Acta Cybernetica **24** (2020) 493–508.

Interval Predictors for a Class of Uncertain Discrete-Time Systems

Nacim Meslem[a] and John J. Martinez[a]

Abstract

This work presents set-valued algorithms to compute tight interval predictions of the state trajectories for a class of uncertain dynamical systems, where the dynamics is described as a sum of a linear and a nonlinear term. Based on interval analysis and the analytic expression of the state response of discrete-time linear systems, non-conservative numerical schemes are proposed. Moreover, under some stability conditions, the convergence of the width of the predicted state enclosures is proved. The performance of the proposed set-valued algorithms is illustrated through two numerical examples, and the results are compared to that obtained with another method selected from the literature.

Keywords: reachability analysis, interval analysis, uncertain systems, set-membership techniques

1 Introduction

One way to check a priori the safety behavior of an uncertain dynamical system is to predict its reachable set [2, 5, 7, 11, 13, 14, 21, 22, 27, 28, 29], namely the set which contains all possible state trajectories of the system generated from a bounded set of initial conditions. More precisely, based on this reachable set it is possible to prove numerically the satisfaction or the violation of desired properties of a dynamical system. Figure 1 illustrates this safety verification method. In this figure, the region of the state space, where the behavior of the system is safe, denoted by $\mathcal{S}$, is delimited by the bold continuous lines. This safe set is described by the set of constraints to be satisfied. Thereby, to guarantee the safe behavior of the system despite the presence of uncertainties in its model, it is sufficient to show that its reachable set, denoted by $\mathcal{R}$ and delimited by the dashed curves, stays inside the safe region $\mathcal{S}$. In addition, by means of reachability analysis, one can verify if a given nominal controller can steer an uncertain system from an initial state set $\mathcal{X}_0$, depicted by the big circle in Figure 1, to a final desired target set

[a]Univ. Grenoble Alpes, CNRS, Grenoble INP*, GIPSA-lab, 38000 Grenoble, France, E-mail: {nacim.meslem, john-jairo.martinez-molina}@grenoble-inp.fr

DOI: 10.14232/actacyb.24.3.2020.12

$\mathcal{T}$, shown by the small circle, without violating the safety and resource limitation constraints [15].

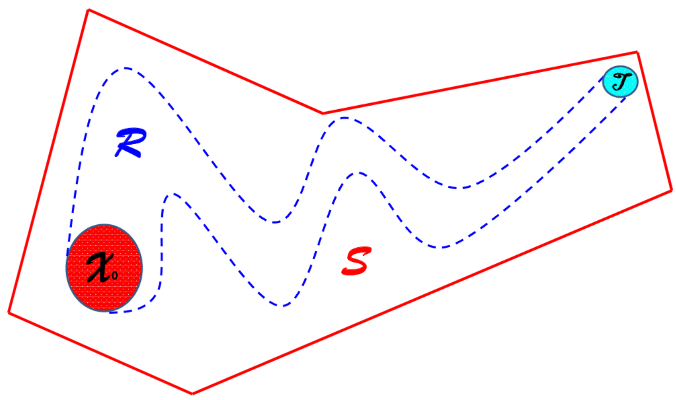

Figure 1: An illustration of viability problems solved by means of reachability analysis. $\mathcal{X}_0$ stands for the initial set, $\mathcal{S}$ is the safety set, $\mathcal{R}$ is the reachable set, and $\mathcal{T}$ stands for the desired target set.

Although the idea behind this safety verification approach is simple, computing the reachable set of an uncertain system is a hard task. Several set-membership methods, based on different geometrical shapes [2, 3, 8, 11, 14, 23, 28], have been proposed in the literature to compute tight outer approximations of the reachable set. Unfortunately, in conjunction with dynamical systems, these set-membership methods have been too conservative. This conservatism is unavoidable for two main reasons. At every iteration, the exact reached set of an uncertain system is overestimated by framing it into a super-set both feasible to construct and to represent on a computer. In the literature, this phenomenon is called the *wrapping effect*, which is well illustrated in the case of interval analysis [1, 4, 19]. The second source of conservatism is the *dependency problem*, which appears when an uncertain variable occurs several times in interval arithmetic.

In this work, effective algorithms for computing tight outer approximations of the reachable set of a class of discrete-time dynamical systems, where the dynamics is defined as a sum of a linear and a nonlinear term, are introduced. To cope with the *wrapping effect*, these algorithms compute at every time instant an enclosure of the reached set directly from the initial set. This means that, due to the proposed numerical schemes, the over-estimations linked to the iterative set computations are avoided. In addition, under some stability assumptions on the system's dynamics, these algorithms provide state enclosures with a converging width. It is worth pointing out that unlike the method introduced in [10], no pseudo-linear transformation is needed to apply the proposed approach. To avoid the *wrapping effect*, an interval version of the analytic expression of the state response of discrete-time linear systems is used, and interval assessments are performed as last computation step. This way to tackle the reachability problem differentiates the proposed method with respect to the other set computation methods that are based on the dynamic equations of the systems.

The remainder of this paper is organized as follows. In Section 2, the considered problem is formulated and a brief introduction to interval analysis is presented. The main contributions of this work are stated in Section 3. The interval predictor algorithms are introduced, and the convergence of the width of the computed state enclosures is analyzed. In Section 4, simulation results obtained on numerical examples are presented. On the one hand, the performance of the proposed algorithms is illustrated, and on the other hand, this performance is compared to that obtained with the method proposed in [16].

2 Problem formulation

Consider the class of uncertain discrete-time dynamical systems described by:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{f}(\mathbf{x}_k, \mathbf{w}_k) + \mathbf{B}\mathbf{u}_k, \tag{1}$$

where $\mathbf{x}_k \in \mathcal{X} \subset \mathbb{R}^n$ is a state vector and $\mathbf{u}_k \in \mathcal{U} \subset \mathbb{R}^m$ is an input vector. The nonlinear term $\mathbf{f}(.,.)$ stands for the poorly-known part of the system (1), which is assumed to be bounded:

$$\forall \mathbf{x}_k \in \mathcal{X} \text{ and } \forall \mathbf{w}_k \in \mathcal{W} \subset \mathbb{R}^p, \quad \mathbf{f}(\mathbf{x}_k, \mathbf{w}_k) \in \mathcal{F}, \tag{2}$$

where $\mathcal{F}$ is a bounded set. The vector $\mathbf{w}_k$ can contain uncertain parameters of the system or unknown exogenous signals (noises, perturbations), which act on the system's dynamics. Likewise, the initial condition of this system $\mathbf{x}_0$ is assumed to be unknown but belonging to a bounded set $\mathcal{X}_0$.

By definition, the reachable set of the uncertain discrete-time system (1), denoted by $\mathcal{R}\big(\{t_0, \ldots, t_k\}, t_0, \mathcal{X}_0\big)$, is the set of all its possible state trajectories generated from the initial bounded set $\mathcal{X}_0 \subset \mathcal{D}$ and solutions to the set of difference equations (1). More formally, $\mathcal{R}\big(\{t_0, \ldots, t_k\}, t_0, \mathcal{X}_0\big)$ stands for a sequence of the reachable sets at the time instants t_k denoted by $\mathcal{R}\big(t_k, t_0, \mathcal{X}_0\big)$. Characterizing the exact reachable set for an uncertain system is a hard task in practice. For this reason, almost all approaches proposed in the literature [3, 9, 10, 11, 12, 14] attempt to compute a tight outer approximation of this reachable set. By construction, an outer approximation of the reachable set of (1), denoted by $\mathcal{Y}\big(\{t_0, \ldots, t_k\}, t_0, \mathcal{Y}_0\big)$, is a set that satisfies the following inclusion:

$$\forall k \geq 0, \quad \mathcal{R}\big(\{t_0, \ldots, t_k\}, t_0, \mathcal{X}_0\big) \subseteq \mathcal{Y}\big(\{t_0, \ldots, t_k\}, t_0, \mathcal{Y}_0\big). \tag{3}$$

In the case of linear dynamical systems, many geometrical forms (e.g. ellipsoids, zonotopes, parallelotopes, boxes) have been used to compute an enclosure of the reachable set. The choice of these forms relies on their feasibility to build and to represent on computers. The common principle of the reachability computation methods consists in propagating in time, according to the uncertain dynamics of the system, these super-sets. These set-membership methods, are step-by-step algorithms, where at each step an optimization problem is solved to obtain a smaller super-set that contains the exact reachable set of the system. More precisely, the provided outer approximation is a union of the guaranteed enclosures $\mathcal{Y}_k$ of the solution to (1) at the time instants $t_k, k = 0, 1, 2 \ldots$.

$$\mathcal{Y}\big(\{t_0, \ldots, t_k\}, t_0, \mathcal{Y}_0\big) = \bigcup_0^k \mathcal{Y}_k. \tag{4}$$

However, this problem can be also solved by means of the theory of monotone dynamical systems [29]. Based on constant or time-varying similarity transformations [6, 16], the dynamics of the system can be represented in a monotone form in the new basis of the state of coordinates. Subsequently, upper and lower systems are designed to generate upper and lower state trajectories that frame all the possible solutions to (1) in the new state basis.

In this work, we propose a new approach, where the state enclosures $\mathcal{Y}_k$ at every time instant t_k are computed directly from $\mathcal{Y}_0$. Notice that, this way to proceed offers a great advantage to avoid *wrapping effect*. In addition, it is worth pointing out that the introduced algorithms are based on interval analysis but it is possible to implement them with any other representations like ellipsoids and zonotopes.

2.1 Interval analysis

At the beginning, interval analysis [1, 19] was introduced to handle uncertainties in numerical computation carried out by computers. Thereafter, it has gained a success in many areas of engineering science [8, 10, 17, 20, 24, 25, 26]. By definition, an interval $[x] = [\underline{x}, \overline{x}]$ is a closed subset of $\mathbb{R}$, where the real numbers $\underline{x}$ and $\overline{x}$ stand for its lower and upper endpoint, respectively. By construction, interval arithmetic is an extension of real arithmetic. For each basic real arithmetic operator $\circ \in \{+, -, \times, \div\}$, its interval counterpart is defined as follows:

$$[x] \circ [y] \triangleq \{a \circ b \mid a \in [x],\ b \in [y]\}, \tag{5}$$

where $[x]$ and $[y]$ are interval operands. Note that, to avoid the undefined division, when the interval $[y]$ contains 0, specific formulas were introduced in [1, 18].

Likewise, 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. All arithmetic operations on real vectors and matrices are extended to the interval case in the same spirit of the definition (5). For instance, the product between an interval matrix $[\mathbf{M}] \subset \mathbb{R}^{n\times m}$ and an interval vector $[\mathbf{x}] \subset \mathbb{R}^m$ returns an interval vector $[\mathbf{y}] \subset \mathbb{R}^m$ defined as follows:

$$[\mathbf{y}] \triangleq \{\mathbf{M}\mathbf{x} \mid \mathbf{x} \in [\mathbf{x}],\ \mathbf{M} \in [\mathbf{M}]\}. \tag{6}$$

The width of an interval is the distance between its endpoints $w([x]) = \overline{x} - \underline{x}$ and the width of an interval vector of dimension n is defined by:

$$w([\mathbf{x}]) \triangleq \max_{1\leq i\leq n} w([x_i]). \tag{7}$$

The absolute value of an interval $[x]$ is the maximum value between the absolute values of its endpoints $|[x]| = \max\{|\underline{x}|, |\overline{x}|\}$. For an interval vector $[\mathbf{x}] \in \mathbb{R}^n$ the absolute value is defined as follows:

$$|[\mathbf{x}]| \triangleq \big(|[x_1]|, |[x_2]|, \ldots, |[x_n]|\big)^T. \tag{8}$$

Moreover, the infinity norm of an interval vector, here denoted by $\|.\|$, is introduced as follows:

$$\big\|[\mathbf{x}]\big\| \triangleq \big\||[\mathbf{x}]|\big\|_\infty = \max_{1\le i\le n} |[x_i]|. \tag{9}$$

Note that, an interval matrix $[\mathbf{M}] \subset \mathbb{R}^{n\times m}$ has n rows and m columns with interval entries $[m_{i,j}] \in \mathbb{R},\ i = 1,\ldots n; j = 1,\ldots m$. Hence, the width and the norm of an interval matrix can be introduced respectively as follows:

$$w([\mathbf{M}]) \triangleq \max_{i,j}\{w([m_{i,j}])\}, \tag{10}$$

$$\big\|[\mathbf{M}]\big\| \triangleq \big\||[\mathbf{M}]|\big\|_\infty \triangleq \max_i\{\sum_j |[m_{i,j}]|\}. \tag{11}$$

In addition, the width of an interval vector $[\mathbf{y}] = [\mathbf{M}][\mathbf{x}]$ can be bounded as:

$$w([\mathbf{y}]) \le \big\|[\mathbf{M}]\big\| w([\mathbf{x}]). \tag{12}$$

3 Main results

The results of this work are widely motivated by the divergence issue observed when interval computation is applied to the iterative rotation operator [1, 4]. To illustrate this problem, consider the following autonomous system,

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k, \tag{13}$$

where the matrix $\mathbf{A}$ is a rotation operator defined as follows:

$$\mathbf{A} = \begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix}. \tag{14}$$

Thus, according to the discrete-time dynamics (13), the initial box of the state variables e.g., $[\mathbf{x_0}] = ([-0.5, 0.5],\ [3.5,\ 4])^T$ rotates at every iteration with an angle, say $\theta = \frac{\pi}{4}$, and preserves its volume constant over all the simulation period. Unfortunately, the naive application of interval analysis to characterize the reachable set of this system leads to too conservative results as shown in Figure 2. The exact reachable set of the system (13) is shown by the thin blue parallelotopes, and its outer approximation computed by the direct interval method is represented by the bold red boxes. In fact, the wrapping procedure introduces an overestimation at every iteration.

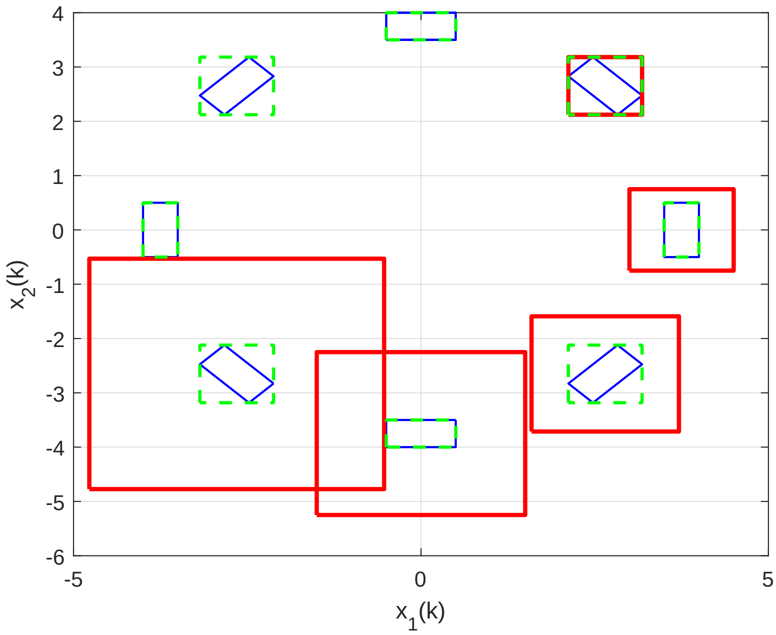

Figure 2: The exact reachable set of the system (13) and its outer approximations.

One way to circumvent this issue is to apply interval analysis to the analytic expression of the state response of this system. That is,

$$\mathbf{x}_k = \mathbf{A}^k \mathbf{x}_0. \tag{15}$$

As shown in Figure 2, the application of this new approach allows to obtain the tightest outer approximation of the exact reachable set of (13). The tightest outer approximation, shown by the dashed green boxes, is constituted by the smallest boxes which contain the exact reachable set. Moreover, as illustrated in this figure, the width of the smallest boxes is bounded.

To generalize the use of this new approach to the class of dynamical systems described by (1), we propose the following interval predictor.

Proposition 1. *Let* $\mathbf{b}_0 = \mathbf{B}\mathbf{u}_0$ *and* $[\mathbf{f}_0] = [\underline{\mathbf{f}}, \overline{\mathbf{f}}] \supseteq \mathcal{F}$. *Then for all* $k \geq 1$, *the interval-based predictor described by:*

$$\begin{array}{rcl} [\mathbf{x}_k] & = & \mathbf{A}^k[\mathbf{x}_0] + [\mathbf{f}_{k-1}] + \mathbf{b}_{k-1} \\ [\mathbf{f}_k] & = & \mathbf{A}^k[\mathbf{f}_0] + [\mathbf{f}_{k-1}] \\ \mathbf{b}_k & = & \mathbf{A}\mathbf{b}_{k-1} + \mathbf{B}\mathbf{u}_k, \end{array} \tag{16}$$

provides a tight outer approximation of the reachable set of the uncertain discrete-time system (1)*:*

$$\bigcup_0^k [\mathbf{x}_k] \supseteq \mathcal{R}\big(\{t_0, \ldots, t_k\}, t_0, \mathcal{X}_0\big), \tag{17}$$

where $[\mathbf{x}_0] \supseteq \mathcal{X}_0$.
In addition, if the matrix $\mathbf{A}$ *is Schur stable, that is its spectral radius* $\rho(A) < 1$*, the width of this outer approximation is bounded.*

Proof. The proof of this proposition is introduced in two stages. First, we prove that the proposed interval predictor encloses in a guaranteed way all possible solutions to (1). Then, under the stability assumption of the matrix $\mathbf{A}$, we show the boundedness of the width of the predicted state enclosures.

1) Framing proof: Here, we will proceed by induction. When $k = 1$,

$$\mathbf{x}_1 = \mathbf{A}\mathbf{x}_0 + \mathbf{B}\mathbf{u}_0 + \mathbf{f}(\mathbf{x}_0, \mathbf{w}_0), \tag{18}$$

where $\mathbf{x}_0 \in [\mathbf{x}_0]$ and $\mathbf{f}(\mathbf{x}_0, \mathbf{w}_0) \in [\underline{\mathbf{f}},\ \overline{\mathbf{f}}]$. Then,

$$\begin{array}{rcl} \mathbf{x}_1 & \in & \mathbf{A}[\mathbf{x}_0] + \mathbf{B}\mathbf{u}_0 + [\underline{\mathbf{f}},\ \overline{\mathbf{f}}] \\ & \in & \mathbf{A}[\mathbf{x}_0] + \mathbf{b}_0 + [\mathbf{f}_0] \\ & \in & [\mathbf{x}_1]. \end{array} \tag{19}$$

Now, we assume that (16) holds for some natural number k, and we have to prove that this statement holds for $k+1$. For a given instant k, the analytic solution to (1) is described by:

$$\mathbf{x}_k = \mathbf{A}^k\mathbf{x}_0 + \sum_{i=0}^{k-1}\mathbf{A}^{k-i-1}\mathbf{B}\mathbf{u}_i + \sum_{i=0}^{k-1}\mathbf{A}^{k-i-1}\mathbf{f}(\mathbf{x}_i, \mathbf{w}_i). \tag{20}$$

Since, (16) is assumed to be true for k, one can state that:

$$\sum_{i=0}^{k-1}\mathbf{A}^{k-i-1}\mathbf{B}\mathbf{u}_i = \mathbf{b}_{k-1}, \tag{21}$$

and

$$\sum_{i=0}^{k-1}\mathbf{A}^{k-i-1}\mathbf{f}(\mathbf{x}_i, \mathbf{w}_i) \in [\mathbf{f}_{k-1}]. \tag{22}$$

Now, for $k+1$ one has:

$$\mathbf{x}_{k+1} = \mathbf{A}^{k+1}\mathbf{x}_0 + \sum_{i=0}^{k}\mathbf{A}^{k-i}\mathbf{B}\mathbf{u}_i + \sum_{i=0}^{k}\mathbf{A}^{k-i}\mathbf{f}(\mathbf{x}_i, \mathbf{w}_i). \tag{23}$$

Notice that, (23) can be rewritten as:

$$\mathbf{x}_{k+1} = \mathbf{A}^{k+1}\mathbf{x}_0 + \mathbf{B}\mathbf{u}_k + \sum_{i=0}^{k-1}\mathbf{A}^{k-i-1}\mathbf{B}\mathbf{u}_i + \mathbf{A}^k\mathbf{f}(\mathbf{x}_0, \mathbf{w}_0) + \sum_{i=0}^{k-1}\mathbf{A}^{k-i-1}\mathbf{f}(\mathbf{x}_i, \mathbf{w}_i). \tag{24}$$

Then, from (21), (22), and (24) one can state that:

$$\begin{array}{rcl} \mathbf{x}_{k+1} & \in & \mathbf{A}^{k+1}[\mathbf{x}_0] + \mathbf{B}\mathbf{u}_k + \mathbf{b}_{k-1} + \mathbf{A}^{k}[\mathbf{f}_0] + [\mathbf{f}_{k-1}] \\ & \in & \mathbf{A}^{k+1}[\mathbf{x}_0] + [\mathbf{f}_k] + \mathbf{b}_k. \end{array} \tag{25}$$

This completes the first part of the proof.

2) Boundedness proof: By construction, the width of the state enclosure at any time instant can be bounded as follows:

$$\begin{array}{rcl} w\big([\mathbf{x}_k]\big) & \leq & \|\mathbf{A}^k\| w\big([\mathbf{x}_0]\big) + w\big([\mathbf{f}_{k-1}]\big), \\ w\big([\mathbf{f}_k]\big) & \leq & \|\mathbf{A}^k\| w\big([\mathbf{f}_0]\big) + w\big([\mathbf{f}_{k-1}]\big). \end{array} \tag{26}$$

Since the matrix $\mathbf{A}$ is assumed to be Schur stable, one has:

$$\lim_{k\to+\infty} \mathbf{A}^k = 0. \tag{27}$$

Then, based on this matrix property, one can claim that:

$$\begin{array}{rcl} \lim_{k\to+\infty} w\big([\mathbf{x}_k]\big) & \leq & \lim_{k\to+\infty} \|\mathbf{A}^k\| w\big([\mathbf{x}_0]\big) + \lim_{k\to+\infty} w\big([\mathbf{f}_{k-1}]\big) \\ & \leq & \lim_{k\to+\infty} w\big([\mathbf{f}_{k-1}]\big) = \lim_{k\to+\infty} w\big([\mathbf{f}_k]\big) \\ & = & \text{a fixed point.} \end{array} \tag{28}$$

This completes the proof. $\square$

Remark 1. Since the matrices $\mathbf{A}$ and $\mathbf{B}$ and the input vector $\mathbf{u}_k$ are assumed to be perfectly known, the last recurrence equation in (16) is not implemented with interval arithmetic.

3.1 Periodically reinitialized interval predictor

In order to improve the precision of the interval-based predictor (16), we propose in this subsection a periodic version of it. Rather than using the global box $[\mathbf{f}_0] \supseteq \mathcal{F}$ to compute the state enclosures at every iteration, we propose in the new algorithm to use the following local boxes:

$$\forall \mathbf{x}_k \in [\mathbf{x}_k] \text{ and } \forall \mathbf{w}_k \in [\mathbf{w}_k] \;\; \mathbf{f}(\mathbf{x}_k, \mathbf{w}_k) \in [\mathbf{f}_k] = [\mathbf{f}]\big([\mathbf{x}_k], [\mathbf{w}_k]\big). \tag{29}$$

Remark 2. Unlike the definition of the boxes $[\mathbf{f}_k]$ used in (16), here the local boxes $[\mathbf{f}_k]$ stand for the *inclusion function* of $\mathbf{f}(.,.)$ assessed over the current boxes $[\mathbf{x}_k]$ and $[\mathbf{w}_k]$.

Hence, it is clear that, from a given state enclosure $[\mathbf{x}_s]$ determined at the time instant s one can predict in a guaranteed way the future state enclosure of the system by means of the following interval equation:

$$[\mathbf{x}_k] = \mathbf{A}^{k-s}[\mathbf{x}_s] + \mathbf{M}\begin{pmatrix} [\mathbf{f}_s] \\ [\mathbf{f}_{s+1}] \\ \vdots \\ [\mathbf{f}_{k-1}] \end{pmatrix} + \widetilde{\mathbf{B}}\begin{pmatrix} \mathbf{u}_s \\ \mathbf{u}_{s+1} \\ \vdots \\ \mathbf{u}_{k-1} \end{pmatrix}, \tag{30}$$

where $\mathbf{M} = \left(\mathbf{A}^{k-1-s}, \mathbf{A}^{k-2-s}, \ldots, \mathbf{I}_n\right)$, $\widetilde{\mathbf{B}} = \left(\mathbf{A}^{k-1-s}\mathbf{B}, \mathbf{A}^{k-2-s}\mathbf{B}, \ldots, \mathbf{B}\right)$ and $k > s$. Here, $\mathbf{I}_n$ stands for the identity matrix of dimension n. Even though this formula is less conservative than the one used in (16), its main drawback is its resource demanding. In fact, the size of the matrices $\mathbf{M}$ and $\widetilde{\mathbf{B}}$ are growing in time. At each iteration the new matrix $\mathbf{M}$ (resp. $\widetilde{\mathbf{B}}$) is constructed by an horizontal concatenation of the matrices $\mathbf{AM}$ and $\mathbf{I}_n$ (resp. $\mathbf{A}\widetilde{\mathbf{B}}$ and $\mathbf{B}$). Thus, the sizes of these matrices in function of k are: $n \times (k * (n+1))$ for $\mathbf{M}$ and $n \times (k * (m+1)$ for $\tilde{\mathbf{B}}$. Consequently, for a large value of k, one obtains matrices of huge dimension, which requires a lot of both computation time and memory space. To overcome this drawback, we propose in the following algorithm to reinitialize periodically this formula. The idea is to reset (30), with the current state enclosure, whenever the sizes of the matrices $\mathbf{M}$ and $\widetilde{\mathbf{B}}$ reach user-defined values.

Form the initial box $[\mathbf{x}_0]$, Algorithm 3.1 computes an outer approximation of the reachable set of (1) over the whole simulation time N, where the formula (30) is reinitialized periodically with user-defined period β.

In line 2, the endpoints of the first period are set. Then, in the beginning of the **while** loop the matrices $\widetilde{\mathbf{A}}$, $\widetilde{\mathbf{B}}$, $\mathbf{M}$, the boxes $[\mathbf{F}]$, $[\mathbf{x}_s]$ and the real vector $\mathbf{U}$ are initialized for each new period β. Based on the interval formula (30), the **for** loop computes the state enclosures $[\mathbf{x}_j]$ at every time instant t_j. To prepare the next iteration, the size varying matrices $\mathbf{M}$ and $\widetilde{\mathbf{B}}$ are recomputed in the lines 9 and 10. The new matrices are column concatenations of the matrices $\{\mathbf{AM},\ \mathbf{I}_n\}$ and $\{\mathbf{A}\widetilde{\mathbf{B}},\ \mathbf{B}\}$, respectively. In the same way, the new vector $\mathbf{U}$ and box $[\mathbf{F}]$ are constructed, in lines 11 and 12, by row concatenations of the vectors $\{\mathbf{U},\ \mathbf{u}_j\}$ and the boxes $\{[\mathbf{F}],\ [\mathbf{f}_j]\}$, respectively.

Remark 3. To be efficient against pessimism propagation the period β has to be chosen such that the following equality is satisfied,

$$\|\mathbf{A}^{\beta}\| \ll 1. \tag{31}$$

4 Illustrative examples

To show the performance of the proposed interval prediction algorithms, two examples borrowed from the literature [6, 16] are considered in this section.

Algorithm 1 Periodic Interval Predictor

Require: $[\mathbf{x}_0]$, N, β;

Set: $T_i := 1$; $j := T_i$; $T_f := \beta$;

while $(T_f < N)$;

- $\widetilde{\mathbf{A}} := \mathbf{I}_n$; $\widetilde{\mathbf{B}} := \mathbf{B}$; $\mathbf{M} := \mathbf{I}_n$;
- $[\mathbf{F}] := [\mathbf{f}_{j-1}]$; $[\mathbf{x}_s] := [\mathbf{x}_{j-1}]$; $\mathbf{U} := \mathbf{u}_{j-1}$;
- **for** $j = T_i$ **to** $j = T_f$
  - $\widetilde{\mathbf{A}} := \widetilde{\mathbf{A}}\mathbf{A}$;
  - $[\mathbf{x}_j] := \widetilde{\mathbf{A}}[\mathbf{x}_s] + \mathbf{M}[\mathbf{F}] + \widetilde{\mathbf{B}}\mathbf{U}$;
  - $\mathbf{M} := \big(\mathbf{A}\mathbf{M},\ \mathbf{I}_n\big)$;
  - $\widetilde{\mathbf{B}} := \big(\mathbf{A}\widetilde{\mathbf{B}},\ \mathbf{B}\big)$;
  - $\mathbf{U} := \big(\mathbf{U};\ \mathbf{u}_j\big)$;
  - $[\mathbf{F}] := \big([\mathbf{F}];\ [\mathbf{f}_j]\big)$;
- **end**
- $T_i := T_f + 1$; $T_f := T_f + \beta$;

end

Return: $[\mathbf{x}_0]$, $[\mathbf{x}_1]$, $\ldots$, $[\mathbf{x}_N]$;

4.1 Example 1

Consider the following nonlinear discrete-time system:

$$\mathbf{x}_{k+1} = \begin{pmatrix} 0.3 & -0.7 \\ 0.6 & -0.5 \end{pmatrix} \mathbf{x}_k + \delta \begin{pmatrix} \sin(0.5kx_{2k}) \\ \sin(0.3k) \end{pmatrix} + \begin{pmatrix} \sin(0.1k) \\ \cos(0.2k) \end{pmatrix}, \tag{32}$$

where the poorly-known nonlinear term in (32) is assumed to be bounded,

$$\forall \delta \in [-0.5,\ 0.5], \forall x_{2k} \in [x_{2k}], \quad \delta \begin{pmatrix} \sin(0.5kx_{2k}) \\ \sin(0.3k) \end{pmatrix} \in \begin{pmatrix} [-0.5,\ 0.5] \\ [-0.5,\ 0.5] \end{pmatrix}. \tag{33}$$

As show in Figure 3, an outer approximation of the reached set of this system generated from the initial state box $\mathbf{x}_0 = [5,\ -5] \times [5,\ -5]$ is computed by the interval predictor (16). This outer approximation is depicted in dashed curves and the elapsed computation time (with a processor Intel(R) Core(TM) i7-8650U CPU @ 1.90 GHz 2.11 GHz) is in the order of millisecond. The thin curves, confined between the dashed curves, represent some possible state trajectories of the system (32) generated randomly from the initial state box $[\mathbf{x}_0]$.

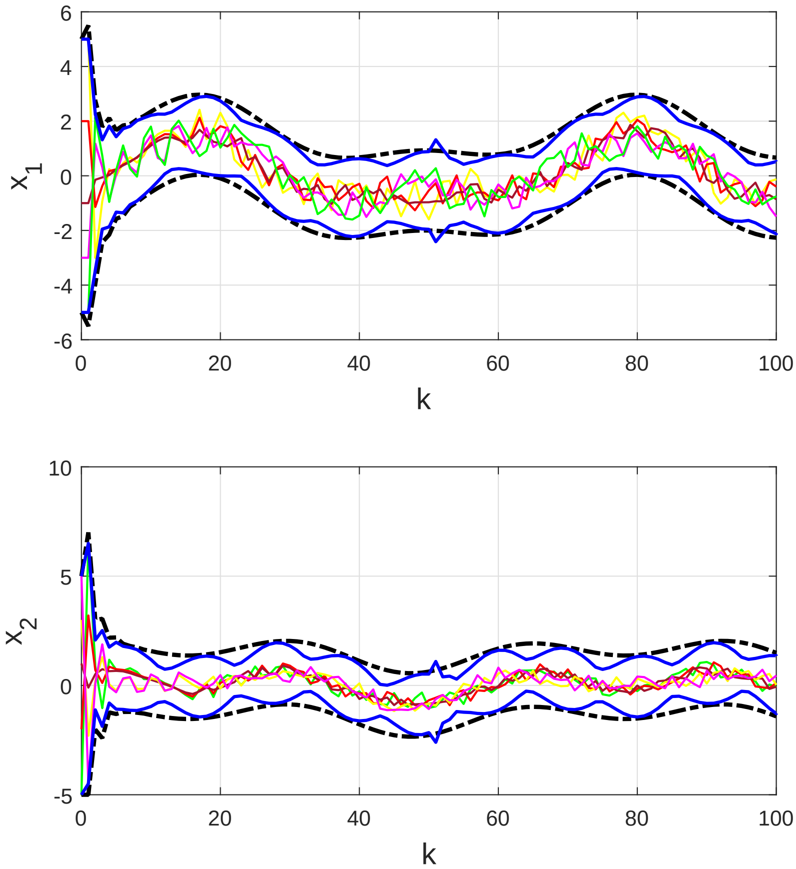

Figure 3: Predicted outer approximations of state trajectories. Dashed lines show the result of the Interval Predictor. Continuous lines present the results of the Periodic Interval Predictor.

As expected, the application of the periodically reinitialized interval predictor allowed to get a tighter outer approximation than that obtained in the first experience. This new outer approximation is depicted with the bold continuous lines in Figure 3. Notice, the observed oscillations in the outer approximation provided by the second algorithm are connected to the sinusoidal boxes used to locally overestimate the nonlinear uncertain term:

$$\forall k \geq 0,\ [\mathbf{f}(\mathbf{x}_k, \mathbf{w}_k)] = \begin{pmatrix} [\delta][\sin(0.5k[x_{2k}]) \\ [\delta]\sin(0.3k) \end{pmatrix}. \tag{34}$$

The paid price to get this improvement is the computation time, which is in this experience slightly larger than that of the first experience. Note that, this algorithm is applied with a re-initialization period $\beta = 50$ and the small inflation observed at the time instant $k = 51$ is due to the re-initialization effect.

4.2 Example 2

In this second example, we compare the performance of our approach to that introduced in [16]. The considered uncertain system in [16] is defined as follows:

$$\mathbf{x}_{k+1} = \frac{1}{2}\begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 1 \\ 1 & -1 & -1 \end{pmatrix}\mathbf{x}_k + \mathbf{f}(w_k), \tag{35}$$

where the poorly-known term $\mathbf{f}(.)$ is assumed to be bounded,

$$\forall k \geq 0, \quad \begin{pmatrix} 0 \\ 2\sin(3k) - 1 \\ 0 \end{pmatrix} \leq \mathbf{f}(w_k) \leq \begin{pmatrix} 0 \\ 2 \\ 0 \end{pmatrix}. \tag{36}$$

The outer approximation of the reachable set of this system, generated from the initial box $[\mathbf{x}_0] = [5,\ -5] \times [7,\ -2] \times [6,\ -3]$ and computed by means of the method proposed in [16], is plotted with the bold continuous lines on the Figure 4. The result of the periodically reset interval predictor is shown by the dashed curves. As observed in this figure, the proposed algorithm yields tighter enclosure of the actual state trajectory of the system depicted by the thin curves. This state trajectory is obtained with,

$$\mathbf{f}(w_k) = \begin{pmatrix} 0 \\ 2\sin(3k) \\ 0 \end{pmatrix}, \tag{37}$$

and started from the initial state $\mathbf{x}_0 = (1, 1, 2)^t$. Note that, the used reset period in this experience is $\beta = 10$ and the elapsed computation time is about 0.012 s. In addition, we highlight that this computation time is lower than that required by the method introduced in [16], which is about 0.027 s.

5 Conclusion

Two algorithms for computing outer approximations of the reachable set of uncertain discrete-time dynamical systems have been introduced in this paper. The main idea of these algorithms is to combine the mathematical expression of the state response of linear discrete-time systems with interval analysis to get tight enclosures of the exact reached set. Indeed, the proposed approach allows to effectively avoid the wrapping effect. Simulation results are shown to illustrate the performance of these algorithms. In addition, it is worth pointing out that these algorithms can be also implemented with other super-sets like ellipsoids and zonotopes.

In forthcoming works, we attempt to combine these reachability algorithms with the set-membership consistency techniques to deal with the safety verification

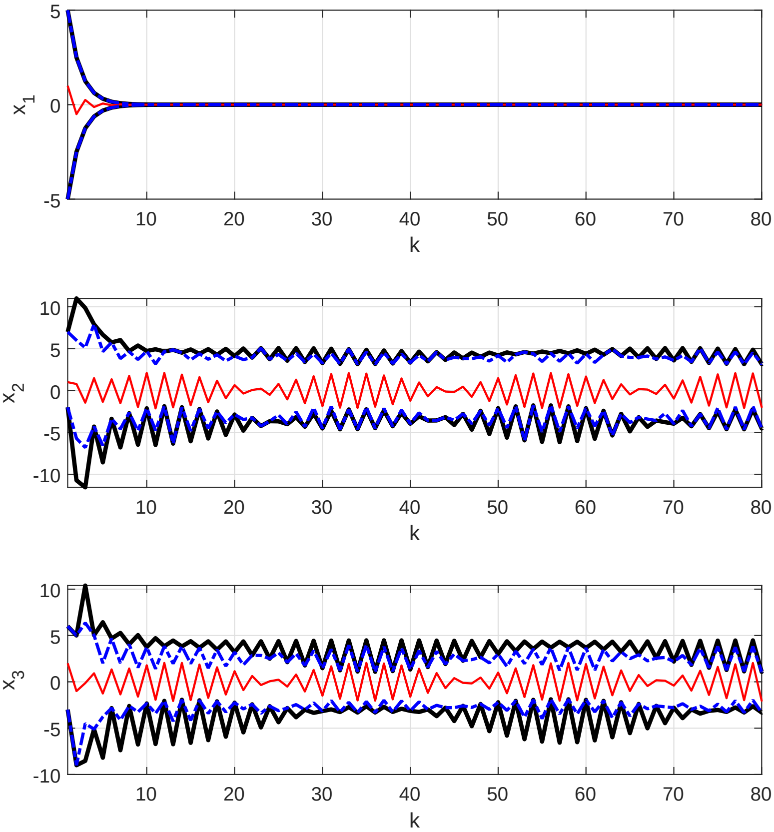

Figure 4: The predicted outer approximations. Bold continuous curves show the results of the method introduced in [16]. Dashed curves illustrate the results of the periodically reset interval predictor.

problem for uncertain dynamical systems. The proposed approach is useful for designing verification algorithms able to provide reliable decisions on the healthy behavior of dynamical systems, despite of the presence of uncertainties in their environments.

References

[1] Alefeld, G. and Mayer, G. Interval analysis: theory and applications. *Journal of Computational and Applied Mathematics*, 121:421–464, 2000. DOI: 10.1016/S0377-0427(00)00342-3.

[2] Chisci, L., Garulli, A., and Zappa, G. Recursive state bounding by parallelotopes. *Automatica*, 32:1049–1055, 1996. DOI:

10.1016/0005-1098(96)00048-9.

[3] Combastel, C. A state bounding observer based on zonotopes. In *Proc. of European Control Conference*, pages 2589–2594, Cambridge, UK, 2003. DOI: 10.23919/ECC.2003.7085991.

[4] Corliss, G. F. Survey of interval algorithms for ordinary differential equations. *Applied Mathematics and Computation*, 31:112–120, 1989. DOI: 10.1016/0096-3003(89)90112-4.

[5] Daryin, A. N., Kurzhanski, A. B., and Vostrikov, I. V. Reachability approaches and ellipsoidal techniques for closed-loop control of oscillating systems under uncertainty. *In Proc. of 51st IEEE Conference on Decision and Control, San Diego, CA, USA*, pages 6390–6395, 2006. DOI: 10.1109/CDC.2006.377784.

[6] Efimov, D., W. Perruquetti, T. Rassi, and Zolghadri, A. On interval observer design for time-invariant discrete-time systems. In *IEEE European Control Conference*, pages 2651–2656, Zurich, Switzerland, 2013. DOI: 10.23919/ECC.2013.6669108.

[7] Guernic, C. Le and Girard, A. Reachability analysis of linear systems using support functions. *Nonlinear Analysis: Hybrid Systems*, 4:250262, 2010. DOI: 10.1016/j.nahs.2009.03.002.

[8] Jaulin, L., Kieffer, M., Didrit, O., and Walter, E. *Applied interval analysis: with examples in parameter and state estimation, robust control and robotics.* Springer-Verlag, London, 2001. DOI: 10.1007/978-1-4471-0249-6.

[9] Kieffer, M., Jaulin, L., and Walter, E. Guaranteed recursive nonlinear state estimation using interval analysis. *International Journal of Adaptative Control and Signal Processing*, 16:193–218, 2002. DOI: 10.1109/CDC.1998.761917.

[10] Krasnochtanova, I., Rauh, A., Kletting, M., Aschemann, H., Hoferd, E. P., and Schoope, K. M. Interval methods as a simulation tool for the dynamics of biological wastewater treatment processes with parameter uncertainties. *Applied Mathematical Modelling*, 34:744–762, 2010. DOI: 10.1016/j.apm.2009.06.019.

[11] Kühn, W. Rigorously computed orbits of dynamical systems without the wrapping effect. *Computing*, 61(1):47–67, 1998. DOI: 10.1007/BF02684450.

[12] Kurzhanski, A. B. and Vályi, I. *Ellipsoidal calculus for estimation and control.* Birkhäuser Boston, 1996. DOI: 10.1007/978-1-4612-0277-6.

[13] Kurzhanski, A. B. and Varaiya, P. On ellipsoidal techniques for reachability analysis. part I: External approximations. *Journal of Optimization Methods and Software*, 17(1):177206, 2002. DOI: 10.1080/1055678021000012426.

[14] Kurzhanski, A. B. and Varaiya, P. Ellipsoidal techniques for reachability analysis of discrete-time linear systems. *IEEE Transactions on Automatic Control*, 52(1):2638, 2007. DOI: `10.1109/TAC.2006.887900`.

[15] Loukkas, N., Meslem, N., and Martinez, J.J. Set-membership tests to evaluate the performance of nominal feedback control laws. In *IEEE Conference on Control Applications (CCA)*, pages 1300–1305, Buenos Aires, Argentina, 2016. DOI: `10.1109/CCA.2016.7587986`.

[16] Mazenc, F., Dinh, T. N., and Niculescu, S. L. Interval observers for discrete-time systems. *International journal of robust and nonlinear control*, 24:2867–2890, 2014. DOI: `10.1002/rnc.3030`.

[17] Meslem, N., Le, V.T.H., Labarre, C., Lecoeuche, S., Kotny, J.L., and Idir, N. Set-membership methods applied to identify high-frequency elements of emi filters. *Control Engineering Practice*, 29:13–22, 2014. DOI: `10.1016/j.conengprac.2014.03.006`.

[18] Moore, R. E. *Interval analysis.* Englewood Cliff, New Jersey: Prentice-Hall, 1966.

[19] Moore, R. E., Kearfott, R. B., and Cloud, M. J. *Introduction to Interval Analysis.* Prentice-Hall, Englewood Cliffs, NJ, 1966. DOI: `10.1137/1.9780898717716`.

[20] Nedialkov, N. S., Jackson, K. R., and Pryce, J. D. An effective high-order interval method for validating existence and uniqueness of the solution of an IVP for an ODE. *Reliable Computing*, 7:449–465, 2001. DOI: `10.1023/A:1014798618404`.

[21] Pico, H. N. V. and Aliprantis, D. C. Voltage ride-through capability verification of DFIG-based wind turbines using reachability analysis. *IEEE Transactions on Energy Conversion*, 51:13871398, 2016. DOI: `10.1109/TEC.2016.2563320`.

[22] Pico, H. N. V. and Aliprantis, D. C. Reachability analysis of linear dynamic systems with constant, arbitrary, and lipschitz continuous inputs. *Automatica*, 95:293–305, 2018. DOI: `10.1016/j.automatica.2018.05.026`.

[23] Polyak, B. T., Nazin, S. A., Durieu, C., and Walter, E. Ellipsoidal parameter or state estimation under model uncertainty. *Automatica*, 40:1171–1179, 2004. DOI: `10.1016/j.automatica.2004.02.014`.

[24] Puig, V. Fault diagnosis and fault tolerant control using set-membership approaches: Application to real case studies. *International Journal of Applied Mathematics and Computer Science*, 20:619–635, 2010. DOI: `10.2478/v10006-010-0046-y`.

[25] Puig, V., Quevedo, J., Escobet, T., Nejjari, F., and Heras, S. De Las. Passive robust fault detection of dynamic processes using interval models. *IEEE Transactions on Control Systems Technology*, 16:1083–1089, 2008. DOI: 10.1109/TCST.2007.906339.

[26] Puig, V., Stancu, A., Escobet, T., Nejjari, F., Quevedo, J., and Patton, R. J. Passive robust fault detection using interval observers: Application to the DAMADICS benchmark problem. *Control engineering practice*, 14:621–633, 2006. DOI: 10.1016/j.conengprac.2005.03.016.

[27] Rakovic, S. V., Kerrigan, E. C., Mayne, D. Q., and Lygeros, J. Reachability analysis of discrete-time systems with disturbances. *IEEE Transactions on Automatic Control*, 51(4):546 – 561, 2006. DOI: 10.1109/TAC.2006.872835.

[28] Ramdani, N., Meslem, N., and Candau, Y. A hybrid bounding method for computing an over-approximation for the reachable set of uncertain nonlinear systems. *IEEE Transactions on Automatic Control*, 24(10):2352–2364, 2009. DOI: 10.1109/TAC.2009.2028974.

[29] Ramdani, N., Meslem, N., and Candau, Y. Computing reachable sets for uncertain nonlinear monotone systems. *Nonlinear Analysis: Hybrid Systems*, 4(2):263–278, 2010. DOI: 10.1016/j.nahs.2009.10.002.