Proceedings of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

State Estimation Based on Self-Triggered Measurements

N. Meslem and C. Prieur

Grenoble Images Speech Signals and Automatics Laboratory (GIPSA-Lab), Grenoble Institute of Technology (INPG), 38402 Saint-Martin-d'Heres Cedex, France (e-mail: {nacim.meslem, christophe.prieur}@gipsa-lab.grenoble-inp.fr).

Abstract: In this work, the problem of state estimation for nonlinear continuous-time systems from discrete data is tackled in a bounded error context. One assumes that all poorly-known system variables belong to a bounded set with known bounds. Then, a self-triggered algorithm is proposed to improve the performance of the classical set-membership state estimator based on the prediction-correction procedures. In order to cope with pessimism propagation linked to the bounding methods, this algorithm triggers the correction step whenever the size of a part of the estimated state enclosure becomes greater than a time-converging threshold a priori defined by the user. The effectiveness of the proposed self-triggered algorithm is illustrated through numerical simulations.

Keywords: Nonlinear systems; state estimation; bounded-error framework; interval analysis; monotone systems; bounding methods.

1. INTRODUCTION

State estimation is an important field of control system theory (Luenberger [1971]),(Isidori [1995]). As a matter of fact, several advanced control and diagnosis approaches are developed under the assumption that the state vector of the continuous-time system is available online. To satisfy this requirement, software sensors called observers are developed to estimate in real-time the state vector (Luenberger [1971]). For example, for linear continuous-time systems one can use the standard Luenberger observer (Luenberger [1971]) or the Kalman filter (Kalman [1960]). Differently, for nonlinear systems, there are different approaches to design nonlinear observers. For instance, one can cite the extended Luenberger observer (Sorenson [1985]), the extended Kalman filter (Misawa and Hedrick [1989]), the high gain observer (Gauthier et al. [1992]) or the sliding mode observer (Drakunov [1983], Slotine et al. [1986]. . .). All these observer design approaches assume that the model of the real system is perfectly known and the measurements are available in continuous-time. In practice, these assumptions are problematic, especially when dealing with biological or biotechnological systems, because the system parameters are poorly-known and the measurements are generally done in discrete time.

To circumvent this problem, prediction-correction set-membership state estimators are developed during these last years (Jaulin [2002], Raïssi et al. [2004], Goffaux et al. [2009], Meslem et al. [2010a]). This kind of estimators are designed in the unknown but bounded error context (Milanese et al. [1996]). They estimate from discrete data an accurate enclosure of the state flow generated by an uncertain system, where all the uncertain variables are represented by boxes (interval vectors) (Moore [1966], Jaulin et al. [2001]). The main contribution of this work consists in endowing the set-membership state estimator by a self-triggered algorithm in order to apply efficiently the correction procedure. In fact, with this algorithm one can master the propagation of pessimism generated by the bounding methods (Kieffer and Walter [2006], Ramdani et al. [2009], Ramdani et al. [2010]). This algorithm is inspired from the event-triggered control strategy applied to continuous-time systems (Meslem and Prieur [2013]).

Note that, to the best of our knowledge, the convergence issue of the state estimation error in the bounded error context is still not well investigated, in particular when applying the prediction-correction algorithm (Jaulin [2002], Raïssi et al. [2004], Goffaux et al. [2009], Meslem et al. [2010a]). In this work, due to this self-triggered algorithm, the proof of the convergence of the state estimation error is provided under some assumptions. This convergence analysis shows more the importance of our findings.

This paper is organized as follows. In Section 2, basic notions about interval computations are introduced. Then, the core idea of the classical prediction-correction state estimator is recalled in Section 3. The main results of this work are stated and proved in Section 4. An illustrative example is given in Section 5 with several simulation tests. Also, a comparative study with the results obtained by an interval observer is presented and commented in this section. Finally, some concluding remarks and future works are discussed.

2. PRELIMINARY NOTIONS ABOUT INTERVAL COMPUTATION

Interval analysis was initially developed to account for the quantification errors introduced by the floating point representation of real numbers with computers and was extended to reliable computations (Moore [1966], Neumaier [1990]). Denote by $[x] = [\underline{x}, \overline{x}]$ a real interval which is a connected and closed subset of $\mathbb{R}$ where the real numbers $\underline{x}$ and $\overline{x}$ are respectively the lower and the upper bound of $[x]$. So, the set of all real intervals of $\mathbb{R}$ is denoted by $\mathbb{IR}$. Over $\mathbb{IR}$ an interval arithmetic was built by an extension of the real arithmetic operations. That means, for each operator $\circ \in \{+, -, \times, \div\}$ and for each couple of intervals $[x]$ and $[y]$ one defines

$$[x] \circ [y] = \{a \circ b \mid a \in [x],\ b \in [y]\} \tag{1}$$

The width of an interval $[x]$ is defined by $w([x]) = \overline{x} - \underline{x}$. As well, an interval vector or box denoted by $[\mathbf{x}]$ is a subset of $\mathbb{R}^n$ defined as the Cartesian product of n closed intervals. The set of all interval vectors of order n will be denoted by $\mathbb{IR}^n$. The width of an interval vector of dimension n is defined by

$$w([\mathbf{x}]) = \max_{1 \le i \le n} w([x_i])$$

Likewise, we define the vector width of an interval vector by

$$\mathbf{w}_v\big([\mathbf{x}]\big) = \big(w([x_1]), w([x_2]), \ldots, w([x_n])\big)^T$$

That is, the components of the real vector $\mathbf{w}_v$ are the widths of each component of the interval vector $[\mathbf{x}]$.

Now, one can describe uncertain parameters by an upper and lower bound, then rigorous bounds on the range of a real function of these parameters are computed using interval arithmetic. Consider the real function $f : \mathbb{R}^n \to \mathbb{R}$. The range of this function over an interval vector $[\mathbf{x}]$ is given by:

$$f([\mathbf{x}]) = \{f(\mathbf{a}) \mid \mathbf{a} \in [\mathbf{x}]\} \tag{2}$$

Then, one calls an inclusion function denoted by $[f]$ for the real function f an interval application that satisfies the following inclusion

$$\forall [\mathbf{x}] \in \mathbb{IR}^n,\ f([\mathbf{x}]) \subset [f]([\mathbf{x}]) \tag{3}$$

In practice, the simplest manner to obtain an inclusion function $[f]$ for real function f consists in replacing each occurrence of a real variable by the corresponding interval and each standard function by its interval counterpart. The resulting function is called the natural inclusion function and the tightness of the enclosure provided by $[f]$ depends on the formal expression of f. In fact, it is well known if the same variable x_i has many occurrences in the mathematical expression of f, the *dependence effect* (Moore [1966], Jaulin et al. [2001]) will induce pessimism while computing an enclosure of the range of the real function. Hence, formal pre-processing of the function expression is advisable in order to minimize the number of variable occurrences.

In the sequel, we will show how the joint use of interval computation and the bounding methods for computing rigorous bounds on the reachable set of uncertain nonlinear systems, allows to solve in guaranteed way the state estimation problem for uncertain nonlinear systems from discrete data.

3. SET-MEMBERSHIP STATE ESTIMATION

3.1 Prediction-Correction state estimator

In this section, we recall briefly the core idea of the classical prediction-correction state estimator of nonlinear continuous-time systems, which are described by

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{p}, \mathbf{u}) \\ \mathbf{y} = \mathbf{g}(\mathbf{x}, \mathbf{p}, \mathbf{u}) \end{cases} \tag{4}$$

where $\mathbf{x} \in \mathbb{R}^n$ is the state vector to be estimated from discrete data. The vectors $\mathbf{u} \in \mathbb{R}^m$ and $\mathbf{y} \in \mathbb{R}^p$ stand for respectively the input and the output of the system. The vector field $\mathbf{f}$ and the output model $\mathbf{g}$ can be linear or nonlinear functions of the state and input with appropriate dimensions. The initial state $\mathbf{x}_0$ and the parameter vector $\mathbf{p} \in \mathbb{R}^{n_p}$ are assumed unknown but bounded with known bounds. That means,

$$\mathbf{x_0} \in [\underline{\mathbf{x}}_0, \overline{\mathbf{x}}_0],\quad \mathbf{p} \in [\underline{\mathbf{p}}, \overline{\mathbf{p}}]$$

where $\overline{\mathbf{x}}_0$, $\underline{\mathbf{x}}_0$, $\overline{\mathbf{p}}$ and $\underline{\mathbf{p}}$ are respectively the known upper and lower bound of the initial state and the parameter vector $\mathbf{p}$. Experimental data $\mathbf{y}_j$ are collected at discrete times $t_j,\ j \in \{1, \ldots, N\}$. And the feasible domain for the output values at each time t_j is given by

$$[\mathbf{y}_j] = \mathbf{y}_j + [\mathbf{e}_j] \tag{5}$$

where the box $[\mathbf{e}_j]$ denotes the feasible domain for output error at time t_j, which includes both deterministic and random error.

*The prediction stage (**Pred**):* In this context the prediction procedure has to compute an outer enclosure of all possible state trajectories generated by the uncertain system (4) between two measurement time instants t_j and t_{j+1}. To accomplish this task, one can use either the validated methods for initial value problems for ordinary differential equations (Rihm [1994], Nedialkov et al. [2001]) based on interval analysis or the hybrid bounding methods (Ramdani et al. [2009], Ramdani et al. [2010]) based on the comparison theorems of differential inequalities (Müller [1926], Marcelli and Rubbioni [1997], Smith [1995]). In this work, the bounding methods are used to carry out the prediction stage. So, an outer enclosure of the state trajectories generated by (4) over a time interval $[t_j, t_{j+1}]$ is obtained by integrating the following bounding system

$$\begin{cases} \dot{\overline{\mathbf{x}}} = \overline{\mathbf{f}}(\overline{\mathbf{x}}, \underline{\mathbf{x}}, \overline{\mathbf{p}}, \underline{\mathbf{p}}, \mathbf{u}),\ \overline{\mathbf{x}}(t_j) = \overline{\mathbf{x}}_j, \\ \dot{\underline{\mathbf{x}}} = \underline{\mathbf{f}}(\overline{\mathbf{x}}, \underline{\mathbf{x}}, \overline{\mathbf{p}}, \underline{\mathbf{p}}, \mathbf{u}),\ \underline{\mathbf{x}}(t_j) = \underline{\mathbf{x}}_j, \end{cases} \tag{6}$$

where the vector functions $\overline{\mathbf{f}}$, $\underline{\mathbf{f}}$ are built in order to frame the field vector $\mathbf{f}$ for all $\mathbf{x} \in [\underline{\mathbf{x}}, \overline{\mathbf{x}}]$ and for all $\mathbf{p} \in [\underline{\mathbf{p}}, \overline{\mathbf{p}}]$. To get more explanations about the construction of the bounding system (6) the reader can refer to Ramdani et al. [2009], Ramdani et al. [2010] and references therein. Thus, one can claim that all possible state trajectories generated by the uncertain system (4) are framed by the solution of the deterministic system (6). That means,

$$\mathbf{x}(t_j) \in [\underline{\mathbf{x}}(t_j), \overline{\mathbf{x}}(t_j)], \mathbf{p} \in [\underline{\mathbf{p}}, \overline{\mathbf{p}}] \\ \Rightarrow \forall t \in [t_j, t_{j+1}], \mathbf{x}(t) \in [\underline{\mathbf{x}}(t), \overline{\mathbf{x}}(t)] \tag{7}$$

To sum up, the prediction stage computes an outer enclosure, here denoted by $[\mathbf{x}(t)]^p$, of the all state trajectories generated by the system (4) on the period $[t_j, t_{j+1}]$.

*The correction stage (**Corr**):* At each measurement time instant t_j, an other outer enclosure of the state vector denoted by $[\mathbf{x}(t_j)]^{inv}$ is computed now by solving the following set inversion problem

$$[\mathbf{x}(t_j)]^{inv} = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{g}(\mathbf{x}, \mathbf{p}, \mathbf{u}) \in [\mathbf{y}(t_j)]\} \tag{8}$$

Then, the inconsistent state vectors belonging to the two outer enclosures are discarded as follows

$$[\mathbf{x}(t_j)]^c = [\mathbf{x}(t_j)]^{inv} \cap [\mathbf{x}(t_j)]^p \tag{9}$$