Interval analysis based learning for fault model identification. Application to control surfaces oscillatory failures.

Renaud Pons*, Carine Jauberthie$^{\dagger *}$, Louise Travé-Massuyès*, Philippe Goupil‡

{rpons,cjaubert,louise}@laas.fr
*LAAS-CNRS
*Université de Toulouse
†Université Paul Sabatier
7, avenue du Colonel Roche
F-31077 Toulouse
France

philippe.goupil@airbus.com
‡Airbus France
Flight Control System department
316, route de Bayonne
31060 Toulouse CEDEX 09
France

Abstract

Interval models may be seen as a trade-off between numerical and qualitative models. They have been often referred as semi-qualitative models. The interval algebra is indeed a specific qualitative algebra with advantageous algebraic properties. This paper presents the application of an interval based parameter estimation method, which is used for learning fault models supporting the detection of Oscillatory Failure Cases (OFC) in Electrical Flight Control System (EFCS) of civil airplanes. The interval estimation method results are guaranteed and computations are performed in finite time. Failures are identified using the fault models which are checked against system input and output measurements.

Introduction

Model based reasoning relies on the soundness of the models supporting the reasoning. This is particularly true for model based fault detection and diagnosis. Nevertheless building models turns out to be an awkward task. At some stage of the process, one may face two kinds of uncertainties. On one side, *unstructured* uncertainties mean that deriving a complete equational model from the physical phenomena is impossible. On the other side, when the structure of the equations is known but some of the parameters are not, uncertainties are said to be *structured*. In addition to these uncertainties, it is not always possible to get informations about disturbances and noises acting on the system. In such cases, assuming bounded uncertainties may be a solution.

Considering structured uncertainties, an interesting way to go is then to use guaranteed estimation methods, which learn the state and/or parameters of the models from data. These methods rely on *interval analysis* that first appeared in (Moore 1966). They are now subject of a growing interest in various communities and are applied for many tasks (Alamo, Bravo, & Camacho 2005; Armengol *et al.* 2001; Guerra, Puig, & Ingimundarson 2006; Jaulin *et al.* 2001; Kieffer & Walter 1998; Kieffer, Jaulin, & Walter 2002; Lesecq, Barraud, & Dinh 2003; Ribot 2006; Ribot, Jauberthie, & Travé-Massuyès 2007).

This paper presents a fault detection method using interval parameter estimation. Parameters of the model are estimated from the input and output measurements of the system. The consistency of this estimation is then checked against parameters computed from a theoretical (possibly faulty) model of the system. Computations use the set inversion algorithm SIVIA (Jaulin & Walter 1993; Jaulin *et al.* 2001). The results are approximated but are bounded in a guaranteed way. The method is applied to detect Oscillatory Failure Cases (OFC) in Electrical Flight Control System (EFCS) of civil airplanes.

The article is organised as follows. Next section positions interval models with respect to qualitative models. Then second section provides an overview of interval analysis, its original purpose and its use for fault detection. The error bounded context is then presented more precisely with parametric estimation using intervals in the fourth section. In fifth section, the case study is presented: we describe what are OFC, and their consequences on the aircraft control surfaces, why such failures must be detected in time and one of the methods currently used on Airbus aircrafts for OFC detection. In sixth section the application and the obtained results are analyzed. Finally some conclusions are outlined in last section.

Qualitative versus interval models

Providing models representing physical systems is a common concern spread over all scientific and engineering communities. Modelling depends on the available knowledge about the physical system. This is why pure numerical models are sometimes disregarded to the benefit of qualitative models which naturally cope with uncertain and inaccurate knowledge. Within the qualitative framework, numerical values are replaced by qualitative values that can be seen as (absolute) *orders of magnitude*[1].

Absolute orders of magnitude are based on partitioning the real line $\mathbb{R}$ into a finite set of basic qualitative val-

[1]Relative orders of magnitude refer to different formalisms based on binary relations used to compare quantities (Dague 1993a; 1993b; Travé-Massuyès *et al.* 2005).

ues. Considering the order relation given by set inclusion, it allows one to build the whole set of qualitative values, organised along to a high semi-lattice (Travé-Massuyès & Piera 1989; Travé-Massuyès, Ironi, & Dague 2003). As an example, (De Kleer & Brown 1984; Forbus 1984; Kuipers 1984) introduced *sign algebra* for which a parameter or a variable x takes values in $\{-, 0, +, ?\}$ depending on whether it is negative, zero, positive, or undetermined. Unfortunately, many operations, *e.g.* $(+) - (+)$, lead to an undetermined result. Absolute order of magnitude algebras were proposed to hinder this problem (Travé-Massuyès, Ironi, & Dague 2003). The real line partitioning defines the *quantity space* of a variable thanks to *landmark values* (Kuipers 1994). It captures the intuition that there are only a few qualitative important values associated to different qualitative behaviors. Whatever partitioning is chosen, an algebra and arithmetical operations can be defined.

The interval algebra can be seen as an extreme case in which the partition elements are provided by every real number and intervals are closed and connected subsets of $\mathbb{R}$. Interval analysis may then be interpreted as a specific case of order of magnitude reasoning.

Interval analysis

Preamble

The key idea of interval analysis is to reason about intervals instead of real numbers and boxes instead of real vectors. The first motivation was to obtain guaranteed results from floating point algorithms and it was then extended to validated numerics (Moore 1959). Let us recall that in computers real numbers can only be represented by a floating point approximation, hence introducing a quantification error. A *guaranteed result* means first that the result set encloses the exact solution. The width of the set, *i.e.* the result precision, may be chosen depending on various criteria among which response time or computation costs. Secondly, it also means that the algorithm is able to conclude on the existence or not of a solution in limited time or number of iterations. The first significant work is due to Moore in its Phd thesis which was the early beginnings of his reference book (Moore 1966).

Main concepts

The matter is to wrap the sets of interest into boxes or union of boxes for which computations may be easier. There are three fundamental operations on intervals which are briefly explained after the definition of an interval.

Interval A real interval $[u] = [\underline{u}, \overline{u}]$ is a closed and connected subset of $\mathbb{R}$ where $\underline{u}$ represents the lower bound of $[u]$ and $\overline{u}$ represents the upper bound. The width of an interval $[u]$ is defined by $w(u) = \overline{u} - \underline{u}$, and its midpoint by $m(u) = (\overline{u} + \underline{u})/2$.

The set of all real intervals of $\mathbb{R}$ is denoted $\mathbb{IR}$.

Two intervals $[u]$ and $[v]$ are equal if and only if $\underline{u} = \underline{v}$ and $\overline{u} = \overline{v}$. Real arithmetic operations are extended to intervals (Moore 1966).

Arithmetic operations on two intervals $[u]$ and $[v]$ can be defined by:

$$\circ \in \{+, -, *, /\},\ [u] \circ [v] = \{x \circ y \mid x \in [u],\ y \in [v]\}.$$

An interval vector (or box) $[X]$ is a vector with interval components and may equivently be seen as a cartesian product of scalar intervals:

$$[X] = [x_1] \times [x_2] \times \ldots \times [x_n].$$

The set of $n-$dimensional real interval vectors is denoted by $\mathbb{IR}^n$.

An interval matrix is a matrix with interval components. The set of $n \times m$ real interval matrices is denoted by $\mathbb{IR}^{n \times m}$. The width $w(.)$ of an interval vector (or of an interval matrix) is the maximum of the widths of its interval components. The midpoint $m(.)$ of an interval vector (resp. an interval matrix) is a vector (resp. a matrix) composed of the midpoint of its interval components.

Classical operations for interval vectors (resp. interval matrices) are direct extensions of the same operations for punctual vectors (resp. punctual matrices) (Moore 1966).

Wrappers Consider a set $\mathbb{U}$ and a set $\mathbb{V}$ of subsets of $\mathbb{U}$. $\mathbb{V}$ is a *set of wrappers* for $\mathbb{U}$ if $\mathbb{U}$ and each singleton of $\mathbb{U}$ belong to $\mathbb{V}$ and $\mathbb{V}$ is closed by intersection.

The figure 1 shows $f([u])$ which is the direct image of a box $[u]$ in $\mathbb{IR}^2$ by a function f, a possible wrapper $[f]([u])$ and the optimal wrapper $[f]^*([u])$. $f([u])$ is called the *range* of f over $[u]$ and is given by:

$$f([u]) = \{f(x) \mid x \in [u]\}.$$

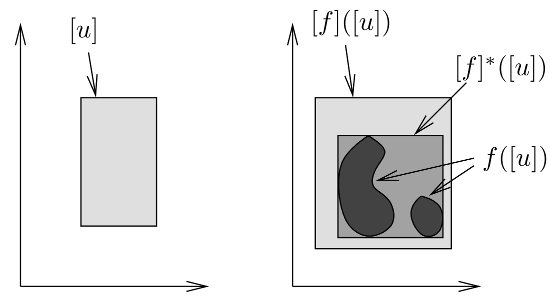

Figure 1: Range of f over $[u]$ and wrappers.

Inclusion function Given $[u]$ a box of $\mathbb{IR}^n$ and a function f from $\mathbb{IR}^n$ to $\mathbb{IR}^m$, the *inclusion function* of f aims at getting an interval containing the image of $[u]$ by f.

An inclusion function of f can be obtained by replacing each occurrence of a real variable by its corresponding interval and by replacing each standard function by its interval evaluation. Such a function is called the natural inclusion function. In practice the inclusion function is not unique, it depends on the syntax of f.

Inclusion test Given a subset $\mathbb{S}$ of $\mathbb{R}^n$, we test if $[x]$ belongs to $\mathbb{S}$, more precisely if $[x] \subset \mathbb{S}$ or $[x] \cap \mathbb{S} = \emptyset$. These tests are used to prove that all points in a given box satisfy a given property or to prove that none of them does.

Contractor The last operation is the *contraction* of $[x]$ with respect to $\mathbb{S}$. This means that we search a smaller box $[z]$ such that $[x] \cap \mathbb{S} = [z] \cap \mathbb{S}$. If $\mathbb{S}$ is the feasibility set of a problem and $[z]$ turns out empty, then the box $[x]$ may not contain the solution (Jaulin *et al.* 2001).

These operations are used to test if a box can or cannot be removed from the solution set. When no conclusion can be drawn, the box may be bisected and each of the sub-boxes can be tested in turn (this corresponds to *branch-and-bound* algorithms).

SIVIA: Set Inversion Via Interval Analysis

Consider the problem of determining a solution set for the unknown quantities u defined by

$$\begin{aligned} S &= \{u \in U \mid \Phi(u) \in [y]\}, \\ &= \Phi^{-1}([y]) \cap U, \end{aligned} \tag{1}$$

where $[y]$ is known a priori, U is an a priori search set for u and Φ a nonlinear function not necessarily invertible in the classical sense. (1) involves computing the reciprocal image of Φ. This can be solved using the algorithm *SIVIA*, which is a recursive algorithm that explores all the search space without loosing any solution. This algorithm makes it possible to derive a guaranteed enclosure of the solution set S as follows:

$$\underline{S} \subseteq S \subseteq \overline{S}. \tag{2}$$

The inner enclosure $\underline{S}$ is composed of the boxes that have been proved feasible. To prove that a box $[u]$ is feasible it is sufficient to prove that $\Phi([u]) \subseteq [y]$. Reversely, if it can be proved that $\Phi([u]) \cap [y] = \emptyset$, then the box $[u]$ is unfeasible. Otherwise, no conclusion can be reached and the box $[u]$ is said undetermined. The latter is then bisected in two sub-boxes that are tested until their size reaches a user-specified precision threshold $\varepsilon > 0$. Such a termination criterion ensures that *SIVIA* terminates after a finite number of iterations.

The algorithm is formally presented below. The functions $L(.)$ and $R(.)$ return respectively the "left" and "right" parts of their interval vector argument once it has been bisected. This bisection may be made using different strategies such as round robin, largest first or random.

Algorithm 1 SIVIA(in: Φ, $[y]$, $[u]$, ε, inout: $\underline{S}$, $\overline{S}$)

if $[\Phi]([u]) \cap [y] = \emptyset$ **then**
    return
end if
if $[\Phi]([u]) \subset [y]$ **then**
    $\underline{S} := \underline{S} \cup [u]$
    $\overline{S} := \overline{S} \cup [u]$
    return
end if
if $width([u]) < \varepsilon$ **then**
    $\overline{S} := \overline{S} \cup [u]$
end if
SIVIA(Φ, $[y]$, $L([u])$, ε, $\underline{S}$, $\overline{S}$)
SIVIA(Φ, $[y]$, $R([u])$, ε, $\underline{S}$, $\overline{S}$)

Fault detection using intervals

Set membership detection uses these concepts to perform state estimation and parameters estimation. In state estimation, a nonlinear dynamical model is approximated by a Taylor expansion (Rihm 1994; Berz & Makino 1998; Nedialkov, Jackson, & Pryce 2001) to compute a box enclosing all possible trajectories of the solution between two successive time steps t_j and t_{j+1}.

The fixed point and Picard-Lindelöf theorems prove the existence and uniqueness of the solution (Rihm 1994). The interval solution becomes obviously wider and wider at each iteration step: this drawback is known as the *wrapping effect*. Numerous methods may circumvent this pessimism: among them one is to use high order Taylor expansion, mean value forms, matrices preconditioning and a predictor-corrector approach (Corliss 1994; Nedialkov 1999; Neumaier 1990; Raïssi, Ramdani, & Candau 2004; Ramdani 1995; Rihm 1994).

Parameter estimation in a bounded error context

Parameters and state estimated from experimental measures are usually obtained within a stochastic framework in which known distribution laws are associated to interferences and noisy measurements. Oppositely, in the bounded error context measures and modeling errors are supposed to be unknown but to stay within known and acceptable bounds.

Errors between measured and predicted outputs may rely on many factors, among them: limited sensors accuracy, interferences, noise, structured uncertainties, ... Some are quantifiable, some are not. We consider here the quantifiable error e, which is added to the model output y. The experimental outputs y_{exp} are given by:

$$y_{\text{exp}}(t_j) = y(t_j) + e(t_j),\ 1 \leq j \leq n. \tag{3}$$

In our context, the error e is supposed to be within an interval whose lower bound is e_{min} and upper bound is e_{max}. An allowable error set $\mathbb{E}$ may be defined as a set of constraints

$$\mathbb{E} = \{e(t_j) \mid e_{\text{min}} \leq e(t_j) \leq e_{\text{max}}\}. \tag{4}$$

These bounds may be considered constant over time as well as variable. They may be established from data given by constructors for electronic parts for example.

Our system has unknown but bounded initial conditions while input and output values are available at any time. The initial conditions belongs to a set, hence the model output y is also a set denoted $[y]$, as well as the error e which is a set $[e]$ that must be in the domain $\mathbb{E}$.

In the same way than for $[e]$, we define an allowable domain $\mathbb{Y}$ for model output $[y]$ such than

$$\begin{aligned} \mathbb{Y} &= \{[y] \mid [y] \subset [y_{\text{exp}}]\}, \\ &= \{[y] \mid [y] \subset [y - e_{\text{max}}, y - e_{\text{min}}]\}. \end{aligned} \tag{5}$$

Interval analysis is used to reject models that are not consistent with data and error bounds.

Numerous approaches have been tested with linear models: ellipsoid shaped methods (Milanese & Vicino 1991;

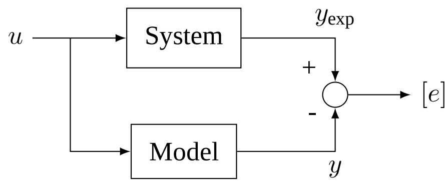

Figure 2: System and model.

Durieu & Walter 2001; Lesecq, Barraud, & Dinh 2003), parallelotopic and zonotopes (Alamo, Bravo, & Camacho 2005).

Consider a nonlinear parametric model described by the following set of equations

$$\begin{cases} \dot{x}(t,p) = f(x(t), u(t), p), \\ y(t,p) = g(x(t), u(t), p), \\ \quad x_0 \in \mathbb{X}_0, \\ \quad p \in \mathbb{P}_0, \end{cases} \tag{6}$$

where

- f and g are continuous nonlinear known functions,
- $x(t) \in \mathbb{R}^n$ is the state vector at time t,
- $u(t) \in \mathbb{R}^m$ is the input vector at time t,
- $y(t) \in \mathbb{R}^p$ is the output vector at time t,
- $\mathbb{X}_0$ is an a priori known set enclosing the initial condition x_0,
- $\mathbb{P}_0$ is the a priori known set enclosing the searched parameter vector p.

A parameter vector p is acceptable if and only if the error between y_{exp} and the model output $[y]$ is bounded in a known way. To estimate system parameters, we have to get the set $\mathbb{P}$ of all parameters p enclosed in the a priori search set $\mathbb{P}_0$ such that error between real data and model outputs denoted

$$[e(p)] = y_{\text{exp}} - [y(p)] \tag{7}$$

belongs to the allowable error set $\mathbb{E}$ whose bounds e_{min} and e_{max} are known:

$$\begin{aligned} \mathbb{P} &= \{p \in \mathbb{P}_0 \mid [e(p)] \in \mathbb{E}\}, \\ &= \{p \in \mathbb{P}_0 \mid e_{\text{min}} \leq [e(p)] \leq e_{\text{max}}\}. \end{aligned} \tag{8}$$

The characterization of the set $\mathbb{P}$ may be defined as a set inversion problem (Raïssi, Ramdani, & Candau 2003; Kieffer & Walter 2005):

$$\mathbb{P} = [e^{-1}](\mathbb{E}). \tag{9}$$

A guaranteed approximation of $\mathbb{P}$ may be computed using the SIVIA algorithm presented previously.

Case study

Problem

One of the tasks devoted to flight control computer is to slave the position of the control surfaces. The control surface motion is driven by an actuator in active or damped mode. There are generally two actuators for one control surface. A *master* computer performs control by sending a command on the active actuator. The other one is set in damped mode and follows the surface motion without opposition. When the master computer detects a failure, it switches the active actuator to damped mode and gives control to a *slave* computer that controls the second actuator which is now in active mode.

All parts in the control chain that contain electronic devices may generate interference signals. These signals make the control surface swing. This is called an *Oscillatory Failure Case (OFC)*. In this paper, only OFC located in the servo-loop control of the moving surfaces are considered, that is, between the *Flight Control Computer* and the control surface, including these two elements (*cf.* Figure 3). When an OFC occurs within the actuator bandwidth , it may have the following consequences:

- coupled with the aeroelastic behaviour of the aircraft, it may lead to unacceptably high loads or vibrations, the worst case corresponds to resonance phenomena with aircraft natural modes ;
- it speeds up actuators stress and reduces their lifetime ;
- it lowers passengers comfort.

The plane is designed to take into account these faults in a limited way, depending on oscillation frequency and range. Taking design actions to counteract these faults would indeed require heavily and costly structure reinforcement. It is then very much advisable to detect them using the flight control computers. Monitoring must be performed to ensure that failures stay within predefined limits. Classical monitoring (e.g. position monitoring, runaway monitoring, etc.) does not guaranty such detections, so specific mechanisms must be added.

When an OFC is detected, the flight computer looses regulation over elevators control. As seen previously, another waiting computer ensures surface control with a redundant servo which switches from damped to active mode.

The problem to solve is to detect in the control loop some OFC with a minimal given range within a given number of periods (the maximal overload does not immediately occur on the structure but after some periods of oscillation). The goal is to detect 1° failures within 3 periods, on a frequency range from 0.2 to 5 Hz. This goal has been chosen for this paper. In real cases, it depends on the aircraft type.

Liquid vs. solid failures

Two different kinds of OFC may occur: liquid or solid ones. As shown in the scheme of figure 4, a liquid failure is a interference signal added to the control loop signal. A solid failure is a signal which replaces the control loop one.

In both cases, a failure is a periodic sinus shaped signal whose frequency, range and phase obey to an uniform law. For both cases of failure, residuals corresponding to estimated position subtracted from real position are shown in Figure 5.

These residuals are used to detect the OFC. The current method used in A380 flight control computers relies

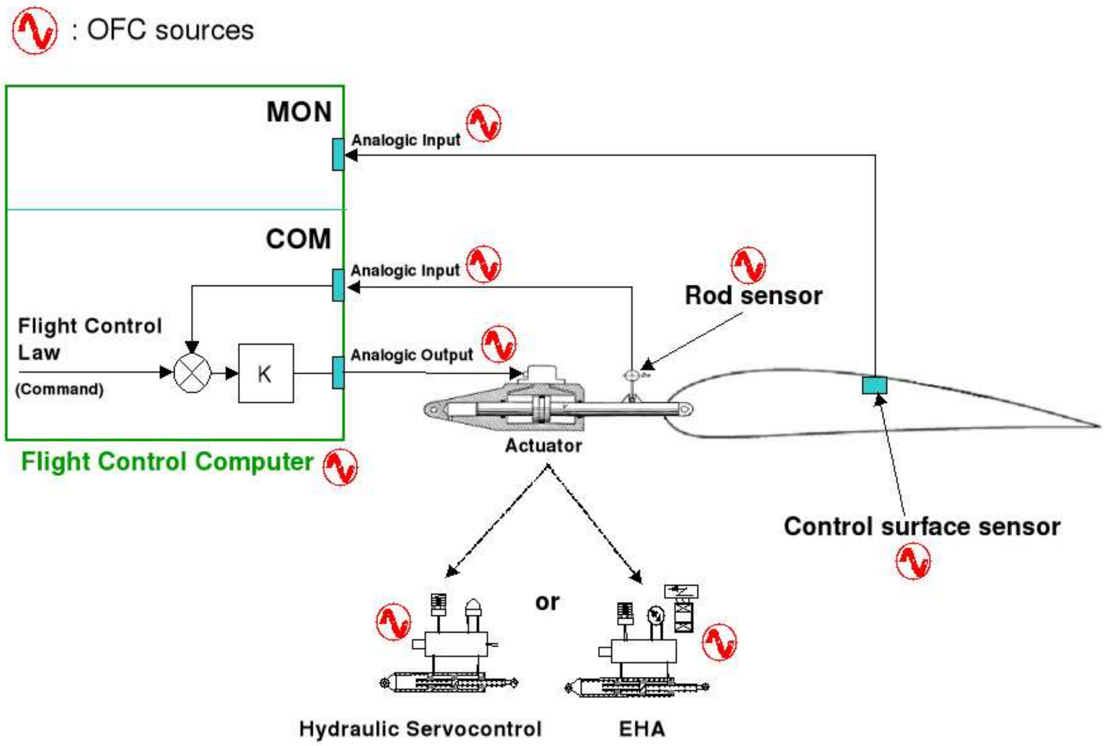

Figure 3: Position control chain.

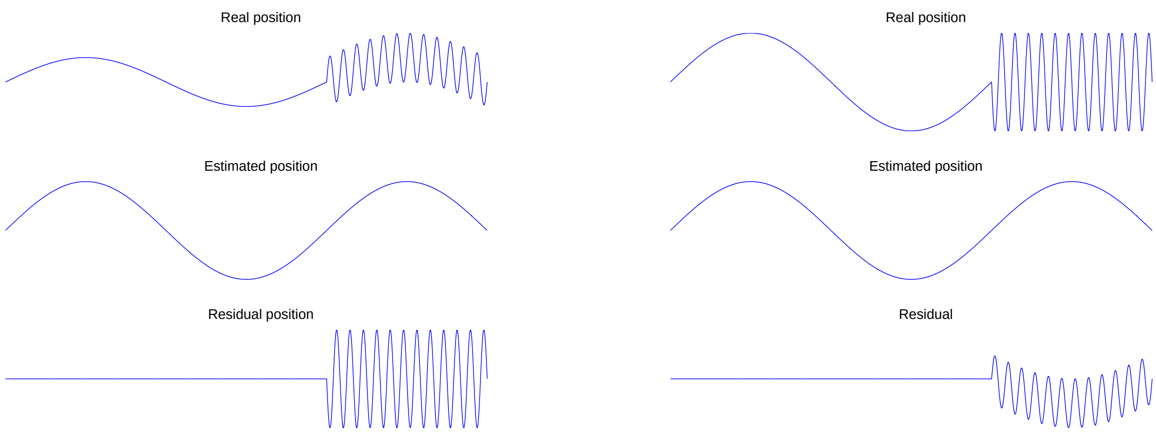

Figure 5: Residuals: liquid failure case on the left side, solid failure case on the right side.

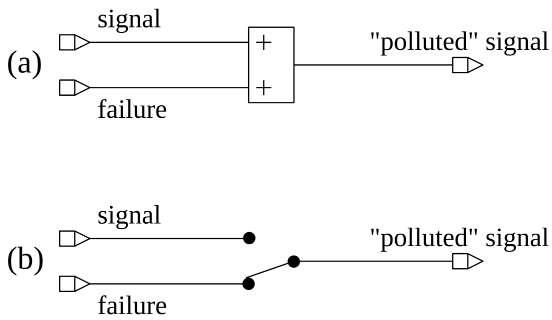

Figure 4: Liquid (a) vs. solid (b) failures.

on residual evaluation by oscillation counting inside spectral subband (Goupil 2007).

Application

In the following section, we address the case of liquid failure with the bounded error parameter estimation method presented previously and use this estimation for detecting OFC. The results are analyzed with respect to the currently used detection method.

Our goal is to perform parameter estimation of the liquid failure model. This fault model defines the shape of the position signal as either a sinus or a triangle. The system to monitor is a simple model of a control surface whose motion is ensured by a hydraulic servo command as presented in figure 6.

In this model, $o(n)$ is the position control signal at time n. The control error ε is given by:

$$\varepsilon(n) = o(n) - \hat{s}(n-1). \tag{10}$$

It is the difference between the position control o at time n and the estimated position $\hat{s}$ at time $n-1$. The estimated current $\hat{i}(n)$ is proportional to the error:

$$\hat{i}(n) = K\varepsilon(n) \tag{11}$$

where K is the constant control gain. A saturation is then applied to the current hence limiting its value within predefined bounds. It is then converted to speed $\hat{v}(n)$ by interpolation with data stored in a look-up table. Finally, the estimated control surface position $\hat{s}(n)$ at time n is computed by integration of the speed.

We ran tests introducing oscillatory failures in the control loop. Two fault models, triangle shaped and sinus shaped, were used. Parameters were estimated over one period of the signal.

Sinus shaped fault

A high noisy sinus-shaped liquid fault signal with a range $A = 1°$ and a frequency of $f = 0.5$Hz is introduced in the control surface model. The initial parameter box is given by $A \times f = [0, 3] \times [0, 10]$.

Figure 7 shows the results provided by the set inversion algorithm when the fault model is supposed to be sinus-shaped. Range parameter A is showed on the horizontal axis while frequency f is on the vertical one. Blue boxes have been rejected, yellow ones have a length inferior to the stop condition set in the algorithm. The red boxes represent the solution. We notice that they concentrate around the real parameter values.

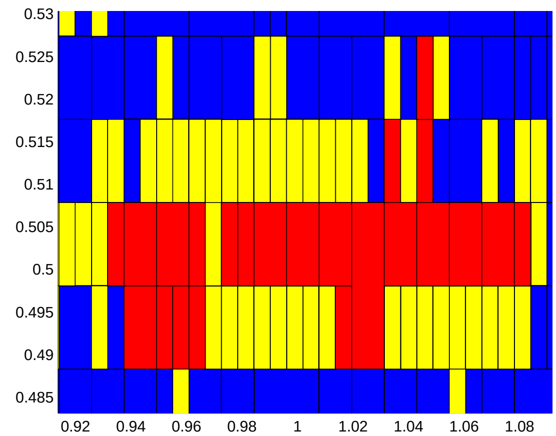

Figure 7: Sinus shaped fault.

When the fault model is triangle-shaped, the algorithm stops after a few iterations and its conclusion is the non-existence of a solution.

Triangle-shaped fault

In this example, the fault is triangle-shaped with a range 2° and a frequency of $f = 0.5$Hz, with a still highly noisy signal. The inital parameter box is now $A \times T = [0, 3] \times [0, 5]$, with $T = 1/f$.

Figure 8 exhibits the obtained results with a triangle-shaped fault model. The parameter A is on the horizontal axis and the period T on the vertical axis. One can notice that the estimation results are fully in accordance with the injected fault.

With a sinus-shaped fault model, the algorithm concludes again to the non-existence of a solution.

Discussion and conclusion

In this paper we presented a method for failure detection using fault models and an error bounded estimation method. The method is based on interval analysis which provides guaranteed results in an error bounded context. It has been applied to solve plane control surfaces oscillatory failures.

The tests show good results for confirming a fault. Now, the real advantage of the method with respect to others is that it is very efficient to prove the non-existence of the solution, that is to discard specific kinds of failures in the real system. In the two case study scenarios, the invalidation of the triangle-shaped (sinus-shaped) fault model is obtained within a few iterations. We should notice that a stochastic method would not invalidate the non relevant fault model but it would conclude to the existence of a solution with a wide confidence range, which is much more difficult to interpret.

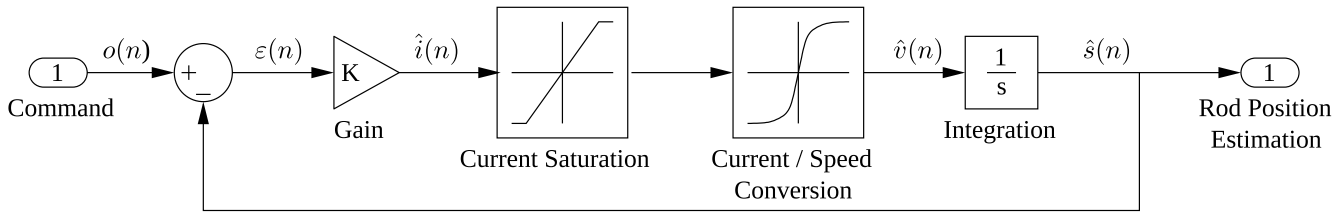

Figure 6: Control surface position estimation model.

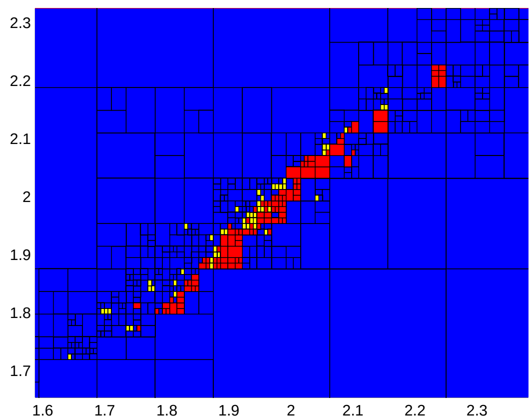

Figure 8: Triangle shaped fault.

Future work will consist in improving the fault model method by studying its properties : response time, false alarm rate, non detection rate and robustness. More simulation tests using alternate fault models against real data will also be performed.

Another direction to go is to use alternate detection methods under the condition to have proper surface control loop models. State estimation and parity state methods, both using interval analysis, should be tested.

References

Alamo, T.; Bravo, J.; and Camacho, E. 2005. Guaranteed state estimation by zonotopes. *Automatica* 41(6):1035–1043.

Armengol, J.; Vehi, J.; Travé-Massuyès, L.; and Sainz, M. A. 2001. Application of modal intervals to the generation of error-bounded envelopes. *Reliable Computing* 7(2):171–195.

Berz, M., and Makino, K. 1998. Verified integration of odes and flows using differential algebraic methods on high-order taylor models. *Reliable Computing* 4:361–369.

Corliss, G. 1994. Guaranteed error bounds for ordinary differential equations. In *Lectures notes at the VI-th SERC Numerical Analysis Summer School*.

Dague, P. 1993a. Numeric reasoning with relative orders of magnitude. In *11th National Conference on Artificial Intelligence, AAAI'93*, 541–547.

Dague, P. 1993b. Symbolic reasoning with relative orders of magnitude. In *International Joint Conference on Artificial Intelligence IJCAI'93*, 1509–1514.

De Kleer, J., and Brown, J. S. 1984. A qualitative physics based on confluences. *Artificial Intelligence* 24:7–83.

Durieu, C., and Walter, E. 2001. *Identification des systèmes*. Paris: Hermès. chapter Estimation ellipsoidales à erreur bornée.

Forbus, K. D. 1984. Qualitative process theory. *Artificial Intelligence* 24:85–168.

Goupil, P. 2007. Oscillatory failure case detection in a380 electrical flight control system by analytical redundancy. In *17th IFAC Symposium on Automatic Control in Aerospace*.

Guerra, P.; Puig, V.; and Ingimundarson, A. 2006. Robust fault detection with state estimators and interval models using zonotopes. In *17th International Workshop on Principles of Diagnosis, DX'06*, 109–116.

Jaulin, L., and Walter, E. 1993. Set inversion via interval analysis for nonlinear bounded-error estimation. *Automatica* 29(4):1053–1064.

Jaulin, L.; Kieffer, M.; Didrit, O.; and Walter, E. 2001. *Applied Interval Analysis, with examples in parameter and state estimation, Robust control and robotics*. Londres: Springer.

Kieffer, M., and Walter, E. 1998. *In A.C. Atkinson, L. Pronzato, H.P. Wynn, Advances Model-Oriented Data Analysis and Experimental Design*. Heidelberg: Physica-Verlag. chapter Interval Analysis for guaranteed nonlinear parameter estimation, 115–125.

Kieffer, M., and Walter, E. 2005. Interval analysis for guaranteed non-linear parameter and state estimation. *Mathematical and Computer Modelling of Dynamical Systems* 11(2):171–181.

Kieffer, M.; Jaulin, L.; and Walter, E. 2002. Guaranteed recursive nonlinear state bounding using interval analysis. *International Journal of Adaptative Control and Signal Processing* 6:191–218.

Kuipers, B. 1984. Commonsense reasoning about causality: deriving behavior from structure. *Artificial Intelligence* 24:169–203.

Kuipers, B. 1994. *Qualitative reasoning, modeling and simulation with incomplete knowledge*. Cambridge, Massachusetts: The MIT Press.

Lesecq, S.; Barraud, A.; and Dinh, K. T. 2003. Numerical accurate computations for ellipsoidal state bounding. In *11th IEEE Mediterranean Conference on Control and Automation, MED'03*.

Milanese, M., and Vicino, A. 1991. Estimation theory for non-linear models and set membership uncertainty. *Automatica* 27(2):403–408.

Moore, R. E. 1959. Automatic error analysis in digital computation. Technical Report LMSD-48421, Lockheed Missiles and Space Co, Palo Alto, CA.

Moore, R. 1966. *Interval Analysis*. Englewood Cliffs: Prentice-Hall.

Nedialkov, N. S.; Jackson, K.; and Pryce, J. 2001. An effective high-order interval method for validating existence and uniqueness of the solution of an ivp for an ode. *Reliable Computing* 7(6):449–465.

Nedialkov, N. 1999. *Computing rigorous bounds on the solution of an initial value problem for an ordinary differential equation*. PhD University of Toronto.

Neumaier, A. 1990. *Interval methods for systems of equations*. Cambridge, UK: Cambridge University Press.

Raïssi, T.; Ramdani, N.; and Candau, Y. 2003. Parameter estimation for nonlinear continuous-time systems in a bounded error context. In *42nd IEEE Conference on Decision and Control, CDC2003*.

Raïssi, T.; Ramdani, N.; and Candau, Y. 2004. Set membership staet and parameter estimation for systems described by nonlinear differential equations. *Automatica* 40:1771–1777.

Ramdani, N. 1995. Méthodes ensemblistes pour l'estimation. Habilitation à diriger des recherches.

Ribot, P.; Jauberthie, C.; and Travé-Massuyès, L. 2007. State estimation by interval analysis for a nonlinear differential aerospace model. In *European Control Conference, ECC'07*, 4839–4844.

Ribot, P. 2006. Estimation d'état et de paramètres en utilisant l'analyse par intervalles. Master recherche smis-eeas, Université Paul Sabatier, Toulouse, France.

Rihm, R. 1994. Interval methods for initial value problems in odes. In Herzberger, J., ed., *Topics in Validated Computations. IMACS-GAMM. International Workshop on Validated Computations*. Amsterdam, New York: University of Oldenburg.

Travé-Massuyès, L., and Piera, N. 1989. The order of magnitude models as qualitative algebras. In *11th International Joint Conference on Artificial Intelligence IJCAI'89*.

Travé-Massuyès, L.; Prats, F.; Sanchez, M.; and Agell, N. 2005. Relative and absolute order-of-magnitude models unified. *Annals of Mathematics and Artificial Intelligence* 45:323–341.

Travé-Massuyès, L.; Ironi, L.; and Dague, P. 2003. Mathematical foundations of qualitative reasoning. *AI Magazine* 24(4):91–106. Special Issue on Qualitative Reasoning.