A TOOLBOX FOR MODEL-BASED FAULT DETECTION AND ISOLATION

L. Tesař*, L. Berec*†, G. Dolanc+, G. Szederkényi, J. Kadlec*, D. Juričić+, K. Hangos** and M. Kinnaert††**

**Institute of Information Theory and Automation AV CR, P.O.B. 18, 182 08 Prague 8, Czech Republic*
†*Department of Theoretical Biology, Institute of Entomology AV CR, Branišovská 31, 370 05 České Budějovice, Czech Republic*
+*Department of Computer Automation and Control, J. Stefan Institute, Jamova 39, SI-1001, Ljubljana, Slovenia*
***Computer and Automation Research Institute MTA, P.O.B. 63, H-1518 Budapest, Hungary*
††*Laboratoire d'Automatique et d'Analyse des Systèmes, Université Libre de Bruxelles, CP 165/55, 50 Av. F.D. Roosevelt, B-1050 Brussels, Belgium, Fax : +322 6502677 and e-mail : kinnaert@labauto.ulb.ac.be*

Keywords : Fault detection, Fault isolation, Computer-aided diagnosis, Software tools.

Abstract

A toolbox for model-based fault detection and isolation (FDI) has been developed in the MATLAB/SIMULINK environment. It includes methods relying on analytical or qualitative models of the supervised process. A demonstration of each approach can be performed on a simulation of either a three-tank system, a cocurrent or a countercurrent heat exchanger. The results are displayed in a common format, which allows performance comparison. A user manual including guidelines for tuning method specific parameters is available.

1 Introduction

In the manufacturing and the process industries, there is a growing awareness of the benefit brought by on-line monitoring of the state of the installation with fault detection and isolation systems. The latter allow one to detect incipient faults, and hence to remedy them in due time, so that breakdown or reduced product quality are avoided.

Fault detection and isolation (FDI) systems differ from classical alarm systems by the fact that they give early warning of faults. Alarm systems essentially process measured signals separately by comparing them to thresholds or by computing their trend. FDI systems take into account the correlation existing between those signals by using a mathematical model of the supervised process.

Thanks to such systems, a preventive maintenance policy can be replaced by a predictive (or on-condition) maintenance policy. The first approach consists of regular inspections and maintenance of components during plant shutdown, according to a periodicity advised by the manufacturers. On the other hand, in a predictive maintenance approach, the maintenance actions are planned on the basis of information deduced from on-line monitoring of the physical state of each component and from prediction of the state evolution. Unnecessary replacements and maintenance operations are thus avoided.

The aim of the toolbox presented in this paper is to provide a user friendly software to design, test and compare FDI systems based on various methods. The emphasis has been to develop MATLAB functions of which the output explicitly indicates whether a fault is present or not. This means that, in the case of observer based fault detection for instance, the residual generator is combined with suitable statistical tests for decision purpose. Indeed, a residual signal has zero mean in the absence of fault, and its mean becomes significantly different from zero upon occurrence of specific faults. It is thus necessary to process it with an adequate detector in order to decide on the possible presence of a fault. It should be noticed that not all the implemented methods can tackle fault isolation, i.e. the location of the faulty component.

The paper is organized as follows. Section 2 gives an overview of the methods implemented in the toolbox. The user interface is described in section 3. Finally the application of some of the methods to the three-tank benchmark is reported in section 4.

2 Fault detection methods

2.1 Methods based on analytical models

These approaches can be separated into two classes:

- parameter identification based methods
- methods based on fixed models (including observer based methods)

For the sake of conciseness, the name indicated in the software menu will be used to designate each approach.

Two methods were considered in the first class : RLS and Hinkley. In the first one, parallel recursive-least-squares estimation of a set of N linear regression models is performed. Each model is estimated with different length of data history. These lengths are defined by a set of exponential weighting factors linearly distributed from φ_{min} to φ_{max}. A time-recursive adaptive computation of the probability $p(h), h = 1, \cdots, N$, that the model h is actually correct is performed using a Bayesian approach. The system is classified to be in faulty/transient state in the time instances where the mean of the probability mass function $p(h)$ is associated to a forgetting factor below φ_{thres}. The constant φ_{thres} is specified by the user. The second method uses a recursive-least-squares algorithm with forgetting to track a slowly time-varying system. Sudden jumps in the identified parameters are detected with a Hinkley test. Upon occurrence of this phenomenon, the forgetting factor is decreased to allow the estimated parameters to reach their new value faster. This method was developed specifically for monitoring the heat transfer coefficient of heat exchangers. Indeed, in an old heat exchanger, big chunks of settled material can break away from the surface, causing the heat transfer coefficient to rise sharply; hence the need for Hinkley test.

The approaches in the second class can be further separated into multi-model and single-model methods. The former typically require a model of the plant in healthy working mode, and a model for every faulty situation, while the latter are based on a single-model in which the faults are represented either by unknown inputs or unknown parameters. Among the multi-model methods, three approaches were considered. The first relies on innovation evaluation by either cumulative sum, χ^2 or sequential probability ratio tests (CUSUM, chi-square and SPRT). The second is based on Bayesian decision making (RMMA). Finally the third is using bootstrap filters (BOOT2 and BOOT3). The single-model approaches can be separated in two categories: a method based on a bootstrap filter (BOOT1), and an innovation generator in the presence of unknown inputs combined with the generalized likelihood ratio (GLR) test (GLR-linear and GLR-bilinear). Each of these methods corresponds to a MATLAB function which is now briefly presented.

CUSUM, χ^2 and SPRT [1] assume that the system is described by linear stochastic models in its healthy working mode, and in faulty conditions. A Kalman filter is designed for each model and the resulting innovations are processed. More precisely, log-likelihood ratios between faulty and healthy state are computed from the innovations. The cumulative sum of log-likelihood ratios is used for decision making. When this cumulative sum is growing above a given threshold, an alarm is triggered to indicate a faulty state. The difference between CUSUM, χ^2 and SPRT lies in the way this cumulative sum is tested.

In RMMA, a set of linear stochastic models is used, one for the healthy mode, and one per each faulty situation. Such a model set can also be seen as a linear time-varying model, the parameter variations corresponding to changes in working mode. At each new acquisition of measurements, the maximum a posteriori estimate of the plant working mode is computed recursively using the innovation of a Kalman filter designed on the basis of the linear time-varying model. Such a computation is based on a Bayesian approach. The problem formulation results in an exponentially growing tree of possibilities, as the process can be in any working mode at each time instant. The less probable branches of the tree are cut in order to keep a tractable algorithm [2].

The two multi-model bootstrap methods, BOOT2 and BOOT3 are similar respectively to RMMA and CUSUM. The main difference is that, unlike RMMA and CUSUM, these methods can work with nonlinear stochastic state space models. Instead of Kalman filters, Bayesian bootstrap filters are used [3], [4]. The state and output equations are given in the form of a sampler and the following two probability density functions: $p_f(x(t+1)|x(t), u(t)), p_g(y(t)|x(t), u(t))$, where $x(t), u(t), y(t)$ denote respectively the plant state, known input and measured output.

The single-model bootstrap method is also based on a Bayesian bootstrap filter. It assumes knowledge of the healthy model only. It can detect any fault that is statistically visible, but cannot distinguish between different faults. The decision is based on recognition of highly improbable outputs.

The GLR-linear and GLR-bilinear methods are based on a stochastic state space model of the supervised process in which the faults appear as step-like inputs of unknown magnitude. The model might also contain unknown inputs, possibly to represent modelling uncertainties. As is obvious from their name, GLR-linear is based on a linear model, while GLR-bilinear uses a bilinear model. Both approaches contain a residual or innovation generator for a system subject to unknown inputs [5]. The resulting innovation vector is processed by a generalized likelihood ratio (GLR) test which provides, upon occurrence of a fault, an estimate of the fault appearance time and of its magnitude [6], [1].

2.2 Methods based on qualitative models

Two methods were developed and implemented in this framework.

The first method supports the purely rule-based approach. It serves to process the diagnostic rules expressed in the form IF *symptom*♯1 AND *symptom*♯2 AND ... AND *symptom*♯*n* THEN $fault\sharp j$. Symptoms are assumed to take qualitative values from the set $\{-1, 0, 1, u\}$ meaning low, none, high and unimportant. Unimportant symptoms are usually addressed to variables insensitive to particular faults. The misleading diagnostic results which might be caused by such variables are thus avoided. Qualitative values of the symptoms are determined by comparing the quantitative values of the corresponding terms with some thresholds. The reasoning mechanism which maps symptoms to faults makes use of the Transferable Belief Model (TBM). This is an approach which derives from the Dempster-Shafer mathematical theory of evidence [7]. Hence, belief masses are associated to each fault of the ranked list, and a measure called strength of conflict indicates the possible occurrence of unmodelled or not foreseen faults [8]. The first method can be applied as a fault isolation module in a number of FDI schemes. The rule base can be derived in many ways, e.g. heuristically or from signed directed graphs (SDG's, [9]).

The second method originated from an attempt to improve the diagnostic resolution of the Multilevel Flow Modelling (MFM) approach described in [10], hence its name "Extmfm". MFM was retained as a modelling framework, but the diagnostic resolution of the method was increased by taking account of the qualitative consistency relationships among the measured process variables expressed in terms of confluences [11]. Confluences are qualitative algebraic equations in which each term takes values from the set $\{-1, 0, 1\}$, the elements meaning respectively negative, zero and positive deviation from the nominal value. "Extmfm" provides a ranked list of fault candidates on the basis of a set of measurement samples and of a qualitative plant model. This is achieved in two steps. In the first step the consistency of each confluence is verified. Then, in the second step, the list of consistent and inconsistent confluences is matched to possible fault sources by making use of the incidence matrix and TBM reasoning. The final result is a list of suspected faults with associated belief masses.

3 Software demonstration

Each of the above mentioned methods has been implemented as a MATLAB function. Typically the arguments of such functions consist of one or several plant models, measured data and a set of design parameters. In order to help the user getting acquainted with the different FDI algorithms, a software demonstration was developed.

All methods have been configured for fault detection and isolation on a three-tank system. This system is depicted in figure 1. It is a pilot plant for which a complete nonlinear model has been determined and validated with experimental data [12]. This model is used to perform a simulation of the process using the MATLAB/SIMULINK environment. Both healthy and faulty working modes can be simulated. Different fault magnitudes and occurrence times can be considered. Three faulty conditions are selected for the demonstration: a leak in the tank R1, a partial clogging in the pipe with pump P2 and a bias on the sensor for measurement of the level in tank R3. Besides, the levels in tanks R1 and R3 are kept at their set point value by PI controllers. Some of the methods have also been configured for leak and fouling detection in countercurrent or cocurrent heat exchangers. These devices have been simulated by standard lumped approximations of the energy balances using the method of lines [13],[14].

Upon starting of the toolbox demonstration, the user first has to choose data from a pull-down menu in order to test one of the FDI methods. Such data can be obtained by running a simulation of the three-tank system, the countercurrent or cocurrent heat exchanger. An alternative to the simulation is to use experimental data recorded on the actual three-tank pilot plant in healthy and faulty working modes. Once the data are loaded, they can be processed by one of the FDI methods available in a second pull-down menu. The results of this operation are provided in a format common to each method (see figure 2). The evaluation criteria that are used have been defined in such a way that they can be easily computed. Hence the terms missed alarm, false alarm and detection delay do not correspond to their classical definitions. They reflect point-wise comparison of the estimated and the true trajectory of the faults. Both trajectories are plotted in the bottom left part of the window. A zero value of the fault signal corresponds to the healthy working mode, while the values 1, 2 and 3 indicate different faulty modes (namely the three above mentioned faults in the case of the three-tank system). The three-dimensional plot on the right gives the "weight" associated to each fault as a function of time. Depending on the algorithm, this weight can be the probability or a belief associated to each fault, or simply a boolean indicating whether the algorithm has detected that particular fault. Besides this standard window, each MATLAB demonstration function also generates other plots which are used to illustrate the

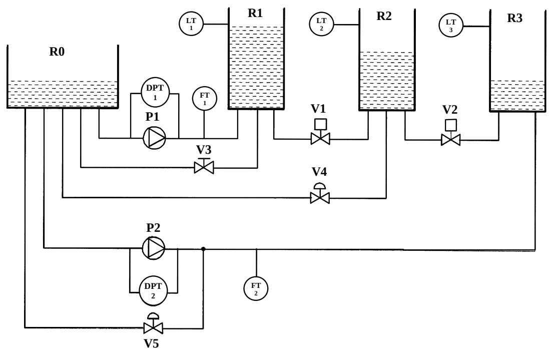

Figure 1: Three-tank system flowsheet

principle behind each method, and to help tuning the design parameters. A few illustrative examples are given in the next section.

4 Method specific displays

Details of the results of the application of two methods, namely RLS and GLR-linear, are given here.

The RLS method offers the advantage of simplicity of configuration and use, when a single fault has to be detected. It is primarily an adaptive detector. For configuration one has to determine from the available measured signals the structure of a regression model of which the transfer function is significantly modified upon occurrence of the fault to be detected. Should several faults have to be detected and isolated, it is further required that the model associated to each fault be insensitive to the other faults. The latter requirement might be difficult to meet and demands insight on the actual system behaviour upon occurrence of faults. As for all system identification based methods, persistency of excitation of the model inputs is crucial. Yet the approach can detect both abrupt and slow changes in the behaviour of the supervised process. Abrupt changes will make the mean of $p(h)$ correspond to forgetting factors very close to φ_{min}, while slower changes are associated to models obtained with longer data length. A suitable adjustment of φ_{thres} allows one to distinguish between normal process changes and faulty working modes. Figure 3 is obtained by processing simulation data from the three-tank system. These data, namely the level in each tank and the flow into the first tank, are plotted in the upper left graph. A bias of water level measurement, a leak in tank R1, and a clog in the pipe with pump P2 are simulated respectively in the time intervals [300, 600], [900,1200] and [1500,1800]. The controller is seen to compensate for the leak fault, as is clear from the step in the flow (the most noisy signal in the

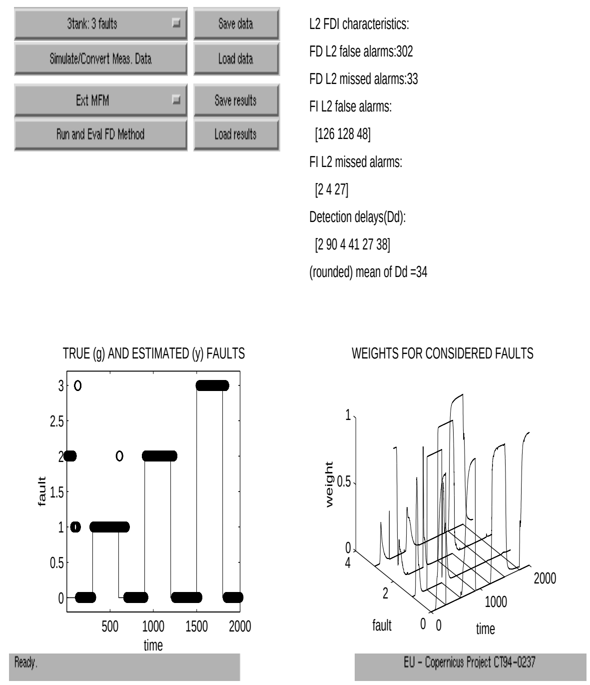

Figure 2: Standard display of results

upper left plot). The other three plots show the evolution of the mean of $p(h)$ for the regression model associated to each fault. The mean of the probability mass function is shifted towards low model numbers (corresponding to low forgetting factors) each time a fault occurs. If its value falls below the indicated threshold, a fault is signaled.

The GLR-linear approach requires the user to determine a linear state space model of the supervised process, which is obviously more demanding than providing the model structure needed for RLS. Such a model can be obtained by linearizing a nonlinear model of the plant around a given set point. It could also be deduced from experimental data using identification methods, provided the data are sufficiently rich. Faulty behaviours should be represented in the model by additive terms. GLR-linear will only work properly in the working range where the linear model is valid. The method consists of two steps. First a residual signal sensitive to one fault and insensitive to the others is generated. Next this residual is processed by a GLR test. A fault is declared when this test crosses its threshold and the estimated fault magnitude is larger than a given tolerance. For the three-tank application the operation is repeated three times, once for each fault. Figure 4 illustrates the approach for the residual generator and the GLR test aimed at detecting and isolating the sensor level bias. The data are the same as in figure 3. The upper left plot in figure 4 shows the residual signal. It is seen to be only sensitive to the sensor fault and not to the leak and clog occurring respectively in the time intervals [900,1200]

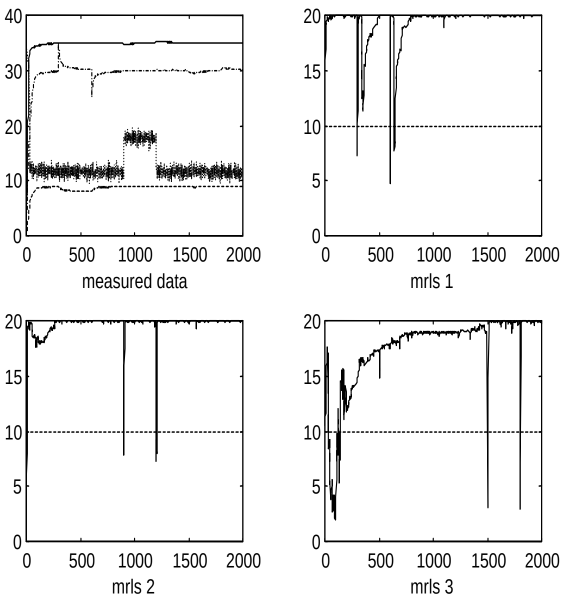

Figure 3: Specific plots displayed for RLS method

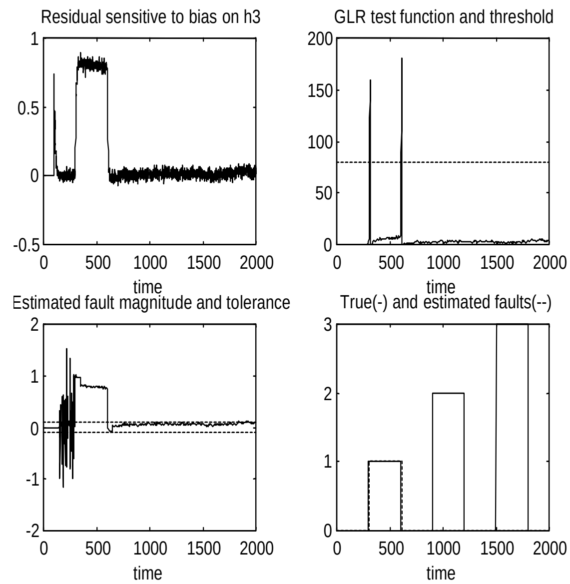

Figure 4: Specific plots displayed for GLR-linear method

and [1500,1800]. The FDI system only starts processing the data after the first 100 time instants. Indeed, a large transient is taking place during this initial time period, and hence the linear model cannot describe the system behaviour accurately in this time interval. That is why the GLR-linear method cannot be properly initialized if it processes the first 100 data samples. The zero value of each plotted function in figure 4 during the first 100 time instants is thus artificial. The upper right plot depicts the GLR test function and the decision threshold. The lower left plot indicates the estimated fault magnitude and the tolerance. Finally the lower right plot presents the true and estimated fault signals. In this case the estimated fault signal indicates that fault 1 (sensor bias) has occurred in the appropriate time period.

Upon occurrence of a fault, the residual is updated by taking into account the estimated fault magnitude, and the GLR test is reinitialized. This mechanism is both a strength and a weakness of the method. Indeed, contrarily to RLS, one does not only detect a change in the process, but one can quantify this change and determine possible fault disappearance. However, the quality of the fault magnitude estimate depends on the validity of the linear model in the faulty mode, and on the signal to noise ratio of the residual. Should this quality be poor, the residual will not be close to zero mean after its update, and the GLR test will immediately be triggered again. This is the reason why GLR-bilinear has been developed [5], [15]. Working with a bilinear model allows one to widen the range of validity of the model, and hence the domain in which the FDI system is working properly [15]. However the modelling task is more involved in the case of a bilinear model.

5 Conclusion

A software toolbox for fault detection and isolation has been developed in the MATLAB/SIMULINK environment. It contains classical methods as well as new original approaches to the problem. A user manual gives guidelines on how to tune the design parameters associated to each method. It also contains a qualitative comparison of the different approaches in the form of tables, which can be used as a first screening tool for the determination of appropriate methods for a given application. Preliminary results on a quantitative study resulting from the application of all algorithms to the three-tank system are also provided in this manual, but they are quite difficult to analyze given that the information needed for configuration of each method is different. For instance, CUSUM, χ^2 and SPRT are configured assuming exact knowledge of the fault magnitude, which is not realistic. Further work is thus needed to analyze the robustness of each approach in the presence of modelling uncertainties. This is a fundamental issue in characterizing the effectiveness of FDI systems. We believe that the FDI toolbox offers a good basis to start such a study.

Acknowledgement The reported work was performed in the framework of Copernicus project CT94-0237. The support of the European Commission for this project is gratefully acknowledged. The bilateral cooperation of

UTIA and IJS was also supported by the Ministry of Education of the Czech Republic and the Ministry of Science and Technology of the Republic of Slovenia.

References

[1] M. Basseville and I. Nikiforov, *Detection of Abrupt Changes - Theory and Applications.* prentice hall, n.j. ed., 1993.

[2] J. Štecha and V. Havlena, “Parameter tracking with alternative noise models,” in *IEEE Workshop CMP'94*, (Prague, Czech Republic), pp. 171–175, 1994.

[3] J. Liu and R. Chen, “Sequential monte carlo methods for dynamic systems,” *Journal of the American Statistical Association*, vol. 93, 1998.

[4] A. F. M. Smith and A. E. Gelfand, “Bayesian statistics without tears,” *The American Statistican*, vol. 46, no. 2, pp. 750–757, 1992.

[5] M. Kinnaert, “Innovation generation for bilinear systems : Application to robust fault detection,” in *Proceedings of the 1998 American Control Conference*, vol. 3, pp. 1595–1599, 1998.

[6] A. Willsky and H. Jones, “A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems,” *IEEE Trans. Automatic Control*, vol. AC-21, pp. pp 108–112, 1976.

[7] A. Rakar, P. Balle, D. Juricic, and D. Fussel, “Residual evaluation in fault diagnosis by means of transferable belief model,” in *Preprints of the Eighth International Workshop on Principles of Diagnosis*, pp. 95–102, 1997. Le Mont-Saint-Michel.

[8] P. A. Y. Peng and R. Hanus, “Quantitative model-based fault diagnosis using belief functions,” in *Prepr. CESA'96 IMACS Multiconference*, vol. 1, (Lille, France), pp. 528–532, 1996.

[9] M. Kramer and B. Palowitch, “A rule-based approach to fault diagnosis using the signed directed graph,” *AIChE Journal*, vol. 33, no. 7, pp. 1067–1078, 1987.

[10] M. Lind, “Modeling goals and functions of complex industrial plants,” *Applied Artificial Intelligence*, vol. 8, pp. 259–283, 1994.

[11] J. de Kleer and J. Brown, “Readings in qualitative reasoning about physical systems,” ch. A qualitative physics based on confluences, Morgan Kaufmann, 1990.

[12] G. Dolanc, D. Juricic, A. Rakar, J. Petrovcic, and D. Vrancic, “Three-tank benchmark test.” Report COPL007R, Copernicus project CT94-0237, 1997.

[13] G. Szederkényi, E. Weyer, and K. Hangos, “Grey box fault detection of heat exchangers,” in *Proceedings of the 3rd IFAC Workshop on On-Line Fault Detection and Supervision in the Chemical Process Industries*, vol. 2, 1998. Institut Français du Pétrole(Eds : P.S. Dhurjati, S. Cauvin).

[14] G. Szederkényi, E. Weyer, and K. Hangos, “Grey box fault detection of heat exchangers,” *Control Engineering Practice*, 1999. to appear.

[15] L. E. Bahir and M. Kinnaert, “Fault detection and isolation for a three tank system based on a bilinear model of the supervised process,” in *Proceedings of the UKACC International Conference on CONTROL 98*, pp. 1486–1491, 1998. Volume 2, Swansea.