INTERVAL ANALYSIS IN NONLINEAR STATE ESTIMATION

ZBRANEK, P[avel] & VESELY, L[ibor]

Abstract: This paper proposes some tools of set observer for estimation the state vector of a nonlinear dynamic system. These tools make it possible to compute outer approximations of the set of all state vectors that are consistent with the model structure, measurements and noise bounds. 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.

Key words: interval analysis, state estimation

1. INTRODUCTION

This paper is concerned with nonlinear state estimation. The main idea of making an estimator is the use of interval analysis. There are functions which enable us to operate and calculate with intervals (Moore 1979), (Di-Loreto et al., 2007) in the same way we approach mathematical operations and calculations with floating-point numbers. Unlike other estimators used today, such as Kalman’s filter, the results of state estimation won’t be absolutely punctual, but we will know the interval of their incidence. The more accurately we want to know the interval, the more calculation effort we need to make.

The retrieval of just the interval itself may seem like a big disadvantage, but compared to the abovementioned Kalman’s filter, the main advantage is that if the calculations are correct, the state values are certainly found in the interval. The Kalman’s filter establishes the states only with a certain probability, which is not exactly 100%.

2. INTERVAL ANALYSIS INTRODUCTION

The key idea of interval analysis is to reason about intervals instead of real numbers and boxes instead of real vectors. The first motivation was to obtain guaranteed results from floating point algorithms and it was then extended to validated numerics (Moore 1959). Let us recall that in computers real numbers can only be represented by a floating point approximation, hence introducing a quantification error. A guaranteed result means first that the result set encloses the exact solution. The width of the set, i.e. the result precision, may be chosen depending on various criteria among which response time or computation costs. Secondly, it also means that the algorithm is able to conclude on the existence or not of a solution in limited time or number of iterations.

2.1 Intervals

Scalar interval \([x] = [\underline{x}, \overline{x}]\) is a closed and connected subset of \(\mathbb{R}\); it may be characterized by its lower and upper bounds \(\underline{x}\) and \(\overline{x}\) or equivalently by its centre \(c([x]) = (\underline{x} + \overline{x})/2\) and width \(w([x]) = (\overline{x} - \underline{x})\).

Interval arithmetical provides an extension to intervals of the usual arithmetical operation on real \(\{+,-,*,/\}\) through the generic formula \(\circ \in \{+,-,*,/\}\). The operations on interval are then defined by:

\[
\forall \circ \in \{+,-,*,/\}, \quad [x] \circ [y] = \{x \circ y \mid x \in [x], y \in [y]\}
\]

2.2 Interval vectors

An interval real vector \([x]\) is a subset of \(\mathbb{R}^n\) that can be defined as the Cartesian product of \(n\) closed intervals. When there is no ambiguity, \([x]\) will simply be called an interval vector, or a box. It will be written as \([x] = [x_1] \times [x_2] \times \ldots \times [x_n]\), or in short, \([x_1, \ldots, x_n]\) for \(i=1, \ldots, n\). (Fig. 1 A box \([x]\) of \(\mathbb{R}^2\).)

2.3 Inclusion function

Definition: 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

\[
[f] : \mathbb{R}^n \rightarrow \mathbb{R}^p, \quad [x] \rightarrow \{f([x]) \mid x \in [x]\}.
\]

\([f](x)\) is thus the smallest box of \(\mathbb{R}^p\) 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 \(F\) satisfying the following definition.

Definition: \(F : \mathbb{R}^n \rightarrow \mathbb{R}^p\) is an inclusion function of \(f : \mathbb{R}^n \rightarrow \mathbb{R}^p\) if

\[
\forall [x] \in \mathbb{R}^n, \quad f([x]) \subseteq F([x]),
\]

and

\[
\lim_{[x] \rightarrow 0} w(F([x])) = 0.
\]

Figure 2 illustrates these two definitions. For any function \(f\) obtained by composition of elementary operations such as \(+, -, *, /, \sin, \cos, \exp, \ldots\), 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 \(f\).
The set $R^k$. In

ternates

is way the

and

ed Cybernetics” and by Freescale

function $f : R^n \rightarrow R^m$, the set $S \subset S_0$ is defined in a way

This is a part of the set-inversion class, which can be
defined as two sets $X \subset R^N$, $Y \subset R^M$ and a function

4. CONCLUSION

A nonlinear state estimation has been presented. The main idea of this

guaranteed estimator is using interval analysis. As in

classical Kalman filtering, this state estimator alternates

prediction and correction. The algorithm was tested on several

simple systems. It was simulated. Simulations are prone to

change parameters and are very time consuming. The algorithm

needs a large amount of computing effort. That is why the

algorithm must be optimized for more complicated systems.

5. ACKNOWLEDGEMENTS

The research has been supported by Czech Science Foundation

under the project GA P103/10/0647 “Intelligent Electrical

Drives Predictive and Robust Control Algorithms”, the

Ministry of Education of the Czech Republic under the project

1M0567 ”Center for Applied Cybernetics” and by Freescale

Semiconductor Czech Republic (former division of Motorola)

6. REFERENCES

time-delay system analysis. In Application of Time-Delay Systems, Lecture Notes in Control and Information Science,

Automatica, 29(4):1053–1064


recursive nonlinear state estimation using interval analysis.

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

3.1 Prediction

Calculating

$X_{k+1} = \{ f_k(x,u_k,v_k) \mid x \in X_k, v_k \in [v_k] \}$ can result

into a direct image evaluation issue, which is at the very core of

interval arithmetic.

Having two compact sets $X \subset R^N$ and $S_0 \subset R^m$ plus a

function $f : R^n \rightarrow R^m$, the set $S \subset S_0$ is defined in a way

that $S = \{ f(x) \in S_0 \mid x \in X \}$.

3.2 Correction

For this step it is necessary to characterize

$X_{k+1} = \{ x \in X_k \mid h_{k+1}(x) \in Y_{k+1} \}$

This problem is a part of the set-inversion class, which can be
defined as two sets $X \subset R^N$, $Y \subset R^M$ and a function

$f : R^n \rightarrow R^m$. The set is then characterized as

$f^{-1}(Y) = \{ x \in X \mid f(x) \in Y \}$.

This particular problem can be sorted out in an approximated,

nonetheless guaranteed fashion by using the Set Inversion Via

Interval Analysis algorithm, or in short the SIVIA, which was

originally designed by Jaulin and Walter (Jaulin et al. 2001).

A fourth approach is to use more sophisticated inclusion functions.

3. STATE ESTIMATION

Let’s think about the non-linear and potential time-varying

system defined as follows:

\[
\begin{align*}
  x_{k+1} &= f_k(x_k, u_k, v_k), \\
  y_k &= h_k(x_k) + w_k,
\end{align*}
\]

where terms $u_k \in R^m$, $x_k \in R^n$ and $y_k \in R^d$ are
defined as input, state and output vector, in the same order they
are mentioned above. The beginning state $x_0$ is considered to
be a part of some preceding compact set $X_0 \subset R^n$. Concerning
terms $\{v_k\}$ and $\{w_k\}$, they are unknown state and
measurement noise sequences, presumed to be part of the known
intervals $\{[v_k]\}$ and $\{[w_k]\}$.

As for the terms $f_k$ and $h_k$, they’re known functions, or finite
algorithms if you like.

The process of state estimation can be summarized in following
idealized algorithm (Jaulin & Walter, 1993), (Kiewref, et al.,
1998):

For $l=0$ to $L$, do

1. Prediction: $X_{l+1} = f_l(X_l, u_l, [v])$.

2. Correction: $X_{l+1} = h_l^{-1}(Y_{l+1} \cap X_{l+1})$.