Fuzzy Inductive Reasoning Model-Based Fault Detection Applied to a Commercial Aircraft

Josep M. Mirats Tur and **Rafael M. Huber Garrido**
Institut de Robòtica i Informàtica Industrial
C. Gran Capità, 2-4, 2ª pl. Ed. Nexus
08034 Barcelona, Spain
E-mail: Jmirats@iri.upc.es; Rhuber@iri.upc.es

This paper concentrates on progress made in fault detection using the Fuzzy Inductive Reasoning *(FIR) methodology. The so-called envelope approach is presented and compared with previous research efforts published by the same group in SIMULATION in 1989 and 1994. The comparison is achieved simulating faults in a B-747 commercial aircraft, the same example that had been used in the previous publications. The previously used crisp detection has been replaced by a new approach, based on the computation of an acceptability interval for each predicted variable. The paper demonstrates the improved capabilities of the new approach to detect faults earlier and in a more reliable manner.*

Keywords: Qualitative simulation and prediction, fault detection, inductive reasoning, decision-making, aircraft

1. Introduction

Modern control technology allows system analysis and control through a variety of techniques in time and frequency domains. Decision making in predefined situations has also been implemented using expert systems and rule-based control systems. For example, in [1], a fault monitoring and diagnosis expert system is described to assist pilots in handling in-flight faults. In [2], an expert system is described for control system design that would be a useful tool to handle sudden structural changes. These so-called intelligent fault-monitoring systems operate as qualitative systems, i.e., they reason by use of a symbolic knowledge representation. Fault monitoring systems must interact with the real physical world that operates on quantitative, i.e., continuous-time, sensory and actuator signals. To this end, sensory information obtained from the plant to be monitored needs to be discretised before it can be used by the reasoning system, and often, the discrete decisions reached by the intelligent system needs to be translated back to smooth real-valued control decisions.

When modelling a system, it is possible to use a mixed quantitative/qualitative knowledge representation [3] that may combine the advantages of both types of approaches and may help solve problems that neither a purely quantitative nor a strictly qualitative model may be able to solve on its own. Modern approaches to fault diagnosis include the so-called model-based approach [4]. This kind of fault detection approach makes explicit use of a quantitative mathematical model of the system. The implementation of on-board digital computers allows the development of fault detection, fault isolation, and fault identification exploiting the use of analytical rather than hardware redundancy.

One of the objectives of this research is to combine the quantitative simulation of a continuous process with a qualitative simulation technique to perform fault detection; fault isolation and fault identification are beyond the scope of this paper. In this article, a new methodological aspect in the mark of the *Fuzzy Inductive Reasoning* methodology is applied to improve fault detection in large-scale systems. An example already employed in previous publications [5–10], namely the numerical model of a B747 aircraft, is used to compare the obtained results.

The organisation of the paper is as follows. In order for the work to be self-contained, Section 2 summarises the *FIR* methodology except for the *qualitative simulation engine* that, for convenience, shall be explained in Section 5. Section 3 provides a short summary of previous research efforts [5] and [9], and then briefly explains the quantitative model of the aircraft as well as the qualitative models obtained with *FIR*. In Section 4, a modified experiment using the same aircraft model is proposed, and then, in Section 4.1, the technique used in [5] and [9] to perform fault detection is applied to the new data for the purpose of comparing its results with the newly presented approach. In Section 5, a new concept in the *FIR* methodology context is introduced, firstly presented in [10], and applied to the detection of faults in the modified example. Section 6 is left for the conclusions.

2. Fuzzy Inductive Reasoning Methodology

Inductive reasoning (*IR*) is an inductive modelling technique designed by Klir [11] as part of his *General System Problem Solving* (*GSPS*) framework. *IR* was first implemented by Uyttenhove [12] during his PhD dissertation research. This implementation was called *Systems Approach Problem Solver* (*SAPS*). An improved version of the original *SAPS* program was developed by Cellier and Yandell [13] and later extended by Li and Cellier [14] to offer fuzzy reasoning capabilities. Accordingly, the enhanced methodology is now called *Fuzzy Inductive Reasoning* [15].

In the sequel, a number of different authors used *FIR* to qualitatively model and simulate different kinds of systems and time series, while constantly improving the methodology [7, 16, 17, 18]. Concretely in [17], FIR is successfully applied to the problem of modelling and simulating univariate time-series, and in [18, 19], a research effort made at the University of Ghent, Belgium, the *FIR* methodology is complemented using tree classification procedures.

FIR offers four main modules and many auxiliary routines. Two of the main modules are computational engines, a *qualitative modelling engine*, and a *qualitative simulation engine*; the other two are data filters, a *fuzzification module*, and a *defuzzification module*. *FIR* operates on observations of input/output behaviour of multi-input, single-output (MISO) systems, and consequently, a multi-input, multi-output system has to be modelled by several parallel *FIR* models, one for each output.

The Fuzzification Module

It is the task of this module to convert quantitative data gathered from the system to its qualitative counterpart. In *FIR*, the process of fuzzification is called *recoding*. In this process, each quantitative data point is mapped into a qualitative triple, containing class, fuzzy membership, and side values. Data are usually recoded into an odd number of classes using equal frequency partitioning to determine the landmarks between neighbouring classes. The fuzzy membership function is a bell-shaped Gaussian curve, which assumes a maximum value of 1.0 at the centre between two landmarks, and a value of 0.5 at each of them. The side value describes whether the real data point lies to the left (side = –1), at the centre (side = 0), or to the right (side = 1) of the maximum of the membership function governing the chosen class. The complete operation of this module has been described in [20].

The Qualitative Modelling Engine

The goal of the *Qualitative Modelling Engine* (*QME*) is to determine the best behaviour system from a given data system and a predetermined output variable. The data system is initially provided to *FIR* in the form of a real-valued matrix. This raw data matrix may take the form:

$$\begin{array}{c} \textbf{\textit{Time}} \\ t-(n_{rec}-1)\,\delta t \\ \vdots \\ t-2\delta t \\ t-\delta t \\ t \end{array} \quad \begin{array}{c} \begin{array}{ccccc} u_1 & u_2 & u_3 & u_4 & y_1 \end{array} \\ \begin{pmatrix} \cdots & \cdots & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \end{pmatrix} \end{array}$$

where δt is the sampling rate. Each row represents one complete observation record, and each column represents one observed variable. In the above example, the data system contains n_{rec} data points for each of the four input variables, $u_1 \dots u_4$, and the single output variable y_1.

The raw data matrix is then converted, by means of the recoding process, into three matrices of equal dimensions: a qualitative class value matrix, a real-valued fuzzy membership matrix with elements in the range [0.5, 1.0], and a ternary side value matrix with elements in the set {−1, 0, +1}. With the recoded data, the *QME* tries to find relationships among the class values that are as deterministic as possible, trying to discover behavioural patterns among the observations using the information stored in the class value matrix. In the case of the above four-input, single-output systems, a possible qualitative relationship that allows the modelling of the output variable from past values of itself as well as current and past values of the inputs, could be:

$$y_1(t) = f\big(u_3(t-2\delta t),\ (u_1(t-\delta t),\\ (u_4(t-\delta t),\ y_1(t-\delta t),\ u_1(t)\big)$$

where f is a qualitative tabular function specified by means of the training data.

The QME returns such a qualitative relationship encoded in the form of another matrix called a *mask* in the context *of* FIR. Masks have the same number of columns as the data system to which they belong, and a certain number of rows, called the *depth* of the mask. Negative elements represent inputs of the qualitative relationship (so-called *m-inputs*), whereas the single positive element represents the mask output (the so-called *m-output*). The number of non-zero elements is the so-called *mask complexity*. The mask corresponding to the previously introduced qualitative relationship is shown below.

$$\begin{array}{c} t/x \\ t-2\delta t \\ t-\delta t \\ t \end{array} \begin{array}{c} \begin{array}{ccccc} u_1 & u_2 & u_3 & u_4 & y_1 \end{array} \\ \begin{pmatrix} 0 & 0 & -1 & 0 & 0 \\ -2 & 0 & 0 & -3 & -4 \\ -5 & 0 & 0 & 0 & +1 \end{pmatrix} \end{array}$$

In the QME, a search of the optimal (the most deterministic, based on Shannon entropy) qualitative relationship (qualitative model) is performed by either an exhaustive search or one of several heuristics applied to a set of mask candidates. The set of mask candidates is encoded in the form of a so-called *candidate mask* that contains −1 elements at the locations of potential *m-inputs*, and a +1 at the location of the *m-output*. If the modeller wants to forbid a relation, a 0 element should be introduced in the given position. A possible candidate mask for the five-variable system is shown below.

$$\begin{array}{c} \\ t-2\delta t \\ t-\delta t \\ t \end{array} \begin{array}{c} \begin{array}{ccccc} u_1 & u_2 & u_3 & u_4 & y_1 \end{array} \\ \begin{pmatrix} -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & 1 \end{pmatrix} \end{array}$$

Candidate mask

A detailed description of how the quality of each mask is computed is given in [20]. The optimal mask offers a good compromise between a *complexity* and an *uncertainty reduction measure*. It chooses those inputs, at given delays, that best model the observed output.

Once the optimal mask has been found, it can be used to flatten dynamic relations into static ones. The mask can be shifted over the matrices that represent the recoded data system; in each mask position, the selected *m-inputs* and *m-output* can be extracted from the data system, and can be written next to each other in a static record. At this point every row of the obtained matrix represent a fuzzy rule. Static records can then be sorted alphabetically. Figure 1 illustrates this process.

This operation is done with the three recoded matrices: the class, membership and side matrices, separately. Consequently, three behaviour matrices are obtained. Together, the three behaviour matrices constitute a set of fuzzy rules that FIR automatically synthesises from the training data and the optimal mask. These behaviour matrices as well as the optimal mask are used in the *qualitative simulation engine*.

The Qualitative Simulation Engine

For convenience, the description of this module is

	u_1	u_2	u_3	u_4	y_1
0	1	2	1	1	3
δt	2	2	1	3	3
2δt	1	1	$\boxed{3}$	3	2
3δt	$\boxed{1}$	1	2	$\boxed{2}$	$\boxed{1}$
4δt	$\boxed{2}$	2	3	3	$\boxed{1}$
5δt	2	1	2	3	1
6δt	1	2	1	3	3

Dynamic relations
(Data system)

i_1	i_2	i_3	i_4	i_5	o_1
1	2	3	3	1	2
1	1	3	2	1	1
3	1	2	1	2	1
2	2	3	1	2	1
3	2	3	1	1	3

Static relations

i_1	i_2	i_3	i_4	i_5	o_1
1	1	3	2	1	1
1	2	3	3	1	2
2	2	3	1	2	1
3	1	2	1	2	1
3	2	3	1	1	3

Sorted static relations
(Behaviour System)

Figure 1. Obtaining static relations by use of a mask

left for Section 5, so the new theoretic approach, the envelopes, can be fully understood.

The Defuzzification Module

It performs the reverse operation of the fuzzification module, converting qualitative triples back to real-valued data. The side value makes it possible to perform the defuzzification of the qualitative into quantitative values unambiguously and without information loss.

3. Previous Research and the Aircraft Model

In a previous research effort at the University of Arizona, qualitative simulation was applied to reason inductively about the behaviour of a quantitatively simulated B-747 aircraft model, to determine online when a malfunction occurs in the quantitative model. A **crisp inductive reasoner** (using a qualitative model computed only on the basis of class values, i.e., without membership and/or side information) was used to recognise that the aircraft had qualitatively changed its behaviour within a few seconds after a simulated malfunction had taken place. Crisp inductive reasoning and **crisp detection** were used to perform fault detection. The results of this study were reported in [5, 6].

Later on, continuing with the research in this area at the Polytechnic University of Catalonia, the former crisp inductive reasoner was replaced by a fuzzy inductive reasoner. This modified reasoning scheme had an enhanced discriminatory power and an improved forecasting capability. The new fuzzy inductive reasoner allows the prediction of a real-valued variable, whereas the crisp inductive reasoner was able to predict class values only. In this research effort, **fuzzy inductive reasoning** (*FIR*) and **crisp detection** (only using information given by the class values) were used to perform fault detection. The results of this study were reported in [7, 8, 9].

An improved method based on fuzzy inductive reasoning is presented in this paper with the aim of fault detection. **Fuzzy inductive reasoning** and what is now called **envelope detection**, to be explained in Section 5, are used here. As will be shown, using envelopes improves the fault detection approach, allowing the detection of faults at an earlier stage, and even the detection of faults that are not detected using either of the previous two approaches.

A numerical model of the B-747 aircraft has been used to generate episodes of the five variables shown in Figure 2. Two input variables, the variation in the elevator deflexion $\Delta\delta_{\mathrm{trim}}$ and the variation in the thrust ΔT_{trim}, and three output variables, the lift L, drag D, and the flight path angle GA, are considered. The mathematical model used is exactly the same that was reported in [5, 6]. This model, named B4, is valid for a B-747 at high altitude horizontal flight.

The mathematical model described in the given references reflects an essentially longitudinal flight restricted to longitudinal deviations from a trimmed reference flight condition. The main characteristic of this reference flight is the requirement that the resultant force and moment acting on the aircraft's centre of mass are zero.

The original aerodynamic parameters of this model were modified to artificially generate faults to test the fault detection methodology (the generated faults do not necessary correspond to real fault situations). Hence different models are found that represent structural changes of the original plane. The main characteristics of these models are [5, 6]:

- Model B4 is the original model representing a B-747 in cruise flight at 20.000 feet altitude. Its aerodynamic parameters are used as reference for the other models.
- Model B13 is characterised by a much more damped step response to the same step inputs. The effect of the angle of attack on the lift coefficient L has been slightly increased. A significant change in the effect of the angle of attack on the drag coefficient D occurred. The effect of the elevator deflection, $\Delta\delta_{trim}$, on the pitching moment has also changed.
- Model B5 represents a change of the original B4 model, which completely alters the influence that the angle of attack has on the aerodynamic response of the aircraft.

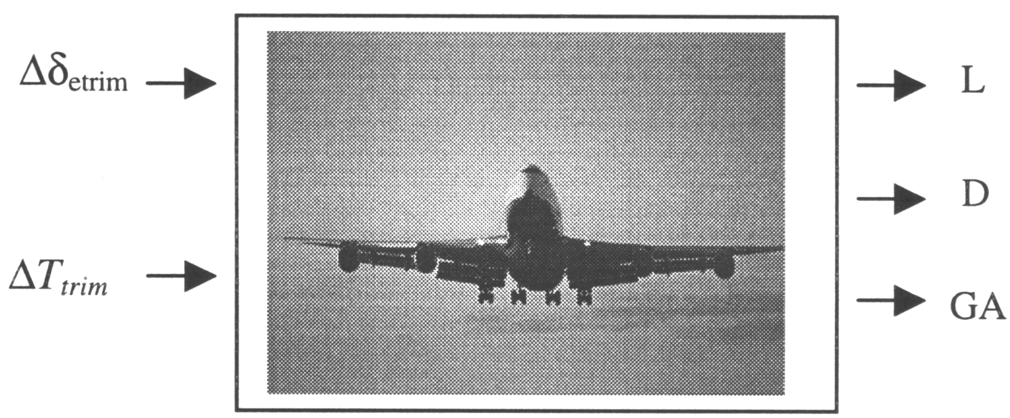

Figure 2. Input and output variables of the aircraft model

- Model 747 represents an enlarged B-747 in cruise flight. In this case, the values of the lift *L*, aerodynamic momentum, drag *D*, and the pitch angle are changed.

In this paper, only two among the aforementioned aircraft models are being used, namely the B4 model, corresponding to the aircraft under normal operation, and the B13 model, representing a fault. Other malfunction situations described in the referenced literature [5] can be tackled in the same way.

The simulation of the quantitative model is performed using *ACSL* (*Advanced Continuous Simulation Language*, [21]). Once the quantitative model has been implemented in *ACSL*, trajectories for the five named variables are generated. The qualitative modelling technique described in Section 2, FIR, is then applied, and a qualitative model of the aircraft is obtained. Details about the process of obtaining the qualitative aircraft model can be found in [10]. As FIR works with MISO systems only, it is necessary to find one model for every output. After performing the optimal mask search (using an exhaustive search), the following masks are found:

$$B4_L = \begin{array}{c} \begin{array}{ccccc} \delta_e & \delta_T & L & D & \gamma \end{array} \\ \begin{pmatrix} -1 & 0 & 0 & 0 & -2 \\ -3 & -4 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array} \quad B4_D = \begin{array}{c} \begin{array}{ccccc} \delta_e & \delta_T & L & D & \gamma \end{array} \\ \begin{pmatrix} -1 & 0 & 0 & 0 & 0 \\ -2 & -3 & 0 & -4 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix} \end{array}$$

$$B4_\gamma = \begin{array}{c} \begin{array}{ccccc} \delta_e & \delta_T & L & D & \gamma \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & -1 & -2 \\ -3 & 0 & 0 & -4 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \end{array}$$

These three masks, together with the behaviour matrices and the fuzzification landmarks, constitute the qualitative model of the B4 aircraft. This model was obtained using SAPS in its present Matlab toolbox version.

Now, with the qualitative model, it is possible to compare trajectories from variables of the real system (*ACSL* simulation) with forecast episodes from the qualitative model. When a structural change has taken place, the model can no longer predict the system. It is at this point that fault detection is achieved. Results obtained with the new methodology are to be compared with those obtained in the previous publications.

4. Smooth Change in the Aircraft Parameters

In order to generate trajectories with accidents, the flight starts with the B4 model (normal situation), and at a given time, a malfunction is numerically simulated by changing some of the structural parameters of the aircraft. In the previous research studies [6, 9], the change of parameters occurs suddenly, i.e., a step perturbation is performed so the change on the parameters of the aircraft is immediate. Such a sudden change is relatively easy to detect as it results in a violent transient behaviour of the aircraft, whereby the qualitative classes of the considered outputs change, making it possible to detect the fault by looking at the class values only. Precisely this technique, looking for discrepancies between the observed and the predicted class values of the output variable, has been employed to perform fault detection in the aircraft model in [6] and [9]. Although the former technique used a crisp inductive reasoner, whereas the latter used an improved fuzzy inductive reasoner, the fault was, in both cases, detected using information about the class values only.

The performed instantaneous parameter change implies a sharp transient in the variables taken into account. This transient was used to detect the aeroplane malfunction, so in some way, the methodology did

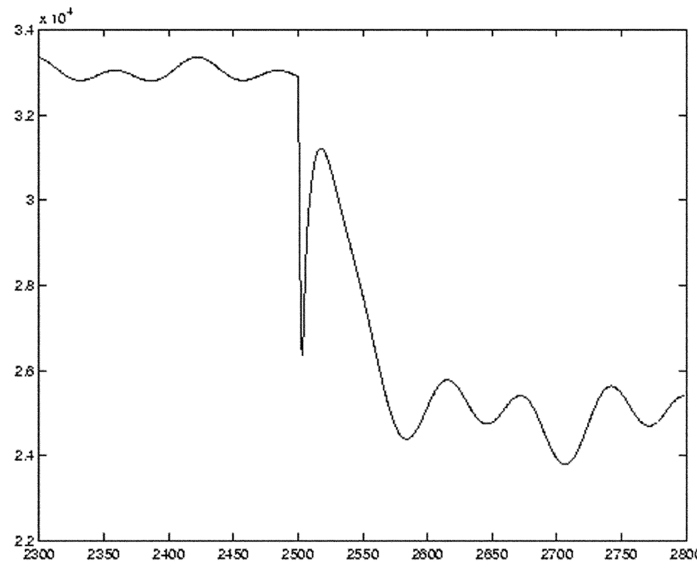

Figure 3. Drag (D): trajectory with a sudden change in the parameters

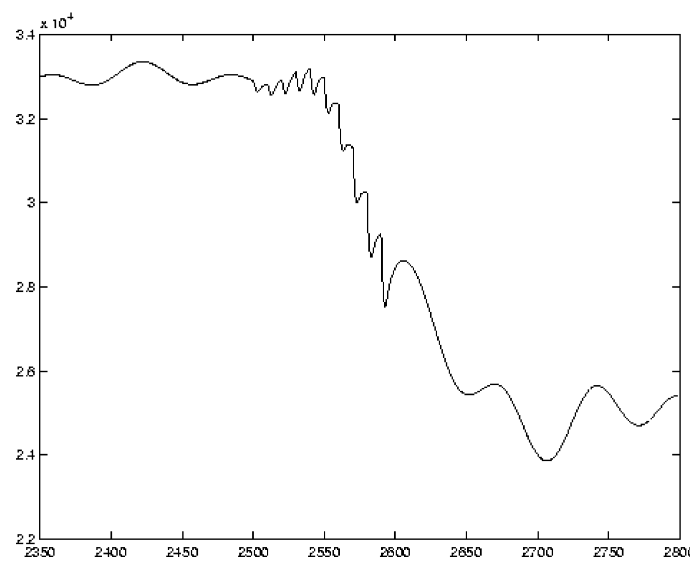

Figure 4. Drag (D): trajectory with a smooth change in the parameter

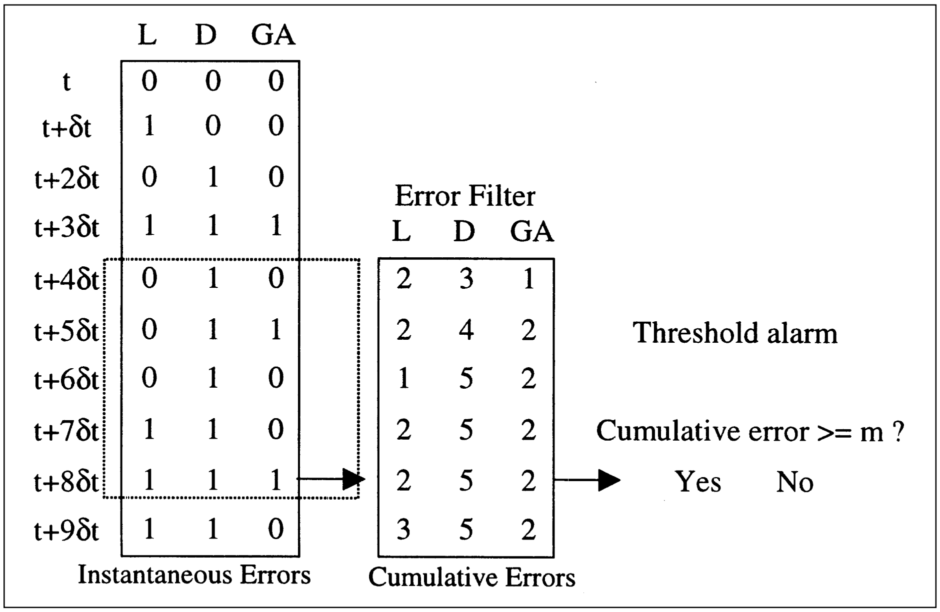

Figure 5. Fault detection scheme

not really detect the model change, but rather the highly dramatic transition phase between the two models. For instance, Figure 3 shows the transient in the output variable *D* resulting from an abrupt B4-B13 model transition.

Although some real malfunctions may indeed involve a sudden change in the parameters, quite often this is not the case, and it may therefore be more realistic to consider gradual rather than sudden changes. To model this situation, a smooth change in the aeroplane parameters has been simulated. The total range of parameter changes is the same as used previously, but now the parameters change gradually by ramping them from their initial to their final values over a period of 10 seconds. Figure 4 shows the change in the output variable *D* when the parameter values change smoothly.

4.1 Crisp Detection Using Smoothed Data

In order to compare the effectiveness of the new fault detection scheme, based on the so-called envelopes, with previous results, in this section the fault detection method presented in [6] and [9] is summarised and applied to the smooth parameter change explained in the previous section.

By using the numerical ACSL aircraft model, quantitative data representing the real system in a fault situation are gathered. The considered fault is the one named B13 in Section 3. The real-valued data obtained by means of a quantitative (ACSL) fault simulation is then converted to qualitative triplets of class, membership, and side values using the fuzzification module of FIR. Afterwards, the qualitative model previously obtained from the normal aircraft operation data (reported in Section 3) is used to predict the future behaviour of the aeroplane using the new situation fault data. Therefore, for each new data point coming from the real system (in the given experiment, the ACSL simulation), a prediction of the considered output variables is computed. The idea behind this is that when a structural change occurs, the qualitative model will receive inputs that have never been seen before. Hence it will no longer be able to correctly predict the behaviour of the system, thereby triggering an alarm vector indicating that a system fault has occurred.

At each sampling interval, an instantaneous error for each system output is computed by subtracting the real-system (fuzzified) class values and the predicted class values using the qualitative model. As long as the prediction is correct, the subtraction results in a value of zero. Hence values different from zero indicate a false prediction that may be interpreted as a potential indicator of a fault having occurred. These errors are stored in a matrix. Then, a moving average error filter is shifted down this matrix, computing, for each output, the sum of instantaneous errors it covers. These cumulative errors are in turn stored in another matrix. If any of these values, at a given point of time, passes the threshold (m) of the alarm module, an alarm is immediately triggered, i.e., a fault has been detected. Figure 5 may illustrate the process.

Table 1 summarises the results obtained when applying the described fault detection method to the new data set, i.e., a gradual change of parameters. The resulting alarm vectors using two different threshold values, $m = 5$ and $m = 2$, in conjunction with a filter depth of 5, are shown.

The change in the parameters starts at time instant $t = 2500$, and lasts 10 seconds. The alarm vectors in the table below show that within the next 18 seconds, the malfunction is not detected when using a threshold in the alarm module of $m = 5$, and some abnormal situation

Table 1. Alarm vectors using the detection approach proposed in [6, 9] with smooth change

Time	Alarm vector, m=5	Alarm vector, m=2
2500	0	0
2501	0	0
2502	0	0
2503	0	0
2504	0	0
2505	0	1
2506	0	1
2507	0	1
2508	0	1
2509	0	1
2510	0	1
2511	0	1
2512	0	1
2513	0	1
2514	0	1
2515	0	1
2516	0	0
2517	0	0
2518	0	0

is reported for time span [2505, 2515] when using $m = 2$. The latter would have been interpreted by the fault monitoring system as a false alarm, because from $t = 2516$ onward, the alarm vector is switched off again. Moreover, using low threshold values implies a higher probability of false alarms. It is interesting to detect faults as early as possible, but only real faults should be reported.

As presented in the next section, this problem can be tackled using the so-called "envelopes."

5. Detection with Envelopes

The concept of "envelopes," in the context of the FIR methodology, is introduced in this section and will be used to detect structural changes in an aircraft model. To understand how these envelopes can be obtained, the qualitative simulation engine of the FIR methodology needs to be explained here.

The Qualitative Simulation Engine. Given an FIR qualitative model, described by means of a mask and three behaviour matrices (constituting a set of fuzzy rules) for every output variable to be modelled, the goal of the *Qualitative Simulation Engine* (*QSE*) is to forecast a value for each of the chosen output variables.

The procedure is as follows. At each sampling instant, the mask is shifted one step forward along the class matrix of the fuzzified values of the (quantitative) variables. Classes of the *m-inputs* are read out from the mask, and a so-called input state vector is constructed. Then the class behaviour matrix is searched for coincidences on that input state vector, and the associated membership and side functions of each record found are compared with the ones of the input state vector. The five nearest neighbours are identified and used to compute the class, membership, and side values of the output. Figure 6 may give a clearer insight of this process using the five-variable example presented in Section 2. Each output is determined as an averaged value from the outputs associated with the five nearest neighbours, where the weights are determined based on the relative relevance (proximity or similarity) of each of the five nearest training data to the testing data record in the *m-input* space [20].

When using an FIR qualitative model to predict a system, a measure of the model error can be obtained comparing the forecast value with the real (fuzzified) value of the concerned variable. Using the B4 aircraft qualitative model presented in Section 3, new values of the outputs can be forecast and compared with the B4 validation data set (values of the variable trajectories not used in the FIR modelling process). If the mean square error is used, $mse = 0.1702$ is the obtained value for the *forecasting average error* of the FIR qualitative model. Although there is a separate forecasting average error for every output variable, in this study, the

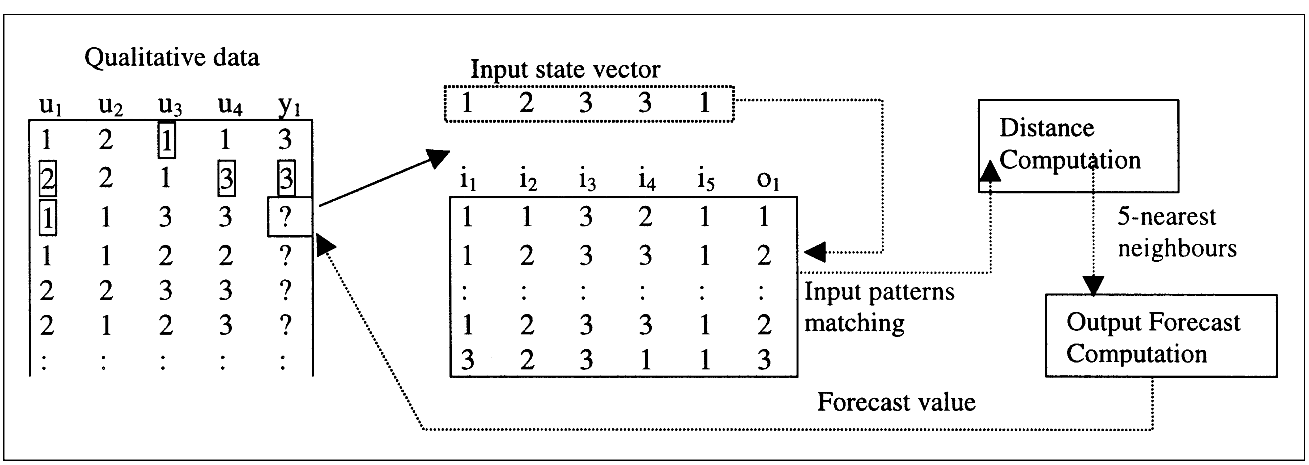

Figure 6. Fuzzy forecasting process

largest of these *mse* values has been used to characterise the envelopes of all three of the output variables.

Envelopes. The idea behind the envelopes approach is to compute, for each forecast value, an *interval of acceptability* of the real trajectory value. Up to now, a single (defuzzified) prediction was made at each point in time, which had been computed as an average of the output values of the five nearest neighbours in the training data base. Yet, it is perfectly defendable to make predictions in different ways. For example, it may make sense to consider the range of predictions made by the five nearest neighbours as an envelope of acceptable predictions. The closer the five nearest neighbours are to each other, i.e., the smaller the dispersion among them, the more narrow that envelope will be. On the other hand, the larger the dispersion among the five nearest neighbours, the wider the envelope will become. Let α and β be the minimum and maximum predictions made by any of the five neighbours, the envelope will then be defined by the range [α, β], a time-varying *interval of forecasting acceptability* associated with the predicted output variable. This information can be exploited for fault monitoring. The SAPS forecasting engine now returns three separate values at each time step: the predicted value of the output (a weighted average), the smallest acceptable prediction, α, and the largest acceptable prediction, β.

Notice that the width of the interval [α, β] provides information about how good the qualitative model is in terms of the dispersion among the five nearest neighbours (small values indicate that the five neighbours are close to each other, whereas a large interval denotes that the neighbours are sparse and that the prediction may possibly be inaccurate). It also provides an indication about whether the training data set size has been large enough.

Yet there is a second source of inaccuracy to be considered, namely the inaccuracy stemming from the reduced information contained in the selected mask. This inaccuracy can be estimated through the *mse* value mentioned earlier in this section. The *mse* value denotes the average inaccuracy of the weighted mean of the five nearest neighbours. To account for this second source of inaccuracy, the interval of forecasting acceptability is widened to the range $[\alpha(1 - mse * P/100), \beta(1 + mse * P/100)]$, where P is the predicted value of the output variable and *mse* is the forecasting average error of the qualitative model. This range is called *the interval of variable acceptability*.

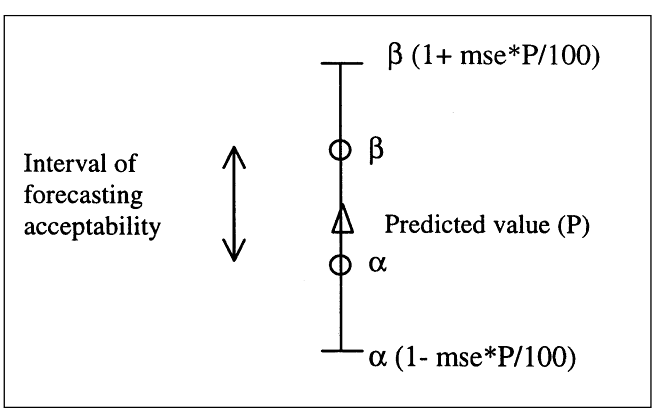

Figure 7. Interval of variable acceptability

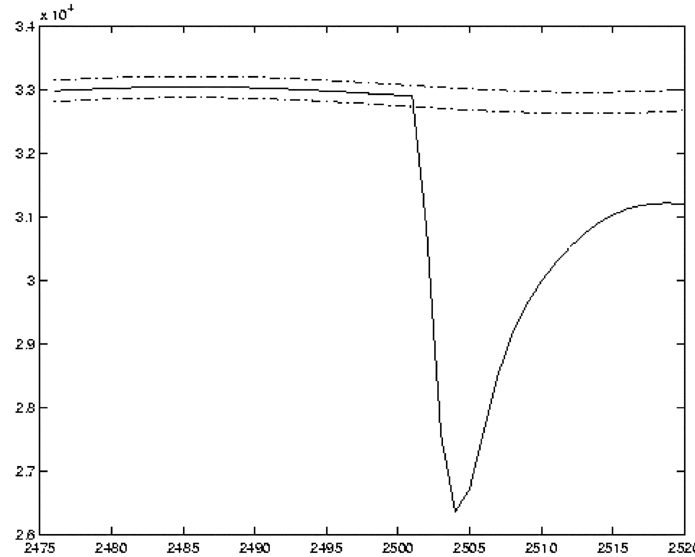

Figure 8. Variable D: real system values and forecast acceptability envelope

Denoting:

$$A = \alpha(1 - mse * P/100)$$

$$B = \beta(1 + mse * P/100),$$

The interval [A,B] of *variable acceptability* is the one within which the forecast and the real values of the output trajectory will be considered to match. Successive values of A and B along the time axis constitute the *variable acceptability envelope* (in short: envelope). Whenever the real value leaves the variable acceptability envelope, an instantaneous error of the concerned output variable is flagged.

Two experiments have been carried out using this new approach. First, the envelopes have been used to monitor a sudden change in the aircraft parameters, as proposed in [6] and [9], and subsequently, they were employed to monitor a smooth change in the aircraft parameters as explained in Section 4.

5.1 Sudden Change Detection

The method of the acceptability envelope of the variables has been applied to the case of an instantaneous change in the aeroplane parameters. Figure 8 shows the trajectory of the real output variable *D* together with the interval of acceptable forecast values (the so-called envelope) obtained by the qualitative FIR model of the B4 aircraft. The figure covers a much shorter time interval than that provided in Figures 3 and 4,

Table 2. Alarm vector obtained when performing fault detection with the envelopes approach in a sudden parameter change situation.

Time	Alarm
2500	0
2501	0
2502	0
2503	1
2504	1
2505	1
2506	1
2507	1
2508	1
2509	1
2510	1
2511	1
2512	1
2513	1
2514	1
2515	1
2516	1
2517	1
2518	1

because the envelope is narrow, indicating that the found qualitative model is of high quality. A wider time window would have made the figure less easily interpretable.

In order to perform fault detection using the envelopes, the method explained in Section 4 is used. An instantaneous error matrix is constructed, where every column is associated with an output variable, and every row corresponds to a sampling instant. A zero value denotes no error (i.e., the value of the real system variable lies within the interval of variable acceptability), and a value of one means an instantaneous prediction error has been registered (i.e., the quantitative real value is outside the range of variable acceptability). Then the error matrix is filtered using a moving average filter, and when the output passes the specified error threshold, an alarm is triggered. Table 2 summarises the results obtained with this method when applied to the situation of a sudden change in aircraft parameter values.

The fault alarm vector has been obtained using a threshold of $m = 5$ and an error window of depth 5. Notice that the accident is detected at time instant 2503, only three seconds after it took place, and the fault remains flagged ever after. The new method of the envelopes is hence reported to detect the malfunction at an earlier time than the approach proposed in [6] and [9]. Moreover, the obtained alarm vector is more stable in the sense that it does not return to a zero value after the transient has taken place, thereby

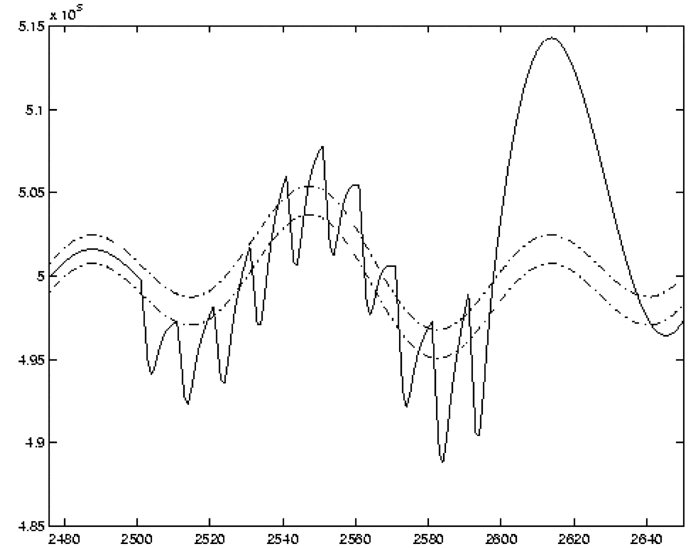

Figure 9. Envelopes with smooth change in variable L

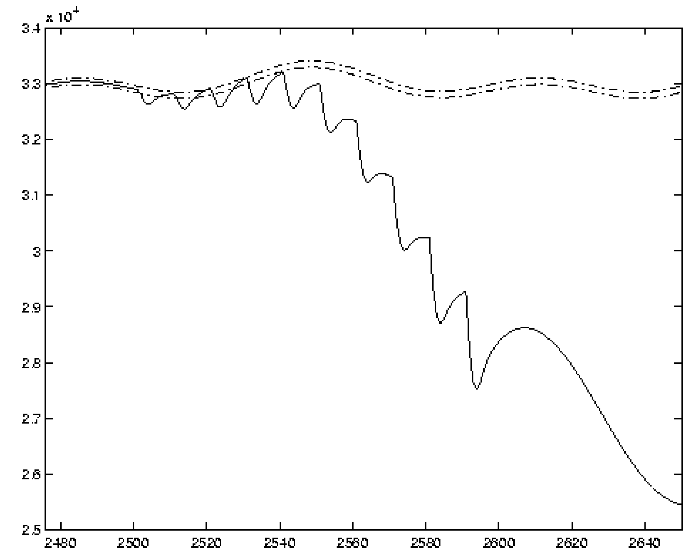

Figure 10. Envelopes with smooth change in variable D

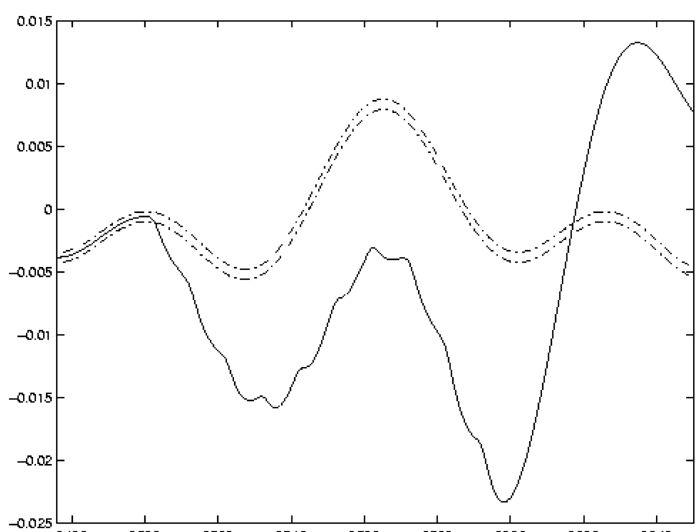

Figure 11. Envelopes with smooth change in variable GA

decreasing the probability of a true emergency being mistaken for a false alarm.

5.2 *Smooth Change Detection*

In this simulation, a smooth parameter change is applied as explained in Section 4, but now, the envelopes approach is applied to fault detection. Figures 9 to 11 show the trajectories of the three output variables together with the forecast envelopes obtained for the qualitative B4 model.

Table 3 summarises the results that have been obtained using the method of the envelopes when using two different alarm thresholds: $m = 5$ and $m = 2$ and an error window of depth 5.

In order to make it easy to compare the results with those obtained using fuzzy inductive reasoning together with crisp fault detection, as presented in [9], the two left-most columns of Table 3 show the results obtained with the method of envelopes, whereas the two right-most columns reproduce the results discussed in Section 4.1.

Using $m = 5$, the smooth fault is detected four seconds after initiating the change of the B4 parameters. If $m = 2$ is applied, an earlier detection is achieved (three seconds instead of four), and conversely to the results obtained in Section 4.1, the fault alarm remains flagged after time instant 2515.

Table 3. Comparing envelopes and crisp detection results for a smooth change situation

	Envelope Method		Crisp Detection Method	
Time	**m=5**	**m=2**	**m=5**	**m=2**
2500	0	0	0	0
2501	0	0	0	0
2502	0	0	0	0
2503	0	1	0	0
2504	1	1	0	0
2505	1	1	0	1
2506	1	1	0	1
2507	1	1	0	1
2508	1	1	0	1
2509	1	1	0	1
2510	1	1	0	1
2511	1	1	0	1
2512	1	1	0	1
2513	1	1	0	1
2514	1	1	0	1
2515	1	1	0	1
2516	0	1	0	0
2517	0	1	0	0
2518	0	1	0	0

6. Conclusions

When a malfunction occurs in a system, it is desirable to detect the malfunction as early as possible, but also in a reliable and robust manner, minimising the number of false alarms that are being interpreted as true emergencies, but also minimising the number of true emergencies that are being interpreted as false alarms. This paper concentrates on both aspects of fault monitoring, using a Boeing 747 aircraft model as a benchmark.

Two *Fuzzy Inductive Reasoning*-based approaches are compared. Both of them use FIR as a qualitative modelling technique, but the former uses a crisp fault detection approach, whereas the latter makes use of a newly proposed envelope detection method. Crisp fault detection had successfully been used in previous research efforts [6, 9] to detect sudden changes in system parameters, such as an engine falling off the aircraft, but is not well suited when confronted with smooth parameter changes that lead to a slow deterioration of the plant, such as ice building up on the wings of the aircraft.

The concepts of interval of forecast acceptability and interval of variable acceptability are introduced, and a fault monitoring method based on these intervals has been presented. In the case of a sudden change of the aircraft parameters, this new method detects the fault earlier. When the parameters of the aircraft are varied smoothly, the malfunction is detected using error threshold values that do not permit a fault detection using the crisp detection method. Moreover, the detection is more robust, since for a given threshold, the existing fault remains flagged, whereas using the crisp method the alarm disappears again, making it likely that the true emergency would be interpreted by the flight engineer as a false alarm.

7. Acknowledgements

The research reported in this paper was sponsored by the Spanish Government under Project TAP96-0882, entitled “Safe Operation of Complex Dynamical Systems.”

8. References

[1] Ali, M. and Scharnhorst, D.A. “Sensor-based Fault Diagnosis in a Flight Expert System.” *Proceedings of the IEEE Conference on Artificial Intelligence Applications*, Miami, FL, pp 49-54, 1985.

[2] Trankle, T.L., Sheu, P., Rabin, U.H. “Expert System Architecture for Control System Design.” *Proceedings of the 1986 American Control Conference*, Seattle, WA, June 18-20, Vol. 2, pp 1163-1169, 1986.

[3] Cellier, F.E., Nebot, A., Mugica, F. and de Albornoz, A. “Combined Qualitative/Quantitative Simulation Models of Continuous-Time Processes using Fuzzy Inductive Reasoning Techniques.” *Proceedings of SICICA'92, IFAC Symposium on Intelligent Components and Instruments for Control Applications*, Malaga, Spain, May 22-24, pp 589-593, 1992.

[4] Chen, J. and Patton, R.J. *Robust Model-based Fault Diagnosis for Dynamic Systems*, Kluwer Academic Publishers, 1998.

[5] Vesanterä P.J. “Qualitative Simulation: A Tool for Global Decision Making.” MS Thesis, Dept. of Electrical Engineering, Univ. of Arizona, Tucson, AZ, 1988.

[6] Vesanterä, P.J. and Cellier, F.E. “Building Intelligence into an Autopilot Using Qualitative Simulation to Support Global Decision Making.” *SIMULATION,* Vol. 52, No. 3, pp 111-121, 1989.

[7] de Albornoz, A. “Inductive Reasoning and Reconstruction Analysis: Two Complementary Tools for Qualitative Fault Monitoring of Large-Scale Systems.” PhD Dissertation, Llenguatges I Sistemes Informàtics, Universitat Politècnica de Catalunya, Barcelona, Spain, 1996.

[8] de Albornoz, A. and Cellier, F.E. “Qualitative Simulation Applied to Reason Inductively about the Behaviour of a Qualitatively Simulated Aircraft Model.” *Proceedings of QUARDET '93, IMACS Intl. Workshop on Qualitative Reasoning and Decision Technologies,* Barcelona, Spain, June 16-18, pp 711-721, 1993.

[9] de Albornoz, A. and Cellier, F.E. “Building Intelligence into an Autopilot Using Qualitative Simulation to Support Global Decision Making.” *SIMULATION*, Vol. 62, No. 6, pp 354-364, 1994.

[10] Mirats Tur, J.M. “Detecció i Reconeixement de Fallades en un Avió Comercial Usant Fuzzy Inductive Reasoning.” Internal Report IRI-Dt-9804, Institut de Robòtica i Informàtica Industrial, Universitat Politècnica de Catalunya, Barcelona, Spain, 1998.

[11] Klir, G.J. *Architecture of System Problem Solving,* Plenum Press, New York, 1985.

[12] Uyttenhove, H.J. “SAPS – System Approach Problem Solver.” PhD Dissertation, SUNY Binghamton, NY, 1979.

[13] Cellier, F.E. and Yandell, D.W. “SAPS II: A New Implementation of the Systems Approach Problem Solver.” *International Journal of General Systems,* Vol. 13, No. 4, pp 307-322, 1987.

[14] Li, D. and Cellier, F.E. “Fuzzy Measures in Inductive Reasoning.” *Proeedings of the Winter Simulation Conference,* New Orleans, LA, pp 527-538, 1990.

[15] Mugica, F. and Cellier, F.E. “A New Fuzzy Inferencing Method for Inductive Reasoning.” *Proceedings of the Sixth International Symposium on Artificial Intelligence, Intelligent Systems in Industry and Business,* Monterrey, Mexico, pp 372-379, 1993.

[16] Nebot, A. “Qualitative Modeling and Simulation of Biomedical Systems Using Fuzzy Inductive Reasoning.” PhD Dissertation, Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, Barcelona, Spain, 1994.

[17] López, J. “Time Series Prediction Using Inductive Reasoning Techniques.” PhD Dissertation, Organització i Control de Sistemes Industrials, Universitat Politècnica de Catalunya, Barcelona, Spain, 1999.

[18] Van Welden, D.F., Kerckhoffs, E.J.H. and Vansteenkiste, G.C. “Extending a Fuzzy Inductive Reasoner with Classification Procedures.” *Proeedings of the European Simulation Symposium,* Nottingham, UK, Oct 26-28, pp 111-116, 1998.

[19] Van Welden, D.F. “Induction of Predictive Models for Dynamical Systems Via Datamining.” PhD Dissertation, Toegepaste Wiskunde en Biometrie, Universiteit Ghent, Belgium, 1999.

[20] Cellier, F.E. *Continuous System Modeling,* Springer-Verlag, New York, 1991.

[21] Mitchell & Gauthier Associates, Inc. *ACSL: Advanced Continuous Simulation Language – Reference Manual,* Edition 10.0, Concord, MA, 1991.