Semiqualitative Simulation using Modal Interval Analysis

New perspectives for imprecise model-based fault detection

Joaquim Armengol[1], Louise Travé-Massuyès[2]

Josep Vehí[1], Miguel Ángel Sáinz[3]

[1]Institut d'Informàtica i Aplicacions. Universitat de Girona & LEA-SICA
Campus Montilivi. E-17071 Girona. Ph. +34 972 418 487
Fax +34 972 418 098 E-mail:{armengol,vehi}@eia.udg.es

[2]LAAS-CNRS & LEA-SICA. 7 Avenue du Colonel-Roche. F-31077 Toulouse
E-mail:louise@laas.fr

[3]Departament d'Informàtica i Matemàtica Aplicada. Universitat de Girona.
Campus Montilivi. E-17071 Girona. E-mail:sainz@ima.udg.es

Abstract

Imprecision and uncertainty in systems can often be expressed with interval models. The result of the simulation of these models in the form of envelope trajectories. These envelopes can be characterised by several properties such as completeness, soundness which lead to the concept of overbounded and underbounded envelopes. Simulation of such interval models can be performed by several means including quantitative, qualitative and semiqualitative techniques. Whereas existing simulators do not provide any information about the "error" with respect to the exact one, a method to obtain error-known envelopes is proposed. It is based on the simultaneous computation of an underbounded and an overbounded envelope. Both envelopes are computed by means of Modal Interval Analysis. A way of controling the error of the envelopes and adjust it to the desired value is presented.

Keywords: semiqualitative simulation, interval analysis, model-based fault detection.

1 INTRODUCTION

Most of the existing simulators need a mathematical model in which the values of the parameters are real numbers. This implies that the user must have a totally deterministic knowledge of the system. However, complex systems are often subjected to uncertainties that make such a model difficult if not impossible to obtain. A precise model cannot represent the behaviour of such systems which require an explicit representation of imprecisions and uncertainties. A special case is when the uncertainties are *structured*: only the parameters undergo imprecisions but the model structure is known. This case can be handled with interval models in which the equation parameter values are allowed to vary within numeric intervals. For instance, such an interval model could be given by the following transfer function with interval parameters which represents a linear differential relation between an input $u(t)$ and an output $y(t)$:

$$F(s) = \frac{Y(s)}{U(s)} = \frac{[2,3]\,s + [1,3]}{[1,2]\,s^2 + [3,5]\,s + [2,4]}$$

in which s is the Laplace variable and $U(s)$ and $Y(s)$ are the input and output Laplace transforms, respectively. Actually, a precise model can be viewed as an interval model in which the interval widths are zero. As interval widths decrease, precision increases [13].

The results of the simulation of such interval models is in the form of envelope trajectories (or *envelopes* for short)[6] [26]. Hence this is of particular interest to the Fault Detection (FD) community as it provides a way to compute automatically and in a model-based sound manner *adaptive alarm thresholds* for every variable. The envelopes can be characterised by several properties, the main ones being completeness and soundness which lead to the concepts of overbounded and underbounded envelopes which have radical consequences on the robustness and sensitivity of the FD system. Indeed, in the most common case, the device is reported as faulty if the observed outputs go out of their respective envelopes [24]. Therefore, if the envelope is overbounded, this can result in not detecting some faults, whereas, on the other hand, if the envelope is underbounded it may result in false alarms.

The simulation of interval models can be performed by several means including quantitative, qualitative and semiqualitative techniques. Existing simulators do not provide any information about the "error" with respect to the exact one. A method to obtain *error-bounded envelopes* is proposed. It is based on the simultaneous computation of an underbounded and an overbounded envelope. Both envelopes are computed by means of Modal Interval Analysis. A way of controling the error of the envelopes and adjust it to the desired value is also provided.

The next section defines the envelopes and their properties in relation to the fault detection problem. Section 3 discusses the related work and provides a summary of the existing simulators that can be used to generate envelopes, putting special emphasis on simulators based on interval arithmetic. In section 4 a method to generate error-bounded envelopes is presented. This method is based on Modal Interval Analysis, which is presented in section 5, and applied to the envelope generation problem in section 6. Finally, some conclusions and directions for the future work are discussed.

2 ENVELOPES AND THEIR PROPERTIES IN RELATION TO MODEL-BASED FAULT DETECTION

When the behaviour of an interval model, which actually represents a whole set of models, is simulated the result cannot obviously be the same as the one given by traditional simulation of a precise system, that is a unique graph trajectory across time for each variable. The behaviour of such systems can be represented in different ways, depending on whether the quantity space of the variables has been abstracted to a set of finite qualitative values or not.

In the first case, it can be represented by a qualitative state sequence. All the possible behaviours of the imprecise system may be represented globally using a graph as the one in figure 1 [7] or a finite automata in the case of a total envisionment.

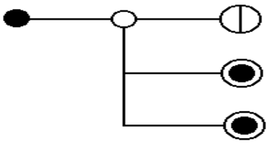

Figure 1: Qualitative simulation

In the second case, the state space is a compact set which can be represented by an envelope trajectory for every variable. All the possible behaviours starting from a specific initial state are compacted within a unique curve. This envelope hence includes a whole family of temporal curves like the one displayed in figure 2. The different temporal curves are indiscernible, so that the qualitative features may not be captured, particularly if the envelope bounding curves do not correspond to a unique temporal trajectory.

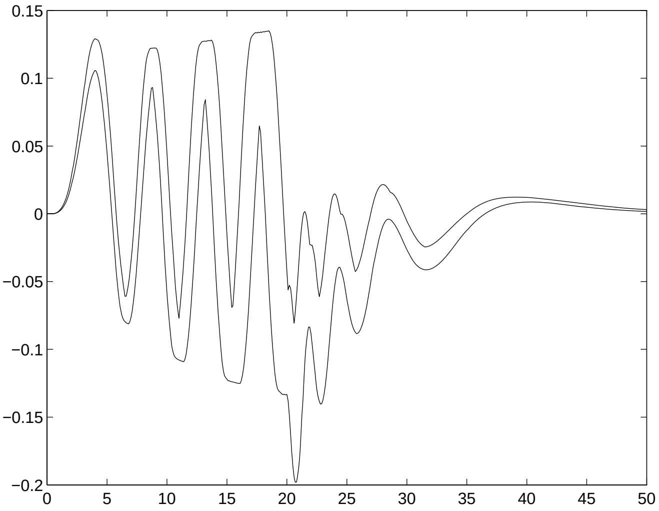

Figure 2: The bounds of the envelopes are two curves

The size of the envelope is critical. If it is too tight there are systems belonging to the model set whose output is outside the envelope, hence the envelope does not include their behaviour. Such an envelope is not *complete*, taking the definition that a complete envelope is one that includes all possible behaviours [23]. On the other hand, if the envelope is too wide it includes zones that cannot be reached by any of the systems belonging to the set. Such an envelope is not *sound*. Our definition of a sound envelope is that every point inside the envelope belongs to the output of at least one of the systems belonging to the set. A complete but not sound envelope is an *overbounded envelope*. A sound but not complete envelope is an *underbounded envelope*.

The ultimate goal is to generate a complete and sound envelope, that is the *exact envelope*. However, more realistic goals are to produce either a minimally overbounded envelope or a minimally underbounded envelope, depending on the use that one wants to make of the envelope. For instance, if the envelope is to be used as reference behaviour for fault detection, the system is reported as faulty if the system output goes out of the envelope. Therefore if the envelope is overbounded, this can result in not detecting some faults. On the other hand, if the envelope is underbounded it may result in false alarms. It is commonly accepted by the FDI community that the overbounded envelopes case is preferable.

Another desirable property of an envelope is *stability*. If the envelope width grows with time, as the input signal remains the same, it is unstable. This means that the imprecision accumulates along the propagation. This is obviously an undesirable property, an unstable envelope becoming useless at some point.

3 RELATED WORK: ENVELOPE SIMULATORS

There are many simulators that can be used to generate envelopes. They can be classified into different groups depending on the information used for the simulation: quantitative, qualitative or semiqualitative [1]. A detailled survey of these simulators can be found in [2].

- Methods based on quantitative or numeric simulation:
  - Superimposition of a threshold (constant or adaptive) arising from statistical data to the behavior of a numeric nominal system [4] [11] [13].
  - Study of a selection of scalar systems [3] [13]. Examples are given by Qua.Si. I and Qua.Si. II [5].
  - Conversion to an optimisation problem. An example is given by Qua.Si. III [5].
- Methods based on qualitative simulation:

When numerical knowledge is available, as in interval systems, there are still notable deficiencies in the attempts that have been made to take advantage of it in qualitative simulators. The resulting methods are called semiqualitative methods. Some examples of systems implemented along this approach are:

  - Q2 [15] which uses QSIM, to which an interval arithmetic module has been added as well as numeric information about the landmarks; Q3 [3].

- SQSIM [13] which intersects the results given by QSIM, Q2, Q3 and NSIM.

- Methods based on interval arithmetic:

Interval arithmetic [20] is interesting as it allows one to consider the whole continuous range of possible instances represented by an interval model. This is due to the natural extension and one of its properties: monotonic inclusion.

Definition 1 *The natural extension of a real function is the one obtained replacing real numbers by standard intervals and real number operations by interval operations.*

Theorem 2 (Monotonic inclusion) *Given $f(x_i)$, a real function, and $F(X_i)$, its natural extension, then $x_i \in X_i$ implies $f(x_i) \subseteq F(X_i)$.*

In consequence, the natural extension gives a guarantee on the result: no function in the class can take values outside the range computed using interval arithmetic. Unfortunately, it does not provide the exact result (complete and sound) in the general case. This comes essentially from the two following problems:

- *The multi-incidence problem*: interval arithmetic considers that each incidence of a variable in a function is independent of the other. Similarly, it is unable to take into account other dependencies. The compiled range is hence overbounded.

- *The wrapping problem*: the state, at some time point, of a system with interval parameters may be represented by an hypercube. However, it may be that the system's state does not evolve into another hypercube at the next time point. In figure 3 an example with two state variables is shown: the hypercube is a rectangle. It transforms into a rhombus at the following time step (it could actually evolve into any two dimensional shape). The projection of this rhombus on the variable axis leads to a new rectangle which obviously includes spurious states, as shown in the figure. Therefore, the obtained envelopes are overbounded.

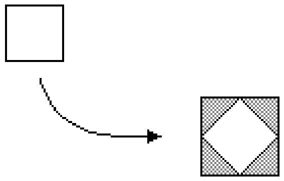

Figure 3: The wrapping problem

Some semiqualitative simulators based on interval arithmetic are:

- The Moore's interval simulator [20] which simply ignores multi-incidences.

- The Markov and Angelov simulator [18].
- AWA [16], that tries to minimise the wrapping effect by changing the system's co-ordinates along the trajectory of the system [13].
- NIS [26].
- NSIM [13].
- Ca$\sim$En (Causal Engine) [6].

All these simulators produce overbounded envelopes. Some methods use fuzzy sets: All of them take the α-cut of the fuzzy sets and end up using interval arithmetic. Some of them are:

- FuSim [22], an extension of QSIM.
- Mycroft [7], which includes a simulator based on QSIM, FuSim and PA.

4 GENERATION AND CONTROL OF ERROR-BOUNDED ENVELOPES

The properties of the simulators presented in the previous section have been exibited in a binary manner. For example, all the interval based simulators are complete but not sound, i.e. they provide overbounded envelopes. A step forwards would be to produce some kind of measure of the *degree of overbounding,* in other word to be able to evaluate the "error" of the obtained envelope with respect to the exact one. However, the exact envelope is of course not known. The exact envelope problem is actually highly complex as it requires global optimisation tools for non linear and non convex functions.

An alternative way to approach this problem is to bound the error, i.e. to determine the maximum distance. This can be achieved by computing both an underbounded envelope and an overbounded envelope. The distance between these envelopes indeed gives the maximum error. These envelopes are defined as *error-bounded envelopes.*

A way of controling the error is by widening the underbounded envelope or by tightening the overbounded one.

The generation of the underbounded and overbounded envelopes using a new approach based on Modal Interval Analysis as well as the way to control the error are presented in the following sections.

The approach used to generate the underbounded and overbounded envelopes is based on the reformulation of the simulation problem into an optimisation problem.

The behaviour of a n-th order dynamic system can be represented by the following difference equation:

$$y_{t+1} = \sum_{i=0}^{n} a_i y_{t-i} + \sum_{j=0}^{m} b_j u_{t-j}$$

in which it can be observed that the output of the system at any time point depends on the values of the previous outputs and inputs. Hence, finding the limits of the envelope at a given time point is equivalent to finding the maximum and the minimum of a function into a parameter space. This is a global optimisation problem.

There are many different methods for global optimisation, but many of them have no guarantee of finding the global optimum. In the case of the envelope generation problem, a local optimum results in an incomplete (underbounded) envelope.

Conversely, global optimisation methods based on the interval arithmetic obtain overbounded results as stated by the monotonic inclusion property (cf. section 3). This comes from the multi-incidence problem. In the following, two types of multi-incidences are pointed out.

The first type of multi-incidence comes from the fact that, in a difference equation like the one shown above, some parameters may appear several times. For instance, the transfer function representation of a generic first order system is:

$$F(s) = \frac{Y(s)}{U(s)} = \frac{k}{\tau s + 1}$$

in which k is the static gain and τ is the time constant. If this transfer function is discretised by the Euler method, the difference equation representation of this system is:

$$y_t = \left(1 - \frac{T}{\tau}\right) y_{t-1} + \frac{kT}{\tau} u_{t-1}$$

in which T is the sampling period. As it can be seen, τ appears more than once in this equation. If the equation is rewritten, renaming the parameters as follows:

$$a = \left(1 - \frac{T}{\tau}\right) \text{ and } b = \frac{kT}{\tau}$$

the situation is even worse because the multi-incidences do not appear explicitely. This is the reason for avoiding intermediate operations: each time an intermediate operation is performed some information is lost. Going deeper, k and τ themselves include implicit multi-incidences with respect to the physical parameters of the device!

The second type of multi-incidences is particular to the simulation mechanism as the equations are taken at different time points. For instance, the difference equation at the time point $t+1$ of the generic first order system used above is:

$$y_{t+1} = \left(1 - \frac{T}{\tau}\right) y_t + \frac{kT}{\tau} u_t$$

As it can be seen, y_t, τ and k appear both in the difference equation for time point t and $t+1$. It should be noticed that these multi-incidences can be treated as independent variables for systems whose parameters are known to vary in time. On the other hand, they must be treated as so if the physical system is assumed to be invariant. In this latter case, the multi-incidences can be made explicit by merging the different equations starting from 0 into a unique expression on which the optimisation is performed.

Modal Interval Analysis is an efficient tool to perform the optimisation task taking the multi-incidences into account.

5 MODAL INTERVAL ANALYSIS

Modal Interval Analysis [8] [9] [10] extends real numbers to intervals, identifying the intervals by the predicates that the real numbers fulfil, unlike classical Interval Analysis which identifies the intervals with the set of real numbers they contain.

In the following, some of the properties of modal intervals that are interesting for envelope generation are stated.

The defining notation for modal intervals is $X := (X', Q)$, where $X' \in I(\mathbf{R})$, which is the set of closed intervals of $\mathbf{R}$, $I(\mathbf{R}) = \{[a,b]' \mid a, b \in \mathbf{R},\ \ a{\le}b\}$, and $Q \in \{E, U\}$, E and U denoting the classical logic qualifiers. The canonical notation is

$$[a_1, a_2] := \left\{ \begin{array}{c} ([a_1, a_2]', E) \text{ if } a_1 \leq a_2 \\ ([a_2, a_1]', U) \text{ if } a_1 \geq a_2 \end{array} \right\}$$

where the quantifiers $Q \in \{E, U\}$ are the *modality* and $[a_1, a_2]'$, called the *extension,* is an interval in the classical sense, i.e. a set of real numbers. Less formally, if the modal intervals are canonically denoted by $[1, 2]$ or $[2, 1]$, when the lower limit is written to the left, like in $[1, 2]$, it means that the interval is just an "interval bound" called "existential interval" or "proper interval". That is to say, the existential interval (the classical one) denotes one value within the interval bounds. When the upper limit is written to the left, like in $[2, 1]$, it indicates that the interval is a "tolerance" and it is called "universal interval" or "improper interval", which denotes all the values within the interval bounds.

The rational operations between modal intervals are extensions of classical interval arithmetic with the addition of the dual operation defined by:

$$Dual([a_1, a_2]) = [a_2, a_1]$$

The dual formulation of the modal intervals allows to define two modal interval extensions of a continuous function f: $f^*(X)$ and $f^{**}(X)$. The modal interval extension represented by $f^*(X)$ may be interpreted as

$$U\left(x_p, X_p^{'}\right) Q\left(z, F\left(X\right)\right) E\left(x_i, X_i^{'}\right) \left(z = f\left(x_p, x_i\right)\right)$$

This can be read: "For all elements belonging to the proper intervals there exists at least one element in the improper intervals that fulfil the function".

Example 3 $[10, 20] + [20, 15] = [30, 35]$ *means*

$$f^*\left(X\right) \subseteq F\left(X\right) \Longleftrightarrow U\left(a, [10, 20]'\right) E\left(f, [30, 35]'\right) E\left(b, [15, 20]'\right) \left(a + b = f\right)$$

Example 4 $[10, 20] + [15, 20] = [25, 40]$ *means*

$$U\left(a, [10, 20]'\right) U\left(b, [15, 20]'\right) E\left(f, [25, 40]'\right) \left(a + b = f\right)$$

On the other hand, the semantic interpretation of the $f^{**}(X)$ extension is dual:

$$F\left(X\right) \subseteq f^{**}\left(X\right) \Longleftrightarrow U\left(x_i, X_i^{'}\right) Q\left(z, Dual\left(F\left(X\right)\right)\right) E\left(x_p, X_p^{'}\right) \left(z = f\left(x_p, x_i\right)\right)$$

These two interpretations are very useful in order to compute the envelopes:

- Overbounded envelope. Its semantics is: "For every (universal quantifier) model parameter, input and initial state, the output belongs to the envelope (existential quantifier)".
- Underbounded envelope. The semantics is dual: "For every output belonging to the envelope there exist parameter, input and initial state values that produce this output"

Therefore, Modal Interval Analysis can be used to compute both envelopes. Unfortunately, the computation of $f^*(X)$ and $f^{**}(X)$ is not always possible. The usual procedure is to find overbounded computations of $f^*(X)$ and underbounded computations of $f^{**}(X)$ which maintain the semantic interpretations. To this respect, an aspect to be taken into consideration is the rounding of computations. Computers work with digital numbers, not with real numbers. In order to maintain the semantic interpretations, direct roundings (up or down) have to be used.

If f is a rational function, there are some theorems in Modal Interval Analysis that allow to obtain the exact range of f in some cases or overbounded computations of $f^*(X)$ and underbounded computations of $f^{**}(X)$ in other cases.

Definition 5 *A modal interval extension $fR(X)$ of f in X is optimal if*

$$f^*(X) = fR(X) = f^{**}(X)$$

The Coercion theorem provides the conditions and the way to obtain optimal extensions.

Theorem 6 (Coercion theorem) *Given $fR(X)$, a rational interval function defined in a parameter space $Prop(X)$, tree-optimal and totally monotonic for each multi-incident component of X. Let XD be an enlarged vector of X obtained considering each multi-incident component as independent and transforming it into its dual if it is antitonic (the monotonicity of this incidence and the monotonicity of the component have opposite senses). In this case,*

$$f^*(X) = fR(XD) = f^{**}(X)$$

Example 7 *Given $f = x^2 - xy$ and the parameter space $x = [2,4]$ and $y = [1,2]$, the range of the function obtained by its natural extension is $f^*(X) \subseteq [-4,14]$. The exact range can be computed applying the Coercion Theorem and is $f^*(X) = f^{**}(X) = [2,4]^2 - Dual([2,4])[1,2] = [0,12]$*

If the function is not monotonic for each multi-incident component, the Coercion Theorem can be partially applied in order to reduce the complexity of the problem. For instance, given an n variable function, the problem of finding the range of this function in a domain in which the function is monotonic with respect to r variables, can be reduced to evaluate the range of an interval function of $n-r$ variables. Therefore, the problem complexity has a lower order.

Example 8 *Given $f = xy - x^2 - 2y$ and the parameter space $x = [1,2]$ and $y = [3,4]$, the range of the function obtained by its natural extension is $f^*(X) \subseteq [-9,1]$. The function is totally monotonic with respect to y*

$$\frac{\partial f}{\partial y} = x - 2 = [-1,0] \leq 0$$

but it is not totally monotonic with respect to x

$$\frac{\partial f}{\partial x} = y - 2x = [-1, 2] \ni 0$$

The Coercion Theorem can be applied to y and hence a better approximation of the range of the function is obtained:

$$f^{*}(X) \subseteq xDual(y) - x^2 - 2y = [-8, -1]$$

A way to obtain even a better approximation is by splitting the parameter space. The advantage is that now only the variable x must be split. Moreover, the range in each sub-space can be computed more exactly because the modality of each incidence of the variable y is already known.

As a conclusion, the number of sub-spaces to be made in order to compute an approximation of the range of the function is smaller when modal intervals are used. This is illustrated in [25], in which modal intervals combined with a branch-and-bound algorithm have been applied to the analysis and design of robust controllers.

6 PRACTICAL APPLICATION TO ENVELOPE GENERATION AND EXAMPLES

In the previous section, some tools of Modal Interval Analysis that are useful for envelope generation have been described. One limitation of these tools is that it is necessary to differentiate the function in order to apply de Coercion Theorem, hence restricting the method to differentiable functions. However, the discrete representation of the system used in our simulation problem is differentiable.

Asimulator based on these tools has been implemented. It uses iMatlab version 5.1 for Unix [19]. Symbolic computations are performed with Maple V r4 [17] through the Symbolic Math Toolbox. Moreover, it uses C++ programs as MEX-files to perform modal interval computations with direct roundings and to accelerate the branch-and-bound algorithm.

The iñplemented simulation algorithm is the following:

Modal Interval Branch-and-bound Algorithm

```
Given a function in a space
Modality of uni-incident variables is not changed
IF all variables are uni-incident THEN
  Exact result
  END
ENDIF
Calculate internal approximation
DO
  External = internal
  DO
    Get subspace
    FOREACH variable with unknown modality
      IF $0 \notin$ first derivative THEN
        fix modality
        ENDIF
      ENDFOREACH
      IF all modalities are known THEN
        Calculate parcial exact
        Internal = internal $\vee$ parcial exact
        External = external $\vee$ parcial exact
      ELSE
        Calculate parcial internal.
        Internal = internal $\vee$ parcial internal.
        Calculate parcial external
        External = external $\vee$ parcial external
        Divide subspace
      ENDIF
    WHILE remaining subspaces
    IF error<$\varepsilon$ THEN
      final=1
    ENDIF
  WHILE final=0
END
```

As an example, figure 4 shows the envelopes obtained for a generic first order system with the following parameters:

- static gain: $k = [0.95, 1.05]$
- time constant: $\tau = [5, 20]$
- initial state: $y_0 = 0$
- sampling time: $T = 1\ s$
- input: steps of different lengths and magnitudes
- maximum error of the envelopes $\varepsilon \leq 0.2$

In this figure, the solid line is the overbounded envelope and the dotted line is the underbounded one.

Another example is the one shown in figure 5, in which the same system as above is excited with a sinusoidal input. On the right, a high frequency white noise has been added to the input, whereas on the left the input is the same but without the noise. The interesting comment about this example is that, as it can be seen on the figures, a significant amount of noise has been "absorbed" by the envelopes. Indeed, the envelopes produced with or without noise have a much higher similarity than the noisy and non noisy inputs. In particular, the envelopes corresponding to the noisy input are not wider than the other ones. The width depends essentially on the imprecision of the model. This means that *semiqualitative simulation acts as a high frequency filter*. This property has not been carefully studied yet but should deserve attention in view of fault detection applications.

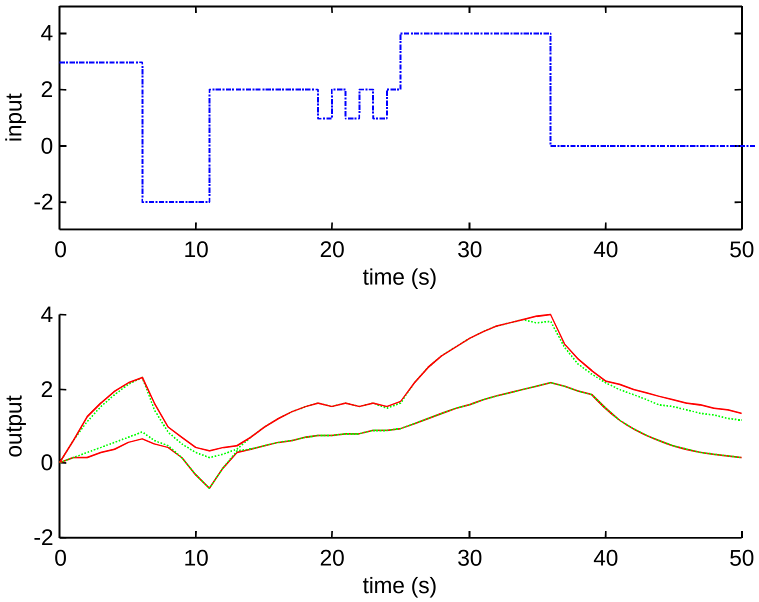

Figure 4: Example of simulation

7 FAULT DETECTION WITH ERROR-BOUNDED ENVELOPES

When th error-bounded envelops are used for fault detection, it is guaranteed that there is a fault when the measure is outside the overbounded envelope, but the measure can be inside this envelope even when the system is faulty.

One reason is that the overbounded envelope, by definition, includes points that cannot be reached by any of the systems represented by the interval model. This zones are commonly referred as spurious points. The size of the spurious points zone can be reduced adjusting the envelopes error bound to a lower value. This implies a higher computation effort that is not necessary if the measure is inside the underbounded envelope. Indeed, all the points within the underbounded envelope are guaranteed to be reachable by a non faulty system. Therefore, the additional computation effort is necessary only when the measure is between the two envelopes. The error must be reduced until the measure appears outside the overbounded envelope or inside the underbounded envelope.

The second reason is that the measure can remain inside the envelope for some time after a fault has occured due to the dynamics of the system. This means that the fault is not detected inmediately. The time necessary to detect the fault is related to the distance between the actual system parameter values and their nominal values. If the distance is small, the time necessary to detect the fault is larger. This time is not reduced if the error between the envelopes is reduced.

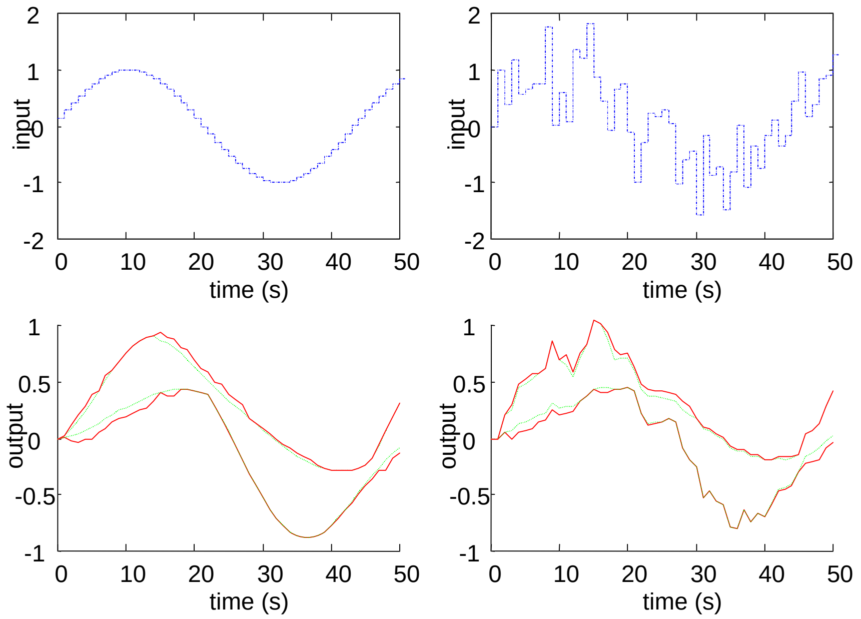

Figure 5: Simulation without (left) and with (right) noise

In conclusion, it is not necessary to work with a small error except in the case when the measure is between the two envelopes. This means that the desired error can be taken as a dynamic parameter of the simulation whose value is updated on-line. In the case of the example given in the previous section, the error, a constant, has been chosen in order to distinguish the two envelopes: it is not very small, but sufficient in most cases.

8 CONCLUSIONS AND FUTURE WORK

In this paper it has been shown that the existing simulators for systems with structured uncertainties provide envelopes which may or may not have properties like completeness, soundness, stability, etc. or not. Sometimes the properties are not even known.

When the properties are known, the error of the envelopes with respect to the exact one is unknown. A method to obtain error-bounded envelopes is proposed. It is based on the simultaneous computation of an underbounded envelope and an overbounded one. Two ways to control this error are proposed as well: tightening the overbounded envelope or widening the underbounded one. Both can be achieved by means of Modal Interval Analysis. The error of our envelopes can hence be adjusted to the desired value. The computation effort of course increases when the error decreases.

This method has been implemented in Matlab and uses Maple and C routines. This facilitates its future integration into a supervision framework that is being developed based on Matlab and Simulink. It is also planned to be used to improve the prediction and fault detection algorithms of the Ca˜En simulator[24].

An inconvenient of this method is that the computation effort increases at each step of the simulation. The exact envelope at a time point t can only be obtained by computing the range of the function that relates the current time point to the initial one, i.e. all the previous states must be considered in order to obtain the exact envelope. This means that the procedure is not incremental. A solution to overcome this problem is the use of a shorter temporal window. Due to the dynamics of the systems, it has been shown that the influence of the previous states over the current one decreases with time[21]. Hence, it should be possible to obtain similar results with a shorter temporal window lengh. Saludes [21] recommends a window length which essentialy depends on the time constant of the system.

A related problem is about the semantics of these results. For instance, given an overbounded envelope and an underbounded one at a time point t, both can be used as the initial state for the function at time point $t + m$. This means that four approximations of the range of the function can be obtained: over and underbounded approximations of the range of the function using over and underbounded approximations of the initial state. Each one of these four approximations has different semantics and the most suitable one, if it exists, has to be decided. This problem is still under investigation. It seems that the solution is simple for overbounded envelopes. This envelope can be obtained computing an overbounded range of the function and using an overbounded initial state. As the length of the temporal window increases, the envelope gets closer to the exact one. The case of underbounded envelopes is more complex and more work remains to be done.

As it has been shown, a very interesting feature of modal intervals is the semantics. A future work is to study whether envelopes with different semantics can be used not only to detect the faults but also to localise the faulty parameter. For instance, if a system has two physical parameters a and b, envelopes with the semantics "for every a there exists b so that..." or "for every b there exists a so that..." can be obtained. If a system is faulty and its output belongs to only one of these two envelopes, it should be possible to determine whether it is a or b that is faulty.

Finally, in [4] it is claimed that it is not necessary to study the evolution of all points belonging to an uncertainty region to know the evolution of the region. The study of the evolution of points belonging to its surface is enough. The possible application of this result to modal interval simulation gives another direction for research.

ACKNOWLEDGEMENTS

This work has been funded by the autonomous government of Catalonia, Generalitat de Catalunya, through CIRIT 1996 XT 00023 and Grup de Recerca Consolidat ref. 1995GR00474 and by the Spanish government through co-ordinated research project CICYT TAP96-1114-C03-03, "Plataformas integradas de CAD de Supervisión y Metodologías".

References

[1] Armengol, J., L. Travé-Massuyès, J. Vehí, J. Ll. de la Rosa. On Modal Interval Analysis for envelope determination within the Ca~En qualitative simulator. Accepted to the 7th Information Processing and Management of Uncertainty in Knowledge-Based Systems Conference (IPMU'98). Paris, France, 1998.

[2] Armengol, J., L. Travé-Massuyès, J. Vehí, J. Ll. de la Rosa. A survey on interval model simulators and their properties related to fault detection. Internal Report. Institut d'Informàtica i Aplicacions. Universitat de Girona. 1998. Submitted to IFAC World Congress 99.

[3] Berleant, D., B. J. Kuipers. Qualitative and Numerical Simulation with Q3. In Faltings, B., P. Struss, eds. Recent advances in qualitative physics. The MIT Press. Cambridge, Massachusetts. 1992.

[4] Bonarini, A., G. Bontempi. A qualitative simulation approach for fuzzy dynamical models. ACM transactions on modeling and computer simulation, Vol. 4, No. 4, pp. 258-313. 1994.

[5] Bontempi, G.. Qua.Si. III : A software tool for simulation of fuzzy dynamical systems. In Proceedings of the European Simulation Multiconference, pp. 615-619. Ghent, Belgium. 1996.

[6] Bousson, K., L. Travé-Massuyès. Putting more numbers in the qualitative simulator CA-EN. In Proceedings of International Conference on Intelligent Systems Engineering (ISE 94). Hamburg-Harburg, Germany. 1994, pp. 62-69.

[7] Coghill, G. Mycroft: a framework for constraint based fuzzy qualitative reasoning. Ph. D. thesis. Department of computing and electrical engineering. Heriot-Watt University. Edinburgh, Scotland. 1996.

[8] Gardeñes, E, H. Mielgo, A. Trepat Modal Intervals: reasons and ground semantics. In Lecture Notes in Computer Science #212 - Interval Mathematics 1985. Springer-Verlag, Berlin, 1986.

[9] Gardeñes, E., H. Mielgo. Modal intervals: functions. In Proceedings of the Polish Symposium on Interval and Fuzzy Mathematics. Zamenkhov's University of Poznan. Poland, 1986.

[10] Gardeñes, E., H. Mielgo, M. A. Sáinz. Presentation of the research group SIGLA/X. Report de recerca IMA 95-10. Universitat de Girona.

[11] Gasca, R. M., J. A. Ortega, M. Toro. Simulación semicualitativa. In Proceedings of Jornades Hispano-Franceses de Sistemes Intel·ligents i Control Avançat. Barcelona, 1996.

[12] Kay, H. Semiquantitative Simulation: Successes, Failures, and Future Directions. 1995.

[13] Kay, H. Refining imprecise models and their behaviors. Ph. D. dissertation. University of Texas at Austin. 1996.

[14] Kuipers, B. J. Qualitative Simulation. Artificial Intelligence. Vol. 29, pp. 289-338. 1986. Also in Weld, D. S., Johan de Kleer, eds. Readings in qualitative reasoning about physical systems. Morgan Kaufmann Publishers, Inc. San Mateo, California. 1990, pp. 236-260.

[15] Kuipers, B. J., D. Berleant. Using Incomplete Quantitative Knowledge in Qualitative Reasoning. In Proceedings of AAAI-88, pp. 324-329. 1988.

[16] Lohner, R.. Einschliessung der Lösung gewöhnlicher Anfangs- und Randwertaufgaben und Anwendungen. Ph. D. thesis. Universität Karlsruhe, Germany. 1988.

[17] Maple V. Waterloo Maple, Inc. 450 Phillip St. Waterloo, ON N2L 5J2, Canada. mailto:info@maplesoft.com, http://www.maplesoft.com.

[18] Markov, S., R. Angelov. An interval method for systems of ODE. In Lecture Notes in Computer Science #212 - Interval Mathematics 1985, pp. 103-108. Springer-Verlag, Berlin, 1986.

[19] Matlab-Simulink. The Math Works, Inc. 24 Prime Park Way. Natick, MA 01760-1500, USA. mailto:info@mathworks.com. http://www.mathworks.com.

[20] Moore, R. E. Interval Analysis. Prentice-Hall, 1966.

[21] Saludes, J., V. Puig, J. Quevedo. Determination of window length for a new algorithm in adaptive threshold generation. In Proceedings of Symposium on Qualitative System Modeling, Qualitative Fault Diagnosis and Fuzzy Logic and Control. TEMPUS S_JEP 07759-94 - MODIFY grant. Budapest, Hungary. 1997.

[22] Shen, Q., R. R. Leitch. Fuzzy qualitative simulation. In IEEE Transactions on Systems, Man and Cybernetics, Vol. 23, No. 4, pp. 1038-1061. 1993.

[23] Struss, P. Problems of interval-based qualitative reasoning. In Weld, D. S., Johan de Kleer, eds. Readings in qualitative reasoning about physical systems. Morgan Kaufmann Publishers, Inc. San Mateo, California. 1990, pp. 288-305.

[24] Travé-Massuyès, L., R. Milne. TIGER: Gas turbine condition monitoring using qualitative model based diagnosis. IEEE Expert Intelligent Systems and Applications. Vol. May-June.

[25] Vehí, J. Anàlisi i disseny de controladors robustos mitjançant intervals modals. Ph. D. Thesis. Universitat de Girona. 1998.

[26] Vescovi, M. R., A. Farquhar, Y. Iwasaki. Numerical Interval Simulation. International Joint Conference on Artificial Intelligence. Montreal, Quebec, Canada. 1995, pp 1806-1813.

[27] Wiegand, M. E. Constructive qualitative simulation of continuous dynamic systems. Ph. D. Thesis. Heriot-Watt University. Edinburgh, Scotland, 1991 05.