Parameter estimation methods for fault detection and isolation

Teresa Escobet*,+

*UPC, Universitat Politècnica de Catalunya
Automatic Control Department. Campus de Terrassa.
Terrassa, Barcelona, Spain
teresa@eupm.upc.es

Louise Travé-Massuyès,+**

**LAAS-CNRS
7 Avenue du Colonel-Roche
31077 Toulouse, France
louise@laas.fr

+LEA-SICA
European Associate Laboratory

1. Introduction

Fault detection via parameter estimation relies in the principle that possible faults in the monitored system can be associated with specific parameters and states of the mathematical model of the system given in the form of an input-output relation:

$$y(t)=f(\mathbf{u},e,\theta,x)$$

where $y(t)$ represents the output vector of the system, $\mathbf{u}(t)$ the input vector, $x(t)$ the state variables which are partially measurable, θ the non measurable parameters which are likely to change on the occurrence of a fault, and $e(t)$ the modeling errors and/or noise terms affecting the process.

The general procedure to detect faults follows the steps below:

(1) Establishment of the mathematical model of the system's normal behavior,

$$y(t)=f(\mathbf{u}(t),\theta)$$

At this stage, allowable tolerances for the system's parameter values are also defined.

(2) Determination of the relationship between the model parameters θ_i and the physical system parameters p_j.

(3) Estimation of the model parameters θ_i from measurements of $y(t)$, $u(t)$ by a suitable estimation procedure,

$$\hat{\theta}(t) = g(y(1),\cdots,y(t),u(1),\cdots,u(t))$$

(4) Calculation of the physical system parameters, via the inverse relationship:

$$\hat{p}(t) = f^{-1}(\hat{\theta}(t))$$

(5) Decision on whether a fault has occurred, based either on the changes Δp_j or on the changes $\Delta\theta_i$ and tolerances limits. If the decision is made based on the $\Delta\theta_i$ the affected p_i's can be easily determined from step 2. This may be achieved with the aid of a fault catalogue in which the relationship between process faults and changes in the coefficients Δp_j has been established. Decision can be made either by simply checking against the predetermined threshold levels, or by using more sophisticated methods from the field of statistical decision theory.

The basis of this class of methods is the combination of theoretical modeling and parameter estimation of continuous time models. The procedure is illustrated in figure 1.

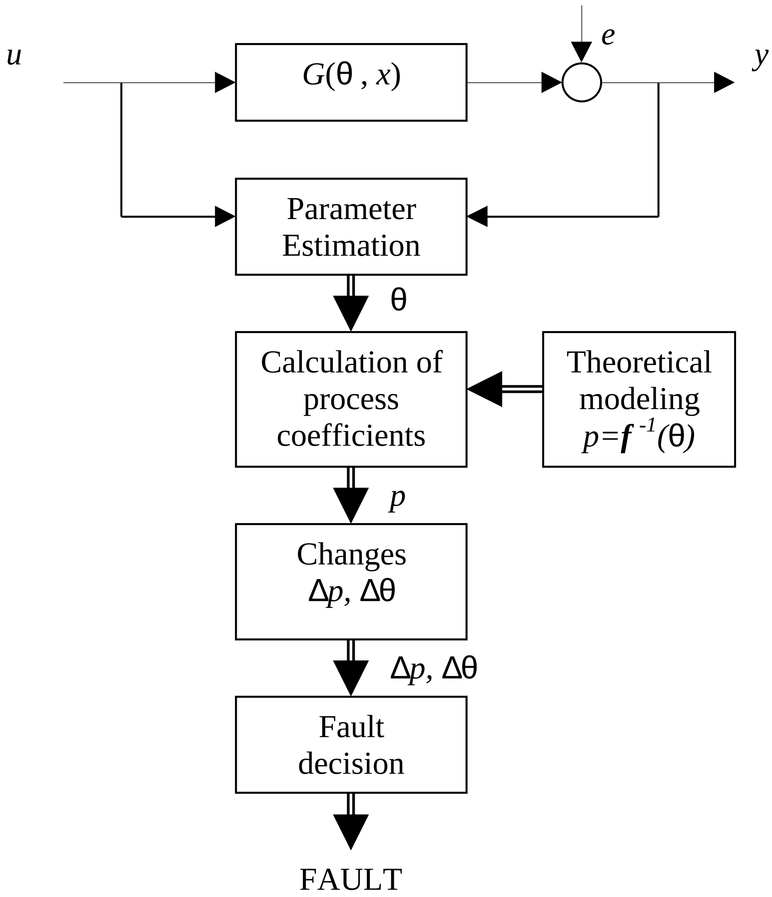

Figure 1.Fault detection based on parameter estimation an theoretical modeling

In this paper, the focus is put on the study of classical parameter estimation methods. The methods explained are then applied to the 3-tanks benckmark *[Lunze, COSY Benchmark Problem]*.

2. Parameter estimation for fault detection

In this section, two approaches for solving a Recursive Least Squares (RLS) algorithm are presented. To apply this method, other important issues like implementation and robustness, not explained here, must be taken into account. Much more information about these techniques can be obtained in *[Pouliezos & Stavrakakis 1994]*.

2.1. Recursive Least Squares algorithms

Given the system represented by the following input-output model:

$$y(t) = a_1 y(t-1) + \cdots + a_{n_a} y(t-n_a) + b_1 u(t-n_k) + \cdots + b_{n_b} u(t-n_b-n_k)$$

where a_{n_a} and b_{n_b} are the order model structure and n_k is the delay.

The Recursive Least Squares algorithm consist of:

$$k(t) = \frac{P(t-1)\phi(t)}{1+\phi^T(t)P(t-1)\phi(t)}$$

$$P(t) = P(t-1) - k(t)\phi^T(t)P(t-1)$$

$$\varepsilon(t) = y(t) - \phi^T(t)\hat{\theta}(t-1)$$

$$\hat{\theta}(t) = \hat{\theta}(t-1) + k(t)\varepsilon(t)$$

where:

$$\phi(t) = \left[-y(t-1)\ldots -y(t-n_a)u(t-nk)\ldots u(t-n_b-nk)\right]^T$$

$$\theta = \left[a_1 \ldots a_{n_a} b_1 \ldots b_{n_b}\right]^T$$

$\varepsilon(t)$ is the innovation error, $P(t)$ is the covariance matrix and $k(t)$ is the innovation gain.

The algorithm needs initial values: $\theta(0)$ and $P(0)$. These can be either provided from knowledge of the system characteristics or calculated from an initial data set using the non recursive least square method.

The result minimises the following expression:

$$V_N(\theta) = \sum_{t=1}^{N} \varepsilon^2(t)$$

where N is the number of samples.

Under the following mild conditions, the LS estimate is consistent, $\hat{\theta}$ i.e. tends to θ as N tends to infinity:

$$E\{\phi(t)\,\phi^T(t)\} \text{ is non singular}$$

$$E\{\phi(t)\,\varepsilon(t)\} = 0$$

The first condition guarantees the excitation level of the output signals and the second guarantees the statistical independence of the output signals and the error.

Tfrom the practical point of view, the convergence speed is generally slow which makes the standard RLS estimation method inadequate for real-time fault detection application. There are several approaches for modifying the RLS algorithm to make it suitable as a real-time fault detection method:

- Use of a forgetting factor
- Use of a virtual Kalman filter
- Use of sliding window data

2.2. Forgetting factor

In this case, the approach is to change the loss function to be minimized. Let the modified loss function be:

$$V_N(\theta) = \sum_{s=1}^{N} \lambda^{t-s}\varepsilon^2(s)$$

This means that the measures that are older than $T_0=1/(1-\lambda)$ samples are included in the criterion with a weight approximately equal to 36% of that of the most recent measurement. The T_0 means the memory time constant of the criterion and reflects the ratio between the time constant of variations in the dynamics and those of the dynamics itself. Typical choice of λ are in the range between 0.98 to 0.995 [Ljung 1986]. The RLS method with forgetting factor is:

$$k(t) = \frac{P(t-1)\phi(t)}{\lambda + \phi^T(t)P(t-1)\phi(t)}$$

$$P(t) = \frac{1}{\lambda}\left(P(t-1) - k(t)\phi^T(t)P(t-1)\right)$$

$$\varepsilon(t) = y(t) - \phi^T(t)\hat{\theta}(t-1)$$

$$\hat{\theta}(t) = \hat{\theta}(t-1) + k(t)\varepsilon(t)$$

Experiences for different values of λ show that a decrease in the value of the forgetting factor has two effects:

(1) The parameter estimates converge to their true value quicker, thus decreasing the fault alarm delay time.

(2) But at the expense of increased sensitivity to noise. If λ is much less than 1 the estimates may even oscillates around their true value.

To solve this problem, there are different approaches [Pouliezos & Stavrakakis, 1994]. Only two of them are presented here: the time-variant forgetting factor and Kalman filters.

· **Time-varying forgetting factor**

One algorithm to implement a variable forgetting factor is the proposed by [Fortescue et al. 1981]. The recursion consists in:

1. Prediction $\hat{y}(t) = \phi^T(t-1)\hat{\theta}(t-1)$
2. Error $\varepsilon(t) = y(t) - \hat{y}(t)$
3. Gain $k(t) = \dfrac{P(t-1)\phi(t)}{1+\phi^T(t)P(t-1)\phi(t)}$
4. Forgetting $\lambda(t) = 1 - \left(1-\phi^T(t-1)k(t)\right)\varepsilon(t)^2/\sigma_0$

   note : if $\lambda(t) < \lambda_{min}$ them $\lambda(t) = \lambda_{min}$
5. Covariance $P(t) = \dfrac{1}{\lambda(t)}\left(P(t-1) - k(t)\phi^T(t)P(t-1)\right)$

In this algorithm, the value of the constant σ_0 is the expected measurement noise variance which must be chosen based on the knowledge of the system. The minimum value for $\lambda(t)$ is also to be chosen by the user.

The intuitive idea behind the time-varying forgetting factor is that the forgetting factor is decreased towards its minimal allowed value as the error increases. In consequence, the data corresponding to a big error is “forgotten” faster.

· **Kalman filters**

The model can be described as a state space equation:

$$y(t) = \phi^T x(t) + e(t)$$

where the ‘state vector’ *x(t)* is given by:

$$x(t) = \left[a_1 \ldots a_{n_a} b_1 \ldots b_{n_b}\right]^T = \theta$$

To model the time-varying ‘states’, the state equation can be described as:

$$x(t+1) = x(t) + v(t); E\left\{v(t)v^T(s)\right\} = R_1\delta_{t,s}$$

This means that the parameter vector is modeled as a non-correlated random drift (which assumes slow variation). The covariance matrix R_1 can be used to describe how fast the different components of θ are expected to vary. The recursive algorithm obtained as a result to apply the Kalman filter to the model is:

$$k(t) = \frac{P(t-1)\varphi(t)}{1+\varphi^T(t)P(t-1)\varphi(t)}$$
$$P(t) = \left(P(t-1) - k(t)\varphi^T(t)P(t-1)\right) + R_1$$
$$\varepsilon(t) = y(t) - \varphi^T(t)\hat{\theta}(t-1)$$
$$\hat{\theta}(t) = \hat{\theta}(t-1) + k(t)\varepsilon(t)$$

In this algorithm R_1 has a similar role as the forgetting factor λ. These design variables should be chosen by trade-off between fast detection (which requires λ "small" or R_1 "large") and reliability (which requires λ close to 1 or R_1 "small").

3 Benchmark process fault detection

The studied system is presented in figure 2 *[Lunze, COSY Benchmark Problem]*.

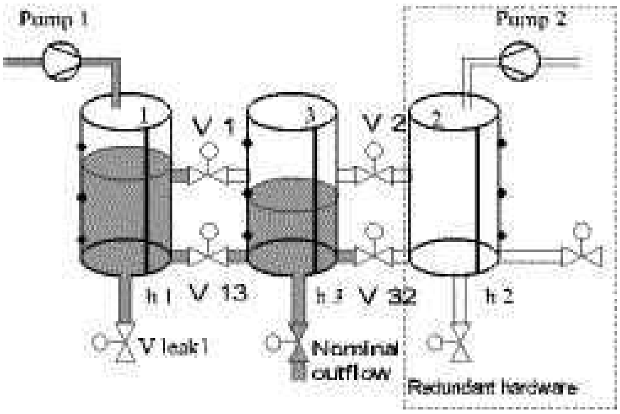

Figure 2. Process configuration and notations

This system can be modeled using the following linear equations:

$$\Delta h_1(z) = \frac{b_{11}z^{-1}}{1-a_1z^{-1}}\Delta Q - \frac{b_{12}z^{-1}}{1-a_1z^{-1}}\Delta S_n$$
$$\Delta h_3(z) = \frac{b_{21}z^{-1}}{1-a_2z^{-1}}\Delta h_1 - \frac{b_{22}z^{-1}}{1-a_2z^{-1}}\Delta S_n$$

where:

Δh_1 is the variation of h_1 around the working point, 0.5 m

Δh_3 is the variation of h_3 around the working point, 0.1 m

ΔQ is the variation of pump flow Q around the working point, $2.76 * 10^{-5}$

ΔS_n is the variation of valve V_1 position at the working point, 0.7

The working point has been chosen as proposed in the benchmark.

The methods used for fault detection have been:

- A RLS with a constant forgetting factor of 0.995, showed in the figures in blue color;
- A RLS algorithm with variable forgetting factor, showed in the figures in green color;
- A Kalman filter modified RLS algorithm, showed in the figures in red color;

These methods have been applied to detect faults in the scenarios I, II and III proposed in the benchmark *[Lunze, COSY Benchmark Problem]*.

4.1. Scenario I

Scenario I corresponds to valve V1 blocked closed from time 1000. The figure 3 shows the estimate values of the parameters.

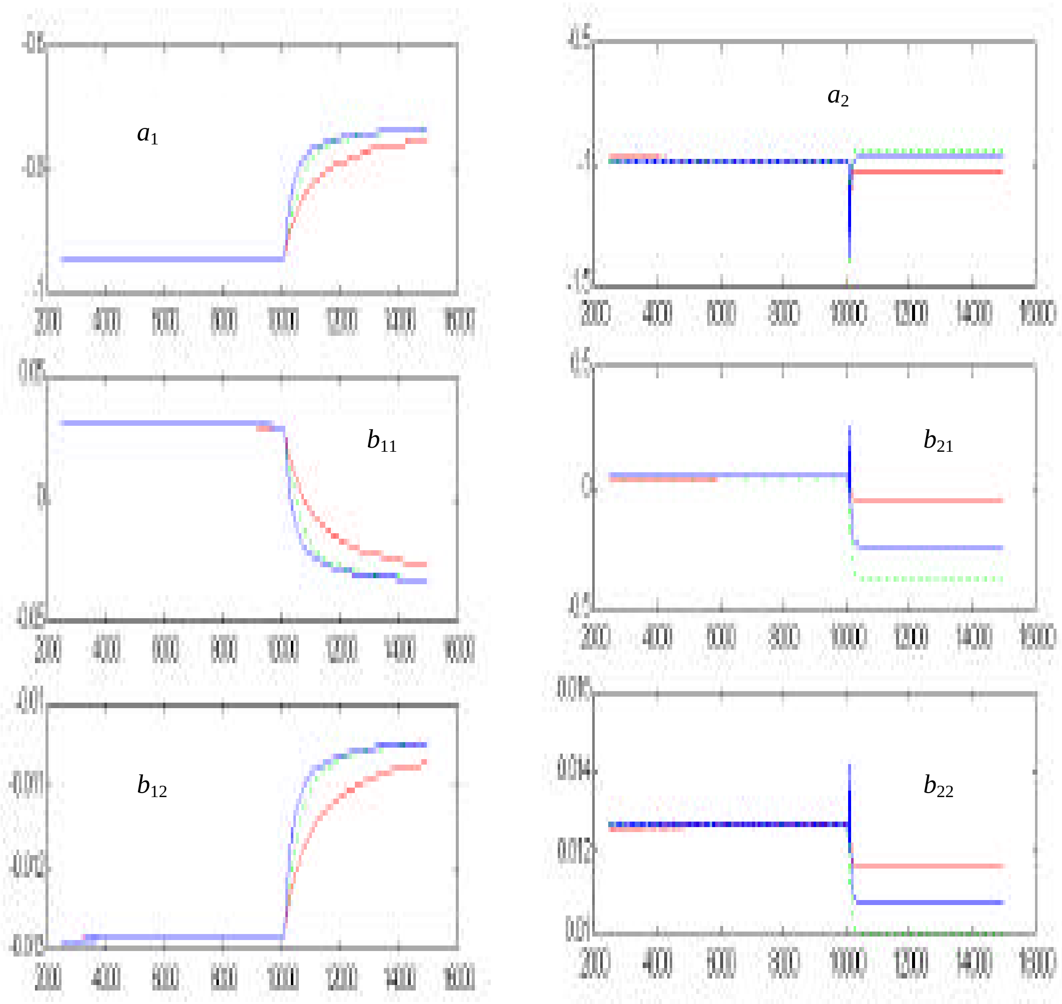

Figure 3. Estimated parameter values using scenario I

When the fault is occurs, the recursive estimation algorithm converges towards the new values. The rate of convergence depends on the weighting factor. In this example the Kalman filter has a rate of convergence significantly slower than the two other ones.

Note that in this scenario, the fault (valve V1 is blocked and closed) changes the model structure. As we maintain the same model structure after the fault, the estimated values of the parameters have no relation with the reality. Even though the change of estimated values is an indication of the fault.

4.2. Scenario II

Scenario II corresponds to valve V1 blocked opened. Figure 4 shows the estimated values of the parameters.

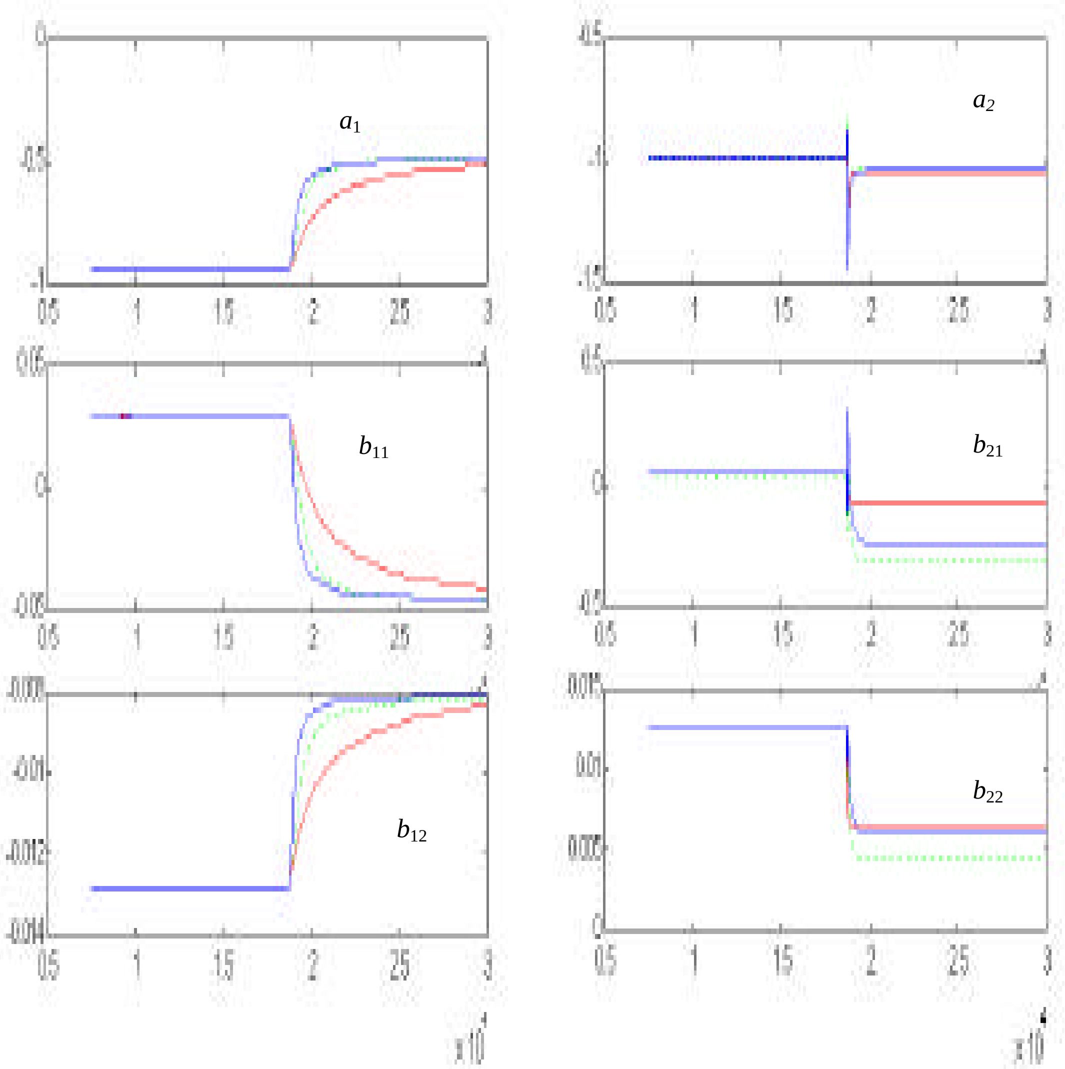

Figure 4. Estimated parameter values using scenario II

As for scenario I, when the fault occurs, the recursive estimation algorithm converges towards the new values. In this example the Kalman filter has also a rate of convergence much slower than the two other ones. It is possible to increase the *R*1 value but in this case the response variance increases.

The major difference with scenario I is that in this case the new value of a_2 is unstable. Actually, both scenarios have convergence problems due to the non excitation of input/output signals.

4.3. Scenario III

Scenario III corresponds to a leak in tank 1, the fault occurs at time 800 seconds. Figure 5 shows the estimated values of the parameters.

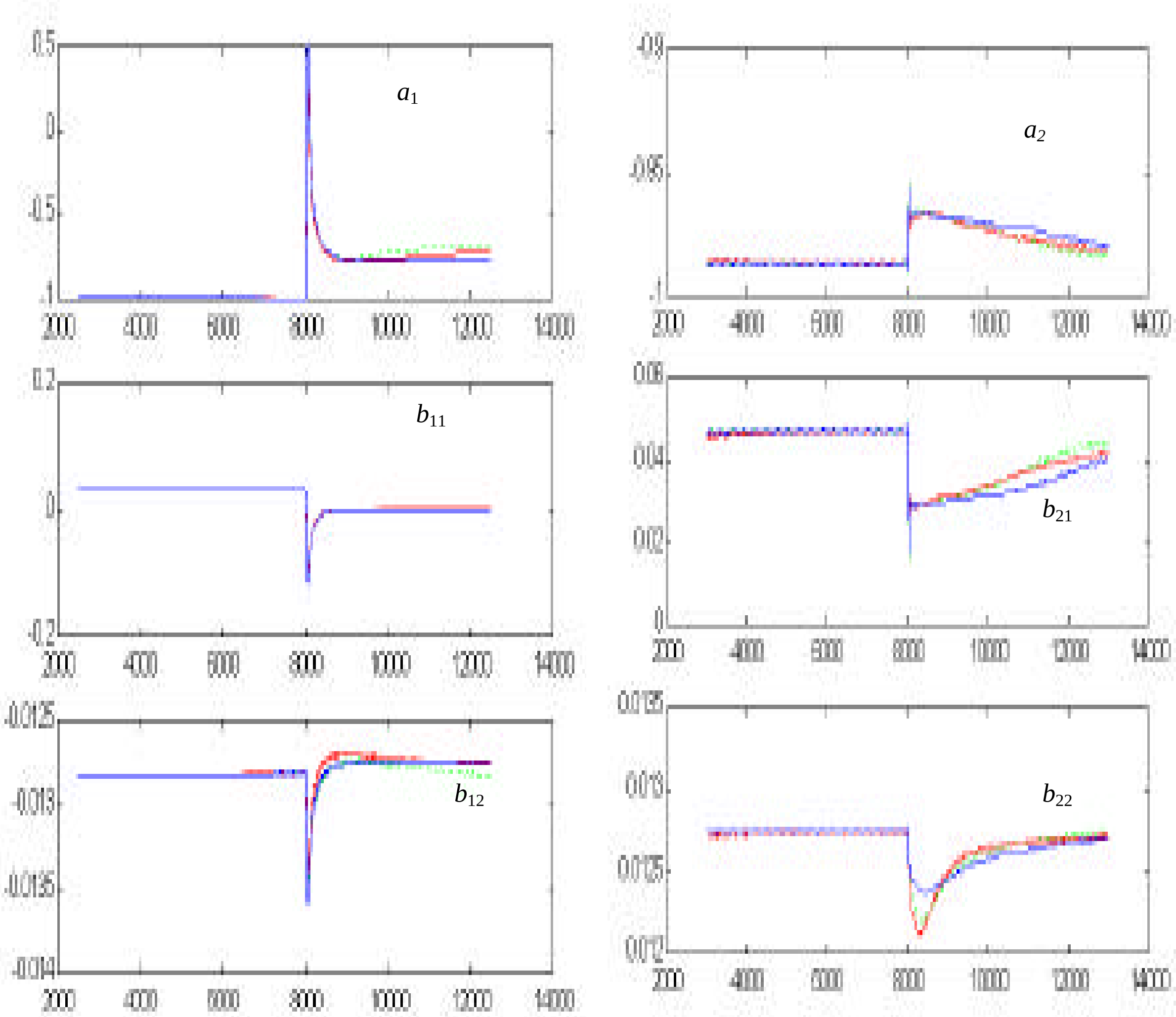

Figure 6. Estimate parameter values using the scenario III

When the fault occurs at time 800, the estimation responds slowly and converging to the new values takes nearly 15 minutes. The rate of convergence of the estimation could be increased by decreasing the weighting factor (or equivalent), but the variance of the estimation response would be larger.

In the new steady state the values of the parameters a_1, b_{11} and $b_{12,}$ which correspond to the first tank, have different values than their initial values whereas the parameters corresponding to the second tank converge back to their initial values. This localizes the fault on the first tank.

5. Conclusions

This paper presents three classical RLS based parameter estimation methods that have been applied to the 3 tanks benchmark.

It should be mentioned that the 3 tanks benchmark is not ideal to test identification methods because two out of three of the proposed faults change the model structure. This is indeed a difficult problem, even with other approaches (observer-based or parity space approaches).

An advantage of these methods is to approach simultaneously the fault detection and fault isolation problems. This is of course conditioned by the knowledge of the inverse relation allowing one to go from model parameters to physical parameters.

A problem that must be reported about estimation methods is their high sensitivity to the parametrization, i.e. in our case, the values of *l* and *R1*, which may completely change the algorithm behavior, turning unstable for instance. Note that other methods exist as a solution to the problem of the covariance matrix being unstable or singular (result of pour excitation signals) *[Pot et al. 1984]*.

Our experiments show that the convergence rate of the algorithms is a critical issue. It should be outlined that our experiments are aimed at illustrating the estimation algorithms behavior but they do not really address the detection problem. Indeed, the problem of detecting changes in a signal (the estimated ones) is a problem by itself. It is often solved by simply checking against predetermined thresholds [*qmin*i, *qmax*i] but there is a large literature on more sophisticated decision procedures to detect changes/faults in signals *[Hägglund 1984][Basseville & Nikiforov 1993].* Also the fault detection robustness issue is addressed, for instance by *[Wahlberg 1990][Kwon and Goodwin 1990]*.

Bibliography

Basseville M., Nikiforov I., (1993) *Detection of Abrupt changes : Theory and application,* Englewood Cliffs, NJ: Prentice Hall.

Fortescue T.R., Kershenbaum L.S. and Ydstie B.E. , (1981). Implementation of seft-tuning regulators with variable forgetting factors. *Automatica,* **17**, 6, 831-835.

Hägglund T., (1984). Adaptive control of systems subject to large parameter changes. *Proceedings, IFAC 9th Triennial World Congress,* Budapest, Hungary, 993-998.

Jung L., (1987). System Identification. Theory for the user. *Prentice Hall Information and System Sciences Series,* United States.

Kwon O.K. and Goodwin G.C., (1990). A fault detection method for uncertain systems with unmodeled dynamics, linearization errors and noisy inputs. *Proceedings, 11th IFAC Triennial World Congress,* Tallinn, Estonia, 367-372.

Lunze J., COSY Benchmark Problem, Laboratory 3 Tank System, Benchmark for the reconfiguration problem version 1.2, in "Projects" of http://www.tu-harburg.de/rts .

Pot J., Falinower V.M. and Irving E., (1984). Regulation multivarible adaptive des fours. *Colloque CNRS "Commande Adaptative. Aspects Practique et Theoriques",* St. Martin d'Heres.

Pouliezos A.D., Stavrakakis G.S. *(1994).* Real Time Fault monitoring of Industrial Processes. *Kluwer Academic Publishers.*

Wahlberg B. (1990). Robust frequency domain fault detection/diagnosis. *Proceedings 11th IFAC Triennial World Congress,* Tallinn, Estonia, 373-378.
LIENHYPERTEXTE