

# Guaranteed estimation of the parameters of nonlinear continuous-time models: Contributions of interval analysis

M. Kieffer<sup>1,2,\*</sup>, † and E. Walter<sup>1</sup>

<sup>1</sup>*Laboratoire des Signaux et Systèmes, CNRS—SUPELEC, Univ Paris-Sud, 3 rue Joliot-Curie, 91192 Gif-sur-Yvette Cedex, France*

<sup>2</sup>*On sabbatical leave at TELECOM ParisTech, CNRS-LTCl, Signal and Image Processing Department, 46 rue Barrault, 75634 Paris, Cedex 13, France*

## SUMMARY

This paper is about guaranteed parameter estimation in two contexts, namely bounded-error and optimal estimation. In bounded-error estimation, one looks for the set of all parameter vectors that are consistent with some prior bounds on the errors deemed acceptable between the model behavior and that of the system. In optimal estimation, one looks for the set of all parameter vectors that minimize some cost function quantifying the discrepancy between the behaviors of the system and its model. In both cases, guaranteed means that *proven* statements are made about the set of interest. The situation is made much more difficult when the model output is assumed to depend nonlinearly in the parameters to be estimated and when dealing with continuous-time models, as here. Important tools based on interval analysis (IA) that contribute to allowing guaranteed estimation in these challenging conditions are presented. Some are absolutely classical in the context of IA but not so well known in the community of parameter estimation at large. Others have been developed recently and were mainly presented in conferences. Some, such as the use of sensitivity functions to reduce more quickly the size of outer approximations of the sets of interest, are new. Challenges for future research in the context of guaranteed nonlinear estimation are mentioned. Copyright © 2010 John Wiley & Sons, Ltd.

Received 24 November 2009; Revised 11 March 2010; Accepted 7 June 2010

**KEY WORDS:** bounded-error estimation; guaranteed estimation; interval analysis; parameter bounding; parameter optimization; sensitivity functions

## 1. INTRODUCTION

Parameter estimation is classically conducted by minimizing some cost function that may be purely heuristic or deduced from information (or hypotheses) about the noise corrupting the data and possibly about the prior distribution for the parameter vector.

Bounded-error parameter estimation, by contrast, does not use a cost function and aims instead at characterizing the set of all values of the parameter vector that are consistent with bounds on the errors one is prepared to accept between the behavior of the model and that of the system to be modeled. It can be traced back to the seminal work of F.C. Schweppe [1, 2] on state estimation for linear models. His idea was recursive to compute ellipsoids *guaranteed* to contain the actual state of the system provided that some bounds on the state perturbations and measurement noise

\*Correspondence to: M. Kieffer, TELECOM ParisTech, CNRS-LTCl, Signal and Image Processing Department, 46 rue Barrault, 75634 Paris, Cedex 13 on sabbatical leave from the L2S—CNRS—SUPELEC, Univ Paris-Sud, France.

†E-mail: kieffer@lss.supelec.fr

were satisfied. This type of parameter or state bounding based on a description of uncertainty by sets is often referred to as *guaranteed estimation*, see, e.g. [3] and the books [4, 5]. We shall see in this paper that *guaranteed* can be given a broader meaning in our context. Of course, other types of containers than ellipsoids could and have been used, such as boxes [6], parallelotopes [7], zonotopes [8] or other limited-complexity polytopes. Matlab toolboxes implementing ellipsoidal or polytopic calculus are readily available [9, 10].

Parameter or state bounding when the model output depends nonlinearly on the quantities to be estimated is of course much more difficult, and nonlinear ellipsoidal calculus remains a largely open subject. A generalization of the ellipsoidal approach to nonlinear state estimation is briefly considered in [4]. It is based on the use of auxiliary (time-varying) linear state equations on which ellipsoidal calculus can be applied. Outer (or inner) approximations of the set of feasible state vectors at any given instant of time can then be computed under the form of ellipsoids using the same methodology as for linear models, provided suitable inclusion relations between the nonlinear and auxiliary models can be established. Along a similar line, one may also try to consider a nonlinear model as a linear one with uncertain regressors, where the uncertainty is described by intervals, and use techniques such as described in [11]. In both cases, the nonlinear problem is transformed into a linear uncertain one to which linear techniques can be applied. Linear models with time-varying coefficients could also be considered, as in [12].

Instead of attempting linearization, one may use the tools provided by interval analysis (IA) directly to compute an approximate but guaranteed characterization of sets in the nonlinear case. In the context of bounded-error parameter estimation, this approach made it possible to derive algorithms that could be used to bracket the set of possible values for the parameter vector between computable unions of boxes in parameter space [13]. At the beginning, these algorithms were limited to models for which a closed-form expression of the model output as a function of the parameters was available, but the concept of guaranteed integration [14–16] made it applicable to models for which no such expression exists [17–19]. This paper will not describe guaranteed integration in any detail, but it may be useful to mention that two approaches at least could be considered. The use of high-order Taylor models is considered in [14–16], while W. Kühn uses a set-valued dynamical system approach based on the description of sets by zonotopes, where again IA plays an important role [20].

The success of IA as a tool to compute guaranteed set estimates in the context of bounded-error estimation seems to have somewhat obscured the fact that the very same tools can also be used to obtain guaranteed estimates in the more traditional approach where a cost function is to be optimized.

Besides presenting our most recent results on the use of IA and guaranteed integration for bounded-error nonlinear parameter estimation in the context of continuous-time models, we would therefore like to show how these tools can also contribute to the more mainstream parameter estimation through optimization. We shall also explain why this is actually more difficult, and point out challenges for future research.

## 2. NOTATION AND PROBLEM STATEMENT

We assume that the model consists of a set of ordinary differential equations in state-space form, involving some vector  $\mathbf{p}$  of parameters to be estimated

$$\begin{aligned}\mathbf{x}'(t) &= \mathbf{f}(\mathbf{x}(t), \mathbf{p}, t), \\ \mathbf{y}_m(\mathbf{p}, t) &= \mathbf{h}(\mathbf{x}(t), \mathbf{p}).\end{aligned}\tag{1}$$

$\mathbf{x}'(t)$  is the derivative of  $\mathbf{x}$  with respect to time and the dependency of  $\mathbf{f}$  in  $t$  may result from the influence of a known, time-varying vector  $\mathbf{u}$  of external inputs. For the sake of simplicity, the initial state  $\mathbf{x}(t_0)$  is assumed here to be known, but it could also be parametrized. With such a model,  $\mathbf{y}_m(\mathbf{p}, t)$ , as computed from (1), is almost always *nonlinear in the parameters*  $\mathbf{p}$ , even when it is linear in some input.

Let  $\mathbf{y}(t_i)$  be the vector of the measurements collected on the system at the instants of time  $t_i$ ,  $i = 1, \dots, n_t$ , and let  $\mathbf{y}$  be the vector obtained by concatenating all  $\mathbf{y}(t_i)$ s

$$\mathbf{y} = (\mathbf{y}^T(t_1), \dots, \mathbf{y}^T(t_{n_t}))^T. \tag{2}$$

Two types of estimation problems will be considered, namely those based on the minimization of some cost function and those involving the characterization of a set of parameter vectors that satisfy error bounds. We shall now state these problems more precisely and explain what we mean by guaranteed in both cases.

2.1. *Guaranteed estimation via optimization*

The most classical statistical estimation methods involve the *optimization of a cost function*  $c(\mathbf{p})$ , which may be deduced from probabilistic assumptions on the noise and parameters, see, e.g. [21, 22]. For example, if one assumes that there exists a *true* value  $\mathbf{p}^*$  of the parameter vector and that the measurements satisfy

$$\mathbf{y} = \mathbf{y}_m(\mathbf{p}^*) + \mathbf{e}, \tag{3}$$

with  $\mathbf{y}_m(\mathbf{p})$  consisting of subvectors  $\mathbf{y}_m(\mathbf{p}, t_i)$  and  $\mathbf{e}$  a zero-mean white Gaussian measurement noise, then the maximum-likelihood estimate of  $\mathbf{p}$  is the argument of the minimum of the cost function

$$c(\mathbf{p}) = (\mathbf{y} - \mathbf{y}_m(\mathbf{p}))^T (\mathbf{y} - \mathbf{y}_m(\mathbf{p})). \tag{4}$$

When  $\mathbf{y}_m(\mathbf{p})$  is linear in  $\mathbf{p}$ , it is easy, at least in principle, to check whether the experimental conditions are such that the global minimizer of (4) is unique, and to compute it when this is the case. When  $\mathbf{y}_m(\mathbf{p})$  is nonlinear in  $\mathbf{p}$ , as is most often the case with models such as (1), optimization becomes much more difficult. The usual local methods (such as quasi-Newton or conjugate gradient algorithms) converge at best to a local minimizer of the cost function. Global optimization based on random search (for instance simulated annealing or genetic algorithms) cannot be guaranteed to locate global minimizers in finite time, so that, most often, no guarantee at all is provided about the estimate. Often, we do not even know how many global minimizers there are.

In this context, guaranteed estimation aims at deriving *proven* statements about the global minimum of the cost function and the associated set of global minimizers (which is also the *set* of optimal estimates).

2.2. *Guaranteed estimation via parameter bounding*

An alternative parameter estimation approach is *parameter bounding*, see [23–26] and the references therein. In parameter bounding, one looks for the *set* of *all* parameter vectors that are consistent with the experimental data, the structure of the model, and the bounds on the errors one is prepared to accept. With each experimental data vector  $\mathbf{y}(t_i)$ , we assume that we can associate some known vector of intervals (or box)  $[\underline{\mathbf{e}}_i, \bar{\mathbf{e}}_i]$  of acceptable errors. The parameter vector  $\mathbf{p}$ , assumed to belong to some known prior box  $[\mathbf{p}]_0$ , is deemed *acceptable* if

$$\underline{\mathbf{e}}_i \leq \mathbf{y}(t_i) - \mathbf{y}_m(t_i, \mathbf{p}) \leq \bar{\mathbf{e}}_i \quad \text{for all } i = 1, \dots, n_t. \tag{5}$$

Parameter estimation then amounts to characterizing the set

$$\begin{aligned} \mathbb{S} &= \{\mathbf{p} \in [\mathbf{p}]_0 \mid \mathbf{p} \text{ is acceptable}\} \\ &= \{\mathbf{p} \in [\mathbf{p}]_0 \mid \mathbf{y}_m(\mathbf{p}) \in [\mathbf{y}]\}, \end{aligned} \tag{6}$$

with

$$[\mathbf{y}] = \begin{pmatrix} [\mathbf{y}(t_1) - \bar{\mathbf{e}}_1, \mathbf{y}(t_1) - \underline{\mathbf{e}}_1] \\ \vdots \\ [\mathbf{y}(t_{n_t}) - \bar{\mathbf{e}}_{n_t}, \mathbf{y}(t_{n_t}) - \underline{\mathbf{e}}_{n_t}] \end{pmatrix} \tag{7}$$

and

$$\mathbf{y}_m(\mathbf{p}) = (\mathbf{y}_m^T(t_1, \mathbf{p}), \dots, \mathbf{y}_m^T(t_{n_t}, \mathbf{p}))^T. \quad (8)$$

In this context, guaranteed estimation aims at deriving *proven* statements about  $\mathbb{S}$ , such as  $\mathbb{S}$  is *void* or  $\mathbb{S}$  is *included in some computed set*  $\tilde{\mathbb{S}}$ . When  $\mathbf{y}_m(\mathbf{p})$  is linear in  $\mathbf{p}$ , a number of methods have been proposed to compute sets guaranteed to contain  $\mathbb{S}$ , which is then a convex polytope. When the model is nonlinear in  $\mathbf{p}$ , as assumed here,  $\mathbb{S}$  may consist of disconnected and nonconvex subsets, which makes the situation much more difficult to handle.

### 2.3. Aims of the paper

This paper is aimed at presenting important tools based on IA and guaranteed integration that contribute to allowing guaranteed estimation in these challenging conditions. Some are absolutely classical in the community of interval analysts but not so well known in the control and statistics communities. Others have been developed recently and were mainly presented in conferences. Some, such as the use of sensitivity functions to reduce more quickly the size of outer approximations of the sets of interest, are new. A seemingly simple but nevertheless challenging example of nonlinear parameter estimation problem will be treated with the optimal and bounding approaches, and will be used to demonstrate the interest of the new approach for set reduction based on the use of sensitivity functions.

## 3. TOOLS FOR GUARANTEED ESTIMATION

In this section, we briefly recall the three main tools to be used to make estimation guaranteed in either of the senses defined above. IA, which is at the core of the other two, is presented first. We then proceed to guaranteed global optimization and guaranteed set characterization.

Specific tools, such as guaranteed integration and sensitivity functions, which are useful to consider the estimation of systems described by continuous-time state-space models, will be introduced in Section 4.

### 3.1. Interval analysis

IA [27–29] provides tools to compute with vectors of intervals (more concisely, *boxes*)  $[\mathbf{x}] = [\underline{\mathbf{x}}, \bar{\mathbf{x}}] \subset \mathbb{R}^n$ . For any function  $\mathbf{f}: \mathcal{D} \subset \mathbb{R}^n \rightarrow \mathbb{R}^m$  defined by combining arithmetical operators and elementary functions, IA makes it possible to build *inclusion functions*  $[\mathbf{f}]$  satisfying

$$\forall [\mathbf{x}] \subset \mathcal{D}, \quad \mathbf{f}([\mathbf{x}]) \subset [\mathbf{f}]([\mathbf{x}]), \quad (9)$$

where  $[\mathbf{f}]([\mathbf{x}])$  is a box.

*Natural inclusion functions*, for example, are obtained by replacing all real variables, operators and elementary functions by their interval counterparts. It thus becomes possible to enclose the set  $\mathbf{f}([\mathbf{x}])$ , which most often cannot be computed, in a computable box  $[\mathbf{f}]([\mathbf{x}])$ .

If  $\mathbf{f}$  is differentiable over  $[\mathbf{x}]$ , the mean-value theorem implies that

$$\forall \mathbf{x} \in [\mathbf{x}], \exists \mathbf{z} \in [\mathbf{x}] \text{ such that } \mathbf{f}(\mathbf{x}) = \mathbf{f}(\mathbf{m}) + \mathbf{J}(\mathbf{z}) \cdot (\mathbf{x} - \mathbf{m}),$$

where  $\mathbf{J}$  is the Jacobian matrix of  $\mathbf{f}$  and  $\mathbf{m} = \text{mid}([\mathbf{x}])$  is the center of  $\mathbf{x}$ . Thus,

$$\forall \mathbf{x} \in [\mathbf{x}], \mathbf{f}(\mathbf{x}) \in \mathbf{f}(\mathbf{m}) + [\mathbf{J}]([\mathbf{x}]) \cdot (\mathbf{x} - \mathbf{m}),$$

where  $[\mathbf{J}]$  is an inclusion function for  $\mathbf{J}$ . So

$$\mathbf{f}([\mathbf{x}]) \subset \mathbf{f}(\mathbf{m}) + [\mathbf{J}]([\mathbf{x}]) \cdot ([\mathbf{x}] - \mathbf{m}).$$

Therefore, the interval function

$$[\mathbf{f}_c]([\mathbf{x}]) = \mathbf{f}(\mathbf{m}) + [\mathbf{J}]([\mathbf{x}]) \cdot ([\mathbf{x}] - \mathbf{m}) \quad (10)$$

is another inclusion function for  $\mathbf{f}$ , called its *centered inclusion function*. This function becomes especially interesting when the width of  $[\mathbf{x}]$  is small, because the overestimation resulting from the interval evaluation of  $[\mathbf{J}](\mathbf{x})$  is reduced by the product with  $[\mathbf{x}] - \mathbf{m}$ , which is a small box centered on zero. The centered inclusion function can be significantly improved at the cost of a slightly more complicated formulation [30].

### 3.2. Guaranteed global optimization

Let  $\mathbb{P} \subset \mathbb{R}^{\dim \mathbf{p}}$  be a (possibly very large but compact) domain of interest in parameter space. For simplicity, here  $\mathbb{P}$  will be a box  $[\mathbf{p}]_0$ . We wish to characterize the set  $\mathbb{S}$  of all the global minimizers of the cost function that are inside  $\mathbb{P}$

$$\mathbb{S} = \arg \min_{\mathbf{p} \in \mathbb{P}} c(\mathbf{p}). \tag{11}$$

These minimizers are not expected to lie on the boundary of  $\mathbb{P}$ , so this is an *unconstrained* minimization. We now present the principles of a simplified version of Hansen’s algorithm, see [29, 31] for more details.

Let  $\mathcal{L}$  be a list of boxes, the union of which contains all global minimizers belonging to  $\mathbb{P}$ . Initially,  $\mathcal{L}$  contains only  $[\mathbf{p}]_0$ . The basic structure of the algorithm is as follows, where  $\varepsilon > 0$  is a tolerance parameter to be tuned by the user.

1. While  $\mathcal{L}$  contains at least one box with width greater than  $\varepsilon$ , do {
  - (a) Pop a box  $[\mathbf{p}]$  with width greater than  $\varepsilon$  out of  $\mathcal{L}$ ,
  - (b) Update the upper bound  $\text{ub}(\hat{c})$  of the value of the global minimum  $\hat{c}$  over  $[\mathbf{p}]_0$ .
  - (c) Try to eliminate  $[\mathbf{p}]$  via the midpoint and monotonicity tests.
  - (d) Should this fail, try to eliminate or at least reduce  $[\mathbf{p}]$  using a contractor.
  - (e) If  $[\mathbf{p}]$  resists elimination and the width of  $[\mathbf{p}]$  is greater than  $\varepsilon$ , then bisect  $[\mathbf{p}]$  and push the two resulting boxes into  $\mathcal{L}$ . Otherwise push  $[\mathbf{p}]$  back as is into  $\mathcal{L}$ .
2. At last eliminate all boxes of  $\mathcal{L}$  that no longer pass the midpoint test.

A good strategy for selecting the box to be considered at Step 1 is to take one of the boxes with the largest potential, i.e. a box  $[\mathbf{p}]$  such that the lower bound of  $[c](\mathbf{p})$  is minimal, with  $[c]$  an inclusion function for the cost function  $c$ . To update  $\text{ub}(\hat{c})$  as required by Step 1(b), it suffices to compute the value of the cost at the center of  $[\mathbf{p}]$ , which is of course an upper bound of the optimal value of the cost, and to compare it with the best upper bound available so far.

Two methods for eliminating the box  $[\mathbf{p}]$  under consideration are tried at Step 1(c). The *midpoint test* is based on the evaluation of  $[c](\mathbf{p})$ . If the lower bound of  $[c](\mathbf{p})$  is greater than  $\text{ub}(\hat{c})$ , then no  $\mathbf{p}$  in  $[\mathbf{p}]$  can be a global minimizer of  $c$  over  $\mathbb{P}$ , so the entire box  $[\mathbf{p}]$  can be eliminated. The *monotonicity test* takes advantage of the fact that if  $c$  is once differentiable then any unconstrained minimizer corresponds to a stationary point at which the gradient  $\mathbf{g}$  of  $c$  takes the value zero. Thus if an inclusion function  $[\mathbf{g}]$  is available for  $\mathbf{g}$  and if  $[\mathbf{p}]$  is such that

$$\mathbf{0} \notin [\mathbf{g}](\mathbf{p}), \tag{12}$$

then no  $\mathbf{p}$  in  $[\mathbf{p}]$  can be an unconstrained minimizer so  $[\mathbf{p}]$  can be eliminated.

If a contractor is available, see Section 3.4, Step 1(d) uses it to reduce the size of  $[\mathbf{p}]$  without bisecting it, which makes Step 1(d) an essential ingredient in the struggle against the curse of dimensionality. A number of potentially useful contractors are presented in [29].

Since the tolerance parameter  $\varepsilon$  is strictly positive, the algorithm stops after a finite number of steps and the set  $\mathbb{S}$  of all global minimizers of  $c$  in  $\mathbb{P}$  is included in the union of the boxes in  $\mathcal{L}$ , each of which has now a width smaller than  $\varepsilon$ .

### 3.3. Set inversion via IA

Let  $\mathbf{f}$  be a possibly nonlinear function from  $\mathbb{R}^{n_p}$  to  $\mathbb{R}^{n_y}$  and  $\mathbb{Y}$  be a *subpaving* of  $\mathbb{R}^{n_y}$ , i.e. a union of nonoverlapping boxes. *Set inversion* is the characterization of the reciprocal image of  $\mathbb{Y}$

$$\mathbb{S} = \{\mathbf{p} \in \mathbb{R}^{n_p} \mid \mathbf{f}(\mathbf{p}) \in \mathbb{Y}\} = \mathbf{f}^{-1}(\mathbb{Y}). \quad (13)$$

Using an inclusion function  $[\mathbf{f}]$  for  $\mathbf{f}$ , two subpavings  $\underline{\mathbb{S}}$  and  $\overline{\mathbb{S}}$  such that

$$\underline{\mathbb{S}} \subset \mathbb{S} \subset \overline{\mathbb{S}}$$

can be obtained with the algorithm SIVIA, for Set Inverter Via IA [13].

A (possibly very large) search box  $[\mathbf{p}]_0$  to which  $\overline{\mathbb{S}}$  is guaranteed to belong must be provided by the user. It will be used as an initial outer approximation to build inner and outer approximations of  $\mathbb{S}$  under the form of subpavings by successive bisections and selections. Consider a box  $[\mathbf{p}]$  obtained by bisection of some box belonging to the current outer subpaving. Several cases may be encountered.

1. First, one may use some contractor  $C$  on  $[\mathbf{p}]$  to make  $[\mathbf{p}]$  smaller by eliminating parts of it that do not satisfy  $\mathbf{f}(\mathbf{p}) \in \mathbb{Y}$ , see Section 3.4.
2. If  $[\mathbf{p}]$  has become empty after contraction, or if  $[\mathbf{f}]([\mathbf{p}])$  has an empty intersection with  $\mathbb{Y}$ , then  $[\mathbf{p}]$  has an empty intersection with  $\mathbb{S}$  and should be discarded.
3. If  $[\mathbf{f}]([\mathbf{p}])$  is included in  $\mathbb{Y}$ , then  $[\mathbf{p}]$  is included in  $\mathbb{S}$ , and should be stored in  $\underline{\mathbb{S}}$  and in  $\overline{\mathbb{S}}$ .
4. If  $[\mathbf{f}]([\mathbf{p}])$  has a nonempty intersection with  $\mathbb{Y}$ , but is not entirely in  $\mathbb{Y}$ , then  $[\mathbf{p}]$  may contain a part of  $\mathbb{S}$  without being included in  $\mathbb{S}$  and is said to be *undetermined*. If it has a width greater than a prespecified precision parameter  $\varepsilon$ , then this undetermined box should be bisected into the boxes  $L[\mathbf{p}]$  and  $R[\mathbf{p}]$ , each of which should be investigated by the algorithm.

The resulting recursive algorithm is given in Table I, where the subpavings  $\underline{\mathbb{S}}$  and  $\overline{\mathbb{S}}$  should be initialized as empty before the first call.

All the boxes of  $\overline{\mathbb{S}}$  that are not in  $\underline{\mathbb{S}}$  form an *uncertainty layer*  $\Delta\mathbb{S}$ . All of them have a width smaller than  $\varepsilon$ , so the accuracy of the description of  $\mathbb{S}$  may be increased by decreasing  $\varepsilon$ , at the cost of more computations.

### 3.4. Contractors

Assume that the parameters to be estimated must satisfy the vector inequality constraint  $\mathbf{k}(\mathbf{p}) \geq \mathbf{0}$ , to be understood componentwise. Assume moreover that  $\mathbf{p}$  is known to belong to  $[\mathbf{p}]$ . A *contractor*  $C_{\mathbf{k}}$  for  $\mathbf{p}$  is then an algorithm to compute a box  $C_{\mathbf{k}}([\mathbf{p}])$  such that

$$\begin{aligned} C_{\mathbf{k}}([\mathbf{p}]) &\subset [\mathbf{p}], \\ \{\mathbf{p} \in [\mathbf{p}] \mid \mathbf{k}(\mathbf{p}) \geq \mathbf{0}\} &\subset C_{\mathbf{k}}([\mathbf{p}]). \end{aligned} \quad (14)$$

The first relation in (14) ensures that  $\mathbf{p}$  is contracted, while the second guarantees that no value of  $\mathbf{p}$  satisfying the constraints is lost. Contractors can be similarly defined in the case of equality constraints.

Table I. SIVIA.

Algorithm SIVIA(in: $\mathbf{f}$ , $\mathbb{Y}$ , $[\mathbf{p}]$ , $\varepsilon$ ; inout: $\underline{\mathbb{S}}$ , $\overline{\mathbb{S}}$ )	
1	$[\mathbf{p}] := C([\mathbf{p}]);$
2	if $[\mathbf{p}] = \emptyset$ or $[\mathbf{f}]([\mathbf{p}]) \cap \mathbb{Y} = \emptyset$ return;
3	if $[\mathbf{f}]([\mathbf{p}]) \subset \mathbb{Y}$ then
3	$\{\underline{\mathbb{S}} := \underline{\mathbb{S}} \cup [\mathbf{p}]; \overline{\mathbb{S}} := \overline{\mathbb{S}} \cup [\mathbf{p}];$ return;};
4	if $w([\mathbf{p}]) < \varepsilon$ then $\{\overline{\mathbb{S}} := \overline{\mathbb{S}} \cup [\mathbf{p}];$ return;};
5	SIVIA( $\mathbf{f}$ , $\mathbb{Y}$ , $L[\mathbf{p}]$ , $\varepsilon$ , $\underline{\mathbb{S}}$ , $\overline{\mathbb{S}}$ );
6	SIVIA( $\mathbf{f}$ , $\mathbb{Y}$ , $R[\mathbf{p}]$ , $\varepsilon$ , $\underline{\mathbb{S}}$ , $\overline{\mathbb{S}}$ ).

Consider first guaranteed global optimization. If an upper bound  $\tilde{c}$  for the global minimum  $\hat{c}$  of  $c(\mathbf{p})$  over  $[\mathbf{p}]_0$  is available, then any global minimizer  $\hat{\mathbf{p}} \in [\mathbf{p}]$  has to satisfy

$$c(\hat{\mathbf{p}}) \leq \tilde{c}. \tag{15}$$

When a closed-form expression of the cost function is available, this constraint may be used to build a contractor for  $\mathbf{p}$  via interval constraint propagation, see [29]. When this is not the case, as usual when considering state-space models such as those described by (1), one may use a centered inclusion function of the cost function over  $[\mathbf{p}] = [p_1] \times \dots \times [p_{n_p}]$  to get from (15)

$$c(\mathbf{m}) + \sum_{i=1}^{n_p} ([p_i] - m_i) \left[ \frac{\partial c}{\partial p_i} \right]([\mathbf{p}]) \subset [0, \tilde{c}], \tag{16}$$

since the cost function is positive. Thus, each  $[p_i]$ ,  $i = 1, \dots, n_p$ , has to satisfy

$$[p_i] \subset \left( [0, \tilde{c}] - c(\mathbf{m}) - \sum_{j \neq i} ([p_j] - m_j) \left[ \frac{\partial c}{\partial p_j} \right]([\mathbf{p}]) \right) / \left[ \frac{\partial c}{\partial p_i} \right]([\mathbf{p}]) + m_i, \tag{17}$$

leading to a contracted box  $[\mathbf{p}]^{\text{new}} = C_c([\mathbf{p}])$ , the components of which are defined as

$$[p_i]^{\text{new}} = [p_i] \cap \left( \left( [0, \tilde{c}] - c(\mathbf{m}) - \sum_{j \neq i} ([p_j] - m_j) \left[ \frac{\partial c}{\partial p_j} \right]([\mathbf{p}]) \right) / \left[ \frac{\partial c}{\partial p_i} \right]([\mathbf{p}]) + m_i \right) \tag{18}$$

for  $i = 1, \dots, n_p$ . The main advantage of this contractor is that it only requires the computation of an inclusion function for the gradient of the cost function, which is easily obtained using sensitivity functions, see Section 4.2.

A division by  $[\partial c / \partial p_i]([\mathbf{p}])$  is performed in (18). If standard interval arithmetic is used, a division by an interval containing zero leads to the whole real line, so if  $0 \in [\partial c / \partial p_i]([\mathbf{p}])$ , no contraction takes place along the component of  $[\mathbf{p}]$  being considered. This situation occurs as soon as  $[\mathbf{p}]$  contains an argument of a minimum of the cost function, but also when  $[\mathbf{p}]$  is large, which may lead to a large overestimation of the range of  $\partial c / \partial p_i$  over  $[\mathbf{p}]$ . Applying extended interval arithmetic [32] when  $0 \in [\partial c / \partial p_i]([\mathbf{p}])$  would allow one to get up to two intervals of the form  $]-\infty, a]$  and  $[b, +\infty[$  when performing the division by  $[\partial c / \partial p_i]([\mathbf{p}])$ . After intersection with  $[p_i]$ , the  $i$ th component of  $C_c([\mathbf{p}])$  would then be empty or consist of one or two subintervals.

Consider now bounded-error parameter estimation. All values of the parameter vector  $\mathbf{p} \in \mathbb{S}$  satisfy

$$\mathbf{y}_m(\mathbf{p}) \in [\mathbf{y}] = [\underline{\mathbf{y}}, \bar{\mathbf{y}}],$$

which leads to

$$\begin{aligned} \mathbf{y}_m(\mathbf{p}) - \underline{\mathbf{y}} &\geq \mathbf{0} \\ -\mathbf{y}_m(\mathbf{p}) + \bar{\mathbf{y}} &\geq \mathbf{0} \end{aligned} \tag{19}$$

Using (19), and provided that a closed-form expression for  $\mathbf{y}_m(\mathbf{p})$  is available, one may use interval constraint propagation to get a contracted domain for  $[\mathbf{p}]$ . When such an expression is not available, one may again resort to a centered inclusion function, now for the model output. For the  $k$ th component  $y_k^m(\mathbf{p})$  of  $\mathbf{y}_m(\mathbf{p})$ , for all  $\mathbf{p} \in \mathbb{S} \subset [\mathbf{p}]$  and  $\mathbf{m} \in [\mathbf{p}]$ , (19) translates into

$$\begin{aligned} y_k^m(\mathbf{m}) + \sum_{j=1}^{n_p} ([p_j] - m_j) \left[ \frac{\partial y_k^m}{\partial p_j} \right]([\mathbf{p}]) - \underline{y}_k &\geq 0, \\ -y_k^m(\mathbf{m}) - \sum_{j=1}^{n_p} ([p_j] - m_j) \left[ \frac{\partial y_k^m}{\partial p_j} \right]([\mathbf{p}]) + \bar{y}_k &\geq 0, \end{aligned}$$

for  $k = 1, \dots, \dim \mathbf{y}_m(\mathbf{p})$ . The  $k$ th output measurement thus leads to a contracted domain  $[\mathbf{p}]^{\text{new}} = C_k([\mathbf{p}])$ , the components of which are defined as

$$[p_i]^{\text{new}} = [p_i] \cap \left( \left( [y_k, \bar{y}_k] - y_k^m(\mathbf{m}) - \sum_{j \neq i} ([p_j] - m_j) \left[ \frac{\partial y_k^m}{\partial p_j} \right]([\mathbf{p}]) \right) / \left[ \frac{\partial y_k^m}{\partial p_i} \right]([\mathbf{p}]) + m_i \right), \quad (20)$$

for  $i = 1, \dots, n_p$ . A similar procedure may be applied for all components of  $\mathbf{y}_m(\mathbf{p})$ . Again, only an inclusion function for the Jacobian matrix of  $\mathbf{y}_m(\mathbf{p})$  is needed to apply the contractor  $C_k$ . For state-space models such as those described by (1), this inclusion function may be obtained using sensitivity functions, see Section 4.2.

With standard interval arithmetic, contraction may only be obtained in (20) when  $0 \notin [\partial y_k^m / \partial p_i]([\mathbf{p}])$ . The conditions leading to  $0 \in [\partial y_k^m / \partial p_i]([\mathbf{p}])$  depend mainly on the structure of the model and the overestimation introduced by the inclusion function for  $\partial y_k^m / \partial p_i$ . They are not necessarily linked to whether  $[\mathbf{p}]$  is in the solution set  $\mathbb{S}$ . Extended intervals could readily be used to improve the efficiency of (20). With the SIVIA algorithm, boxes tend to accumulate on the boundary of  $\mathbb{S}$  [13]. For boxes that are very small and across the boundary of  $\mathbb{S}$ , no significant contraction may be obtained. However, for the first iterations of SIVIA, very large boxes may be efficiently eliminated or their size may be strongly reduced using  $C_k$ .

Many other contractors may be considered. For example the Newton contractor, which can be seen as an interval counterpart to the Newton algorithm, can be used to reduce the search domain in global optimization. Assuming that the cost function is sufficiently differentiable, it exploits the constraint  $\mathbf{g}(\mathbf{p}) = \mathbf{0}$ , which must be satisfied by any  $\mathbf{p}$  to qualify as an unconstrained global minimizer. This, however, requires an inclusion function for the Hessian of the cost function, which significantly complicates implementation and execution. Other interesting contractors include the Krawczyk contractor and contractors based on constraint propagation. For more details, see [29].

#### 4. DEALING WITH CONTINUOUS-TIME MODELS

Apart from the evaluation of an inclusion function of the cost function  $c$ , the simplified version of Hansen's algorithm presented in Section 3.2 requires the evaluation of an inclusion function of its gradient  $\mathbf{g}$ . When a closed-form expression of the model output is available, a formal expression of the gradient is easily obtained by formal differentiation and the standard methods for obtaining inclusion functions apply. In general, however, it is not possible to obtain closed-form solutions for models such as those described by (1), which makes the obtention of inclusion functions for  $c$  and  $\mathbf{g}$  particularly difficult.

To get an inclusion function for  $c$ , the naive approach would be to use one of the guaranteed ODE solvers provided by IA, e.g. AWA [33], COSY [34] or VNODE [16]. Unfortunately, these solvers prove unable to provide accurate enclosures for the solutions when there are uncertain parameters, as here. The main idea to bypass this difficulty is to bound the solutions of *uncertain* dynamical systems using *deterministic* dynamical systems. Efficient guaranteed numerical integrators can then be used to compute the corresponding bounding solutions. Preliminary results were presented in [35, 36] in the context of cooperative dynamical models, i.e. models such as (1) for which the off-diagonal terms of the Jacobian matrix of  $\mathbf{f}$  are positive. These results were inspired by the interval observer proposed in [37]. Müller's theorems [38], which have been recently used in the context of guaranteed simulation [39], make it possible to bound the solutions of more general dynamical models.

##### 4.1. Müller's theorem

In order to apply one of the theorems in [38] to the bounding of the solutions of uncertain dynamical models such as (1) in the presence of uncertain parameters  $\mathbf{p} \in [\underline{\mathbf{p}}, \bar{\mathbf{p}}_0]$  and uncertain initial conditions  $\mathbf{x}(t_0) \in [\underline{\mathbf{x}}_0, \bar{\mathbf{x}}_0]$ , we reformulate it as follows.

*Theorem 1*

Assume that  $\mathbf{f}(\mathbf{x}, \mathbf{p}, t)$  in (1) is continuous on

$$\mathbb{T} : \begin{cases} \omega(t) \leq \mathbf{x} \leq \Omega(t) \\ \underline{\mathbf{p}}_0 \leq \mathbf{p} \leq \bar{\mathbf{p}}_0 \\ t_0 \leq t \leq t_1 \end{cases}$$

where  $\omega_i(t)$  and  $\Omega_i(t)$ ,  $i = 1 \dots n_x$ , are continuous on  $[t_0, t_1]$  and such that

1.  $\omega(t_0) = \underline{\mathbf{x}}(t_0)$  and  $\Omega(t_0) = \bar{\mathbf{x}}(t_0)$ ,
2. the left derivatives  $D^- \omega_i(t)$  and  $D^- \Omega_i(t)$  and right derivatives  $D^+ \omega_i(t)$  and  $D^+ \Omega_i(t)$  of  $\omega_i(t)$  and  $\Omega_i(t)$  satisfy, for  $i = 1 \dots n_x$  and  $t \in [t_0, t_1]$ ,

$$D^\pm \omega_i(t) \leq \min_{\underline{\mathbb{T}}_i(t)} f_i(\mathbf{x}, \mathbf{p}, t) \quad \text{and} \quad D^\pm \Omega_i(t) \geq \max_{\bar{\mathbb{T}}_i(t)} f_i(\mathbf{x}, \mathbf{p}, t),$$

where  $\underline{\mathbb{T}}_i(t)$  is the subsets of  $\mathbb{T}$  defined by

$$\underline{\mathbb{T}}_i(\tau) : \begin{cases} x_i = \omega_i(\tau), \\ \omega_j(\tau) \leq x_j \leq \Omega_j(\tau), \quad j \neq i, \\ \underline{\mathbf{p}}_0 \leq \mathbf{p} \leq \bar{\mathbf{p}}_0, \\ t = \tau, \end{cases}$$

and  $\bar{\mathbb{T}}_i(t)$  is the subset of  $\mathbb{T}$  defined by

$$\bar{\mathbb{T}}_i(\tau) : \begin{cases} x_i = \Omega_i(\tau) \\ \omega_j(\tau) \leq x_j \leq \Omega_j(\tau), \quad j \neq i, \\ \underline{\mathbf{p}}_0 \leq \mathbf{p} \leq \bar{\mathbf{p}}_0, \\ t = \tau. \end{cases}$$

Then, for any given  $\mathbf{x}(t_0) \in [\underline{\mathbf{x}}_0, \bar{\mathbf{x}}_0]$  and  $\mathbf{p} \in [\underline{\mathbf{p}}_0, \bar{\mathbf{p}}_0]$ , a solution to (1) exists such that

$$\omega(t) \leq \mathbf{x}(t) \leq \Omega(t) \quad \forall t \in [t_0, t_1],$$

and this solution is equal to  $\mathbf{x}(t_0)$  at  $t = t_0$ .

Moreover, if for any  $\mathbf{p} \in [\underline{\mathbf{p}}_0, \bar{\mathbf{p}}_0]$  and  $t \in [t_0, t_1]$ ,  $\mathbf{f}(\mathbf{x}, \mathbf{p}, t)$  is Lipschitz with respect to  $\mathbf{x}$ , then for any given  $\mathbf{x}(t_0) \in [\underline{\mathbf{x}}_0, \bar{\mathbf{x}}_0]$  and  $\mathbf{p} \in [\underline{\mathbf{p}}_0, \bar{\mathbf{p}}_0]$  this solution is unique.

This theorem allows the evaluation of lower and upper bounds for the solution of (1) provided that two functions  $\omega(t)$  and  $\Omega(t)$  are available. The interval function  $[\Phi](t) = [\omega(t), \Omega(t)]$  can then be seen as an *inclusion function* for all solutions  $\mathbf{x}(t)$  of the state equation in (1).

The construction of  $\omega(t)$  and  $\Omega(t)$  is usually easy on a case-by-case basis, as illustrated in Section 5.1.

An inclusion function for  $\mathbf{g}$  may similarly be obtained by using the sensitivity functions of the output with respect to the parameters, see Section 4.2.

*4.2. Sensitivity functions*

Denote the first-order sensitivity of  $x_j$  with respect to  $p_k$  by  $s_{jk}$

$$s_{jk}(\mathbf{p}, t) = \frac{\partial x_j}{\partial p_k}(\mathbf{p}, t). \tag{21}$$

Assume, for the sake of simplicity, that the model output is linear in the state and given by

$$\mathbf{h}(\mathbf{x}(t), \mathbf{p}) = \mathbf{M}\mathbf{x}(t), \tag{22}$$

where  $\mathbf{M}$  is a known matrix. The Jacobian matrix of  $\mathbf{h}(\mathbf{x}(t), \mathbf{p})$  is then given by

$$\mathbf{J}_h(\mathbf{p}, t) = \mathbf{M} \frac{\partial \mathbf{x}(\mathbf{p}, t)}{\partial \mathbf{p}}, \quad (23)$$

with

$$\frac{\partial \mathbf{x}(\mathbf{p}, t)}{\partial \mathbf{p}} = (s_{jk}(\mathbf{p}, t)), \quad j = 1, \dots, \dim \mathbf{x}, \quad k = 1, \dots, \dim \mathbf{p}. \quad (24)$$

When considering all measurements between  $t_1$  and  $t_{n_t}$ , as in (8), one gets

$$\frac{d\mathbf{y}_m(\mathbf{p})}{d\mathbf{p}} = \begin{pmatrix} \mathbf{M} \frac{\partial \mathbf{x}(\mathbf{p}, t_1)}{\partial \mathbf{p}} \\ \vdots \\ \mathbf{M} \frac{\partial \mathbf{x}(\mathbf{p}, t_{n_t})}{\partial \mathbf{p}} \end{pmatrix}. \quad (25)$$

The gradient of  $c$  may then be obtained as

$$\frac{dc}{d\mathbf{p}}(\mathbf{p}) = -2(\mathbf{y} - \mathbf{y}_m(\mathbf{p}))^T \frac{d\mathbf{y}_m(\mathbf{p})}{d\mathbf{p}},$$

where  $d\mathbf{y}_m(\mathbf{p})/d\mathbf{p}$  may be evaluated using first-order sensitivity functions.

To compute  $s_{jk}$ , differentiate the  $j$ th row of (1) with respect to  $p_k$  to get

$$s'_{jk} = \frac{\partial f_j(\mathbf{x}, \mathbf{p})}{\partial x_j} s_{jk} + \frac{\partial f_j(\mathbf{x}, \mathbf{p})}{\partial p_k}. \quad (26)$$

Since  $\mathbf{x}(t_0)$  is assumed to be known, the initial conditions are

$$s_{jk}(t_0) = \frac{\partial \mathbf{x}(t_0)}{\partial p_k} = 0.$$

The sensitivity function may then be obtained by considering the *extended* state-space model consisting of the dynamical part of (1), and of all differential equations (26) satisfied by the sensitivity functions. Müller's theorem will then turn out to be especially useful, as the extended state-space model is seldom cooperative, even if this is the case of the initial state-space model, see Section 5.

## 5. EXAMPLES

Consider the seemingly simple compartmental model [40] described by Figure 1. Although it is linear from an input–output point of view, its output is highly nonlinear in its parameters. Its state equation is readily obtained from conservation law as

$$\mathbf{x}' = \mathbf{f}(\mathbf{x}, \mathbf{p}, u), \quad (27)$$

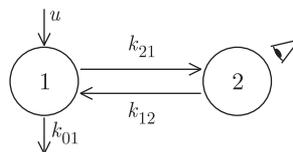


Figure 1. Two-compartment model.

where  $\mathbf{p} = (k_{21}, k_{12}, k_{01})^T$  and

$$\mathbf{f}(\mathbf{x}, \mathbf{p}, u) = \begin{pmatrix} -(p_1 + p_3)x_1 + p_2x_2 + u \\ p_1x_1 - p_2x_2 \end{pmatrix}. \tag{28}$$

The quantity of material in Compartment 2 is assumed to be observed, so

$$y_m(t_i, \mathbf{p}) = x_2(t_i, \mathbf{p}), \quad i = 1, \dots, n_t.$$

Assume that there is no input ( $u \equiv 0$ ) and that the initial condition is known to be  $\mathbf{x}_0 = (1, 0)^T$ .

Differentiating (28) with respect to each of the parameters in turn, one gets the differential equations

$$\begin{aligned} s'_{11} &= -(p_1 + p_3)s_{11} + p_2s_{21} - x_1, \\ s'_{21} &= p_1s_{11} - p_2s_{21} + x_1, \\ s'_{12} &= -(p_1 + p_3)s_{12} + p_2s_{22} + x_2, \\ s'_{22} &= p_1s_{12} - p_2s_{22} - x_2, \\ s'_{13} &= -(p_1 + p_3)s_{13} + p_2s_{23} - x_1, \\ s'_{23} &= p_1s_{13} - p_2s_{23}. \end{aligned} \tag{29}$$

Since  $\mathbf{x}_0$  does not depend on  $\mathbf{p}$ , the initial conditions for all of these sensitivity equations are zero. Note that the 8th-order extended state equation to be solved for computing  $\mathbf{x}$  and the  $s_{jk}$ s is not cooperative, which makes the Müller theorem recalled in Section 4.1 particularly useful.

Since all  $p_k$ s and  $x_j$ s are positive, it is easy, using Müller's theorem, to build bounding dynamical systems to evaluate guaranteed enclosures of the model output  $\mathbf{y}_m(\mathbf{p})$ , of the cost  $c$  and of its gradient  $\mathbf{g}$  with respect to  $\mathbf{p}$ . One may, for instance, compute guaranteed enclosures of  $\mathbf{x}$ ,  $s_{11}$  and  $s_{21}$  via the simulation of the two 6th-order *deterministic* ODEs

$$\begin{aligned} \underline{x}'_1 &= -(\underline{p}_1 + \underline{p}_3)\underline{x}_1 + \underline{p}_2\underline{x}_2, \\ \underline{x}'_2 &= \underline{p}_1\underline{x}_1 - \underline{p}_2\underline{x}_2, \\ \bar{x}'_1 &= -(\underline{p}_1 + \underline{p}_3)\bar{x}_1 + \bar{p}_2\bar{x}_2, \\ \bar{x}'_2 &= \bar{p}_1\bar{x}_1 - \underline{p}_2\bar{x}_2, \\ \underline{s}'_{11} &= -(\bar{p}_1 + \bar{p}_3)\underline{s}_{11} + \underline{p}_2\underline{s}_{21} - \bar{x}_1, \\ \underline{s}'_{21} &= \underline{p}_1\underline{s}_{11} - \bar{p}_2\underline{s}_{21} + \underline{x}_1 \end{aligned} \tag{30}$$

and

$$\begin{aligned} \underline{x}'_1 &= -(\bar{p}_1 + \bar{p}_3)\underline{x}_1 + \underline{p}_2\underline{x}_2, \\ \underline{x}'_2 &= \underline{p}_1\underline{x}_1 - \bar{p}_2\underline{x}_2, \\ \bar{x}'_1 &= -(\underline{p}_1 + \underline{p}_3)\bar{x}_1 + \bar{p}_2\bar{x}_2, \\ \bar{x}'_2 &= \bar{p}_1\bar{x}_1 - \underline{p}_2\bar{x}_2, \\ \bar{s}'_{11} &= -(\underline{p}_1 + \underline{p}_3)\bar{s}_{11} + \bar{p}_2\bar{s}_{21} - \underline{x}_1, \\ \bar{s}'_{21} &= \bar{p}_1\bar{s}_{11} - \underline{p}_2\bar{s}_{21} + \bar{x}_1. \end{aligned} \tag{31}$$

The same procedure can be applied for the sensitivities with respect to  $p_2$  and  $p_3$ , and one could, at least in principle, build a single 16th-order deterministic ODE, the solution of which would provide all required bounding functions. This is not to be recommended, however, as the complexity of guaranteed ODE solvers depends heavily on the state dimension, so one is better of dealing with six 6th-order deterministic ODEs.

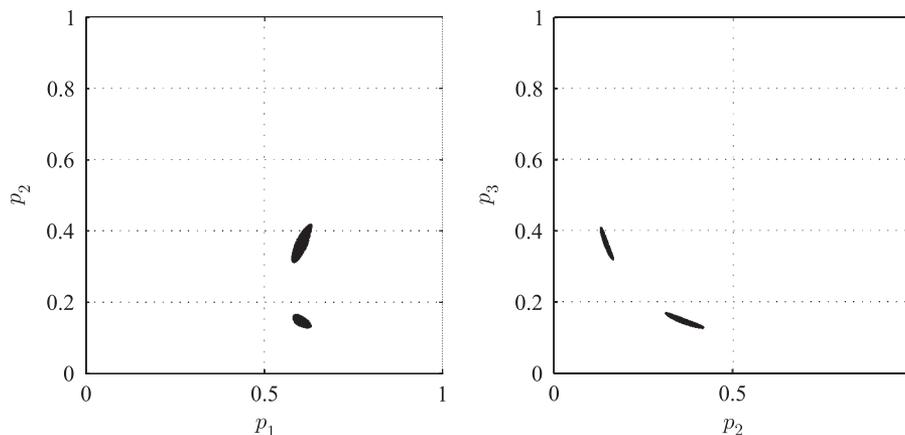


Figure 2. Projection onto the  $(p_1, p_2)$  plane (left) and  $(p_2, p_3)$  plane (right) of a guaranteed outer approximation of the set of all global minimizers ( $\varepsilon=0.001$ ).

To generate artificial data, a two-compartment model with a ‘true’ value of the parameter vector  $\mathbf{p}^* = (0.6, 0.15, 0.35)^T$  has been simulated. The data were then obtained by rounding the value of  $x_2(t_i)$  to the nearest two-digit number for  $t_i = i\Delta t$ , with  $\Delta t = 1$  s and  $i = 1, \dots, 15$ . The initial search domain is  $[\mathbf{p}]_0 = [0.01, 1]^{\times 3}$ .

### 5.1. Global optimization

The cost function is written as

$$c(\mathbf{p}) = \sum_{i=1}^{n_t} (x_2(\mathbf{p}, t_i) - y(t_i))^2,$$

where  $x_2(p, t_i)$  is computed by guaranteed numerical integration of (27). For a precision parameter  $\varepsilon=0.001$ , the deterministic global optimization algorithm described in Section 3.2 with the contractor  $C_c$  presented in Section 3.4 provides a list of boxes whose projections onto the  $(p_1, p_2)$  plane and  $(p_2, p_3)$  plane are shown in Figure 2. Only boxes for which it was not possible to prove that they do not contain any global minimizer are represented, so this is an outer approximation. This result has been obtained in 3 h on a Pentium IV at 2 GHz. The symmetry of the projection onto the  $(p_2, p_3)$ -plane of the solution boxes suggests an identifiability problem that could be confirmed by a theoretical study [22]: the values of the parameters  $p_2$  and  $p_3$  may be exchanged without modifying the model output. This identifiability study was not used to get the results presented here.

### 5.2. Bounded-error estimation

The SIVIA algorithm described in Section 3.3 is now employed. To be tolerated, the error must belong to  $[e] = [-0.005, 0.005]$  for all measurements. Three versions have been considered. In NIF, the natural inclusion function is used; CF uses the centered form, and CF-CP uses the contractors  $C_k$  on  $[\mathbf{p}]$  presented in Section 3.4, with  $\mathbf{m} = (\mathbf{p} + \bar{\mathbf{p}})/2$  for all measurements.

Using the same data as in the optimization case, we obtain the results summarized in Tables II and III. The guaranteed numerical integration toolbox VNODE [16] has been used to obtain guaranteed solutions to (30), (31), and to similar systems of ODEs used for evaluating the sensitivities with respect to  $p_2$  and  $p_3$ .

For a given value of  $\varepsilon$ , NIF is the most efficient in terms of computational complexity, see Table II, since only cooperativity is used, and no sensitivity function is involved. The price to be paid is, by far, the worst performance in terms of accuracy, measured by the volume of the outer approximation to the solution set, see Table III. When considering a centered form, since the accuracy of the inclusion function is much better, much less undetermined boxes have to be

Table II. Computing times in seconds for three versions of the SIVIA algorithm.

$\varepsilon$	0.01	0.005	0.001	0.0005
NIF	21	48	294	823
CF	650	1265	7720	28328
CF-CP	180	336	5818	27876

Table III. Volumes ( $\times 10^3$ ) of guaranteed outer approximations of the solution set for three versions of the SIVIA algorithm.

$\varepsilon$	0.01	0.005	0.001	0.0005
NIF	1.03	0.31	0.032	0.014
CF	0.23	0.057	0.093	0.0051
CF-CP	0.049	0.016	0.0050	0.0040

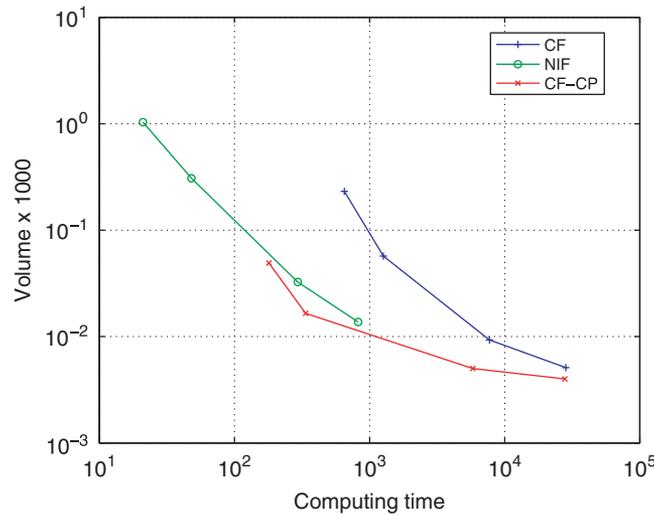


Figure 3. Volume of the solution set as a function of computing time (in seconds).

considered. The volume of the solution set is then much smaller for a given  $\varepsilon$ . Nevertheless, the centered form is much more costly to obtain, since additional ODEs have to be solved to get the sensitivity functions. Using the contractor in conjunction with the centered inclusion function allows one to reduce computing time significantly, at least for large values of  $\varepsilon$ . The volume of the solution set is also significantly reduced, when compared with the other two approaches. For larger values of  $\varepsilon$ , CF-CP behaves similarly to CF in terms of computing time, but the quality of the solution is still better for CF-CP. The advantage brought by the contractors reduces when  $\varepsilon$  decreases, since many small boxes accumulate at the boundary of the solution set. The boxes within the solution set and those at the boundary cannot be reduced by contraction. This explains why the number of simulations become similar.

Figure 3 plots the volume of the guaranteed outer approximation of the solution set as a function of computing time. CF turns out to be the worst solution, always bettered by NIF or CF-CP.

Figures 4 and 5 show planar projections of the guaranteed outer approximations  $\overline{\mathbb{S}}_\varepsilon$  to the solution set  $\mathbb{S}$  obtained by the three versions of the SIVIA algorithm for four values of the precision parameter. Presentation is in the same order as in Tables II and III. With NIF, the precision parameter  $\varepsilon$  has to be very small (less than 0.001) to make it possible to see that the actual solution set  $\mathbb{S}$  consists of two subsets, suggesting an identifiability problem. This conclusion is already reached with  $\varepsilon = 0.005$  for CF and with  $\varepsilon = 0.01$  for CF-CP.

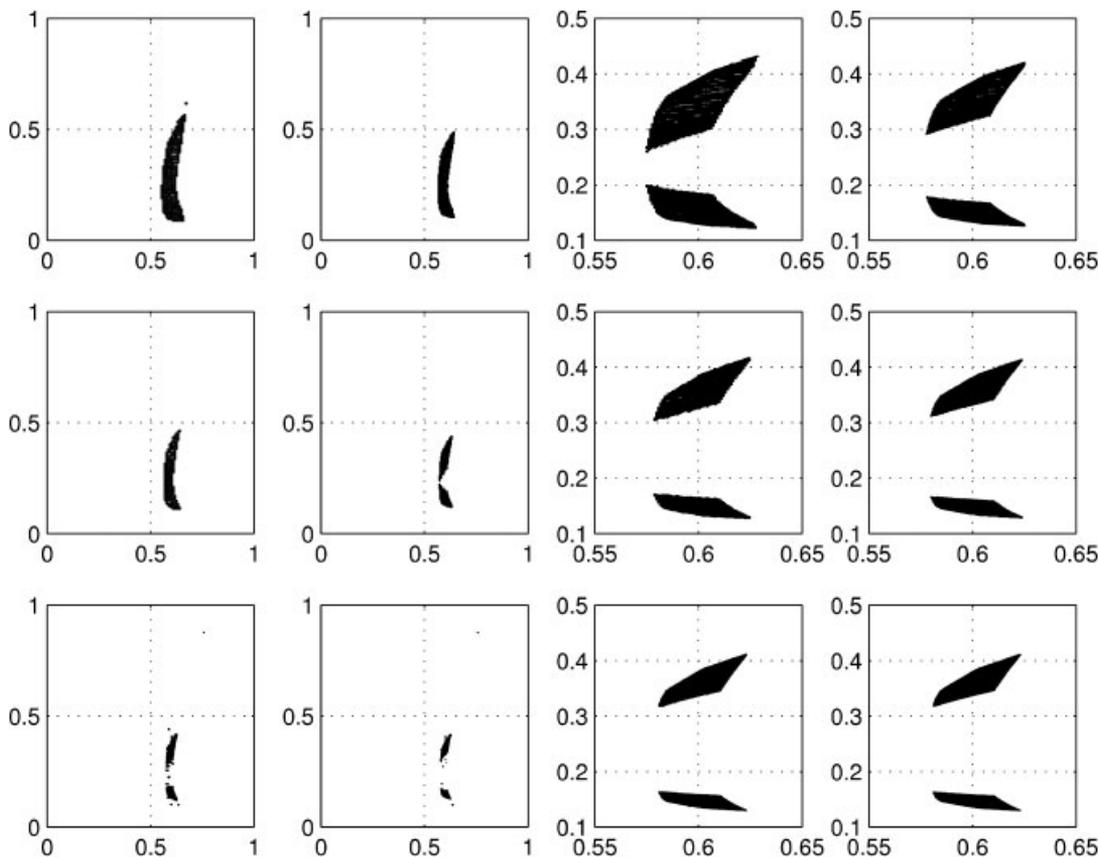


Figure 4. Projection on the  $(p_1, p_2)$ -plane of outer-approximations of the solution set obtained for various values of the precision parameter  $\varepsilon$  (from left to right,  $\varepsilon=0.01$ ,  $\varepsilon=0.005$ ,  $\varepsilon=0.001$ , and  $\varepsilon=0.0005$ ), and for three versions of SIVIA (from top to bottom, natural inclusion function (NIF), centered form (CF), and centered form with constraint propagation (CF-CP)); note the scale of the subfigures of the two right-most columns.

The results of Tables II and III suggest combining CF-CP and NIF within a single search algorithm. For each box  $[\mathbf{p}]$ , CF-CP should only be used when the contractor  $C_k$  applied on  $[\mathbf{p}]$  may produce a contraction significant enough to compensate for the additional computing time required to obtain the sensitivity functions needed by  $C_k$ . If obtaining a NIF takes  $\alpha\%$  of the time required to obtain a CF-CP, one may stop using CF-CP as soon as the ratio of the volume of the contracted box to that of the initial box is larger than  $\alpha\%$ . Many other strategies may be considered.

## 6. CONCLUSIONS AND PERSPECTIVES

Interval analysis (IA) is an attractive option for anyone who feels disturbed by the fact that the usual techniques for nonlinear parameter estimation are unable to deliver any guarantee about the estimates that they produce. In this paper, we have presented some already available tools that allow global and guaranteed results to be obtained in nontrivial cases involving continuous-time state-space models. Guaranteed simulation, in combination with Müller's theorems, allow one to deal with ODEs for which no closed-form solution is available, in a more general setting than the notion of cooperativity previously used.

It should be clear, however, that guaranteed estimation is far from being applicable to all nonlinear estimation problems of interest. In the general case, the characterization of the set of

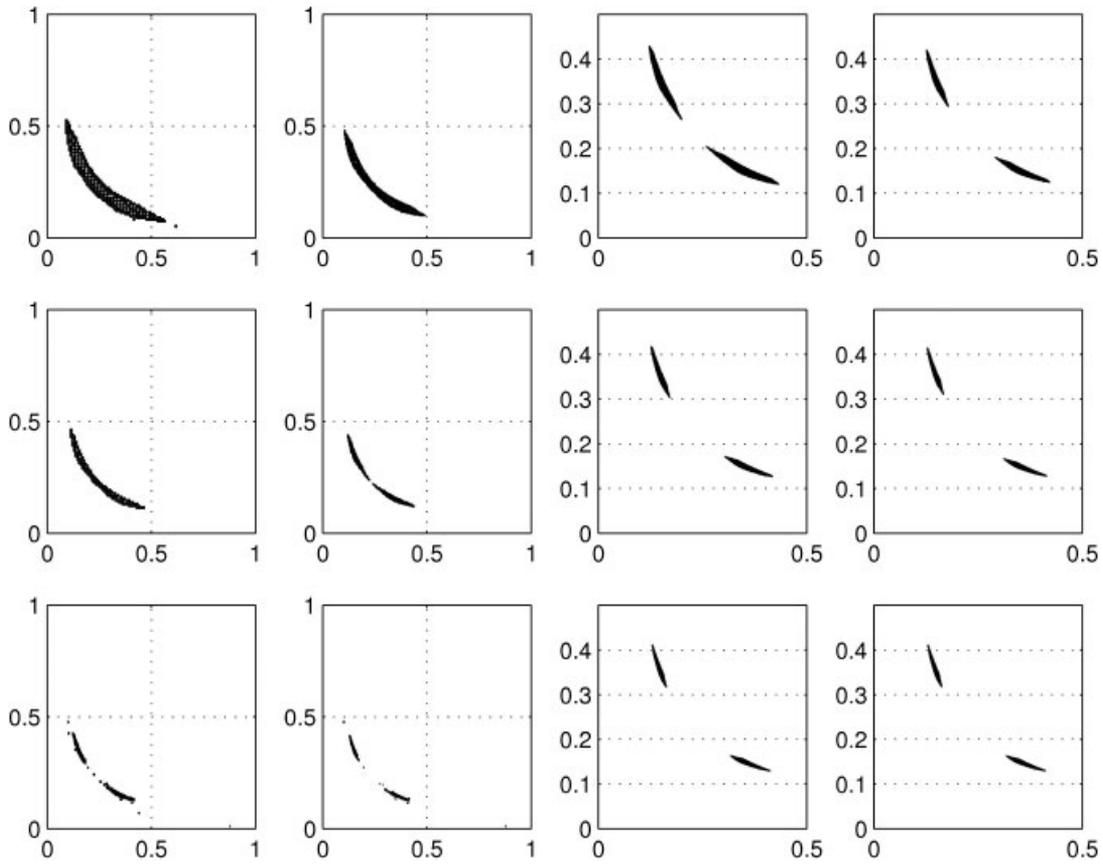


Figure 5. Projection on the  $(p_2, p_3)$ -plane of outer-approximations of the solution set obtained for various values of the precision parameter  $\varepsilon$  (from left to right,  $\varepsilon = 0.01$ ,  $\varepsilon = 0.005$ ,  $\varepsilon = 0.001$ , and  $\varepsilon = 0.0005$ ), and for three versions of SIVIA (from top to bottom, natural inclusion function (NIF), centered form (CF), and centered form with constraint propagation (CF-CP)); note the scale of the subfigures of the two right-most columns.

all optimal or acceptable parameter vectors is an NP-complete problem, and the struggle against the curse of dimensionality is absolutely crucial. The use of contractors, which reduces the size of boxes of interest without bisecting them or losing any solution, is an important ally in this endeavor. We have shown how sensitivity functions, combined with guaranteed integration and one of Müller’s theorems, can be used to build such contractors.

Guaranteed parameter optimization turns out to be more difficult than guaranteed parameter bounding. The presence of a sum of terms depending on the uncertain parameters in the cost function entails overestimation in the evaluation of the associated inclusion function. This is because all interval quantities are treated as independent, even if they actually correspond to the same box in parameter space. More efficient contractors than those employed here could be considered to address this difficulty. A first option, which we plan to investigate, is to build contractors for each term in the sum, taking into account the fact that each of them must comply with the constraint (15). Convexity tests based on second-order sensitivity functions could also be considered.

Moreover, significant improvement in computing times can be expected by an appropriate management of the switching between NIF and CP-CF, which may differ from box to box and from time to time.

ACKNOWLEDGEMENTS

This work has been partly supported by the ANR CPP and Digiteo-DIM-LSC PASO projects.

## REFERENCES

1. Schweppe FC. Recursive state estimation: unknown but bounded errors and system inputs. *IEEE Transactions on Automatic Control* 1968; **13**(1):22–28.
2. Schweppe FC. *Uncertain Dynamic Systems*. Prentice-Hall: Englewood Cliffs, NJ, 1973.
3. Chernousko FL. Optimal guaranteed estimates of indeterminacies with the aid of ellipsoids. *Engineering Cybernetics* 1980; **18**:1–9.
4. Chernousko FL. *State Estimation for Dynamic Systems*. CRC Press: Boca Raton, FL, 1994.
5. Kurzhanski A, Valyi I. *Ellipsoidal Calculus for Estimation and Control*. Birkhäuser: Boston, MA, 1997.
6. Milanese M, Belforte G. Estimation theory and uncertainty intervals evaluation in presence of unknown but bounded errors: linear families of models and estimators. *IEEE Transactions on Automatic Control* 1982; **27**(2):408–414.
7. Chisci L, Garulli A, Zappa G. Recursive state bounding by parallelotopes. *Automatica* 1996; **32**:1049–1056.
8. Alamo T, Bravo JM, Camacho EF. Guaranteed state estimation by zonotopes. *Automatica* 2005; **41**(6):1035–1043.
9. Kurzhanski AA, Varaiya P. Ellipsoidal toolbox. *Technical Report UCB/EECS-2006-46*, EECS Department, University of California, Berkeley, May 2006.
10. Veres SM, Mayne DQ. Geometric bounding toolbox. *Technical Report*, SysBrain, 2001. Available from: <http://sysbrain.com/gbt/>.
11. Nazin S, Polyak B. Interval parameter estimation under model uncertainty. *Mathematical and Computer Modelling of Dynamical Systems* 2005; **11**(2):225–237.
12. Kurzhanski AB, Varaiya P. Ellipsoidal techniques for reachability analysis. *Optimization Methods and Software* 2002; **17**(2):177–237.
13. Jaulin L, Walter E. Set inversion via interval analysis for nonlinear bounded-error estimation. *Automatica* 1993; **29**(4):1053–1064.
14. Lohner R. Enclosing the solutions of ordinary initial and boundary value problems. In *Computer Arithmetic: Scientific Computation and Programming Languages*, Kaucher E, Kulisch U, Ullrich C (eds). BG Teubner: Stuttgart, 1987; 255–286.
15. Berz M, Makino K. Verified integration of ODEs and flows using differential algebraic methods on high-order Taylor models. *Reliable Computing* 1998; **4**(4):361–369.
16. Nedialkov NS, Jackson KR. Methods for initial value problems for ordinary differential equations. In *Perspectives on Enclosure Methods*, Kulisch U, Lohner R, Facius A (eds). Springer: Vienna, 2001; 219–264.
17. Kieffer M, Walter E. Guaranteed parameter estimation for cooperative systems. In *Positive Systems*, Benvenuti L, De Santis A, Farina L (eds). Lecture Notes in Control and Information Science, vol. 294. Springer: Berlin, 2003; 103–110.
18. Raissi T, Ramdani N, Candau Y. Set membership state and parameter estimation for systems described by nonlinear differential equations. *Automatica* 2004; **40**(10):1771–1777.
19. Walter E, Kieffer M. Guaranteed nonlinear parameter estimation in knowledge-based models. *Journal of Computational and Applied Mathematics* 2007; **199**(2):1458–1468.
20. Kühn W. Rigorously computed orbits of dynamical systems without the wrapping effect. *Computing* 1998; **61**(1):47–67.
21. Kay SM. *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice-Hall: Englewood Cliffs, NJ, 1993.
22. Walter E, Pronzato L. *Identification of Parametric Models from Experimental Data*. Springer: London, 1997.
23. Walter E (ed.). Special issue on parameter identification with error bounds. *Mathematics and Computers in Simulation* 1990; **32**(5–6):447–607.
24. Norton JP (ed.). Special issue on bounded-error estimation: issue 1. *International Journal of Adaptive Control and Signal Processing* 1994; **8**(1):1–118.
25. Norton JP (ed.). Special issue on bounded-error estimation: issue 2. *International Journal of Adaptive Control and Signal Processing* 1995; **9**(1):1–132.
26. Milanese M, Norton J, Piet-Lahanier H, Walter E (eds). *Bounding Approaches to System Identification*. Plenum Press: New York, NY, 1996.
27. Moore RE. *Methods and Applications of Interval Analysis*. SIAM: Philadelphia, PA, 1979.
28. Neumaier A. *Interval Methods for Systems of Equations*. Cambridge University Press: Cambridge, U.K., 1990.
29. Jaulin L, Kieffer M, Didrit O, Walter E. *Applied Interval Analysis*. Springer: London, 2001.
30. Hansen ER. On solving systems of equations using interval arithmetic. *Mathematical Computing* 1968; **22**:374–384.
31. Hansen ER. *Global Optimization Using Interval Analysis*. Marcel Dekker: New York, NY, 1992.
32. Klatte R, Kulisch U, Wieth A, Lawo C, Rauch M. *C-xsc: A C++ Class Library For Extended Scientific Computing*. Springer: Berlin, Germany, 1993.
33. Lohner R. Computation of guaranteed enclosures for the solutions of ordinary initial and boundary value-problem. In *Computational Ordinary Differential Equations*, Cash JR, Gladwell I (eds). Clarendon Press: Oxford, 1992; 425–435.
34. Hoefkens J, Berz M, Makino K. Efficient high-order methods for ODEs and DAEs. In *Automatic Differentiation: From Simulation to Optimization*, Corliss G, Faure C, Griewank A (eds). Springer: New York, NY, 2001; 341–351.

35. Walter E, Kieffer M. Interval analysis for guaranteed nonlinear estimation. *Proceedings of the 13th IFAC Symposium on System Identification (SYSID)*, Rotterdam, 2003; 259–270.
36. Kieffer M, Walter E. Interval analysis for guaranteed nonlinear parameter and state estimation. *Mathematical and Computer Modelling of Dynamic Systems* 2005; **11**(2):171–181.
37. Gouzé JL, Rapaport A, Hadj-Sadok ZM. Interval observers for uncertain biological systems. *Journal of Ecological Modelling* 2000; **133**(1–2):45–56.
38. Müller M. Über das fundamentaltheorem in der theorie der gewöhnlichen differentialgleichungen. *Mathematische Zeitschrift* 1926; **26**:619–645.
39. Gennat M, Tibken B. Simulation of uncertain systems with guaranteed bounds. *Eleventh GAMM—IMACS International Symposium on Scientific Computing, Computer Arithmetic, and Validated Numerics*, Fukuoka, Japan, 4–8 October 2004.
40. Kieffer M, Walter E. Interval analysis for guaranteed nonlinear parameter estimation. In *MODA 5-Advances in Model-Oriented Data Analysis and Experiment Design*, Atkinson AC, Pronzato L, Wynn HP (eds). Physica-Verlag: Heidelberg, 1998; 115–125.