



Brief Paper

Proving set inclusion via intervals:
application to parametric robust stability¹

L. Jaulin*, J. Burger

Laboratoire d'Ingénierie des Systèmes Automatisés, Université d'Angers, 2 boulevard Lavoisier, 49 045 Angers, France.

Received 7 January 1997; revised 25 February 1998; received in final form 7 September 1998

Abstract

Proving that an uncertain parametric model is stable amounts to prove the inclusion of two sets: the set A of all feasible parameters and the set B of all parameters for which the model is stable. In this paper, a new algorithm, able to decide whether or not A is included in B , is presented. The method is based on interval analysis which is a numerical tool able to deal with inequalities in a global and guaranteed way. Convergence properties of the algorithm are provided. The algorithm is then applied to the robust stability of a discrete-time model where the information on the parameters is given through bounded-error data. The behavior of the algorithm with respect to the number of parameters is illustrated on a continuous-time model. © 1999 Elsevier Science Ltd. All rights reserved.

Keywords: Bounded-error estimation; Interval analysis; Parametric model; Robust stability; Set estimation; Set inclusion

1. Introduction

The parametric approach to robust stability analysis has received a great attention in the last decade (see, e.g. Barmish, 1994; Bhattacharyya et al., 1995 and the references therein). One of the main problems considered in this approach is to decide on the stability of an uncertain parametric model $\mathcal{M}(\mathbf{p})$, where the parameter vector \mathbf{p} is only known to belong to some known feasible set A . The model $\mathcal{M}(\mathbf{p})$ is assumed to be input–output linear. Let us denote by

$$P(\mathbf{p}, s) = \alpha_d(\mathbf{p})s^d + \alpha_{d-1}(\mathbf{p})s^{d-1} + \dots + \alpha_1(\mathbf{p})s + \alpha_0(\mathbf{p}), \quad (1)$$

its characteristic polynomial. When the functions $\alpha_j(\mathbf{p})$, $j = 0 \dots d$, are affine with respect to \mathbf{p} , and when A is a box or a polytope, extreme point methods (see, e.g. Barmish and Kang, 1993), based on Kharitonov's theorem (Kharitonov, 1978) and other extreme-point results

such as the Edge theorem (Barlett et al., 1988), can decide on the stability of $\mathcal{M}(\mathbf{p})$ in a guaranteed way. When $\alpha_j(\mathbf{p})$ is polynomial and when the feasible set A is a box, approaches based on Bernstein polynomial are available (Vicino et al., 1990) but branch-and-bound methods have to be used to avoid immoderate pessimism. When $\alpha_j(\mathbf{p})$ is any nonlinear function, interval methods (Moore, 1979) are able to deal with the stability of $\mathcal{M}(\mathbf{p})$ in a guaranteed way. In the context of parametric models, interval analysis has already been used by Kolev (1988) to build a guaranteed optimization algorithm for robust control, by Walter and Jaulin (1994) for the characterization of stability domains and by Malan et al. (1996) and Jaulin and Blalter (1994) to characterize the set of all parametric controllers that assure some given performances to an uncertain system.

All the methods that have been presented in the literature require a feasible set A with a simple shape (generally a box or a polytope) that contains the actual parameter vector in a guaranteed way. Such a feasible set can be computed by using bounded-error parametric estimation (see, e.g., Walter, 1990; Norton, 1994, 1995; Milanese et al., 1996 and the references therein). This approach makes it possible to enclose the prior feasible set, i.e., the set of all parameter vectors that are consistent with bounded-error data, in a guaranteed way. Now, the prior feasible set can generally be defined by nonlinear

* Corresponding author. Tel.: 0033 02 41735 417; fax: 0033 02 41735 352, e-mail: jaulin@univ-angers.fr.

¹ This paper was not presented at any IFAC meeting. This paper was recommended for publication in revised form by Associate Editor A. Vicino under the direction of Editor T. Söderström.

inequalities and dealing with them makes it possible to avoid the pessimism introduced by an outer approximation. For such sets defined by nonlinear inequalities, classical robust approach cannot be used in a guaranteed way and a specific methodology has to be developed.

In this paper, we shall consider the general situation where A is described by a set of nonlinear inequalities. Since the model is input–output linear, the stability domain B (set of all p associated with a stable model) can also be defined by nonlinear inequalities that may be obtained by using the Routh criterion for linear continuous-time models or the Jury criterion for linear discrete-time models. The problem of proving the stability of $\mathcal{M}(p)$ amounts to prove the inclusion of two inequality sets, namely $A \subset B$. Interval analysis (Moore, 1979) will be shown to be particularly suited to solve this problem.

Basic notions of interval analysis are introduced in the following section. In Section 3, a new algorithm capable of proving the inclusion of two inequality sets is presented. A convergence analysis is provided in Section 4. In Section 5, the algorithm is applied to the stability analysis of an uncertain discrete-time model where some informations are available on the parameters through bounded-error data. The complexity of the algorithm with respect to the number of parameters is illustrated with the robust stability analysis of an uncertain continuous-time model.

2. Interval analysis

Interval analysis is a numerical tool originally developed to quantify the effect of finite-precision arithmetic on results obtained by a computer (Moore, 1979). At present, it is also used in a finite-dimensional context for global optimization (see e.g. Hansen, 1992) or to prove formal inequalities (see e.g. Moore, 1979). Interval analysis is based on the notions of boxes and inclusion functions that are now introduced.

A *box* or *vector interval* \mathbf{X} of R^n is the Cartesian product of n intervals:

$$\mathbf{X} = [x_1^-, x_1^+] \times \cdots \times [x_n^-, x_n^+] = X_1 \times \cdots \times X_n. \quad (2)$$

In the sequel, intervals are written with upper-case letters, boxes with bold upper-case letters and vectors with bold lower-case letters. The set of all boxes of R^n is denoted by \mathbb{R}^n . A principal plane of \mathbf{X} is a symmetry plane of \mathbf{X} normal to a side of maximum length. To bisect a box \mathbf{X} means to cut it along one of its principal planes. A bisection generates two non-overlapping boxes \mathbf{X}_1 and \mathbf{X}_2 such that $\mathbf{X} = \mathbf{X}_1 \cup \mathbf{X}_2$. Let \mathbf{f} be a vector function mapping R^n into R^m . A set-valued function \mathbf{F} , defined

from \mathbb{R}^n into \mathbb{R}^m , is an inclusion function of \mathbf{f} if:

$$\forall \mathbf{X} \in \mathbb{R}^n, \quad \mathbf{f}(\mathbf{X}) \subset \mathbf{F}(\mathbf{X}). \quad (3)$$

Let $w(\mathbf{X})$ be the width of the box \mathbf{X} , i.e. the length of its largest side(s). \mathbf{F} is convergent if, for any sequence of boxes $\mathbf{X}(k)$ of \mathbb{R}^n ,

$$\lim_{k \rightarrow \infty} w(\mathbf{X}(k)) = 0 \Rightarrow \lim_{k \rightarrow \infty} w(\mathbf{F}(\mathbf{X}(k))) = 0. \quad (4)$$

The computation of a convergent inclusion function associated with any continuous function, defined by an explicit formal expression, is simple (see, e.g. Moore, 1979) and routinely performed by commercially available languages such as C-XSC (Hammer et al., 1996).

3. Proving set inclusion

Let A and B be two compact (closed and bounded) sets of R^n included in a box \mathbf{X}_0 . The problem to be solved is that of deciding whether $A \subset B$. Sets A and B are assumed to be defined by finite sets of inequalities:

$$A = \{\mathbf{x} \in R^n \mid \mathbf{f}(\mathbf{x}) \leq 0\}, \quad (5)$$

$$B = \{\mathbf{x} \in R^n \mid \mathbf{g}(\mathbf{x}) \leq 0\},$$

where \mathbf{f} and \mathbf{g} are continuous functions. The inequalities have to be understood componentwise. In what follows, \mathbf{R}^- and \mathbf{R}^{-*} denote the sets $] - \infty, 0]^q$ and $] - \infty, 0[^q$, respectively, where q is the appropriate dimension. If \mathbf{f}^{-1} and \mathbf{g}^{-1} denote the *reciprocal function* (in a set-theoretic sense) of \mathbf{f} and \mathbf{g} , then,

$$A = \{\mathbf{x} \in R^n \mid \mathbf{f}(\mathbf{x}) \in \mathbf{R}^- \} = \mathbf{f}^{-1}(\mathbf{R}^-), \quad (6)$$

$$B = \{\mathbf{x} \in R^n \mid \mathbf{g}(\mathbf{x}) \in \mathbf{R}^- \} = \mathbf{g}^{-1}(\mathbf{R}^-).$$

A possible approach that could be considered for proving that $A \subset B$ is to bracket A and B between two subpavings (i.e., union of boxes), using set-inversion approaches (Moore, 1992; Jaulin and Walter, 1993). Proving the inclusion then becomes trivial. We shall here consider a more efficient approach that avoid, an unnecessary in-depth characterization of sets A and B .

A vector \mathbf{x} is said to be *bad* if $\mathbf{x} \in A$ and $\mathbf{x} \notin B$. If a bad vector exists, $A \not\subset B$. A vector that is not bad is said to be *good*. A box \mathbf{X} is *bad* if it contains at least one bad vector, otherwise \mathbf{X} is *good*. Note that if $\mathbf{X} \subset B$, then \mathbf{X} is good because it cannot contain one \mathbf{x} such that $\mathbf{x} \notin B$. In the same manner, if $\mathbf{X} \cap A = \emptyset$, then \mathbf{X} cannot contain one \mathbf{x} in A and therefore \mathbf{X} is good. The principle of the algorithm to be presented is to partition the prior box of interest \mathbf{X}_0 into a set of nonoverlapping boxes K , such that any box \mathbf{X} in K is good. If it succeeds in performing such a partition, and since A and $B \subset \mathbf{X}_0$, then $A \subset B$.

Two convergent inclusion functions \mathbf{F} and \mathbf{G} for \mathbf{f} and \mathbf{g} , are assumed to be available. The following theorem provides three different basic tests for deciding whether a given box \mathbf{X} is good or bad.

Theorem 1. *If \mathbf{X} is a box of \mathbf{R}^n the center of which is denoted by $\bar{\mathbf{x}}$, then*

- (i) $\mathbf{G}(\mathbf{X}) \subset \mathbf{R}^- \Rightarrow \mathbf{X}$ is good,
- (ii) $\mathbf{F}(\mathbf{X}) \cap \mathbf{R}^- = \emptyset \Rightarrow \mathbf{X}$ is good,
- (iii) $\mathbf{f}(\bar{\mathbf{x}}) \in \mathbf{R}^-$ and $\mathbf{g}(\bar{\mathbf{x}}) \notin \mathbf{R}^- \Rightarrow \mathbf{X}$ is bad

Proof. (i) If $\mathbf{G}(\mathbf{X}) \subset \mathbf{R}^-$, then, $\mathbf{g}(\mathbf{X}) \subset \mathbf{R}^-$. Therefore, $\mathbf{g}^{-1}(\mathbf{g}(\mathbf{X})) \subset \mathbf{g}^{-1}(\mathbf{R}^-)$. Now, $\mathbf{X} \subset \mathbf{g}^{-1}(\mathbf{g}(\mathbf{X})) \Rightarrow \mathbf{X} \subset \mathbf{g}^{-1}(\mathbf{R}^-)$ i.e. $\mathbf{X} \subset B$. Hence, \mathbf{X} is good. (ii) If $\mathbf{F}(\mathbf{X}) \cap \mathbf{R}^- = \emptyset$, $\mathbf{f}(\mathbf{X}) \cap \mathbf{R}^- = \emptyset$. Now, $A = \mathbf{f}^{-1}(\mathbf{R}^-)$, therefore $\mathbf{f}(A) \subset \mathbf{R}^-$. Consequently, $\mathbf{f}(\mathbf{X}) \cap \mathbf{f}(A) = \emptyset$, i.e. $\mathbf{X} \cap A = \emptyset$. Hence, \mathbf{X} is good. (iii) If $\mathbf{f}(\bar{\mathbf{x}}) \in \mathbf{R}^-$ and $\mathbf{g}(\bar{\mathbf{x}}) \notin \mathbf{R}^-$, then $\bar{\mathbf{x}} \in A$ and $\bar{\mathbf{x}} \notin B$. Therefore $\bar{\mathbf{x}}$ is bad. Hence \mathbf{X} is bad. \square

In the algorithm to be presented, Q is a queue of boxes, i.e. it has a first-in-first-out (FIFO) structure. It contains all boxes still to be studied. The initial box \mathbf{X}_0 is assumed to enclose the two compact sets A and B .

Step 0: $\mathbf{X} := \mathbf{X}_0$; $Q := \emptyset$.

Step 1: If $\mathbf{G}(\mathbf{X}) \subset \mathbf{R}^-$, go to Step 6.

Step 2: If $\mathbf{F}(\mathbf{X}) \cap \mathbf{R}^- = \emptyset$, go to Step 6.

Step 3: $\bar{\mathbf{x}} = \text{center}(\mathbf{X})$.

Step 4: If $\mathbf{f}(\bar{\mathbf{x}}) \in \mathbf{R}^-$ and $\mathbf{g}(\bar{\mathbf{x}}) \notin \mathbf{R}^-$, return “ $A \not\subset B$ ”; END.

Step 5: Bisect \mathbf{X} and push the two resulting boxes at the end of Q .

Step 6: If $Q \neq \emptyset$, pull the first box of Q into \mathbf{X} and go to Step 1.

Step 7: Return “ $A \subset B$ ”. END.

A box \mathbf{X} such that neither of the three conditions (i)–(iii) are satisfied is said to be *indeterminate*. Since the three basic tests are pessimistic, an indeterminate box might be good or bad. Note that an indeterminate box is necessarily bisected by the algorithm.

4. Convergence analysis

If we assume that the algorithm ends, then two conclusions may be reached. If it ends at Step 7, then the initial box \mathbf{X}_0 has been partitioned into a set K of non-overlapping good boxes, and therefore $A \subset B$. If the algorithm stops at Step 4, then a point \mathbf{x} is found such that $\bar{\mathbf{x}} \in A$, and $\bar{\mathbf{x}} \notin B$. Thus $A \not\subset B$.

Let $C \neq \{\mathbf{x} \in \mathbf{R}^n | \mathbf{h}(\mathbf{x}) \leq \mathbf{0}\}$ where \mathbf{h} is assumed to be continuous. The border C^0 of C is the set defined by

$$C^0 = \{\mathbf{x} \in C | \exists i \in \{1, \dots, \dim \mathbf{h}\} | h_i(\mathbf{x}) = 0\}. \quad (8)$$

If ∂C denotes the boundary of C , then $\partial C \subset C^0$ and, in generic situations, $\partial C = C^0$. For $h(\mathbf{x}) = (x_1^2 + x_2^2)(x_1^2 + x_2^2 - 1)$ where $\dim h = 1$, we have $\mathbf{0} \in C^0$ but $\mathbf{0} \notin \partial C$. It is therefore an example of a degenerated situation. A vector that belongs to C^0 , even if it is deep inside C , can come out of C if \mathbf{h} is infinitesimally moved.

Theorem 2. *Let A^0 and B^0 be the borders of A and B . If $A \subset B$ and $A \cap B^0 = \emptyset$, the algorithm proves the inclusion in a finite time.*

Proof. The proof is by contradiction. If the algorithm never stops, it generates a nested sub-sequence of indeterminate boxes $\mathbf{X}(k)$ that accumulates over a point $\hat{\mathbf{x}}$. Since $\mathbf{X}(k)$ is bisected, it does not satisfy any of the conditions required by Steps 1, 2 and 4 i.e. $\mathbf{X}(k)$ satisfies the three following conditions:

$$\mathbf{G}(\mathbf{X}(k)) \not\subset \mathbf{R}^-$$

$$\mathbf{F}(\mathbf{X}(k)) \cap \mathbf{R}^- \neq \emptyset, \quad (9)$$

$$(\mathbf{f}(\bar{\mathbf{x}}(k)) \notin \mathbf{R}^-) \text{ or } (\mathbf{g}(\bar{\mathbf{x}}(k)) \in \mathbf{R}^-).$$

Moreover, since (i) $w(\mathbf{X}(k)) \rightarrow 0$, (ii) $\forall k, \hat{\mathbf{x}} \in \mathbf{X}(k)$ and (iii) \mathbf{F} and \mathbf{G} are convergent, the accumulation point $\hat{\mathbf{x}}$ satisfies:

$$\mathbf{g}(\hat{\mathbf{x}}) \notin \mathbf{R}^{-*},$$

$$\mathbf{f}(\hat{\mathbf{x}}) \in \mathbf{R}^-, \quad (10)$$

$$\mathbf{f}(\hat{\mathbf{x}}) \notin \mathbf{R}^{-*} \text{ or } \mathbf{g}(\hat{\mathbf{x}}) \in \mathbf{R}^-.$$

Let us denote by \bar{A} and \bar{B} the complementary sets of A and B . Since (i) $\mathbf{g}(\hat{\mathbf{x}}) \notin \mathbf{R}^{-*} \Leftrightarrow \hat{\mathbf{x}} \in \bar{B} \cup B^0$, (ii) $\mathbf{f}(\hat{\mathbf{x}}) \in \mathbf{R}^- \Leftrightarrow \hat{\mathbf{x}} \in A$, (iii) $\mathbf{f}(\hat{\mathbf{x}}) \notin \mathbf{R}^{-*} \Leftrightarrow \hat{\mathbf{x}} \in \bar{A} \cup A^0$, and (iv) $\mathbf{g}(\hat{\mathbf{x}}) \in \mathbf{R}^- \Leftrightarrow \hat{\mathbf{x}} \in B$, Eq. (10) is equivalent to $\hat{\mathbf{x}} \in D$, where

$$D = (\bar{B} \cup B^0) \cap A \cap (\bar{A} \cup A^0 \cup B). \quad (11)$$

Using Boolean notation, $D = (\bar{B} + B^0)A(\bar{A} + A^0 + B) = (\bar{B}A + B^0A)(\bar{A} + A^0 + B) = \bar{B}A\bar{A} + \bar{B}AA^0 + \bar{B}AB + B^0A\bar{A} + B^0AA^0 + B^0AB$. Since $A\bar{A} = \emptyset$, $AA^0 = A^0$, $B\bar{B} = \emptyset$, $BB^0 = B^0$, and $A^0 + A = A$, we get $D = A^0\bar{B} + AB^0$. Now, $A \subset B$ and $A \cap B^0 = \emptyset$. Therefore, D is empty and the accumulation point $\hat{\mathbf{x}}$ cannot exist. The algorithm is thus a finite algorithm. \square

Theorem 3. *If $\exists \mathbf{x} | \mathbf{x} \in A - A^0$ and $\mathbf{x} \notin B$ the algorithm proves that $A \not\subset B$ in a finite time.*

Proof. Since $\mathbf{x} \in A - A^0$ and $\mathbf{x} \notin B$, we have

$$\mathbf{f}(\mathbf{x}) \in \mathbf{R}^{-*} \text{ and } \mathbf{g}(\mathbf{x}) \notin \mathbf{R}^-. \quad (12)$$

Note that any box that contains \mathbf{x} cannot be eliminated via Step 1 or Step 2. If the algorithm never stops, the

algorithm generates a nested subsequence $\mathbf{X}(k)$ of indeterminate boxes that contain \mathbf{x} . Thus, the sequence $\bar{\mathbf{x}}(k) = \text{center}(\mathbf{X}(k))$ converges to \mathbf{x} . Since \mathbf{f} and \mathbf{g} are continuous, $\mathbf{f}(\bar{\mathbf{x}}(k)) \rightarrow \mathbf{f}(\mathbf{x})$ and $\mathbf{g}(\bar{\mathbf{x}}(k)) \rightarrow \mathbf{g}(\mathbf{x})$. From (12), there exists an integer k_0 such that $\mathbf{f}(\bar{\mathbf{x}}(k_0)) \in \mathbf{R}^-$ and $\mathbf{g}(\bar{\mathbf{x}}(k_0)) \notin \mathbf{R}^-$. For such a k_0 , at Step 4, the algorithm returns “ $A \not\subset B$ ”. This is in contradiction to the fact that the algorithm never stops. \square

5. Test case

Test case 1: Consider the discrete-time model

$$y(i) = -\frac{1}{p_1}(y(i-1) + p_2y(i-2)) + 2y(i-3) + y(i-4), \tag{13}$$

where $\mathbf{p} = (p_1, p_2)^T$ is the parameter vector. This model is taken from Exercise 16.33 of Rivoire et al. (1992) related to Jury criterion. The initial conditions are given by

$$y(-3) = y(-2) = y(-1) = y(0) = 1. \tag{14}$$

Moreover, five experimental data have been collected on the system at times $i \in \{1, 2, 3, 4, 5\}$. The associated data vector is

$$\mathbf{y}^\# = (-2, -0.8, -0.08, 1.55, 0.85)^T. \tag{15}$$

It has been obtained by simulation of Eq. (13) for $p_1 = 2.5$ and $p_2 = 1$. Let us denote by $\mathbf{y}^m(\mathbf{p})$ the vector of all model

outputs homogeneous to the data vector $\mathbf{y}^\#$. To be feasible, \mathbf{p} should satisfy $|y_i^m(\mathbf{p}) - y_i^\#| \leq e_{\max}, i \in \{1, 2, 3, 4, 5\}$. The feasible set $A(e_{\max})$ is then defined by a set of nonlinear inequalities $\mathbf{f}(\mathbf{p}) \leq \mathbf{0}$ where its i th component is $f_i(\mathbf{p}) = |y_i^m(\mathbf{p}) - y_i^\#| - e_{\max}$. The Jury criterion, a formal procedure that transforms a problem of stability for a linear discrete-time system into a set of nonlinear inequalities, is now applied to the model (13). We obtain that necessary and sufficient conditions for stability are (Rivoire et al., 1992):

$$\begin{pmatrix} 2 - p_1 - p_2 \\ 1 - |p_1| \\ |1 - 2p_1| - |1 - p_1^2| \\ |p_2(1 - p_1)(1 - p_1^2) - (1 - 2p_1)(2 - p_1)| - |(1 - p_1^2)^2 - (1 - 2p_1)^2| \end{pmatrix} \leq \mathbf{0}.$$

Over the box $\mathbf{P}_0 = [10^{-8}, 10^8] \times [-10^8, 10^8]$, for $e_{\max} = 0.302$, in 7.2 s on a DX4-100 computer, the algorithm proves that all feasible models are stable, i.e. it proves that $A(0.302) \subset B$. 2455 good boxes have been generated. For $e_{\max} = 0.303$, in 7.3 s, the algorithm proves that $A(0.303) \not\subset B$. The model associated with $\mathbf{p} = (2.305 \ 1.188)^T$ is found to be feasible and unstable. The generated paving around \mathbf{p} is represented in Fig. 1. Dark gray boxes are proved to be unfeasible and light gray boxes are proved to be stable. The algorithm finds a bad box, drawn in white. In the black region are located all boxes that have not been studied, but whatever happen in this region cannot change the conclusion.

Assume now that 20 measurements have been generated (instead of 5). The algorithm proves that $A_{20}(0.302) \subset B$ in

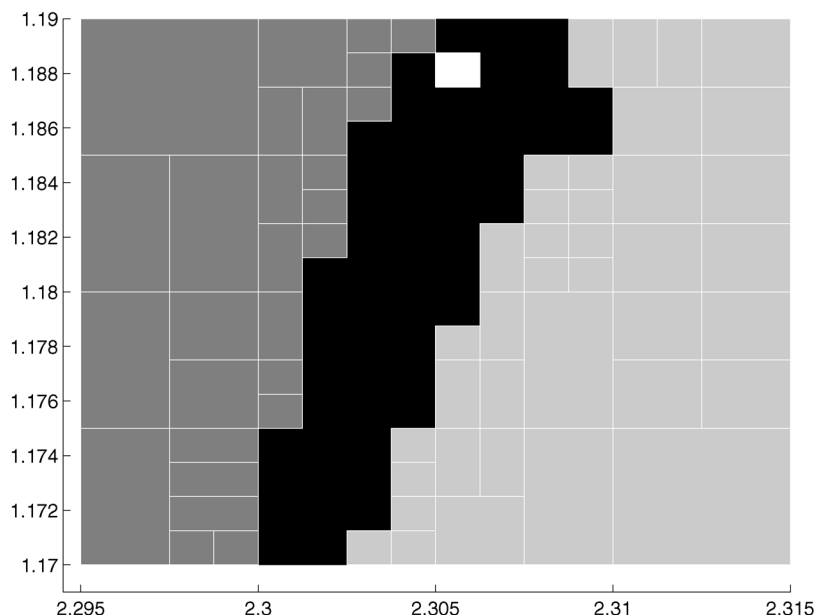


Fig. 1. Paving generated, for $e_{\max} = 0.303$, in the (p_1, p_2) -space, for proving that the model may be unstable.

6.5 s (the subscript for A indicates the number of measurements) instead of 7.2 s. The number of boxes is now reduced to 999: more informations are available for p , i.e. $A_{20} \subset A_5$, and it is much more easy for our algorithm to prove that $A \subset B$. Note that since the number of measurements is bigger, the computing time for one box has increased.

Test case 2: The aim of this test case is to show the behavior of the algorithm with respect to the dimension of the parameter space. Consider a continuous-time model, the characteristic polynomial of which is given by

$$P(\mathbf{p}, s) = s^n + p_n s^{n-1} + \dots + p_2 s + p_1. \quad (16)$$

The nominal value $\hat{\mathbf{p}}$ for \mathbf{p} is obtained by setting $P(\hat{\mathbf{p}}, s) = (s + 1)^n$. For example, if $n = 3$, $\hat{\mathbf{p}} = (3, 3, 1)^T$. Let us assume that the parameter vector \mathbf{p} of the model is known to satisfy

$$\|\mathbf{p} - \hat{\mathbf{p}}\| - 0.2 \leq 0 \quad (17)$$

The uncertain model $\mathcal{M}(\mathbf{p})$ is guaranteed to be stable if the set A defined by the inequality (17) is included in the set B defined by inequalities derived from (16) by using the Routh criterion. On a DX4-100 computer, the algorithm proved the inclusion for $n = 1-8$. The computing times are given by Table 1.

6. Conclusion

The information available on an uncertain parametric model, as well as its stability domain, can generally be described by nonlinear inequalities that should be satisfied by the parameter vector. The problem of proving the stability of the uncertain model amounts to prove that the feasible set is included in the stability domain. Interval analysis has been used to develop a new algorithm able to prove the inclusion. Its convergence properties have been provided and it has been shown that the algorithm is finite, except for degenerated cases. To illustrate the efficiency of the method, two test cases have been solved. The first one deals with a discrete-time parametric model where the information for the parameters is available under the form of bounded-error data. Discrete-time models involve stability domains defined by nonsmooth inequalities and the feasible set generated by a bounded-error approach is also defined by nonlinear

inequalities. This test case is just an illustration of the large class of parametric robust stability problems that can be handled by the method. To the best of our knowledge, no others methods in the literature are able to solve this test case in a guaranteed way. The second one illustrates the behavior of our algorithm when the number of parameters increases. As expected, the complexity seems to be exponential with respect to the number of parameters. The approach considered in this paper seems therefore to be restricted to problems with reasonable dimensions.

References

- Barlett, A. C., Hollot, C. V., & Huang, L. (1988). Root locations of an entire polytope of polynomial: It suffices to check the edges. *Math. Control, Signals Systems, 1*, 61–71.
- Barmish, B. R., & Kang, H. I. (1993). A survey of extreme point results for robustness control systems. *Automatica, 29*, 13–35.
- Barmish, B. R. (1994). *New tools for robustness of linear systems*. New York: Macmillan Publishing Co.
- Bhattacharyya, S. P., Chapellat, H., & Keel, L. H. (1995). *Robust control, the parametric approach*, Englewood Cliffs, NJ: Prentice-Hall.
- Hammer, R., Hocks, M., Kulisch, U., & Ratz, D. (1995). *C++ Toolbox for verified computing*. Heidelberg: Springer.
- Hansen, E. (1992). *Global optimization using interval analysis*, New York: Marcel Dekker.
- Jaulin, L., & Walter, E. (1993). Set inversion via interval analysis, *Automatica, 29*(4), 1053–1064.
- Jaulin, L., & Walter, E. (1996). Guaranteed tuning, with application to robust control and motion planning. *Automatica, 32*(8), 1217–1221.
- Kharitonov, V. L. (1978). Asymptotic stability of an equilibrium position of a family of systems of linear differential equations. *Differentsial'nye Uravneniya, 14*, 2086–2088.
- Kolev, L. V. (1988). An interval first-order method for robustness analysis. *IEEE Int. Sym. Circuits and Systems*. (pp. 2522–2524). Chicago.
- Malan, M., Milanese, M., & Taragna, M. (1996). Robust stability and design of control systems using interval arithmetic. *Proc. IFAC 13th Triennial World Congress* (pp. 25–30), San Francisco.
- Milanese, M., Norton, J., Piet-Lahanier, H., & Walter, E. (Eds.) (1996). *Bounding approaches to system identification*. New York: Plenum.
- Moore, R. E. (1979). *Methods and applications of interval analysis*. Philadelphia: SIAM.
- Moore, R. E. (1992). Parameter set for bounded-error data. *Math. Comput. Simulation, 34*, 113–119.
- Norton, J.-P. (Ed.) (1994). Special issue on bounded-error estimation, 1. *Int. J. Adapt. Control Signal Process* 8(1), 1–118.
- Norton, J.-P. (Ed.) (1995). Special issue on bounded-error estimation, 2. *Int. J. Adapt. Control Signal Process*, 9(1), 1–132.
- Rivoire, M., Ferrier, J. L., & Groleau, J. (1992). *Exercices d'automatique, commande par calculateur et identification*, Tome 3, Paris: Eyrolles.
- Vicino, A., Tesi, A., & Milanese, M. (1990). Computation of nonconservative stability perturbation bounds for systems with nonlinearly correlated uncertainties. *IEEE Trans. Automat. Control, AC-35*, 835–841.
- Walter, E. (Ed.) (1990). Special issue on parameter identification with error bound. *Math. Comput. Simulation, 32*(5&6), 447–607.
- Walter, E., & Jaulin, L., (1994). Guaranteed characterization of stability domains via set inversion. *IEEE Trans. on Automat. Control, AC-39*, 886–889.

Table 1
Computing time with respect to the dimension in Test-case 2

n	1	2	3	4	5	6	7	8
t (s)	0.02	0.05	0.1	0.2	0.2	0.4	0.7	1.1



Luc Jaulin received the Ph.D. degree in automatic control from the University of Orsay, France in 1993. He is currently Assistant Professor of physics at the University of Angers, France since 1993. He does his research at the *Laboratoire d'Ingénierie des Systèmes Automatisés* (LISA) on robust estimation and control using interval methods.



Jacques Burger is a Professor at the Faculty of Sciences at the University of Angers. He is in charge of the scientific computation module of the department of physics. He does his research works at the *Laboratoire d'Ingénierie des Systèmes Automatisés*, where he supervises the research group on the control of hybrid dynamical systems.