection algorithm based on PCA applied to ARX parameters the following values were obtained. For the nominal region, the eigenvalues obtained are $\lambda = [3.36\ 1.07\ 4.3e\text{-}1\ 1.39e\text{-}1\ 4.06e\text{-}9]$. The first two principal components ($a = 2$) justify around 89 % of the total unitary variance; this value justify why only two principal components were retained in the PCA model. To compute the threshold for the nominal region, a value $F_\alpha (a,n\text{-}a) = 12.97$ was obtained from an F-distribution

table; this value was obtained for $n = L = 15$ (the sliding window length) and $a = 2$ (two principal components), and was multiplied by a gain (1.2), in order to reduce the false alarm rate. Only one PCA model has been build, in off-line, for the nominal region.

The fault isolation task only performs after a fault is detected. Fault isolation signal becomes j when fault j is isolated, accordingly the influence matrix method. For the fault isolation algorithm based on the influence matrix method, [12], two parametric faults were considered in this work. Fault F1a is an increase ($R \rightarrow 1.5\,R$) in the armature resistance. Fault F1b is a decrease ($R \rightarrow 0.5\,R$) in the armature resistance.

All the signals in the simulations were normalized to be in the range [0;1]. The sampling time used is $T_S = 1/3$ [s]. The duration of the simulation experiments is 1200 [s].

IV. SIMULATION RESULTS

A. Faults on motor resistance

The first fault considered is fault F1a, corresponding to an increase ($R \rightarrow 1.5\,R$) in the armature resistance. In Fig. 2, from top to bottom can be observed the following signals: the reference signal $r(k)$, the sensor signal $y(k)$, the actuator signal $u(k)$, the distance between the centroid and the origin for the PCA model $cd(k)$, the angle between $\Delta\theta$ and each influence vector $an(k)$, the fault magnitude $fm(k)$, the fault detection signal $fd(k)$, and finally the fault isolation signal $fi(k)$. In the figure can be observed that fault occurs at time instant 840 s, and the detection time delay is 30 s; this time depends on the transient behavior of the RLS algorithm. After the fault is detected, less than 1 s is sufficient to be well isolated; the minimum angle tends to zero, and that indicates an increase in the physical parameter, R. Accordingly to $fm(k)$, the estimated fault magnitude is about 0.6, a value near the real value (0.5). The dotted line in the plot for fd represents the real fault signal. A positive signal represented by a dotted line in $fi(k)$ graphic indicates an increase in R, and a negative signal indicates a decrease.

The fault F1b is a decrease ($R \rightarrow 0.5\,R$) in the armature resistance, and the simulations can be observed in Fig. 3. The PI controller cannot stabilize the system, but the diagnosis performance is better than the F1 case. The detection delay is 2 s, and 1 s is sufficient to isolate the fault. The angle tends to 180 degree, meaning a decrease in R, as shown by the negative (dotted) signal in $fi(k)$ graphic.

In the experiments were considered two influence vectors, Ω_i. One for the fault in the motor resistance, R, and the other for the fault in the proportional gain of the controller, K_p. The signals $an(k)$ and $fm(k)$ shows the features of each influence vector.

Fig. 4 shows the evolution of the PCA scores, for the faulty case F1b. The ellipse defines the threshold in the scores space.
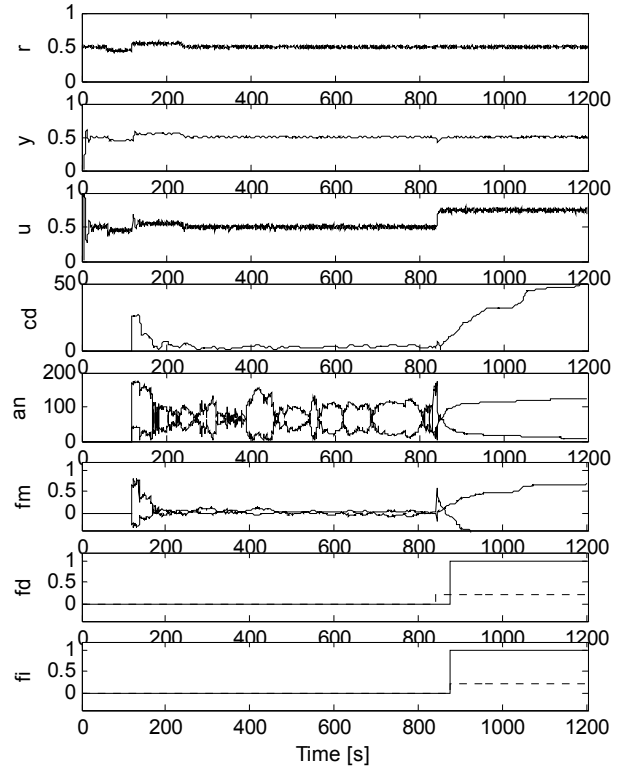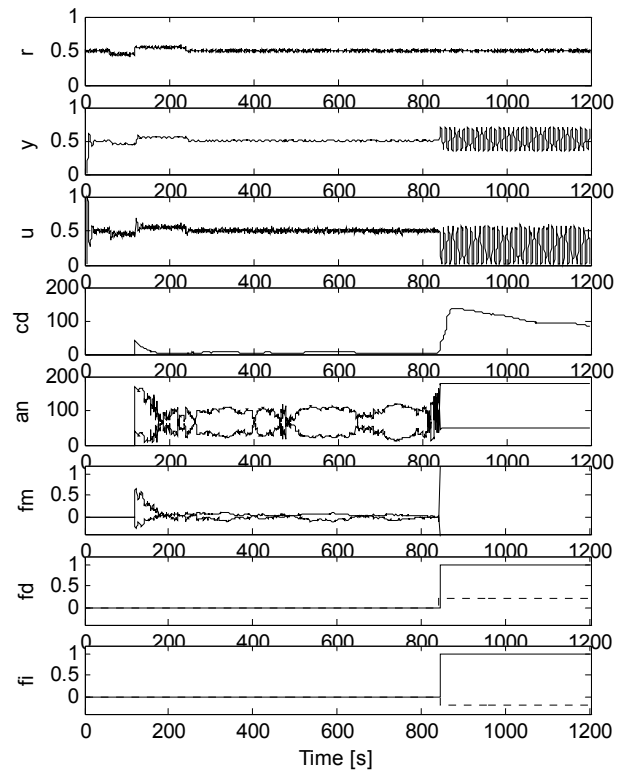

Fig. 2. Signals for fault F1a.

Fig. 3. Signals for fault F1b.

Fig. 4. Evolution of PCA scores for fault F1b.

For the faulty case F1b, Fig. 5 shows the influence vectors, Ω_i (labels iv1 and iv2), and the feature deviation vectors, $\Delta\theta$ (label dth), only in the plane {a1;a2} of ARX parameters; the nominal ARX parameters, θ^{nom}, have the label thnom. Other figures can be depicting to others planes ({a1;a3}; {a2;a3}; etc). The first influence vector iv1 is associated with fault on motor resistance, R, and the second influence vector is associated with fault on controller proportional gain, K_p. Can be observed that the angle between vectors iv1 and dth is approximately 180 degree; this is expected, since the fault F1b concerns a decrease in the motor resistance.

Fig. 5. Influence vectors and feature deviation vector for Fault F1b.

V. CONCLUSION

A combined approach to fault diagnosis was proposed. The proposed method requires a rich input signal, and an on-line identification of an ARX model, for the both tasks of fault detection and fault isolation.

A fault detection scheme based on the application of a data-driven technique (statistical PCA) to ARX parameters was proposed. Our experience shown that the threshold must be greater than the value given by the F-distribution table; the reason is due to the influence of noise in the ARX parameter estimation via RLS.

The influence matrix method for fault isolation reveals a reasonable performance. A good performance can be obtained if exits a linear relationship between model parameters and physical parameters.

The overall performance of the proposed methodology is good, and has the great advantage of do not require a precise mathematical model of the system. The application of the proposed methodology to control systems will improve the reliability and reduce the down-time.

The future work will be related to the improvement of the overall performance of this methodology, and the extension to nonlinear systems.

REFERENCES

[1] K. J. Astrom, and T. Hagglund, *Automatic Tuning of PID Controllers*. Instrument Society of America, 1988.
[2] M. Berry, and G. Linoff, *Data Mining Techniques*. Wiley, 1997.
[3] J. Chen, and R. Patton, *Robust Model-Based Fault Diagnosis for Dynamic Systems*. Kluwer Academic Publishers, 1999.
[4] L. H. Chiang, E. L. Russell, and R. D. Braatz, *Fault Detection and Diagnosis in Industrial Systems*. Springer, 2001.
[5] R. Doraiswami, and M. Stevenson, "A robust influence matrix approach to fault diagnosis," *IEEE Trans. on Control Syst. Tech.*, vol. 4, no. 1, pp. 29–39, Jan. 1996.
[6] J. Gertler, *Fault Detection and Diagnosis in Engineering Systems*. Marcel Dekker, 1998.
[7] R. Isermann, "Supervision, Fault detection and fault diagnosis methods – an introduction," in *Control Eng. Practice*, vol. 5, no. 5, pp. 639-652, May. 1997.
[8] J. Jackson, *A User's Guide to Principal Components*. Wiley, 2003.
[9] B. C. Kuo, *Automatic Control Systems*. Prentice Hall, 1995.
[10] T. Ono, T. Kumamaru, A. Maeda, S. Sagara, and K. Kumamaru, "Influence matrix approach to fault diagnosis of parameters in dynamic systems," *IEEE Trans. on Industrial Electronics*, vol. IE-34, no. 2, pp. 285–291, May 1987.
[11] L. B. Palma, F. V. Coito, and R. N. Silva, "Robust fault diagnosis approach using analytical and knowledge based techniques applied to a water tank system ," *International Journal of Engineering Intelligent Systems for Electrical Engineering and Communications – Special Issue on Knowledge Engineering*, to be published.
[12] J. Poshtan, R. Doraiswami, and M. Stevenson, "A real-time fault diagnosis and parameter tracking scheme," in *Proc. of the American Control Conference*, New Mexico - Albuquerque, 1997, pp. 483–487.
[13] T. Soderstrom, and P. Stoica, *System Identification*. Prentice Hall, 1989.
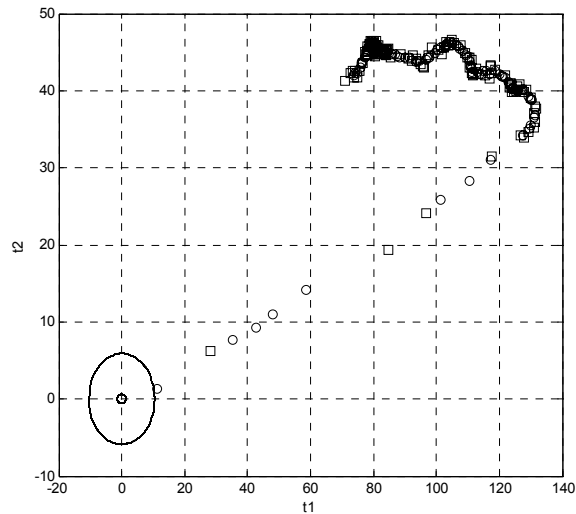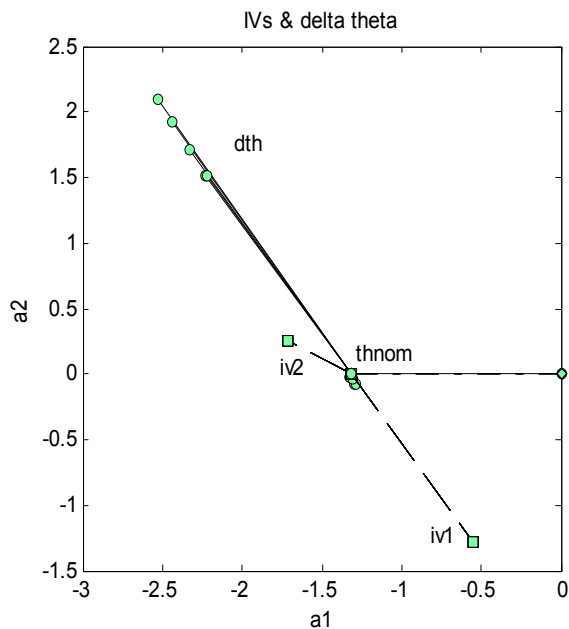[14] B. Wise, "Adapting Multivariate Analysis for Monitoring and Modeling of Dynamic Systems," Ph.D. dissertation, Dept. Chem. Eng., Univ., of Washington, 1991.