Interval analysis for non-linear parameter and state estimation: contributions and limitations

Michel Kieffer, Eric Walter, Isabelle Braems, Luc Jaulin

INTERVAL ANALYSIS FOR NONLINEAR PARAMETER AND STATE ESTIMATION: CONTRIBUTIONS AND LIMITATIONS

Michel Kieffer * **Eric Walter** * **Isabelle Braems** *
Luc Jaulin **

* *Laboratoire des Signaux et Systèmes*
CNRS – SUPELEC – Université Paris-Sud
91192 Gif-sur-Yvette FRANCE
** *Laboratoire d'Ingénierie des Systèmes Automatisés*
Université d'Angers
49045 Angers FRANCE

Abstract: Parameter or state estimation should take into account the fact that the model is an approximation of reality and that the data are corrupted by noise. In this paper, each uncertain quantity is assumed to belong to a known set. The problem, known as *parameter or state bounding*, is then to characterize the *set* of all parameter or state vectors that are consistent with the model structure, data and error bounds. A description of how interval analysis can be used to find guaranteed estimates in a nonlinear context is provided. The main notions of interval analysis are first recalled very briefly. The simpler problem of parameter estimation is then considered. State estimation, which contains parameter tracking as a special case, is treated next. A simple illustrative example is finally presented.

Keywords: bounding methods, constraints, guaranteed methods, identification, intervals, nonlinear estimation, parameter estimation, state estimation

1. INTRODUCTION

Parameter or state estimation should take into account the fact that the model is an approximation of reality and that the data are corrupted by noise. Two main approaches are available for this purpose. One is based on the *minimization of a cost function*, which may be deduced from a probabilistic description of uncertainty, for instance by a maximum-likelihood or Bayesian approach, see, *e.g.*, (Walter and Pronzato 1997). The problem is then to characterize the set $\mathbb{S}$ of *all global* minimizers of this cost function. The other approach assumes that each uncertain quantity belongs to a known set, which leads to *set-membership estimation*, also known as *parameter or state bounding*. The problem is then to characterize the set $\mathbb{S}$ of all parameter or state vectors that are consistent with the model structure, data and error bounds, see, *e.g.*, (Chernousko 1994), (Milanese *et al.* 1996) and (Kurzhanski and Valyi 1997).

In both approaches, the algorithms used to perform these tasks differ radically depending on whether the model output is linear in the unknown parameters or state variables to be estimated. The linear case is much simpler and well documented. In the nonlinear case considered here, difficulties accumulate. Iterative optimization algorithms may get trapped and converge to local minimizers of the cost function associated with maximum-likelihood estimation. Moreover, the propagation of a Gaussian state perturbation in a nonlinear model results in a loss of normality which makes a probabilistic interpretation of the results questionable. In set-membership estima-

tion $\mathbb{S}$ is the set of all unfalsified vector estimates, with no other probabilistic interpretation, so this difficulty is avoided, but $\mathbb{S}$ may be nonconvex or even nonconnected and thus particularly difficult to characterize in a guaranteed way.

This paper describes how interval analysis (IA) can be used to find guaranteed estimates in this difficult nonlinear context. The main notions of interval analysis are first recalled very briefly. The simpler problem of parameter estimation is then considered, and the reason why set-membership estimation is particularly suitable for treatment by IA is explained. State estimation, which contains parameter tracking as a special case, is then treated in the bounding context. A simple illustrative example is finally presented, which is solved using software downloadable from our web site at `www.lss.supelec.fr/books/interval`.

2. INTERVAL ANALYSIS

Let $\mathbb{IR}^n$ be the set of all the vector intervals or *boxes* of $\mathbb{R}^n$ and let $\mathbb{S}$ be a set of $\mathbb{R}^n$. IA allows guaranteed conclusions to be reached about the properties of $\mathbb{S}$ after a finite number of floating-point computations, although the vectors in $\mathbb{S}$ may not even be denumerable. This is made possible by replacing calculation on $\mathbb{S}$ by computation on boxes. Three basic tasks are (i) computing a box guaranteed to contain the image of a box $[\mathbf{x}]$ by a function $\mathbf{f}(.)$, (ii) testing whether $[\mathbf{x}] \subset \mathbb{S}$ or whether $[\mathbf{x}] \cap \mathbb{S} = \emptyset$, (iii) contracting $[\mathbf{x}]$ with respect to $\mathbb{S}$, *i.e.*, replacing $[\mathbf{x}]$ by a smaller box $[\mathbf{z}]$, such that $[\mathbf{x}] \cap \mathbb{S} = [\mathbf{z}] \cap \mathbb{S}$.

The key to Task (i) is the notion of *inclusion function*. Let $\mathbf{f}(.)$ be a function from $\mathbb{R}^n$ to $\mathbb{R}^{n'}$. The function $[\mathbf{f}](.)$ from $\mathbb{IR}^n$ to $\mathbb{IR}^{n'}$ is an inclusion function for $\mathbf{f}(.)$ if for any $[\mathbf{x}]$ in $\mathbb{IR}^n$, $\mathbf{f}([\mathbf{x}]) \subset [\mathbf{f}]([\mathbf{x}])$. Various methods can be employed to build inclusion functions for a huge class of functions, which may be defined by analytic expressions, by algorithms or by differential equations, see (Moore 1979), (Ratschek and Rokne 1984), (Berz and Makino 1998). At least in principle, Task (ii) is trivial when an inclusion function is available. Important tools to perform Task (iii) are the interval Newton method (for the case $n' = n$) and *interval constraint propagation* (ICP) for the general case, see (Davis 1987), (Sam-Haroud and Faltings 1996) and (Jaulin 2000).

Based on (i) to (iii), it becomes possible to obtain guaranteed solutions to problems such as finding all the global minimizers $\widehat{\mathbf{x}}$ of a cost function in a given search box $\mathbb{X}$, or the set $\mathbb{S}$ of all vectors $\mathbf{x}$ in $\mathbb{X}$ satisfying a set of equality and inequality constraints. Guaranteed means here that outer (and sometimes inner) approximations of the solution set are being built, and the quality of these approximations can be improved at the cost of increasing the computational effort. Solving sets of equations or inequalities and optimizing cost functions are, of course, at the core of many problems in engineering and control (Jaulin *et al.* (to appear)); we shall concentrate here on parameter and state estimation.

3. PARAMETER ESTIMATION

To estimate the vector $\mathbf{x}$ of the parameters of a mathematical model from a numerically known vector $\mathbf{y}$ of experimental data, the usual approach is to define a cost function $c(\mathbf{x}, \mathbf{y})$ to be minimized with respect to $\mathbf{x}$. This cost function may be deduced by a maximum-likelihood approach from prior information or hypotheses about the noise corrupting the data, but other approaches may also be employed. Except in some very special cases, there is no explicit solution to this minimization problem, and no guarantee that an estimate obtained by iterative local techniques is indeed a global minimizer of $c(., \mathbf{y})$. There may even be several values of $\mathbf{x}$ that lead to exactly the same value of the cost. Random search may avoid being trapped around local minimizers but again no guarantee can usually be provided about results obtained in finite time. On the other hand, optimization algorithms based on IA compute a list $\mathcal{L}$ of boxes in parameter space that are guaranteed to contain all the global minimizers. Hansen's algorithm for unconstrained optimization, for instance, uses inclusion functions for the cost, its gradient and the diagonal entries of its Hessian matrix to eliminate boxes and the interval Newton method to contract boxes that resist elimination. The surviving contracted boxes are then split into subboxes before being put back in the list of boxes still to be studied, and the algorithm stops when all remaining boxes are deemed too small to be worth further bisection. For more details, see (Hansen 1992), entirely devoted to the subject.

Very often, $c(\mathbf{x}, \mathbf{y})$ is a possibly weighted sum of the squares of the differences between the components of $\mathbf{y}$ and a corresponding model output $\boldsymbol{\eta}(\mathbf{x})$, which means that there are many occurrences of $\mathbf{x}$ in the formal expression for $c(\mathbf{x}, \mathbf{y})$. This situation is not favorable to interval analysis, because in interval computation the fact that $\mathbf{x}$ takes the same value in all of these occurrences is forgotten, and each occurrence of $[\mathbf{x}]$ is treated as if it were independent of the others. This is an additional incentive to considering the alternative approach known as set-membership estimation or parameter bounding. In this approach, it is assumed that the set $\mathbb{S}$ to be estimated is defined by a set of inequalities, for instance,

$$\mathbb{S} = \{\mathbf{x} \in \mathbb{X} \subset \mathbb{R}^n \mid |y_i - \eta_i(\mathbf{x})| \leqslant \overline{e}_i, \quad i = 1, \ldots, \overline{i}\} \tag{1}$$

where $\mathbb{X}$ is some (possibly very large) prior feasible set also defined by inequalities and $\overline{e}_i$ is some bound on the acceptable absolute error between the ith datum and the corresponding model output. It is then assumed that an inclusion function $[\mathbf{e}](.)$ is available for the function $\mathbf{e}(.)$ defined by

$$\mathbf{e}(\mathbf{x}) = \mathbf{y} - \boldsymbol{\eta}(\mathbf{x}). \tag{2}$$

Note that the number of occurrences of $\mathbf{x}$ in each component of $\mathbf{e}(\mathbf{x})$ is much lower than in $c(\mathbf{x}, \mathbf{y})$, so more efficient inclusion functions can be obtained. $\mathbb{S}$ can now be defined as

$$\begin{aligned}\mathbb{S} &= \{\mathbf{x} \in \mathbb{X} \subset \mathbb{R}^n \mid \mathbf{e}(\mathbf{x}) \in \mathbb{E}\} \\ &= \mathbf{e}^{-1}(\mathbb{E}) \cap \mathbb{X},\end{aligned} \tag{3}$$

where $\mathbb{X}$ is some prior search domain included in $\mathbb{R}^n$ and $\mathbb{E}$ is the prior feasible set for the error $\mathbf{e}$. Characterizing $\mathbb{S}$ can then be viewed as a problem of set inversion.

The algorithm SIVIA (for set inversion via interval analysis) partitions $\mathbb{X}$ into three subpavings (unions of nonoverlapping boxes). These subpavings are $\mathbb{S}_{\text{in}}$ contained in $\mathbb{S}$, $\mathbb{S}_{\text{out}}$ such that its intersection with $\mathbb{S}$ is empty and $\mathbb{S}_{\text{bound}}$ for which no conclusion could be reached (Jaulin and Walter 1993*a*), (Jaulin and Walter 1993*b*). $\mathbb{S}$ is thus bracketed (in the sense of inclusion) between $\mathbb{S}_{\text{in}}$ and $\mathbb{S}_{\text{in}} \cup \mathbb{S}_{\text{bound}}$. The volume of the uncertainty subpaving $\mathbb{S}_{\text{bound}}$ may be reduced at the cost of increasing the computational effort. The results are global within $\mathbb{X}$, as illustrated by the example in (Braems *et al.* 2001), where the parameters of a nonuniquely identifiable model were estimated.

4. STATE ESTIMATION

Consider now a nonlinear and possibly time-varying system described by the state equation

$$\mathbf{x}_{\ell+1} = \mathbf{f}_\ell(\mathbf{x}_\ell, \mathbf{w}_\ell), \quad \ell = 0, 1, \ldots \tag{4}$$

and the observation equation

$$\mathbf{y}_\ell = \mathbf{h}_\ell(\mathbf{x}_\ell) + \mathbf{v}_\ell, \quad \ell = 0, 1, \ldots \tag{5}$$

where $\mathbf{x}_\ell \in \mathbb{R}^n$ is the state vector and $\mathbf{y}_\ell \in \mathbb{R}^p$ the output vector. Assume that the state perturbation $\mathbf{w}_\ell$ belongs to the known box $[\mathbf{w}]_\ell$, and that the measurement noise $\mathbf{v}_\ell$ belongs to the known box $[\mathbf{v}]_\ell$. The noise-free output $\mathbf{y}_\ell - \mathbf{v}_\ell$ is then guaranteed to belong to

$$\mathbb{Y}_\ell = \mathbf{y}_\ell - [\mathbf{v}]_\ell \tag{6}$$

The functions $\mathbf{f}_\ell$ and $\mathbf{h}_\ell$ are known, and may be defined by finite algorithms. Two problems are considered in this section. The first one is the *causal estimation* of a set guaranteed to contain $\mathbf{x}_\ell$ based on the information available up to the time indexed by ℓ, and the second one is the *noncausal estimation* of $\mathbf{x}_\ell$ using all the information available, including measurements posterior to the time indexed by ℓ. Of course, in real time, only causal estimation can be employed, but when the data are to be processed off-line, there is no reason for not using the entire set of available measurements to estimate the state at any given time.

Remark 1. The function $\mathbf{f}_\ell$ may of course depend on a vector of inputs $\mathbf{u}_\ell \in \mathbb{R}^m$, omitted in (4) for the sake of simplicity. ■

Remark 2. The more general problem of joint state and parameter estimation can easily be treated in this context. It suffices to replace $\mathbf{x}_\ell$ by an *extended state* $\mathbf{x}_\ell^{\text{e}} = (\mathbf{x}_\ell^{\text{T}}, \mathbf{p}_\ell^{\text{T}})^{\text{T}}$ incorporating the unknown parameter vector $\mathbf{p}_\ell \in \mathbb{R}^q$. An evolution equation for the parameters is then needed; it may, for instance, be $\mathbf{p}_{\ell+1} = \mathbf{p}_\ell + \mathbf{w}_\ell^p$, with $\mathbf{w}_\ell^p$ in some known box $[\mathbf{w}^p]_\ell$. ■

4.1 *Causal state estimation*

Figure 1 illustrates the principle of one iteration of the recursive causal state estimator (RCSE) described by Table 1. At ℓ, the state is known to belong to $\mathbb{X}_\ell$. The *predicted set* $\mathbb{X}_{\ell+} = \mathbf{f}_\ell(\mathbb{X}_\ell, [\mathbf{w}]_\ell)$ thus contains all possible values of $\mathbf{x}_{\ell+1}$. When a measurement becomes available at $\ell + 1$, $\mathbf{h}_{\ell+1}^{-1}(\mathbb{Y}_{\ell+1})$ contains all state vectors that could have led to a noise-free output belonging to $\mathbb{Y}_{\ell+1}$. Thus, the state belongs to the *corrected set* $\mathbb{X}_{\ell+1} = \mathbb{X}_{\ell+} \cap \mathbf{h}_{\ell+1}^{-1}(\mathbb{Y}_{\ell+1})$.

Table 1. Recursive causal state estimator

Algorithm RCSE(in: $\mathbb{X}_0$; out: $\mathbb{X}_1, \ldots, \mathbb{X}_{\overline{\ell}}$)	
1	for $\ell := 0$ to $\overline{\ell} - 1$,
2	$\mathbb{X}_{\ell+} = \mathbf{f}_\ell(\mathbb{X}_\ell, [\mathbf{w}]_\ell)$; // prediction
3	wait for $\mathbb{Y}_{\ell+1}$;
4	$\mathbb{X}_{\ell+1} = \mathbb{X}_{\ell+} \cap \mathbf{h}_{\ell+1}^{-1}(\mathbb{Y}_{\ell+1})$; // correction

To estimate $\mathbf{x}_\ell$, it suffices to supplement a correction step based on the algorithm SIVIA for parameter estimation presented in Section 3 with a prediction step. This prediction step is implemented using the algorithm IMAGESP (Kieffer *et al.* 2001*a*). IMAGESP encloses the direct image of a set by a given function in a subpaving, with a precision that can be chosen by the user. Alternating prediction with IMAGESP and correction with SIVIA, one gets a guaranteed nonlinear counterpart to Kalman filtering (Kieffer *et al.* 1998).

Remark 3. It is not required that each of the estimated sets consist of a single connected com-

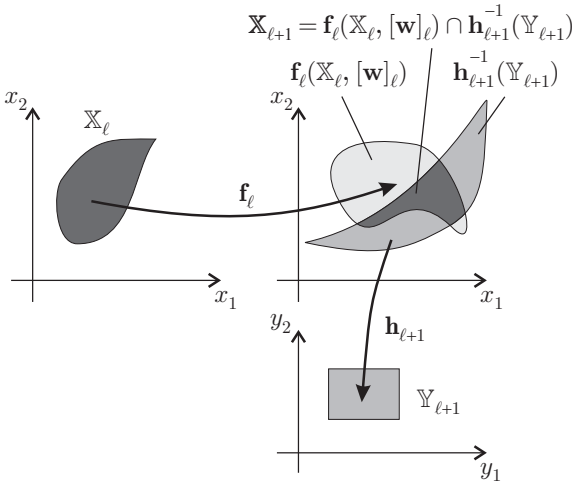

Fig. 1. Principle of recursive state estimation

ponent. Because of the global nature of the results provided, it thus becomes possible to estimate the state of an unobservable model, as illustrated by some problems in robotics (Kieffer *et al.* 2000), (Kieffer *et al.* 2001*b*). ■

4.2 *Noncausal estimation*

The noncausal case is treated in two phases. The *forward phase* computes the set $\mathbb{X}_\ell$ of all the state vectors $\mathbf{x}_\ell$ that are consistent with the past measurements and the initial domain $\mathbb{X}_0$, for $\ell = 1, 2, \ldots$. This corresponds to the algorithm RCSE of Table 1. The *backward phase* eliminates all parts of $\mathbb{X}_\ell$ that are not consistent with the set estimates obtained for $k > \ell$. The resulting procedure is described in Table 2. For more details, see (Jaulin et al. (to appear)).

Table 2. Noncausal set state estimator

	Algorithm NCSE(inout $\mathbb{X}_0$; in: $\mathbb{Y}_1, \ldots, \mathbb{Y}_{\overline{\ell}}$; out $\mathbb{X}_1, \ldots, \mathbb{X}_{\overline{\ell}}$)	
1	for $\ell := 0$ to $\overline{\ell} - 1$,	// forward phase
2	$\mathbb{X}_{\ell+} = \mathbf{f}_\ell(\mathbb{X}_\ell)$;	// prediction
3	$\mathbb{X}_{\ell+1} = \mathbb{X}_{\ell+} \cap \mathbf{h}_{\ell+1}^{-1}(\mathbb{Y}_{\ell+1})$;	// correction
4	for $\ell := \overline{\ell}$ down to 1,	// backward phase
5	$\mathbb{X}_{\ell-} := \mathbf{f}_{\ell-1}^{-1}(\mathbb{X}_\ell)$;	// postdiction
6	$\mathbb{X}_{\ell-1} := \mathbb{X}_{\ell-1} \cap \mathbb{X}_{\ell-}$;	// correction

Remark 4. In the presentation of NCSE, the influence of the state perturbation $\mathbf{w}_\ell$ has been neglected for the sake of brevity. Actually, $\mathbf{f}_\ell$ is a function from $\mathbb{A}$ to $\mathbb{B}$, with $\mathbb{A}$ the Cartesian product of the state and perturbation spaces, and $\mathbb{B}$ the state space. For $\mathbf{f}_{\ell-1}^{-1}$, the roles of $\mathbb{A}$ and $\mathbb{B}$ are interchanged. To compute $\mathbb{X}_{\ell-1}$, instead of going through Steps 5 and 6, one should therefore first compute an approximation of the image of $\mathbb{X}_\ell$ by $\mathbf{f}_{\ell-1}^{-1}$ in the $\mathbb{A}$ space, then intersect it with $\mathbb{X}_{\ell-1} \times [\mathbf{w}_{\ell-1}]$ and finally project the result onto the state space. Note that intersection and projection are not commutative. ■

5. EXAMPLE

The following example will show how the approach of Section 4 can be used to estimate the state of a linear uncertain or nonlinear discrete-time model under the hypothesis that all the uncertain quantities belong to known intervals or boxes. Consider a discrete-time model described by

$$\mathbf{x}_{\ell+1} = \mathbf{A}(w_\ell)\mathbf{x}_\ell, \tag{7}$$

with $\mathbf{x}_\ell = (x_1(\ell)\ \ x_2(\ell))^{\mathrm{T}}$ and

$$\mathbf{A}(w_\ell) = \begin{pmatrix} w_\ell \cos \pi/4 & -w_\ell \sin \pi/4 \\ w_\ell \sin \pi/4 & w_\ell \cos \pi/4 \end{pmatrix}. \tag{8}$$

The initial state vector $\mathbf{x}_0$ is only known to belong to $\mathbb{X}_0 = [4, 5] \times [-1, 1]$. When w_ℓ is treated as a state perturbation, (7) describes an uncertain linear system. If w_ℓ is included in an extended state vector $\mathbf{x}_\ell^{\mathrm{e}}$, then the model becomes nonlinear.

In what follows, all computing times are given for a Pentium 233 MHz personal computer. In each algorithm, the precision of set description is controlled by the value given to a tuning parameter ε, which, roughly speaking, corresponds to the width of the smallest box that can be used. Of course, the smaller ε is and the more complicated the computation becomes, so a compromise must be struck between accuracy and complexity.

5.1 Long-range state prediction

The parameter w_ℓ in (8) is uncertain and only known to belong to the interval $[w_\ell] = [0.8, 0.9]$. With the help of IMAGESP, one can evaluate a subpaving $\overline{\mathbb{X}}_\ell$ guaranteed to contain $\mathbb{X}_\ell$ for any $\ell > 0$. The evolution of $\overline{\mathbb{X}}_\ell$ for $\varepsilon = 0.1$ and $\ell = 0, \ldots, 10$ is represented on Figure 2. Table 3 shows the influence of ε on the computing time and on the quality of description, as quantified by the volume of $\overline{\mathbb{X}}_{10}$.

Table 3. Influence of ε on computing time and pessimism

ε	0.2	0.1	0.05	0.025
time (s)	1.8	10	76	750
volume of $\overline{\mathbb{X}}_{10}$	4.91	2.71	2.05	1.79

For $\varepsilon = 0.05$, $\overline{\mathbb{X}}_{1000}$ is computed in 180 s, and included in the box $[-0.23, 0.20] \times [-0.22, 0.19]$. Note that prediction over 1000 steps requires less than three times the time needed for prediction over 10 steps. This is due to the fact that the sets to be characterized become smaller as time goes by.

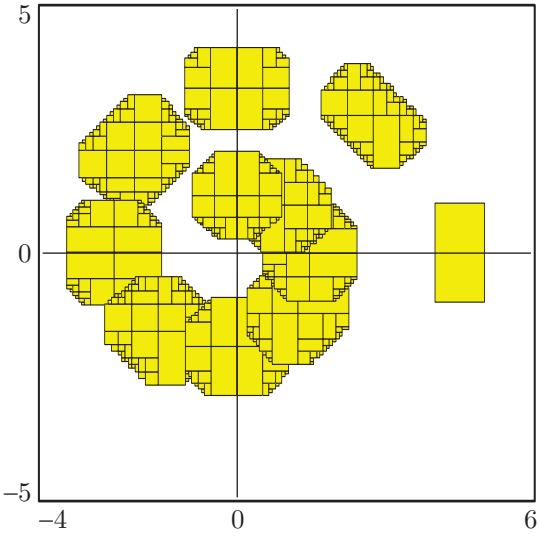

Fig. 2. Evolution of $\overline{\mathbb{X}}_\ell$ for $\ell = 0, \ldots, 10$; $w_\ell \in [0.8, 0.9]$, $\varepsilon = 0.1$

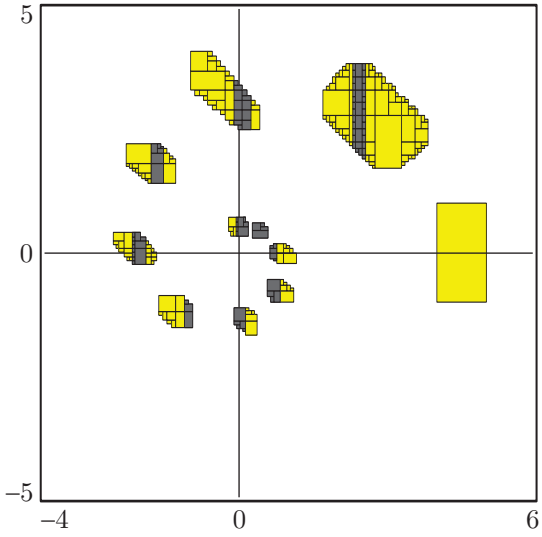

Fig. 3. Causal state estimation; the predicted sets are in light grey, the corrected sets are in dark grey

5.2 Causal state estimation

Assume now that the first component of the state vector is observed, according to

$$y_\ell = \begin{pmatrix} 1 & 0 \end{pmatrix} \mathbf{x}_\ell + v_\ell, \tag{9}$$

where v_ℓ is a realization of some additive measurement noise belonging to $[-0.1, 0.1]$. A state estimator is implemented for this system, alternating prediction and correction as explained in Section 4.1. The data y_ℓ used in this example are generated by simulating (7) and (9) with w_ℓ picked at random in $[0.8, 0.9]$, v_ℓ picked at random in $[-0.1, 0.1]$, starting from the (unknown) initial state $\mathbf{x}_0 = (4.5, 0)^{\mathrm{T}}$.

Figure 3 presents the results computed by the state estimator with $\varepsilon = 0.1$. The predicted sets $\overline{\mathbb{X}}_{\ell+}$ are in light grey, and the corrected sets $\overline{\mathbb{X}}_{\ell+1}$ in dark grey. Computing all these sets from $\ell = 0$ to $\ell = 10$ takes about 1.3 s. Estimation at $\ell = 0$ and $\ell = 1$ takes most of this time, because the sets manipulated are larger. After $\ell = 1$, each iteration of RCSE takes less than 0.03 s. Much less time is needed than in Section 5.1, because the sets to be manipulated are smaller.

5.3 Causal joint parameter and state estimation

The unknown parameter w_ℓ is now assumed to be constant, and should be estimated jointly with the state vector. For this purpose, w_ℓ is appended to the state vector to form the extended state vector

$$\mathbf{x}^{\mathrm{e}}_\ell = \left(\mathbf{x}_\ell^{\mathrm{T}}, w_\ell\right)^{\mathrm{T}}. \tag{10}$$

The discrete-time equation satisfied by the extended state is

$$\mathbf{x}^{\mathrm{e}}_{\ell+1} = \begin{pmatrix} \mathbf{A}(w_\ell)\mathbf{x}_\ell \\ w_\ell \end{pmatrix} = \mathbf{f}^{\mathrm{e}}\left(\mathbf{x}^{\mathrm{e}}_\ell\right). \tag{11}$$

The observation equation is still assumed to be (9). The (unknown) initial extended state is $\mathbf{x}^{\mathrm{e}}_0 = (4.5, 0, 0.85)^{\mathrm{T}}$. It is only assumed to belong to $\mathbb{X}^{\mathrm{e}}_0 = [4, 5] \times [-1, 1] \times [0.5, 1.5]$. Figure 4 depicts the projection of the sets $\overline{\mathbb{X}}^{\mathrm{e}}_\ell$ onto the $\mathbf{x}$-plane, as obtained with RCSE, with $\varepsilon = 0.1$ and $\ell = 0, \ldots, 10$. The computation takes 34 s in total. Again, the first three iterations take most of this time. After $\ell = 2$, each iteration takes less than 0.25 s.

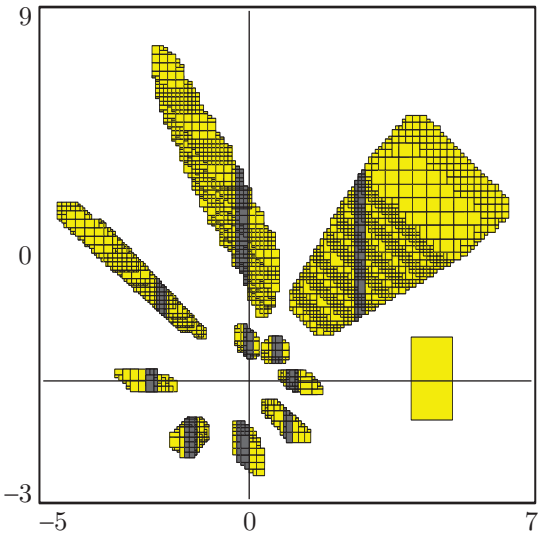

Fig. 4. Causal extended state estimation; projection onto the $\mathbf{x}$-plane of $\overline{\mathbb{X}}^{\mathrm{e}}_\ell$ (dark grey) and $\overline{\mathbb{X}}^{\mathrm{e}}_{\ell+}$ (light grey)

The last component of the extended state corresponds to the parameter w_ℓ. The projection of $\overline{\mathbb{X}}^{\mathrm{e}}_\ell$ onto the w_ℓ-axis is an interval corresponding to a guaranteed estimate of w_ℓ, represented on Figure 5 as a function of ℓ.

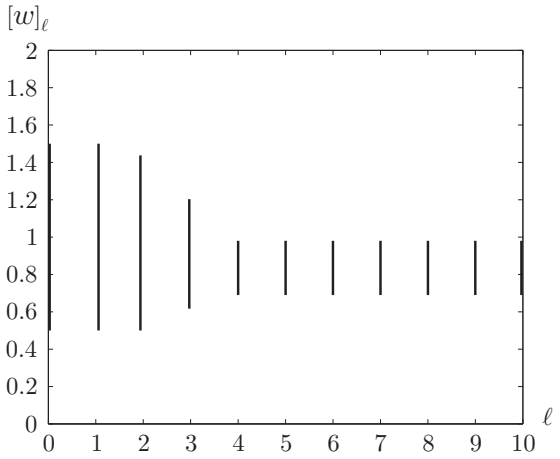

Fig. 5. Intervals enclosing the actual value of w_ℓ

5.4 Noncausal estimation

Assume now that all measurements are available at the outset. The projections onto the $\mathbf{x}$-plane of the sets $\overline{\mathbb{X}}_\ell^c$ obtained by NCSE are depicted on Figure 6. Note that the improvement is particularly significant when ℓ is small.

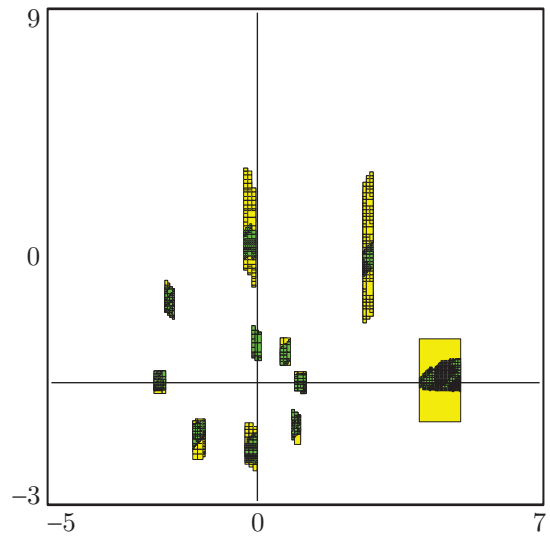

Fig. 6. Noncausal extended state estimation: projection onto the $\mathbf{x}$-plane of the sets $\overline{\mathbb{X}}_\ell^c$; the corrected sets obtained with the causal estimator are in light grey, and the corrected sets obtained with the noncausal estimator in dark grey

6. REFERENCES

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

Braems, I., F. Berthier, L. Jaulin, M. Kieffer and E. Walter (2001). Guaranteed estimation of electrochemical parameters by set inversion using interval analysis. Journal of Electroanalytical Chemistry **495**, 1–9.

Chernousko, F. L. (1994). State Estimation for Dynamic Systems. CRC Press. Boca Raton.

Davis, E. (1987). Constraint propagation with interval labels. Artificial Intelligence **32**, 281–331.

Hansen, E. R. (1992). Global Optimization using Interval Analysis. Marcel Dekker. New York.

Jaulin, L. (2000). Interval constraint propagation with application to bounded-error estimation. Automatica **36**, 1547–1552.

Jaulin, L. and E. Walter (1993a). Guaranteed nonlinear parameter estimation from bounded-error data via interval analysis. Mathematics and Computers in Simulation **35**, 1923–1937.

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

Jaulin, L., M. Kieffer, O. Didrit and E. Walter (to appear). Applied Interval Analysis. Springer-Verlag. London.

Kieffer, M., L. Jaulin and E. Walter (1998). Guaranteed recursive nonlinear state estimation using interval analysis. In: Proceedings of the 37th IEEE Conference on Decision and Control. Tampa. pp. 3966–3971.

Kieffer, M., L. Jaulin and E. Walter (2001a). Guaranteed recursive nonlinear state bounding using interval analysis. To appear in International Journal of Adaptative Control and Signal Processing.

Kieffer, M., L. Jaulin, E. Walter and D. Meizel (2000). Robust autonomous robot localization using interval analysis. Reliable Computing **6**, 337–362.

Kieffer, M., L. Jaulin, E. Walter and D. Meizel (2001b). Localisation et suivi robustes d'un robot mobile grâce à l'analyse par intervalles. Traitement du Signal **17**(3), 207–219.

Kurzhanski, A. and I. Valyi (1997). Ellipsoidal Calculus for Estimation and Control. Birkhäuser. Boston.

Milanese, M., Norton, J., Piet-Lahanier, H. and Walter, E., Eds.) (1996). Bounding Approaches to System Identification. Plenum Press. New York.

Moore, R. E. (1979). Methods and Applications of Interval Analysis. SIAM. Philadelphia.

Ratschek, H. and J. Rokne (1984). Computer Methods for the Range of Functions. Halsted Press. New York.

Sam-Haroud, D.J. and B. Faltings (1996). Consistency techniques for continuous constraints. Constraints **1**, 85–118.

Walter, E. and L. Pronzato (1997). Identification of Parametric Models from Experimental Data. Springer-Verlag. London.