Mathematics and Computers in Simulation 35 (1993) 123–137
North-Holland

MATCOM 932

Guaranteed nonlinear parameter estimation from bounded-error data via interval analysis

Luc Jaulin and Eric Walter

Laboratoire des Signaux et Systèmes, CNRS — Ecole Supérieure d'Electricité, Gif-sur-Yvette, France

Abstract

Jaulin, L. and E. Walter, Guaranteed nonlinear parameter estimation from bounded-error data via interval analysis, Mathematics and Computers in Simulation 35 (1993) 123–137.

This paper deals with parameter estimation in the bounded-error context. A new approach, based on interval analysis, is proposed to compute guaranteed estimates of suitable characteristics of the set $\mathbb{S}$ of all values of the parameter vector such that the error between the experimental data and the model outputs belongs to some predefined feasible set. This approach is especially suited to models whose output is nonlinear in their parameters, a situation where most available methods fail to provide any guarantee as to the global validity of the results obtained. After a brief presentation of interval analysis, an algorithm is proposed, which makes it possible to obtain guaranteed estimates of characteristics of $\mathbb{S}$ such as its volume or the smallest axis-aligned box that contains it. Properties of this algorithm are established, and illustrated on a simple example.

1. Introduction to bounded-error estimation

Let $y \in \mathbb{R}^{n_y}$ be the vector of *all* measurements performed on a system to be studied. This vector may correspond for example to n_y scalar measurements performed at various times on a dynamical system, or to a set of measurements corresponding to various experimental conditions on a static process. We assume that these data are to be described by the vector $y_m \in \mathbb{R}^{n_y}$ of the outputs of a model with a fixed structure but unknown parameter vector $p \in \mathbb{R}^{n_p}$. Although both y and y_m usually depend on inputs (or operating mode, or experimental conditions), this dependency will be omitted here for simplicity of notation. The purpose of parameter estimation is to find p such that $y_m(p)$ fits y best in a sense to be specified. A recent special issue of *Mathematics and Computers in Simulation* [19] has been devoted to the special case of bounded-error estimation. In this context (see, e.g., [21] for a survey), the parameters are considered admissible if the error $e(p)$ defined as

$$e(p) = y - y_m(p) \tag{1}$$

Correspondence to: E. Walter, Laboratoire des Signaux et Systèmes, Centre National de la Recherche Scientifique, Ecole Supérieure d'Electricité, Plateau de Moulon, 91192 Gif-sur-Yvette Cedex, France.

belongs to some prior compact set of admissible errors $\mathbb{E} \subset \mathbb{R}^{n_y}$ defined by n_c inequalities (n_c as number of constraints). $\mathbb{E}$ may for instance be the box defined as

$$\mathbb{E} = \{e \mid e^- \leqslant e \leqslant e^+\}, \tag{2}$$

where e^- and e^+ are prior bounds and the $n_c = 2n_y$ inequalities are to be taken componentwise. Other types of compact sets could be considered as well, as more general polytopes or ellipsoids,. for example when there is only a quadratic constraint on e ($n_c = 1$). One is then primarily interested in finding the set $\mathbb{S}$ of *all* values of p such that the error is admissible:

$$\mathbb{S} = \{p \mid e(p) \in \mathbb{E}\}. \tag{3}$$

This set has been called by various authors *membership set, likelihood set* and *(posterior) feasible set*. If the data have been generated by a model $y_m(p^*)$, where p^* is some true value of the parameters, and if $e(p^*) \in \mathbb{E}$, then $\mathbb{S}$ contains p^*. Thus, $\mathbb{S}$ provides us with an accurate description of the uncertainty with which p^* is estimated.

Remark 1.1. If the prior feasible set for p is a subset of $\mathbb{R}^{n_p}$ defined by inequality constraints, these inequalities can readily be incorporated in the set of those defining the posterior feasible set $\mathbb{S}$.

If the reciprocal function of y_m (in a set-theoretic sense) is denoted by y_m^{-1}, $\mathbb{S}$ can equivalently be defined as

$$\mathbb{S} = y_m^{-1}(y - \mathbb{E}) = y_m^{-1}(\mathbb{Y}), \tag{4}$$

where $\mathbb{Y} = y - \mathbb{E}$ is the *measurement set*. For any $p \in \mathbb{S}$, there exists $e \in \mathbb{E}$ such that

$$y = y_m(p) + e. \tag{5}$$

When y_m is linear or affine in p and $\mathbb{E}$ is a polyhedron, $\mathbb{S}$ is a convex polyhedron (most often a convex polytope). It can then be characterized exactly and recursively (see, e.g., [4,13,20]). Techniques have also been proposed to bound $\mathbb{S}$ by ellipsoids [1,7] or boxes [11].

When y_m is not affine in p, one may think [2] of linearizing it around some value of the parameters estimated beforehand and then using any method for linear models. Scanning the parametric space using random search has also been considered (see, e.g., [17]). None of these techniques, however, offers any guarantee as to the global nature of the results obtained.

Whether or not y_m is affine in p, computing the smallest axis-aligned orthotope containing $\mathbb{S}$ can be performed by solving $2n_p$ problems of mathematical programming. Each of them corresponds to the maximization or minimization of a component of p subject to the n_c inequality constraints that define $\mathbb{S}$. In a large number of problems of practical interest (such as the estimation of the parameters of an ARMA model or of a discrete state space model) these problems can be set in the framework of signomial programming and techniques have been proposed that allow one to get *global* results [6,12]. Here, we suggest an alternative approach — based on interval analysis — which can also be used to obtain guaranteed estimates of suitable characteristics of $\mathbb{S}$ in the nonlinear case. First interval analysis is very briefly presented in Section 2. Section 3 then describes how it can be used in the context of

bounded-error estimation. An algorithm is proposed, and some of its properties are established. An illustrative example is treated in Section 4.

2. Interval analysis

Interval analysis has been a very active field in scientific computation for the last twenty years (see, e.g., [10,14,16,18]). There are now commercially available extensions of FORTRAN and PASCAL that include interval arithmetic among their features [8,10]. After recalling the notions needed to understand the algorithm proposed in Section 3, we shall establish a result useful for the analysis of its complexity.

2.1. Interval arithmetic

Interval arithmetic treats intervals as a new kind of numbers, on which all classical arithmetical operations can be performed. Intervals are basic constituents of the description of uncertainty in the bounded-error context, as probability laws are for a statistical description. Interval arithmetic makes it possible to propagate the uncertainty on numbers to the results obtained by making numerical computations on these numbers.

Definition 2.1. An *interval* $[x]$ of $\mathbb{R}$ (or *scalar interval*) is a closed, bounded and connected set of real numbers

$$[x] = [x^-, x^+] = \{x \mid x^- \leqslant x \leqslant x^+\}.$$

The set of all intervals of $\mathbb{R}$ will be denoted by $\mathbb{IR}$.

Definition 2.2. A *box* $[\boldsymbol{x}]$ of $\mathbb{R}^n$ (or *vector interval*) is the Cartesian product of n intervals of $\mathbb{IR}$.

The set of all boxes of $\mathbb{R}^n$ will be denoted by $\mathbb{IR}^n$. Boxes will be specified indifferently in any of the three following ways:

$$[\boldsymbol{x}] = [x_1^-, x_1^+] \times [x_2^-, x_2^+] \times \cdots \times [x_n^-, x_n^+],$$
$$[\boldsymbol{x}] = [x_1] \times [x_2] \times \cdots \times [x_n],$$
$$[\boldsymbol{x}] = [\boldsymbol{x}^-, \boldsymbol{x}^+].$$

Remark 2.3. Vectors $\boldsymbol{x}$ of $\mathbb{R}^n$ will also be considered as belonging to $\mathbb{IR}^n$, with $\boldsymbol{x}^- = \boldsymbol{x}^+ = \boldsymbol{x}$.

Definition 2.4. The *width* of a box $[\boldsymbol{x}] \in \mathbb{IR}^n$, denoted by $w([\boldsymbol{x}])$, is given by

$$w([\boldsymbol{x}]) = \max\{x_i^+ - x_i^-, i = 1, \ldots, n\}.$$

Any vector of $\mathbb{R}^n$ has a zero width, so that w is only a semi-norm.

Definition 2.5. The *enveloping box* of a compact subset $\mathbb{A} \subset \mathbb{R}^n$, denoted by $[\mathbb{A}]$, is the smallest box of $\mathbb{IR}^n$ that contains $\mathbb{A}$:

$$[\mathbb{A}] = \bigcap \{[\boldsymbol{x}] \in \mathbb{IR}^n \mid \mathbb{A} \subset [\boldsymbol{x}]\}.$$

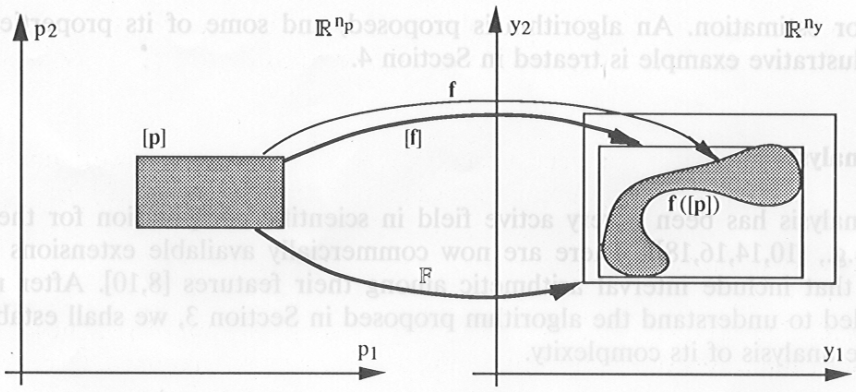

Fig. 1. Minimal inclusion function $[f]$ and inclusion function $\mathbb{F}$ of a function f.

Definition 2.6. Let f be a function from $\mathbb{R}^n$ to $\mathbb{R}^p$. The *minimal inclusion function* of f, denoted by $[f]$, is defined as

$$[f]:\mathbb{IR}^n \to \mathbb{IR}^p, \quad [x] \to \left[\{f(x) \mid x \in [x]\}\right].$$

$[f]([x])$ is thus the smallest box of $\mathbb{IR}^p$ that contains $f([x])$, i.e., the enveloping box of $f([x])$. It is easy to compute for usual elementary functions. When no efficient algorithm exists for the computation of $[f]$, it can be approximated by an inclusion function $\mathbb{F}$ satisfying the following definition.

Definition 2.7. $\mathbb{F}:\mathbb{IR}^n \to \mathbb{IR}^p$ is an *inclusion function* of $f:\mathbb{R}^n \to \mathbb{R}^p$ if

$$\forall [x] \in \mathbb{IR}^n, \ f([x]) \subset \mathbb{F}([x]) \tag{6}$$

and

$$w([x]) \to 0 \Rightarrow w(\mathbb{F}([x])) \to 0. \tag{7}$$

Figure 1 illustrates Definitions 2.6 and 2.7. For any function f obtained by composition of elementary operations such as $+$, $-$, $*$, $/$, sin, cos, exp, ..., it is easy to obtain an inclusion function by replacing each of these elementary operations by its minimal inclusion function in the formal expression of f [14]. Relaxing Definition 2.7 by discarding condition (7), it is also possible to take into account the effect of rounding in the computation so as to obtain intervals guaranteed to contain the exact mathematical solutions. It must be noted, however, that the algorithm presented in this paper is guaranteed to converge only if (7) is valid.

2.2. Pavings

Most methods for estimating or optimizing parameters are based on computations performed at point values of the parameter vector. In the absence of external knowledge (e.g., on the unimodality of the problem), convergence to the global result is then not guaranteed. The main interest of the notion of paving is to make it possible to replace point values of vectors by

subsets of the parameter space. For simplicity, we shall use pavings based upon boxes, but other types of coverings based, e.g., on more complex polyhedrons or ellipsoids could be used as well.

Definition 2.8. A *paving* of a compact subset $\{\mathbb{P}\} \subset \mathbb{R}^n$ is a set $\mathbb{P}$ of nonoverlapping boxes of $\mathbb{IR}^n$ with nonzero width such that the union of these boxes corresponds to $\{\mathbb{P}\}$.

Definition 2.9. A *subpaving* $\mathbb{K}$ of the paving $\mathbb{P}$ is a subset of $\mathbb{P}$.

Definition 2.10. The *accumulation set* of a subpaving $\mathbb{K}$ is the limit of the subset of $\mathbb{R}^n$ formed by the union of the boxes of $\mathbb{K}$ with width lower than ϵ when ϵ tends to zero.

Remark 2.11. Since pavings only contain boxes with nonzero width, the accumulation set of a finite paving is necessarily void.

Definition 2.12. A sequence of subpavings $(\mathbb{K}(k))_{k \in \mathbb{N}}$ *accumulates* on a subset $\mathbb{A}$ of $\mathbb{R}^n$ if $\mathbb{A}$ is the accumulation set of $\mathbb{K}(\infty)$.

The set $\mathbb{R}^n$ will be equipped with the distance defined by $d_\infty(x, y) = w([x, y])$. The distance separating two sets $\mathbb{A}$ and $\mathbb{B}$ will then be the minimum distance (in the sense of d_∞) between vectors belonging to $\mathbb{A}$ and $\mathbb{B}$. Note that d_∞ is not a distance on the set $\mathscr{C}(\mathbb{R}^n)$ of all compact sets of $\mathbb{R}^n$, since the distance separating any two connected sets is zero. This is why $\mathscr{C}(\mathbb{R}^n)$ will be equipped with a distance similar to the Hausdorff distance (see, e.g., [3]) and defined by $h_\infty(\mathbb{A}, \mathbb{B}) = \min\{r \mid \mathbb{A} \subset \mathbb{B} + r\mathbb{U} \text{ and } \mathbb{B} \subset \mathbb{A} + r\mathbb{U}\}$ where $\mathbb{U}$ is the unit sphere in $(\mathbb{R}^n, d_\infty)$.

Definition 2.13. The *precision of description* ϵ_d of a compact set $\mathbb{A}$ by the paving $\mathbb{P}$ is the h_∞-distance between $\mathbb{A}$ and the smallest subpaving of $\mathbb{P}$ that covers $\mathbb{A}$.

Definition 2.14. Let n_b be the minimum number of boxes of $\mathbb{IR}^n$ with width smaller than ϵ necessary to cover $\mathbb{A} \subset \mathbb{R}^n$. The *fractal dimension* of $\mathbb{A}$ is defined [5] by

$$m = \lim_{\epsilon \to 0^+} \sup \frac{\ln(n_b)}{\ln(1/\epsilon)}.$$

Remark 2.15. We shall only consider manifolds $\mathbb{A}$ that are regular enough for m to be equal to their classical dimension.

Define a weak partial ordering over the set of real sequences by

$$u \leftarrow v \Leftrightarrow \lim_{n \to \infty} \frac{\ln(u_n)}{\ln(v_n)} \leqslant 1. \tag{8}$$

This ordering induces an equivalence relation $\sim$ defined by

$$u \sim v \Leftrightarrow u \leftarrow v \text{ and } v \leftarrow u. \tag{9}$$

Remark 2.16. If ϵ is a positive sequence converging to zero and if n_b is defined as in Definition 2.14, then $n_b \sim 1/\epsilon^m$.

Theorem 2.17. *Let ϵ_d be a positive sequence tending to zero. Let $\mathbb{P}$ be a paving of a compact set $\{\mathbb{P}\}$ that describes a compact submanifold $\mathbb{A} \subset \{\mathbb{P}\} \subset \mathbb{R}^n$ with a precision of ϵ_d, and such that no box of $\mathbb{P}$ has an edge smaller than ϵ_d. Let m be the dimension of $\mathbb{A}$ ($m < n$). Then $1/(\epsilon_d)^m \leftarrow \text{card}(\mathbb{P}) \leftarrow 1/(\epsilon_d)^n$ almost always.*

Proof. As $m < n$, almost always on the set $\mathbb{A}$ the boxes of $\mathbb{P}$ have a width of the same order of magnitude as ϵ_d. From Remark 2.16, the minimum number of boxes with width smaller than ϵ_d necessary to cover $\mathbb{A}$ is therefore $n_b \sim 1/(\epsilon_d)^m$, so that $1/(\epsilon_d)^m \leftarrow \text{card}(\mathbb{P})$. Since no box of $\mathbb{P}$ has an edge smaller than ϵ_d, card($\mathbb{P}$) is smaller than the number of hypercubes with width ϵ_d required to cover $\{\mathbb{P}\}$, i.e., $\text{vol}(\{\mathbb{P}\})/(\epsilon_d)^n$. Therefore $\text{card}(\mathbb{P}) \leftarrow 1/(\epsilon_d)^n$. □

Theorem 2.17 will be useful for the analysis of the complexity of the algorithm proposed in Section 3.2. The intuition behind it can be understood by considering a box of $\mathbb{R}^m$ covered with hypercubes of width ϵ. When the width of these hypercubes is divided by 2, the number of hypercubes required is multiplied by 2^m.

Remark 2.18. In Theorem 2.17, "almost always" is meant to exclude situations such as when $\mathbb{A}$ is a finite union of boxes, because then card($\mathbb{P}$) remains finite.

3. Characterization of $\mathbb{S}$ via interval analysis

3.1. Application of interval analysis to bounded-error estimation

Although the algorithm to be presented could be used to study other properties of $\mathbb{S}$, the characteristics of $\mathbb{S}$ that we shall consider more specifically are its volume and its enveloping box $[\mathbb{S}]$. In $\mathscr{P}(\mathbb{R}^{n_p})$ equipped with the partial ordering $\subset$, the algorithm encloses $\mathbb{S}$ between two compact sets corresponding to two subpavings, hence the proposed acronym SEVIA (Set Enclosure Via Interval Analysis).

SEVIA applies to any model structure such that an inclusion function $\mathbb{Y}_m$ can be computed for $\boldsymbol{y}_m$, where $\boldsymbol{y}_m$ is considered as a function from $\mathbb{R}^{n_p}$ into $\mathbb{R}^{n_y}$. Note that this class is extremely large, since inclusion functions exist for the solution of differential equations. We shall say that a box $[\boldsymbol{p}]$ of $\mathbb{R}^{n_p}$ is *feasible* if $[\boldsymbol{p}] \subset \mathbb{S}$ and *unfeasible* if $[\boldsymbol{p}] \cap \mathbb{S} = \emptyset$; else, $[\boldsymbol{p}]$ is *ambiguous*.

The principle of SEVIA is as follows.

(i) Define a prior feasible box $[\boldsymbol{p}](0)$ within which the search will be performed.

(ii) Compute a paving $\mathbb{P}$ of $[\boldsymbol{p}](0)$.

(iii) Compute $\mathbb{Y}_m([\boldsymbol{p}])$ for each box of this paving. Three situations must then be considered (Fig. 2).

$$(1) \qquad \mathbb{Y}_m([\boldsymbol{p}]) \subset \mathbb{Y} \Rightarrow [\boldsymbol{p}] \subset \mathbb{S}, \text{ so that } [\boldsymbol{p}] \text{ is feasible.} \tag{10}$$

(The implication becomes an equivalence relation if $\mathbb{Y}_m = [\boldsymbol{y}_m]$ and $\mathbb{E}$ is a box.)

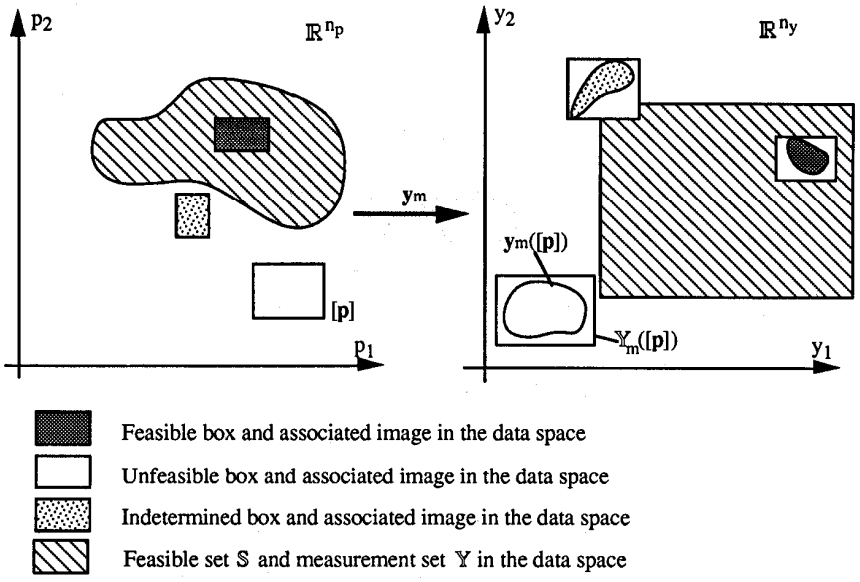

Fig. 2. Feasibility of boxes.

(2) $\mathbb{Y}_m([\boldsymbol{p}]) \cap \mathbb{Y} = \emptyset \Rightarrow [\boldsymbol{p}] \cap \mathbb{S} = \emptyset$, so that $[\boldsymbol{p}]$ is unfeasible. (11)

(3) Otherwise, $[\boldsymbol{p}]$ is indetermined (and might be ambiguous).

We shall partition $\mathbb{P}$ into three subpavings $\mathbb{K}_f$, $\mathbb{K}_u$ and $\mathbb{K}_i$, corresponding respectively to the sets of all feasible, unfeasible and indetermined boxes (Fig. 3).

If we denote by $\{\mathbb{K}\}$ the subset of $\mathbb{R}^{n_p}$ generated by the union of all boxes of the subpaving $\mathbb{K}$, then these subpavings satisfy

$$\{\mathbb{K}_f\} \subset \mathbb{S} \subset \{\mathbb{K}_f\} \cup \{\mathbb{K}_i\}, \tag{12}$$

$$\partial\mathbb{S} \subset \{\mathbb{K}_i\}, \tag{13}$$

$$\text{vol}(\{\mathbb{K}_f\}) \leqslant \text{vol}(\mathbb{S}) \leqslant \text{vol}(\{\mathbb{K}_f\}) + \text{vol}(\{\mathbb{K}_i\}), \tag{14}$$

$$[\{\mathbb{K}_f\}] \subset [\mathbb{S}] \subset [\{\mathbb{K}_f\}] \cup [\{\mathbb{K}_i\}], \tag{15}$$

$$\{\mathbb{K}_f\} \text{ nonconnected and } \{\mathbb{K}_f\} \cup \{\mathbb{K}_i\} \text{ nonconnected} \Rightarrow \mathbb{S} \text{ nonconnected}. \tag{16}$$

SEVIA uses a recursive implementation of the principle that has just been described.

3.2. SEVIA

Let $[\boldsymbol{p}](k)$ be the box considered at iteration k. Define the *required accuracy* ϵ_r for the paving $\mathbb{P}$ as the maximum width that an indetermined box can have. SEVIA makes an extensive use of a stack of boxes. A stack is a dynamical structure on which only three operations are possible. One may stack, i.e., put an element on top of the stack, unstack, i.e., remove the element located on top of the stack or test the stack for emptiness. We shall call *principal plane* of a box a symmetry plane of this box that is orthogonal to an axis $i \in \{j \mid w([\boldsymbol{p}]) = w([p_j])\}$.

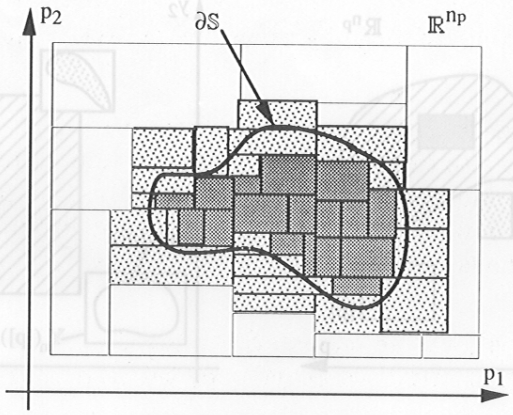

Fig. 3. Enclosure of $\mathbb{S}$ by two subpavings (convention for hatching is as on Fig. 2).

We shall assume that a characteristic $\mathfrak{C}$ of $\mathbb{S}$ is to be estimated, such as its volume, its enveloping box $[\mathbb{S}]$ or its connexity. Let $\mathfrak{E}(\mathbb{K})$ be an exhaustive summary of a subpaving $\mathbb{K}$ with respect to the characteristic $\mathfrak{C}$, i.e., $\mathfrak{E}(\mathbb{K})$ contains all the information contained in $\mathbb{K}$ and needed to compute an estimate of $\mathfrak{C}(\mathbb{S})$. The quantities $\mathfrak{E}(\mathbb{K}_f)$, $\mathfrak{E}(\mathbb{K}_i)$ and $\mathfrak{E}(\mathbb{K}_u)$ evolve at each iteration depending on whether $[\boldsymbol{p}]$ turns out to belong to $\mathbb{K}_f$, $\mathbb{K}_i$ or $\mathbb{K}_u$. This evolution is characterized by the operator $\mathfrak{O}$ that depends on $\mathfrak{E}$ which itself depends on $\mathfrak{C}$.

SEVIA can be described as follows:

Program inputs

Data: $\boldsymbol{y}$;
Inclusion function: $\mathbb{Y}_m$;
Feasible error set: $\mathbb{E}$;
Prior feasible box: $[\boldsymbol{p}](0)$;
Required accuracy for the paving: ϵ_r;
Characteristic of $\mathbb{S}$ to be estimated: $\mathfrak{C}$;
Exhaustive summary to be used: $\mathfrak{E}$;
Evolution operator: $\mathfrak{O}$;

Initialization

$\mathbb{Y} = \boldsymbol{y} - \mathbb{E}$;
stack $= \emptyset$;
$k = 0$;
$[\boldsymbol{p}] = [\boldsymbol{p}](0)$;
$\mathfrak{E}(\mathbb{K}_f) = \mathfrak{E}(\mathbb{K}_i) = \mathfrak{E}(\mathbb{K}_u) = \mathfrak{E}(\emptyset)$;

Iteration k

Step 1: If $\mathbb{Y}_m([\boldsymbol{p}](k)) \subset \mathbb{Y}$, then $\mathfrak{E}(\mathbb{K}_f) = \mathfrak{O}(\mathfrak{E}(\mathbb{K}_f), [\boldsymbol{p}](k))$.
Step 2: Else, if $\mathbb{Y}_m([\boldsymbol{p}](k)) \cap \mathbb{Y} = \emptyset$, then $\mathfrak{E}(\mathbb{K}_u) = \mathfrak{O}(\mathfrak{E}(\mathbb{K}_u), [\boldsymbol{p}](k))$.
Step 3: Else, if $w([\boldsymbol{p}](k)) \leqslant \epsilon_r$, then $\mathfrak{E}(\mathbb{K}_i) = \mathfrak{O}(\mathfrak{E}(\mathbb{K}_i), [\boldsymbol{p}](k))$.
Else bisect $[\boldsymbol{p}](k)$ along a principal plane and stack the two resulting boxes.
Step 4: If the stack is not empty, then unstack and store the resulting box in $[\boldsymbol{p}](k+1)$, increment k by one and go to Step 1.

Step 5: Compute the estimate of $\mathfrak{C}(\mathbb{S})$ by using $\mathfrak{E}(\mathbb{K}_f)$, $\mathfrak{E}(\mathbb{K}_u)$ and $\mathfrak{E}(\mathbb{K}_i)$.
End.

Special care must be taken to avoid memorizing unnecessary information by use of suitable exhaustive summaries. Otherwise the quantity of memory required to store the paving of $\mathbb{S}$ will increase linearly at each iteration, which may result into a memory overflow even for problems of modest dimension. Note however that the paving must be explored by SEVIA even if it is not stored, so that using exhaustive summaries saves memory but no computing time.

Example 3.1. If $\mathfrak{C}(\mathbb{S})$ is vol($\mathbb{S}$), we can choose $\mathfrak{E}(\mathbb{K}) = \text{vol}(\{\mathbb{K}\})$ and obtain an interval estimate by using (14). The instruction $\mathfrak{E}(\mathbb{K}) = \mathfrak{O}(\mathfrak{E}(\mathbb{K}), [\boldsymbol{p}](k))$ translates into $\text{vol}(\{\mathbb{K}\}) = \text{vol}(\{\mathbb{K}\}) + \text{vol}([\boldsymbol{p}](k))$.

Example 3.2. If $\mathfrak{C}(\mathbb{S})$ is $[\mathbb{S}]$, we can choose $\mathfrak{E}(\mathbb{K}) = [\{\mathbb{K}\}]$ and obtain an interval estimate by using (15). The instruction $\mathfrak{E}(\mathbb{K}) = \mathfrak{O}(\mathfrak{E}(\mathbb{K}), [\boldsymbol{p}](k))$ translates into $[\{\mathbb{K}\}] = [[\{\mathbb{K}\}] \cup [\boldsymbol{p}](k)]$.

Remark 3.3. It frequently happens that the parameter space is not isotropic because the sensitivities of $\boldsymbol{y}_m$ with respect to the various components of $\boldsymbol{p}$ are not of the same order of magnitude. The basic bisection technique suggested in the description of the algorithm may then not be efficient enough. The problem is then to choose a bisection policy that results in a convergence as rapid as possible. An idea would be to use weighted widths so as make the problem as isotropic as possible, but it seems difficult to suggest a rational policy for the choice of the corresponding weights. Limiting ourselves to bisections of $[\boldsymbol{p}]$ into boxes $[\boldsymbol{p}_1]$ and $[\boldsymbol{p}_2]$ along a symmetry plane, we suggest the bisection that minimizes $\text{vol}(\mathbb{Y}_m([\boldsymbol{p}_1])) + \text{vol}(\mathbb{Y}_m([\boldsymbol{p}_2]))$. If $[\boldsymbol{p}]$ is not ambiguous, this policy will thus tend to avoid classifying $[\boldsymbol{p}_1]$ and $[\boldsymbol{p}_2]$ as indetermined. Preliminary experiments tend to indicate that this can improve the efficiency of SEVIA quite spectacularly when the anisotropy is severe.

3.3. Properties of SEVIA

3.3.1. Convergence

Let k_f be the total number of iterations performed. From Step 3, we have

$$\text{if } [\boldsymbol{p}] \in \mathbb{K}_i(k_f), \text{ then } w([\boldsymbol{p}]) \leqslant \epsilon_r. \tag{17}$$

Assume that a Lipschitz constant κ can be associated with the inclusion function for the model output. It satisfies

$$\kappa > \lim_{\epsilon_r \to 0} \sup_{[\boldsymbol{p}]_{\epsilon_r} \subset [\boldsymbol{p}](0)} \frac{w\left(\mathbb{Y}_m([\boldsymbol{p}]_{\epsilon_r})\right)}{\epsilon_r}, \tag{18}$$

where $[\boldsymbol{p}]_{\epsilon_r}$ is a box with width smaller than ϵ_r. For ϵ_r small enough, we have

$$w\left(\mathbb{Y}_m([\boldsymbol{p}]_{\epsilon_r})\right) \leqslant \kappa \epsilon_r = \epsilon_y. \tag{19}$$

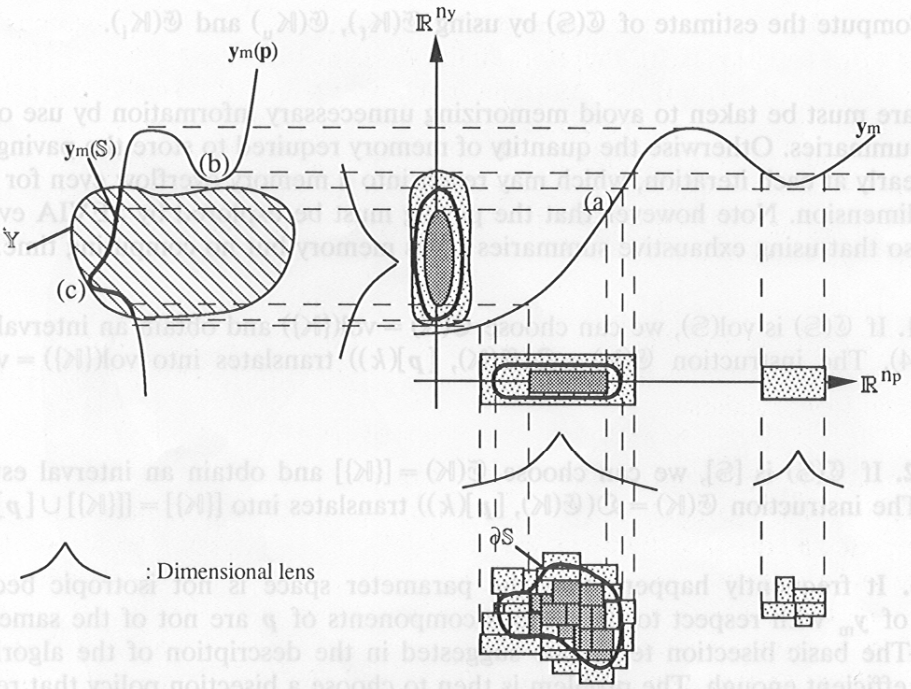

Fig. 4. Discontinuity of $y_m^{-1}(\mathbb{Y})$ (convention for hatching is as on Fig. 2).

Theorem 3.4. *If* $\boldsymbol{p} \in \{\mathbb{K}_i(k_f)\}$, *then* $d_\infty(\boldsymbol{y}_m(\boldsymbol{p}), \partial\mathbb{Y}) \leqslant \epsilon_y$.

Proof. If $\boldsymbol{p} \in \{\mathbb{K}_i(\boldsymbol{k}_f)\}$, then from (17), $\exists[\boldsymbol{p}]_{\epsilon_r} \in \mathbb{K}_i(k_f)$ such that $\boldsymbol{p} \in [\boldsymbol{p}]_{\epsilon_r}$. As $[\boldsymbol{p}]_{\epsilon_r}$ is indetermined, $[\boldsymbol{p}]_{\epsilon_r}$ does not satisfy (11) and therefore $\mathbb{Y}_m([\boldsymbol{p}]_{\epsilon_r}) \cap \mathbb{Y} \neq \emptyset$. The box $\mathbb{Y}_m([\boldsymbol{p}]_{\epsilon_r})$ connects $\boldsymbol{y}_m(\boldsymbol{p})$ and $\mathbb{Y}$. Using (19) and the definition of d_∞, we have $d_\infty(\boldsymbol{y}_m(\boldsymbol{p}), \mathbb{Y}) \leqslant \epsilon_y$. Similarly, if $\overline{\mathbb{Y}}$ denotes the complement of $\mathbb{Y}$ in the data space, if $\boldsymbol{p}$ belongs to $\{\mathbb{K}_i(k_f)\}$, then $d_\infty(\boldsymbol{y}_m(\boldsymbol{p}), \overline{\mathbb{Y}}) \leqslant \epsilon_y$, which completes the proof. □

Remark 3.5. When it is possible to compute a Lipschitz constant κ and information is available on the reliability of the error bounds, the relation $\epsilon_y = \kappa\epsilon_r$ can be used as an aid to choose ϵ_r, as illustrated by the upper right side of Fig. 4. In this simple case, the Lipschitz constant κ may correspond to the slope around (a). For any prespecified value of ϵ_y, we can therefore compute ϵ_r.

Theorem 3.6. *For almost any* $\mathbb{Y}$, $\mathbb{K}_f(k_f)$ *and* $\mathbb{K}_i(k_f)$ *satisfy*:

$$\{\mathbb{K}_f(k_f)\} \xrightarrow{\subset} \mathbb{S}, \tag{20}$$

$$\{\mathbb{K}_f(k_f) \cup \mathbb{K}_i(k_f)\} \xrightarrow{\supset} \mathbb{S}, \tag{21}$$

$$\{\mathbb{K}_i(k_f)\} \xrightarrow{\supset} \partial\mathbb{S}, \tag{22}$$

$$\mathrm{vol}(\{\mathbb{K}_i(k_f)\}) \to 0, \tag{23}$$

when ϵ_r *tends to zero.*

Proof. The detailed proof uses techniques similar to those used in [18] and exceeds the space available for this paper. We shall therefore limit ourselves to sketching it. Let $[\boldsymbol{p}]_{\epsilon_r}$ be a box with width ϵ_r tending to zero. If the algorithm splits this box into two smaller ones, this means that $[\boldsymbol{p}]_{\epsilon_r}$ is indetermined, i.e., that $\mathbb{Y}_m([\boldsymbol{p}]_{\epsilon_r})$ does not belong to the measurement set $\mathbb{Y}$ and that the intersection between $\mathbb{Y}_m([\boldsymbol{p}]_{\epsilon_r})$ and $\mathbb{Y}$ is not void. Since $\mathbb{Y}_m([\boldsymbol{p}]_{\epsilon_r})$ is infinitely small, from Theorem 3.4 it must be d_∞-close to $\partial\mathbb{Y}$, so that $[\boldsymbol{p}]_{\epsilon_r}$ is almost surely d_∞-close to $\partial\mathbb{S}$. We then have (22), which implies (23) since the volume function is continuous in $\mathscr{C}(\mathbb{R}^n)$. From (12), this in turn implies (20) and (21). □

Remark 3.7. An atypical situation where Theorem 3.6 does not apply is when $\partial\mathbb{Y}$ contains a nonempty open subset of the image submanifold of $\boldsymbol{y}_m$. Then $\mathrm{vol}(\boldsymbol{y}_m^{-1}(\partial\mathbb{Y})) \neq 0$, so that (20), (22) and (23) are no longer satisfied.

3.3.2. Discontinuity of the reciprocal model $\boldsymbol{y}_m^{-1}$

From (12), the quality of the knowledge of $\mathbb{S}$ provided by the algorithm could be quantified by computing the h_∞-distance between $\{\mathbb{K}_f\}$ and $\{\mathbb{K}_f\} \cup \{\mathbb{K}_i\}$. It must be stressed that this distance can be large even when ϵ_r is small. This is due to the possible discontinuity of the reciprocal model that we shall now consider. Any d_∞-continuous model $\boldsymbol{y}_m : \mathbb{R}^{n_p} \to \mathbb{R}^{n_y}$ can be extended as an h_∞-continuous model $\boldsymbol{y}_m : \mathscr{C}(\mathbb{R}^{n_p}) \to \mathscr{C}(\mathbb{R}^{n_y})$. Even when $\boldsymbol{y}_m$ is h_∞-continuous, $\boldsymbol{y}_m^{-1}$ can be h_∞-discontinuous as illustrated by situation (b) on Fig. 4; it is possible to find $\mathbb{Y}$ and a neighbor $\mathbb{Y} + d\mathbb{Y}$ such that $h_\infty(\mathbb{Y} + d\mathbb{Y}, \mathbb{Y})$ tends to zero while $h_\infty(\boldsymbol{y}_m^{-1}(\mathbb{Y} + d\mathbb{Y}), \boldsymbol{y}_m^{-1}(\mathbb{Y}))$ does not. On the other hand, a small variation $d\mathbb{Y}$ of $\mathbb{Y}$ around (c) could result in a hole in $\mathbb{S}$, but $h_\infty(\boldsymbol{y}_m^{-1}(\mathbb{Y} + d\mathbb{Y}), \boldsymbol{y}_m^{-1}(\mathbb{Y}))$ would remain small. Note that if $\boldsymbol{y}_m$ is d_∞-continuous, the h_∞-discontinuity zone has a zero measure, so that $\boldsymbol{y}_m^{-1}$ is h_∞-piecewise continuous. Situations such as described on Fig. 4(b) do not contradict Theorem 3.6, which is valid only in the continuity zones. For almost any *given* $\mathbb{Y}$, it is possible to choose ϵ_r small enough for the problem to disappear.

3.3.3. Complexity

At each iteration k, $\mathbb{P}(k) = [\boldsymbol{p}](k) \cup \mathrm{stack}(k) \cup \mathbb{K}_f(k) \cup \mathbb{K}_i(k) \cup \mathbb{K}_u(k)$ is a paving of $[\boldsymbol{p}](0)$. If ϵ_r tends to zero, then the paving $\mathbb{P}(\epsilon_r) = \mathbb{P}(k_f(\epsilon_r))$ accumulates on $\partial\mathbb{S}$ which is generally of dimension $n_p - 1$. As $\mathbb{P}$ is a paving of $[\boldsymbol{p}](0)$ that describes the compact submanifold $\partial\mathbb{S}$ with a precision of ϵ_r, and such that no box of $\mathbb{P}$ has an edge smaller than ϵ_r, Theorem 2.17 then indicates that $1/(\epsilon_r)^{n_p-1} \leftarrow \mathrm{card}(\mathbb{P}) \leftarrow 1/(\epsilon_r)^{n_p}$. The main limitation of SEVIA lies in the increase of the computing time (which is proportional to $\mathrm{card}(\mathbb{P})$) with the number of parameters, and is due to the necessity of accumulating the paving on the boundary of $\mathbb{S}$. It seems impossible to give a precise limit to the complexity of the problems that can be handled, because it depends on a number of factors, such as the characteristics of $\mathbb{S}$ (shape, size, area of $\partial\mathbb{S}, \ldots$) and the relative position of $\boldsymbol{y}_m(\mathbb{R}^{n_p})$ and $\mathbb{Y}$.

3.3.4. Memory used

In the cases considered here, such as the computation of $[\mathbb{S}]$ or $\mathrm{vol}(\mathbb{S})$, only the stack takes a significant place in memory. This place is extraordinarily small as evidenced by the following theorem.

Theorem 3.8. *The number of elements in the stack satisfy*

$$\text{card(stack)} < n_p \text{int}(\log_2(w([\boldsymbol{p}](0))) - \log_2(\epsilon_r) + 1).$$

Proof. If $\text{card(stack)} \geqslant kn_p$, then the smallest box (which is stored on the top of the stack) has a width $\epsilon \leqslant 2^{-k} w_1$, where w_1 is the width of the largest box (which is stored on the bottom of the stack). Since $w_1 \leqslant w([\boldsymbol{p}](0))$, we then have the following proposition:

$$\text{card(stack)} \geqslant kn_p \Rightarrow \exists [\boldsymbol{p}] \in \text{stack}, w([\boldsymbol{p}]) \leqslant 2^{-k} w([\boldsymbol{p}](0)). \tag{24}$$

The contrapositive of (24) is

$$\forall [\boldsymbol{p}] \in \text{stack}, w([\boldsymbol{p}]) > 2^{-k} w([\boldsymbol{p}](0)) \Rightarrow \text{card(stack)} < kn_p. \tag{25}$$

Now $\forall [\boldsymbol{p}] \in \text{stack}, w([\boldsymbol{p}]) \geqslant \epsilon_r > 2^{-k} w([\boldsymbol{p}](0))$ for any $k > \log_2(w([\boldsymbol{p}](0))) - \log_2(\epsilon_r)$ and therefore for $k = \text{int}(\log_2(w([\boldsymbol{p}](0))) - \log_2(\epsilon_r) + 1)$. Using proposition (25), $\text{card(stack)} < n_p \text{int}(\log_2(w([\boldsymbol{p}](0))) - \log_2(\epsilon_r) + 1)$. □

Even for a very large number of parameters, the maximum size of the stack will remain quite reasonable. For instance, if $n_p = 100$, $w([\boldsymbol{p}](0)) = 10^4$ and $\epsilon_r = 10^{-10}$, Theorem 3.8 implies that $\text{card(stack)} < 4600$.

4. Example

To illustrate the behavior of SEVIA, we consider a two-parameter estimation problem, which makes it possible to draw pictures of the paving obtained. In this example, the vector of all available data $\boldsymbol{y}$ has been taken from [12]:

$$\boldsymbol{y} = (7.39, 4.09, 1.74, 0.097, -2.57, -2.71, -2.07, -1.44, -0.98, -0.66)^t. \tag{26}$$

It corresponds to ten scalar measurements, taken at times

$$\boldsymbol{t} = (0.75, 1.5, 2.25, 3, 6, 9, 13, 17, 21, 25)^t. \tag{27}$$

The error between the model output and the corresponding data is defined as

$$\boldsymbol{e}(\boldsymbol{p}) = \boldsymbol{y} - \boldsymbol{y}_m(\boldsymbol{p}). \tag{28}$$

The vector $\boldsymbol{y}_m(\boldsymbol{p})$ of all model outputs is given by

$$\boldsymbol{y}_m(\boldsymbol{p}) = 20 \exp(-p_1 \boldsymbol{t}) - 8 \exp(-p_2 \boldsymbol{t}), \tag{29}$$

where the ith component of $\boldsymbol{y}_m(\boldsymbol{p})$ is computed for the ith component of $\boldsymbol{t}$. To be acceptable, the measurement error $\boldsymbol{e}$ must satisfy

$$\boldsymbol{e} \in \mathbb{E} = [\boldsymbol{e}] = [-\boldsymbol{e}_{\max}, \boldsymbol{e}_{\max}], \tag{30}$$

with

$$\boldsymbol{e}_{\max} = 0.5 |\boldsymbol{y}| + \boldsymbol{1}, \tag{31}$$

References

[1] G. Belforte, B. Bona and V. Cerone, Parameter estimation algorithms for set membership description of uncertainty, *Automatica* 26 (1990) 887–898.

[2] G. Belforte and M. Milanese, Uncertainty interval evaluation in presence of unknown-but-bounded errors: nonlinear families of models, in: *Proc. 1st IASTED Internat. Symp. Modeling, Identification, Control*, Davos (1981) 75–79.

[3] M. Berger, *Espace Euclidien, Triangle, Cercles et Sphères; Géometrie 2* (Cedic/Fernand Nathan, Paris, 1979) 127–129.

[4] V. Broman and M.J. Shensa, A compact algorithm for the intersection and approximation of N-dimensional polytopes, *Math. Comput. Simulation* 32 (5&6) (1990) 469–480.

[5] G. Cherbit, *Fractals* (Masson, Paris, 1987) 96–97.

[6] J.E. Falk, Global solutions for signomial programs, Tech. Report T-274, George Washington Univ., Washington, DC, 1973.

[7] E. Fogel and Y.F. Huang, On the value of information in system identification — bounded noise case, *Automatica* 18 (1982) 229–238.

[8] IBM, High-accuracy arithmetic subroutine library (ACRITH), Program description and user's guide, SC 33-6164-02, 3rd ed., 1986.

[9] D.E. Knuth, *Fundamental Algorithms* (Addison-Wesley, Stanford, CA, 2nd ed., 1977) 305–346.

[10] U. Kulisch, ed., *PASCAL-SC: A PASCAL Extension for Scientific Computation, Information Manual and Floppy Disks* (Teubner, Stuttgart, 1987).

[11] M. Milanese and G. Belforte, Estimation theory and uncertainty intervals evaluation in presence of unknown-but-bounded errors. Linear families of models and estimators, *IEEE Trans. Automat. Control* AC-27 (1982) 408–414.

[12] M. Milanese and A. Vicino, Estimation theory for nonlinear models and set membership uncertainty, *Automatica* 27 (1991) 403–408.

[13] S.H. Mo and J.P. Norton, Fast and robust algorithm to compute exact polytope parameter bounds, *Math. Comput. Simulation* 32 (5&6) (1990) 481–493.

[14] R.E. Moore, *Methods and Applications of Interval Analysis*, SIAM Stud. Appl. Math. (SIAM, Philadelphia, PA, 1979).

[15] R. Moore, Parameter sets for bounded-error data, *Math. Comput. Simulation* 34 (2) (1992) 113–119.

[16] A. Neumaier, *Interval Methods for Systems of Equations* (Cambridge Univ. Press, Cambridge, 1990).

[17] H. Piet-Lahanier and E. Walter, Characterization of non-connected parameter uncertainty regions, *Math. Comput. Simulation* 32 (5&6) (1990) 553–560.

[18] A. Ratschek and J. Rokne, *New Computer Methods for Global Optimization* (Ellis Horwood, Chichester, UK/Wiley, New York, 1988).

[19] E. Walter, ed., Special Issue on Parameter Identifications with Error Bound, *Math. Comput. Simulation* 32 (5&6) (1990) 447–607.

[20] E. Walter and H. Piet-Lahanier, Exact recursive polyhedral description of the feasible parameter set for bounded error models, *IEEE Trans. Automat. Control.* AC-34 (1989) 911–915.

[21] E. Walter and H. Piet-Lahanier, Estimation of parameters bounds from bounded-error data: a survey, *Math. Comput. Simulation* 32 (5&6) (1990) 449–468.