Acta Cybernetica **27** (2026) 405–428.

A Muller Approach for Reachability

Luc Jaulin[ab]

Abstract

This paper presents an approach to deal with the prediction of dynamical systems in case of interval uncertainties. These uncertainties can be both on the initial state vector, on the time-dependent inputs and on the evolution function. The approach is based on the Muller theorem often used in an interval context to perform the prediction of cooperative systems. We show here that the Muller approach can be used for general non-cooperative systems. We also show the benefit we can obtain by using a conditioning approach in order to reduce the overestimation.

Keywords: differential inclusion, interval analysis, interval integration, reachability, Muller theorem

1 Introduction

Reachability has been studied by many authors using set-membership tools [4, 6, 9, 11, 18, 28]. Often, the objective of reachability is to predict the future of a dynamical system under uncertainties [27].

The dynamical system we consider has the form $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$, where $\mathbf{x}$ is the state vector and $\mathbf{u}$ is the input vector. The input satisfies $\mathbf{u}(\cdot) \in [\mathbf{u}](\cdot)$ which means that, for all t, $\mathbf{u}(t) \in [\mathbf{u}](t)$, where $[\mathbf{u}](t)$ is a box which depends on t. The initial state vector satisfies $\mathbf{x}(0) \in \mathbb{X}_0$. We want to compute the set

$$\mathbb{X}(t) = \left\{\mathbf{a} | \, \exists \mathbf{x}(0) \in \mathbb{X}_0, \exists \mathbf{u}(\cdot) \in [\mathbf{u}](\cdot), \, \mathbf{a} = \varphi_{t,\mathbf{u}(\cdot)}(\mathbf{x}(0))\right\} \tag{1}$$

where $\varphi_{t,\mathbf{u}(\cdot)}$ is the flow, as illustrated by Figure 1. In this figure, three feasible trajectories are represented. The red is *true* one. The blue trajectory starts from the true $\mathbf{x}(0)$, but deviates from the red due to the uncertainty on $\mathbf{u}$. The black trajectory has the same input but starts from a feasible $\mathbf{x}(0)$ which is not the true one.

From a mathematical point of view, solving a reachability problem amounts to integrating a differential inclusion [5] and requires the use of interval analysis [19]. A general approach to solve this problem has been proposed in [14] but

[a]Robex, Lab-STICC, ENSTA-Bretagne, France
[b]E-mail: lucjaulin@gmail.com, ORCID: 0000-0002-0938-0615

DOI: 10.14232/actacyb.323265

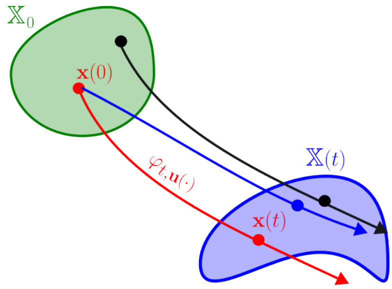

Figure 1: Reach set $\mathbb{X}(t)$ that we want to enclose

the understanding of the approach is not easy and requires the computation of logarithmic norms. Another tool for reachability is CORA [1] with a nice user interface and detailed tutorials. General bounding results for computing reachable sets using differential inequalities can also be found in [26]. See also [7] for an approach based on using mixed monotonicity. A comparison between several tools for reachability can be found in [10].

Probably the first approach to solve this problem is proposed in 1927 by Muller [20]. Due to the fact that interval computation were not know at these times, the use of the Muller's approach was limited to cooperative systems, even if the Muller results are valid for much more general systems. The cooperative property means that the evolution function $\mathbf{f}(\mathbf{x}, \mathbf{u})$ has some monotony properties that are met by only few systems. Since, several people have used the Muller approach, [23, 15, 12] but again, these contributions remain in a context of cooperative systems, even if some change of coordinates are needed to meet cooperativity (see, e.g., [22])).

The contributions of this paper are the following:

- Clearly underline the cooperativity property is not required at all, when using a Muller approach.
- Show a way to transform the reachability problem into a simple ordinary differential equations (ODE).
- Show that conditioning techniques, classically used in the interval algorithms, could reduce drastically the pessimism, in case of small uncertainties.
- Provide an enclosure which is asymptotically minimal [13], *i.e.*, which is minimal when the uncertainties are infinitesimal.

The paper is organized as follows. Section 2 introduces *interval dynamical systems* (IDS) which is a class of differential inclusions that can be integrated as an ODE. Section 3 shows how interval computation combined with the Muller can provide an enclosure of the reachability problem. Section 4 defines conditioners, combines them with a Muller enclosure method and gives some theoretical results on the asymptotic minimality of conditioned IDS. Section 5 gives an application in robotics. Section 6 concludes the paper.

2 Interval dynamical system

In this section, we give the definition of interval dynamical systems, show how they can be used to represent a dynamical system with interval uncertainties and explain how they can be simulated using the Muller theorem.

2.1 Principle

An *interval* $[x] = [x^-, x^+]$ is a connected closed subset of $\mathbb{R}$. The *lower bound* of $[x]$ is $\text{lb}([x]) = x^-$ and its *upper bound* is $\text{ub}([x]) = x^+$. A *box* of $\mathbb{R}^n$ is the Cartesian product of n intervals:

$$[\mathbf{x}] = [x_1^-, x_1^+] \times \cdots \times [x_n^-, x_n^+]. \tag{2}$$

Denote by $\mathbb{IR}$ the set of intervals and by $\mathbb{IR}^n$ the set of all boxes of $\mathbb{R}^n$. The width of a box $[\mathbf{x}]$ is denoted by $w([\mathbf{x}])$.

Definition 2.1. *As illustrated by Figure 2, an* interval dynamical system *(IDS) is a system of the form*

$$\begin{array}{rcl} \dot{\mathbf{z}}(t) & = & \mathbf{f}_z(\mathbf{z}(t), \mathbf{u}^-(t), \mathbf{u}^+(t), t) \\ [\mathbf{x}](t) & = & \mathbf{h}(\mathbf{z}(t), t) \end{array} \tag{3}$$

where $\mathbf{h} : \mathbb{R}^q \times \mathbb{R} \mapsto \mathbb{IR}^n$ is a continuous one to one function. The vector $\mathbf{z} \in \mathbb{R}^q$ is called the epistemic state vector*. It is qualified as* epistemic*, since, as we will see later, it represents the knowledge we have of the real state $\mathbf{x}$ of a dynamical system. The evolution function for $\mathbf{z}$, denoted by $\mathbf{f}_z$, is assumed to be continuous and locally Lipschitz in $\mathbf{x}$. It should not be confused with the evolution function $\mathbf{f}$ for $\mathbf{x}$ introduced in the previous section. The quantities $[\mathbf{u}](t) = [\mathbf{u}^-(t), \mathbf{u}^+(t)]$ and $[\mathbf{x}](t) = [\mathbf{x}^-(t), \mathbf{x}^+(t)]$ are time dependent boxes or* tubes *[17, 25]. The IDS should satisfy the* thinness *property*

$$\left.\begin{array}{r} w([\mathbf{x}](0)) = 0 \\ \forall t, w([\mathbf{u}](t)) = 0 \end{array}\right\} \Rightarrow \forall t, w([\mathbf{x}](t)) = 0 \tag{4}$$

which means that if both the initial state $\mathbf{x}(0)$ and the input $\mathbf{u}(0)$ are perfectly known, the IDS does not produce any uncertainty.

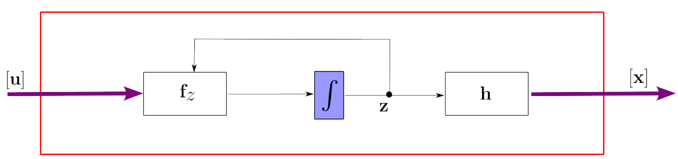

Figure 2: Interval dynamical system (IDS). Thick arrows are interval valued

Definition 2.2. *Consider a dynamical system*

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \tag{5}$$

where $\mathbf{f}$ *is continuous and differentiable in* $\mathbf{x}, \mathbf{u}$*, locally Lipschitz in* $\boldsymbol{x}$ *and where* $\mathbf{u}$ *is piecewise continuous, so that a single solution exists for any initial state vector. The IDS (3) is said to be* an enclosing IDS *of (5) if we have*

$$\left.\begin{array}{c} \mathbf{x}(0) \in [\mathbf{x}](0) \\ \forall t, \mathbf{u}(t) \in [\mathbf{u}^-(t), \mathbf{u}^+(t)] \end{array}\right\} \Rightarrow \forall t, \mathbf{x}(t) \in [\mathbf{x}](t). \tag{6}$$

Note that the condition $\mathbf{x}(0) \in [\mathbf{x}](0)$ implies that $\mathbf{h}(\mathbf{z}(0)) \in [\mathbf{x}](0)$.

Example 2.1. The IDS

$$\begin{array}{rcl} \dot{z}_1 & = & z_1 + u^- \\ \dot{z}_2 & = & z_2 + u^+ \\ [x] & = & h(\mathbf{z}) = [z_1, z_2] \end{array} \tag{7}$$

encloses the system $\dot{x} = x + u$ for $u \in [u] = [u^-, u^+]$.

2.2 Muller IDS

In the literature, IDS are generally given in a more specific case which we call here the Muller form [20].

Definition 2.3. *An IDS is of* Muller type *if it has the form*

$$\left\{ \begin{array}{ccc} \underbrace{\left(\begin{array}{c} \dot{\mathbf{x}}^-(t) \\ \dot{\mathbf{x}}^+(t) \end{array} \right)}_{\dot{\mathbf{z}}(t)} & = & \underbrace{\left(\begin{array}{c} \mathbf{f}^-(\mathbf{x}^-(t), \mathbf{x}^+(t), \mathbf{u}^-(t), \mathbf{u}^+(t)) \\ \mathbf{f}^+(\mathbf{x}^-(t), \mathbf{x}^+(t), \mathbf{u}^-(t), \mathbf{u}^+(t)) \end{array} \right)}_{\mathbf{f}_z(\mathbf{z}(t), \mathbf{u}^-(t), \mathbf{u}^+(t))} \\ [\mathbf{x}](t) & = & \underbrace{[\mathbf{x}^-(t), \mathbf{x}^+(t)]}_{\mathbf{h}(\mathbf{z}(t))} \end{array} \right. . \tag{8}$$

The thinness *property becomes*

$$\mathbf{f}^-(\mathbf{x}, \mathbf{x}, \mathbf{u}, \mathbf{u}) \quad = \quad \mathbf{f}^+(\mathbf{x}, \mathbf{x}, \mathbf{u}, \mathbf{u}). \tag{9}$$

IDS generate interval trajectories $[\mathbf{x}^-(t), \mathbf{x}^+(t)]$ (represented by the epistemic state vector $\mathbf{z}$) that are supposed to enclose the trajectory of an uncertain dynamical system. It corresponds to an ordinary differential equation (ODE) which can be integrated using a Runge-Kutta method or using an interval integration [14, 21].

The enclosure property (see Definition 2.2) is illustrated for Muller IDS by Figure 3. The *true* trajectory (blue) $\mathbf{x}(t)$ is between the two red trajectories $\mathbf{x}^-(t)$ and $\mathbf{x}^+(t)$, or equivalently,

$$\forall t, \mathbf{x}(t) \in [\mathbf{x}^-(t), \mathbf{x}^+(t)] = [\mathbf{x}](t). \tag{10}$$

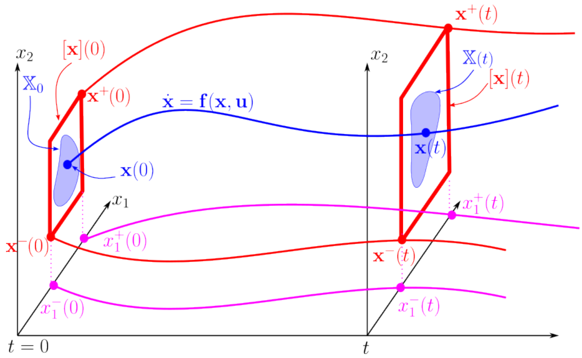

Figure 3: A Muller IDS enclosing a dynamical system

Proposition 2.1. *Consider a Muller IDS (see Equation 8) which satisfies*

$$\frac{\partial f_i^-}{\partial x_i^+} = 0, \ \frac{\partial f_i^+}{\partial x_i^-} = 0 \tag{11}$$

$$\frac{\partial f_i^-}{\partial x_j^-} \geq 0, \ \frac{\partial f_i^-}{\partial x_j^+} \leq 0, \ \frac{\partial f_i^+}{\partial x_j^-} \leq 0, \ \frac{\partial f_i^+}{\partial x_j^+} \geq 0, \quad \textit{for } i \neq j \tag{12}$$

and

$$\frac{\partial f_i^-}{\partial u_\ell^-} \geq 0, \ \frac{\partial f_i^-}{\partial u_\ell^+} \leq 0, \ \frac{\partial f_i^+}{\partial u_\ell^-} \leq 0, \ \frac{\partial f_i^+}{\partial u_\ell^+} \geq 0 \tag{13}$$

as soon as $\mathbf{x}^- \leq \mathbf{x}^+$ *and* $\mathbf{u}^- \leq \mathbf{u}^+$*. The IDS encloses the system* $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$ *with*

$$\mathbf{f}(\mathbf{x}, \mathbf{u}) = \mathbf{f}^-(\mathbf{x}, \mathbf{x}, \mathbf{u}, \mathbf{u}) \quad = \quad \mathbf{f}^+(\mathbf{x}, \mathbf{x}, \mathbf{u}, \mathbf{u}). \tag{14}$$

Note that these conditions on $\mathbf{f}$ are close to the notion of *mixed monotonicity*, introduced in [7]. Before giving the proof, let us give a simple illustrative example.

Example 2.2. An enclosing IDS for the dynamical system

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} x_1 x_2 \\ x_1 + u \end{pmatrix} \tag{15}$$

is

$$\begin{pmatrix} \dot{x}_1^- \\ \dot{x}_1^+ \\ \dot{x}_2^- \\ \dot{x}_2^+ \end{pmatrix} = \begin{pmatrix} \min(x_1^- x_2^-, x_1^- x_2^+) \\ \max(x_1^+ x_2^-, x_1^+ x_2^+) \\ x_1^- + u^- \\ x_1^+ + u^+ \end{pmatrix}. \tag{16}$$

Indeed,

$$\mathbf{f}^-(\mathbf{x}, \mathbf{x}, \mathbf{u}, \mathbf{u}) = \mathbf{f}^+(\mathbf{x}, \mathbf{x}, \mathbf{u}, \mathbf{u}) = \mathbf{f}(\mathbf{x}, \mathbf{u}) = \begin{pmatrix} x_1 x_2 \\ x_1 + u \end{pmatrix}. \tag{17}$$

Moreover, we have

$$\left\{\begin{array}{lll} \frac{\partial f_1^-}{\partial x_2^-} & =\max(x_1^-,0) & \geq 0 \\ \frac{\partial f_1^-}{\partial x_2^+} & =\min(x_1^-,0) & \leq 0 \\ \frac{\partial f_1^+}{\partial x_2^-} & =\min(x_1^+,0) & \leq 0 \\ \frac{\partial f_1^+}{\partial x_2^+} & =\max(x_1^+,0) & \geq 0 \\ \frac{\partial f_2^-}{\partial x_1^-} & =1 & \geq 0 \\ \frac{\partial f_2^-}{\partial x_1^+} & =0 & \leq 0 \\ \frac{\partial f_2^+}{\partial x_1^-} & =0 & \leq 0 \\ \frac{\partial f_2^+}{\partial x_1^+} & =1 & \geq 0 \end{array}\right. \text{ and } \left\{\begin{array}{lll} \frac{\partial f_1^-}{\partial u^-} =0 & \geq & 0 \\ \frac{\partial f_1^-}{\partial u^+} =0 & \leq & 0 \\ \frac{\partial f_1^+}{\partial u^-} =0 & \geq & 0 \\ \frac{\partial f_1^+}{\partial u^+} =0 & \leq & 0 \\ \frac{\partial f_2^-}{\partial u^-} =1 & \geq & 0 \\ \frac{\partial f_2^-}{\partial u^+} =0 & \leq & 0 \\ \frac{\partial f_2^+}{\partial u^-} =0 & \leq & 0 \\ \frac{\partial f_2^+}{\partial u^+} =1 & \geq & 0 \end{array}\right. . \quad (18)$$

Proof. (of Proposition 2.1). The proof is by induction. First, by assumption (see (2)), we have $\mathbf{x}(0) \in [\mathbf{x}^-(0), \mathbf{x}^+(0)]$. Assume that the proposition is true for t. Let us prove that it is also true for $t+dt$, where dt is infinitesimal. Thus, we have to prove that

$$\mathbf{x}(t+dt) \in [\mathbf{x}^-(t+dt), \mathbf{x}^+(t+dt)] \quad (19)$$

i.e., for all i,

$$x_i^-(t+dt) \leq x_i(t+dt) \leq x_i^+(t+dt). \quad (20)$$

or equivalently

$$\begin{array}{llll} (i) & x_i^+(t+dt) - x_i(t+dt) & \geq & 0 \\ (ii) & x_i^-(t+dt) - x_i(t+dt) & \leq & 0. \end{array} \quad (21)$$

We just check (i) for $i=1$, *i.e.*, we want to prove that

$$x_1^+(t+dt) - x_1(t+dt) \geq 0. \quad (22)$$

The same reasoning can be done to prove (ii) and for $i>1$.

Since $x_1^+(t) - x_1(t) \geq 0$, we have to consider two cases

• Case 1. $x_1^+(t) - x_1(t) > 0$. By continuity of the function $e_1^+(t) = x_1^+(t) - x_1(t)$ with respect to t, we deduce that $e_1^+(t+dt) = x_1^+(t+dt) - x_1(t+dt) > 0$, since dt is infinitesimal.

• Case 2. $x_1^+(t) - x_1(t) = 0$. In this case, $x_1^+(t+dt) - x_1(t+dt) \geq 0$ if

$$f_1^+(\mathbf{x}^-(t), \mathbf{x}^+(t), \mathbf{u}^-(t), \mathbf{u}^+(t)) - f_1(\mathbf{x}(t), \mathbf{u}(t)) \geq 0 \quad (23)$$

This is what we will now prove. From (9), we have

$$f_1^+\begin{pmatrix} x_1, x_2, \ldots, x_n \\ x_1, x_2, \ldots, x_n \\ u_1, u_2, \ldots, u_m \\ u_1, u_2, \ldots, u_m \end{pmatrix} - f_1\begin{pmatrix} x_1, x_2, \ldots, x_n \\ u_1, u_2, \ldots, u_m \end{pmatrix} = 0. \quad (24)$$

Moreover, since

$$\begin{cases} \frac{\partial f_1^+}{\partial x_j^-} \leq 0, & \frac{\partial f_1^+}{\partial x_j^+} \geq 0 & \text{for } j \neq 1 & \text{(see (12))} \\ \frac{\partial f_1^+}{\partial u_\ell^-} \leq 0, & \frac{\partial f_1^+}{\partial u_\ell^+} \geq 0 & \forall \ell & \text{(see (13))} \\ x_j^- \leq x_j \leq x_j^+, & & \forall j & \\ u_\ell^- \leq u_\ell \leq u_\ell^+, & & \forall \ell & \end{cases}$$

we get

$$f_1^+ \begin{pmatrix} x_1, x_2^-, \ldots, x_n^- \\ x_1, x_2^+, \ldots, x_n^+ \\ u_1^-, u_2^-, \ldots, u_m^- \\ u_1^+, u_2^+, \ldots, u_m^+ \end{pmatrix} - f_1 \begin{pmatrix} x_1, x_2, \ldots, x_n \\ u_1, u_2, \ldots, u_m \end{pmatrix} \geq 0. \tag{25}$$

Since $x_1^+ = x_1$, and since f_1^+ does not depend on x_1^- ($\partial f_1^+ / \partial x_1^- = 0$, see (11)), we get Inequality (23). □

3 Interval analysis to get an enclosing interval dynamical system

In this section, we show how we can use interval computation to get a Muller dynamical system which encloses a given dynamical system.

3.1 Inclusion function

An *inclusion function* $[\mathbf{f}]$ for $\mathbf{f} : \mathbb{R}^n \mapsto \mathbb{R}^m$, is a function

$$[\mathbf{f}] : \begin{array}{ccc} \mathbb{IR}^n & \to & \mathbb{IR}^m \\ [\mathbf{x}] & \to & [\mathbf{f}]([\mathbf{x}]) \end{array} \tag{26}$$

such that [19]

$$\mathbf{f}([\mathbf{x}]) \subset [\mathbf{f}]([\mathbf{x}]). \tag{27}$$

An inclusion function $[\mathbf{f}]$ is *thin* if for any singleton $[\mathbf{x}] = \{\mathbf{x}\}$, $[\mathbf{f}]([\mathbf{x}])$ is also a singleton. It is *inclusion monotonic* if

$$[\mathbf{x}] \subset [\mathbf{y}] \Rightarrow [\mathbf{f}]([\mathbf{x}]) \subset [\mathbf{f}]([\mathbf{y}]). \tag{28}$$

3.2 Epistemic transform

We present here the epistemic transform which will be used to reformulate a differential inclusion into an ODE. The main idea is to explore the boundary of $[\mathbf{x}]$ to compute a box enclosing $\mathbf{f}([\mathbf{x}])$. This makes sense when the function is known to be one to one in $[\mathbf{x}]$. The idea of using the boundary has been explored by several authors in the context of dynamical systems (see, *e.g.*, [29]).

Definition 3.1. *The* epistemic transform *of the interval function*

$$[\mathbf{f}]: \begin{array}{ccc} \mathbb{IR}^n & \to & \mathbb{IR}^m \\ [\mathbf{x}] & \to & [\mathbf{f}]([\mathbf{x}]) \end{array} \tag{29}$$

is given by

$$\boldsymbol{\Gamma}([\mathbf{f}]): \begin{pmatrix} x_1^- \\ x_1^+ \\ x_2^- \\ x_2^+ \\ \vdots \\ x_n^- \\ x_n^+ \end{pmatrix} \mapsto \begin{pmatrix} lb(& [f_1](x_1^-, [x_2], [x_3], \dots, [x_n]) &) \\ ub(& [f_1](x_1^+, [x_2], [x_3], \dots, [x_n]) &) \\ lb(& [f_2]([x_1], x_2^-, [x_3], \dots, [x_n]) &) \\ ub(& [f_2]([x_1], x_2^+, [x_3], \dots, [x_n]) &) \\ & \vdots & \\ lb(& [f_m]([x_1], [x_2], [x_3], \dots, x_n^-) &) \\ ub(& [f_m]([x_1], [x_2], [x_3], \dots, x_n^+) &) \end{pmatrix}. \tag{30}$$

We will write

$$\widehat{[\mathbf{f}]} = \boldsymbol{\Gamma}([\mathbf{f}]). \tag{31}$$

As defined above, the epistemic transform is a functional operator which transforms an interval function $[\mathbf{f}] : \mathbb{IR}^n \mapsto \mathbb{IR}^m$ into another function $\widehat{[\mathbf{f}]} : \mathbb{R}^{2n} \mapsto \mathbb{R}^{2n}$.

Example 3.1. The epistemic transform of

$$[\mathbf{f}]([\mathbf{x}]) = \begin{pmatrix} [x_1] \cdot [x_2] \\ \exp\left([x_1] - \sin[x_2]\right) \end{pmatrix} \tag{32}$$

is

$$\widehat{[\mathbf{f}]} \begin{pmatrix} x_1^- \\ x_1^+ \\ x_2^- \\ x_2^+ \end{pmatrix} = \begin{pmatrix} \min(x_1^- x_2^-, x_1^- x_2^+) \\ \max(x_1^+ x_2^-, x_1^+ x_2^+) \\ \exp\left(x_1^- - \sin x_2^-\right) \\ \exp\left(x_1^+ - \sin x_2^+\right) \end{pmatrix}. \tag{33}$$

Definition 3.2. *Define the* epistemic canonical injection *as the function*

$$\boldsymbol{\gamma}: \begin{array}{ccc} \mathbb{IR}^n & \to & \mathbb{R}^{2n} \\ [\mathbf{x}] & \to & \left(x_1^-, x_1^+ \quad , \dots, x_n^-, x_n^+ \right)^T. \end{array} \tag{34}$$

We will write

$$\widehat{[\mathbf{x}]} = \boldsymbol{\gamma}([\mathbf{x}]). \tag{35}$$

which means that the hat *operator transforms a box of* $\mathbb{R}^n$ *into an equivalent* $2n$*-vector. The epistemic canonical injection* $\boldsymbol{\gamma}$ *generates a vector containing the bounds of its interval inputs. This function has a unique reciprocal* $\boldsymbol{\gamma}^{-1}$ *for which*

$$dom(\boldsymbol{\gamma}^{-1}) = \left\{ \left(x_1^-, x_1^+ \quad , \dots, x_n^-, x_n^+ \right) \mid \forall i, x_i^- \leq x_i^- \right\} \tag{36}$$

i.e., $\boldsymbol{\gamma}^{-1} \circ \boldsymbol{\gamma}([\mathbf{x}]) = [\mathbf{x}]$, *for all* $[\mathbf{x}] \in \mathbb{IR}^n$. *Note that*

$$\boldsymbol{\gamma}^{-1} \circ \widehat{[\mathbf{f}]} \circ \boldsymbol{\gamma}([\mathbf{x}]) \subset [\mathbf{f}]([\mathbf{x}]). \tag{37}$$

The reason for this inclusion is that $\boldsymbol{\gamma}^{-1} \circ \widehat{[\mathbf{f}]} \circ \boldsymbol{\gamma}$ evaluates the box $[\mathbf{x}]$ its boundary (via each of its $2n$ faces) whereas $[\mathbf{f}]$ evaluates $[\mathbf{x}]$ in its boundary and its interior. As a consequence, there may exist a $\mathbf{x} \in [\mathbf{x}]$ such that $\mathbf{f}(\mathbf{x})$ is not inside $\boldsymbol{\gamma}^{-1} \circ \widehat{[\mathbf{f}]} \circ \boldsymbol{\gamma}([\mathbf{x}])$. It may happen if for one i, the function f_i has a local extremum inside the interior of $[\mathbf{x}]$.

3.3 Enclosing IDS

The following proposition shows that from a general differential inclusion $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$,$\mathbf{u} \in [\mathbf{u}]$, we can derive an enclosing IDS.

Proposition 3.1. *An enclosing IDS for*

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t) \tag{38}$$

is

$$\begin{pmatrix} \dot{x}_1^- \\ \dot{x}_1^+ \\ \dot{x}_2^- \\ \dot{x}_2^+ \\ \vdots \\ \dot{x}_n^- \\ \dot{x}_n^+ \end{pmatrix} = \begin{pmatrix} lb(& [f_1](x_1^-, [x_2], [x_3], \dots, [x_n], [\mathbf{u}], t) &) \\ ub(& [f_1](x_1^+, [x_2], [x_3], \dots, [x_n], [\mathbf{u}], t) &) \\ lb(& [f_2]([x_1], x_2^-, [x_3], \dots, [x_n], [\mathbf{u}], t) &) \\ ub(& [f_2]([x_1], x_2^+, [x_3], \dots, [x_n], [\mathbf{u}], t) &) \\ & \vdots & \\ lb(& [f_n]([x_1], [x_2], [x_3], \dots, x_n^-, [\mathbf{u}], t) &) \\ ub(& [f_n]([x_1], [x_2], [x_3], \dots, x_n^+, [\mathbf{u}], t) &) \end{pmatrix} \tag{39}$$

or equivalently

$$\begin{array}{rcl} \dot{\mathbf{z}} & = & \widehat{[\mathbf{f}]}(\mathbf{z}, [\mathbf{u}], t) \\ \mathbf{z} & = & \boldsymbol{\gamma}([\mathbf{x}]) \end{array} \tag{40}$$

where $[f_i]$ are thin inclusion functions for the f_i's that are assumed to be inclusion monotonic. The box $[\mathbf{u}]$ and time t are seen as parameters and are not involved in the epistemic transform.

Figure 4 illustrates the equation $\dot{x}_1^- = \mathrm{lb}([f_1](x_1^-, [x_2], \mathbf{u}))$ in a two dimensional state space. We see that the evolution of the left face of box $[\mathbf{x}]$ only depends on the vector field $\mathbf{f}(\mathbf{x}, \mathbf{u})$ on the left face $(x_1^-, [x_2])$. More than this, the evolution depends only on the first component $f_1(\mathbf{x}, \mathbf{u})$ of $\mathbf{f}(\mathbf{x}, \mathbf{u})$. We take the lower bound to be sure that the left face will not go faster than any feasible trajectory.

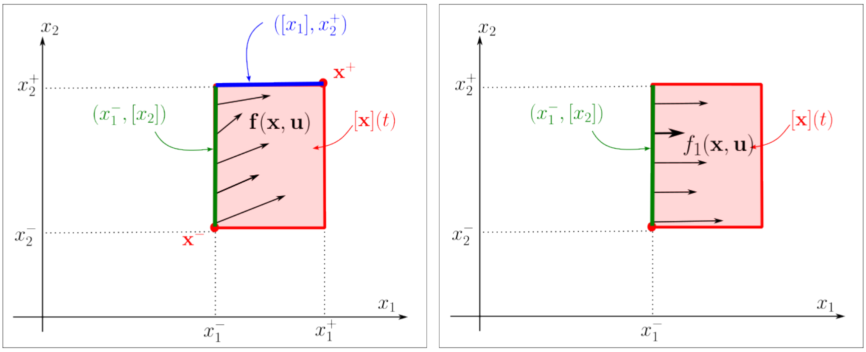

Figure 4: The left face should not go faster than the bold arrow

Proof. To prove the proposition, we apply Proposition 2.1. First, in (39), since for all i, $\dot{x}_i^-$ do not depend on x_i^+ and $\dot{x}_i^+$ do not depend on x_i^-, Condition 11 is satisfied. To prove (12) and (13), consider first an interval function $[f] = [f^-, f^+]$ which is inclusion monotonic, we have

$$\begin{array}{rcc} [a, b^-] \subset [a, b^+] & \Rightarrow & [f]([a, b^-]) \subset [f]([a, b^+]) \\ & \Leftrightarrow & [f^-([a, b^-]), f^+([a, b^-])] \subset [f^-([a, b^+]), f^+([a, b^+])] \\ & \Leftrightarrow & \left\{ \begin{array}{lcl} f^-([a, b^-]) & \geq & f^-([a, b^+]) \\ f^+([a, b^-]) & \leq & f^+([a, b^+]) \end{array} \right. \end{array}$$

which means that f^- is decreasing with respect to the upper bound b of its interval argument and f^+ is increasing with respect to this bound. Equivalently, we get that f^+ is increasing with respect to the lower bound a of its interval argument and f^+ is decreasing with respect to a. If we apply this reasoning to $[f_i]$ which is inclusion monotonic we get (12) and (13). Property 5 comes from the fact that the inclusion function is thin. □

3.4 Example: the pendulum

An enclosing IDS for the pendulum

$$\begin{array}{rcl} \dot{x}_1 & = & x_2 \\ \dot{x}_2 & = & -\sin x_1 - x_2 + u \end{array} \tag{41}$$

is

$$\begin{array}{rcl} \dot{\mathbf{z}} & = & \widehat{[\mathbf{f}]}(\mathbf{z}, [u]) \\ \mathbf{z} & = & \boldsymbol{\gamma}([\mathbf{x}]) \end{array} \tag{42}$$

where

$$\begin{array}{rcl} \mathbf{f}(\mathbf{x},u) & = & \begin{pmatrix} x_2 \\ -\sin x_1 - x_2 + u \end{pmatrix} \\ \widehat{[\mathbf{f}]}(\mathbf{z},[u]) & = & \underbrace{\begin{pmatrix} z_3 \\ z_4 \\ \mathrm{lb}(-\sin([z_1,z_2])) - z_3 + u^- \\ \mathrm{ub}(-\sin([z_1,z_2])) - z_4 + u^+ \end{pmatrix}}_{f_z(z,[u])} \\ \mathbf{z} & = & \begin{pmatrix} x_1^- & x_1^+ & x_2^- & x_2^+ \end{pmatrix}^{\mathrm{T}} \\ [\mathbf{x}] & = & [z_1,z_2]\times[z_3,z_4]. \end{array} \tag{43}$$

We simulate our IDS for $t \in [0,6]$, a sampling time $dt = 0.001$ and an initial state box $[\mathbf{x}](0) = [1-\varepsilon, 1+\varepsilon], \times[0.2-\varepsilon, 0.2+\varepsilon]$ where $\varepsilon = 0.01$ is a small positive number representing some uncertainties. For the input box, we have taken $[u] = [-\varepsilon^2, \varepsilon^2]$. The simulation yields Figures 5 and 6. Even if the pendulum is a stable system, we observe an instability of the IDS, where the all bounds $x_1^-, x_1^+, x_2^-, x_2^+$ diverge.

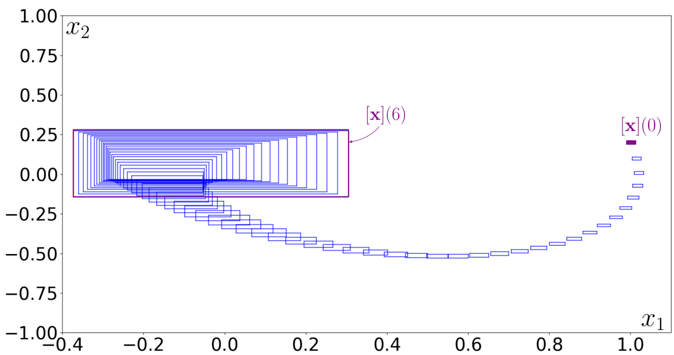

Figure 5: Boxes $[\mathbf{x}](t)$ generated by the Muller IDS

4 Conditioner

In this section, we propose a method to get an accurate IDS, that we call *asymptotically minimal* (see [13] for more details concerning the asymptotic minimality). More precisely, if the width $w([\mathbf{x}](0))$ of the initial box $[\mathbf{x}](0)$ is in $O(\varepsilon)$ (where ε is a tiny positive number), if the width of the input box is small compared to ε, *i.e.*, $w([\mathbf{u}]) = o(\varepsilon)$, then we will have an overestimation introduced by the IDS which is small compared to ε, *i.e.*, in $o(\varepsilon)$.

A conditioner can be seen as a change of coordinates moving along a reference trajectory. The principle is classical when we want to use linear enclosure methods to reduce the conservatism [14, 16, 8].

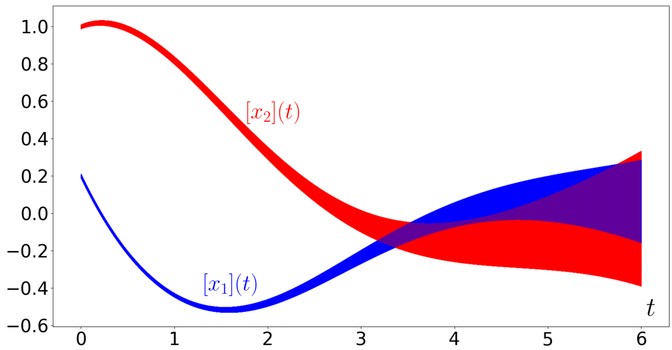

Figure 6: Tubes $[x_1](t)$ and $[x_2](t)$ generated by the Muller IDS

4.1 Principle

Consider again the dynamical system

$$\begin{array}{c} \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \\ \mathbf{x}(0) \in [\mathbf{x}](0) \\ \mathbf{u}(t) \in [\mathbf{u}](t) \end{array} \qquad (44)$$

for which we want to compute get an IDS.

We propose to add a *conditioner* as illustrated by Figure 7. Here, conditioner has to be understood in its generic form, *i.e.*, something that improves the enclosure, making it more accurate.

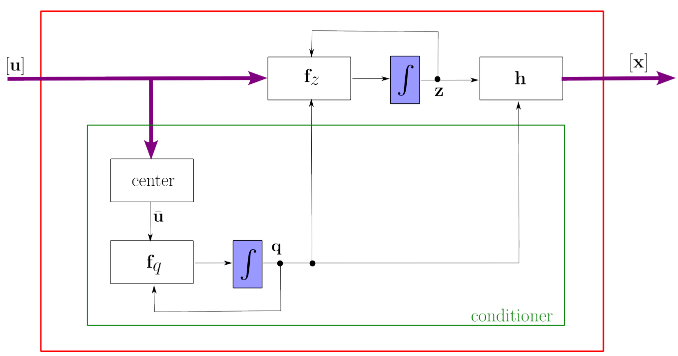

Figure 7: The conditioner improves the accuracy of the interval integration

As a results, we get

$$\begin{array}{rcl}\dot{\mathbf{z}} & = & \mathbf{f}_z(\mathbf{z},\mathbf{q},[\mathbf{u}]) \\ \dot{\mathbf{q}} & = & \mathbf{f}_q(\mathbf{q},\text{center}([\mathbf{u}])) \\ [\mathbf{x}] & = & \mathbf{h}(\mathbf{z},\mathbf{q})\end{array} \tag{45}$$

i.e.,

$$\begin{array}{rcl}\dot{\mathbf{z}} & = & \mathbf{f}_z(\mathbf{z},[\mathbf{u}],t) \\ [\mathbf{x}] & = & \mathbf{h}(\mathbf{z},t)\end{array} \tag{46}$$

where $\mathbf{f}_z(\mathbf{z},[\mathbf{u}],t)$ and $\mathbf{h}(\mathbf{z},t)$ correspond to $\mathbf{f}_z(\mathbf{z},\mathbf{q}(t),[\mathbf{u}])$ and $\mathbf{h}(\mathbf{z},\mathbf{q}(t))$, respectively. This is clearly an IDS which can be integrated as an ODE, as it will be shown now.

4.2 IDS with the conditioner

To find the right conditioner, we compute one trajectory $\mathbf{a}(t)$ which corresponds to an approximation of the feasible trajectories. For instance, we can chose $\mathbf{a}(t)$ such that

$$\begin{array}{c}\dot{\mathbf{a}} = \mathbf{f}(\mathbf{a},\bar{\mathbf{u}}) \\ \mathbf{a}(0) = \text{center}([\mathbf{x}](0)) \\ \bar{\mathbf{u}}(t) = \text{center}([\mathbf{u}](t)).\end{array} \tag{47}$$

Then we will make a change of coordinates at the neighborhood of the trajectory $\mathbf{a}(t)$.

The following theorem corresponds to the main contribution of the paper. It shows that for a large class of differential inclusion $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{u}),\mathbf{u} \in [\mathbf{u}]$, we can generate an enclosing IDS with some nice asymptotic properties. The main idea of this theorem is to define an error $\mathbf{e}$ between $\mathbf{x}$ and $\mathbf{a}$ in the new local frame. Then we build an IDS for the evolution of the error $\mathbf{e}$ introducing the epistemic state vector $\mathbf{z}$ associated with the box $[\mathbf{e}]$. This will allow us to build a parallelepiped $\langle\mathbf{x}\rangle(t)$ which encloses the trajectory $\mathbf{x}(t)$.

Theorem 4.1. *An enclosing IDS for the system* $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{u})$ *is (see Figure 8)*

$$\begin{array}{rcl}\dot{\mathbf{J}} & = & \left(\frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{a},\bar{\mathbf{u}})\right)\cdot\mathbf{J} \\ \dot{\mathbf{a}} & = & \mathbf{f}(\mathbf{a},\bar{\mathbf{u}}) \\ \dot{\mathbf{z}} & = & \widehat{[\mathbf{f}_e]}(\mathbf{z},[\mathbf{u}]) \\ [\mathbf{e}] & = & \boldsymbol{\gamma}(\mathbf{z}) \\ [\mathbf{x}] & = & \mathbf{J}\cdot[\mathbf{e}]+\mathbf{a}\end{array} \tag{48}$$

where

$$\begin{array}{rcl}\widehat{[\mathbf{f}_e]} & = & \boldsymbol{\Gamma}([\mathbf{f}_e]) \\ [\mathbf{f}_e]([\mathbf{e}]) & = & \mathbf{f}_e(\bar{\mathbf{e}})+\mathbf{J}_e\cdot([\mathbf{e}]-\bar{\mathbf{e}}) \\ \mathbf{f}_e(\mathbf{e}) & = & \mathbf{J}^{-1}\left(-\dot{\mathbf{J}}\mathbf{e}+\mathbf{f}(\mathbf{J}\mathbf{e}+\mathbf{a},[\mathbf{u}])-\dot{\mathbf{a}}\right) \\ \mathbf{J}_e([\mathbf{e}]) & = & -\mathbf{J}^{-1}\dot{\mathbf{J}}+\left(\frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{J}\cdot[\mathbf{e}]+\mathbf{a},[\mathbf{u}])\right)\cdot\mathbf{J} \\ \bar{\mathbf{e}} & = & \textit{center}([\mathbf{e}]).\end{array} \tag{49}$$

For the initialization, we take $\mathbf{J}(0) = \mathbf{I}$*,* $\mathbf{a}(0) = center([\mathbf{x}](0))$ *and* $[\mathbf{e}](0) = [\mathbf{x}](0) - \mathbf{a}(0)$*.*

Moreover if $w([\mathbf{x}](0)) = O(\varepsilon)$ *and* $w([\mathbf{u}](t)) = o(\varepsilon)$*, where* ε *is infinitesimal, then, for a given* t*,*

(i) The parallelepiped $\langle \mathbf{x} \rangle (t) = \{\mathbf{x} | \exists \mathbf{e} \in [\mathbf{e}](t), \mathbf{x} = \mathbf{J}(t) \cdot \mathbf{e} + \mathbf{a}(t)\}$ *is such that the Hausdorff distance* $H([\mathbf{x}](t)], \mathbb{X}(t))$ *between the box* $[\mathbf{x}](t)]$ *and the reach set*

$$\mathbb{X}(t) = \left\{ \mathbf{a} | \, \exists \mathbf{x}(0) \in [\mathbf{x}](0), \exists \mathbf{u}(\cdot) \in [\mathbf{u}], \, \mathbf{a} = \varphi_{t, \mathbf{u}(\cdot)}(\mathbf{x}(0)) \right\} \tag{50}$$

at time t *is* $o(\varepsilon)$*.*

(ii) The box $[\mathbf{x}](t)$ *generated by the conditioned IDS (48) is such that the Hausdorff distance* $H([\mathbf{x}](t)], [\mathbb{X}(t)])$ *between* $[\mathbf{x}](t)]$ *and the hull box* $[\mathbb{X}(t)]$ *of* $\mathbb{X}(t)$ *at time* t *is* $o(\varepsilon)$*.*

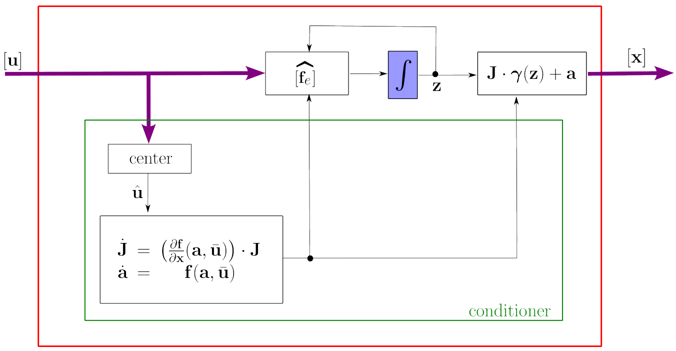

Figure 8: The conditioner increases the accuracy of the interval integrator

Remark 4.1. The two first equations of (48) are named the *variational equations* [2]. The pair $(\mathbf{a}, \mathbf{J})$ follows the flow, *i.e.*, the function $\mathbf{\Phi}_t(\mathbf{x}_0)$ which returns the state vector $\mathbf{x}(t)$ reached at time t assuming that $\mathbf{x}(0)$ has been initialized at $\mathbf{x}_0$. The matrix $\mathbf{J}$ satisfies $\mathbf{J}(t) = \frac{\partial \mathbf{\Phi}_t(\mathbf{x}_0)}{\partial \mathbf{x}}$. The principle of the conditioner is to store all uncertainties in the error $\mathbf{e}$ and propagate them through a single box $[\mathbf{e}]$ which will be shown to be almost static. More precisely, the inflation is in $o(\varepsilon)$.

Proof. Consider a trajectory of the system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$. Since $\mathbf{f}$ is C_1 in $\mathbf{x}$, the Jacobian matrix $\mathbf{J}(t)$ is invertible for all t [3]. This is a consequence of Liouville's formula for linear systems:

$$\frac{d}{dt} \det \mathbf{J}(t) = \mathrm{tr} \left(\frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{a}, \bar{\mathbf{u}}) \right) \cdot \det \mathbf{J}(t). \tag{51}$$

This scalar differential equation yields

$$\det \mathbf{J}(t) = \exp\left(\int_0^t \mathrm{tr} \left(\frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{a}, \bar{\mathbf{u}}) \right) \, ds \right). \tag{52}$$

which never vanishes. Define the vector (seen as an error)

$$\mathbf{e} = \mathbf{J}^{-1}(t)\,(\mathbf{x} - \mathbf{a}(t)) \tag{53}$$

as illustrated by Figure 9. The first step of the proof is to show that $\mathbf{e}$ satisfies $\dot{\mathbf{e}} = \mathbf{f}_e(\mathbf{e})$ where $\mathbf{f}_e$ is given in (49), and to show that $\dot{\mathbf{e}} = o(\varepsilon)$.

By differentiating the relation $\mathbf{Je} = \mathbf{x} - \mathbf{a}$ with respect to t, we get:

$$\begin{array}{rcl} \mathbf{J}\dot{\mathbf{e}} + \dot{\mathbf{J}}\mathbf{e} & = & \dot{\mathbf{x}} - \dot{\mathbf{a}} \\ & = & \mathbf{f}(\mathbf{x}, \mathbf{u}) - \dot{\mathbf{a}} \\ & = & \mathbf{f}(\mathbf{Je} + \mathbf{a}, \mathbf{u}) - \dot{\mathbf{a}}. \end{array} \tag{54}$$

Thus,

$$\dot{\mathbf{e}} \;=\; \underbrace{\mathbf{J}^{-1}\left(-\dot{\mathbf{J}}\mathbf{e} + \mathbf{f}(\mathbf{Je} + \mathbf{a}, \mathbf{u}) - \dot{\mathbf{a}}\right)}_{\mathbf{f}_e(\mathbf{e})}. \tag{55}$$

Note that

$$\begin{array}{rcl} \mathbf{f}_e(\mathbf{e}) & = & \mathbf{J}^{-1} \cdot \left(- \underbrace{\dot{\mathbf{J}}}_{=\left(\frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{a},\bar{\mathbf{u}})\right)\cdot \mathbf{J}} \mathbf{e} + \underbrace{\mathbf{f}(\mathbf{Je} + \mathbf{a}, \mathbf{u})}_{=\mathbf{f}(\mathbf{a},\mathbf{u}) + \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{a},\mathbf{u})\cdot(\mathbf{Je}) + o(\|\mathbf{e}\|)} - \mathbf{f}(\mathbf{a}, \bar{\mathbf{u}}) \right) \\ & = & \mathbf{J}^{-1} \cdot \underbrace{(\mathbf{f}(\mathbf{a}, \mathbf{u}) - \mathbf{f}(\mathbf{a}, \bar{\mathbf{u}}))}_{o(\varepsilon)} + \underbrace{\left(\frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{a}, \mathbf{u}) - \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{a}, \bar{\mathbf{u}}) \right)}_{o(\varepsilon)} \cdot (\mathbf{Je}) + o(\|\mathbf{e}\|) \\ & = & o(\varepsilon). \end{array}$$

A direct consequence of the fact that $\mathbf{f}_e(\mathbf{e}) = o(\varepsilon)$ is that $\mathbf{\Gamma}([\mathbf{f}_e]) = o(\varepsilon)$, since we used the centered form. We would have $\mathbf{\Gamma}([\mathbf{f}_e]) = O(\varepsilon)$ if we had used the natural interval extension [19]. Thus $\dot{\mathbf{z}} = o(\varepsilon)$, whereas $\mathbf{z}(0) = O(\varepsilon)$.

Define the error set $\mathbb{E}(t) = \{\mathbf{e} | \exists \mathbf{x} \in \mathbb{X}(t), \mathbf{e} = \mathbf{J}^{-1}(t)(\mathbf{x} - \mathbf{a}(t))$. Due to the fact that $\dot{\mathbf{z}} = o(\varepsilon)$, we have $H(\mathbb{E}(t), [\mathbf{e}](t)) = o(\varepsilon)$ and therefore, $H(\mathbb{X}(t), \langle \mathbf{x} \rangle (t)) = o(\varepsilon)$ which corresponds to (ii). We immediately get that $H([\mathbb{X}(t)], [\mathbf{x}](t)) = o(\varepsilon)$ which corresponds to (i). □

4.3 Pendulum

Consider again the system given in Subsection 3.4, with $t_{\max} = 20$ instead of $t_{max} = 6$. Our conditioned IDS yields Figure 10. The trajectories painted yellow have been generated with $u = 0$ and starting with each corner of the initial box $[\mathbf{x}](0)$. With two different level of zooms (red and magenta), we illustrate the asymptotic minimality of the enclosure. For large t, we also observe a stability at the neighborhood of $\mathbf{0}$. More precisely, we can show that the state will enter inside small zone at the neighborhood of the origin and will stay in this zone forever.

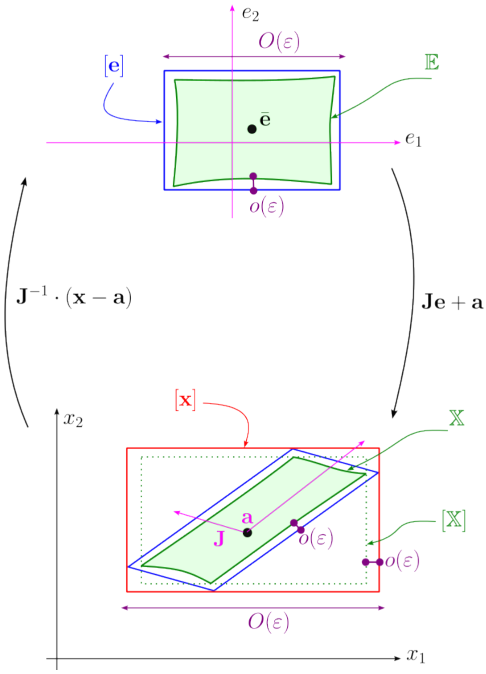

Figure 9: The conditioner increases the accuracy of the interval integrator

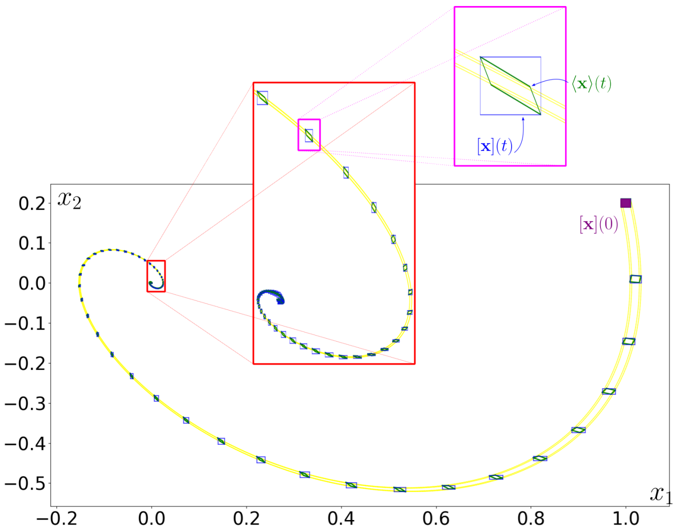

Figure 10: Boxes $[\mathbf{x}](t)$ and parallelepipeds $\langle\mathbf{x}\rangle(t)$ generated by our asymptotically minimal IDS

5 Application

5.1 Dead reckoning

Consider a dynamical system (for instance a vehicle such as a car or a boat) described by the state equations

$$\begin{array}{rcl} \dot{\mathbf{x}}_v & = & \mathbf{f}_v(\mathbf{x}_v, \mathbf{u}_v) \\ \mathbf{y}_v & = & \mathbf{g}_v(\mathbf{x}_v) + \mathbf{e}_v, \end{array} \tag{56}$$

where $\mathbf{x}_v$ is the state vector of the vehicle, $\mathbf{u}$ is the input vector, $\mathbf{y}$ is the measured output and $\mathbf{e}$ is the measured noise. Assume that we have a controller of the form

$$\begin{array}{rcl} \dot{\mathbf{x}}_c & = & \mathbf{f}_c(\mathbf{x}_c, \mathbf{y}_v) \\ \mathbf{u}_v & = & \mathbf{g}_c(\mathbf{x}_c) + \mathbf{e}_u. \end{array} \tag{57}$$

We have the extended system

$$\underbrace{\begin{pmatrix} \dot{\mathbf{x}}_v \\ \dot{\mathbf{x}}_c \end{pmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{pmatrix} \mathbf{f}_v(\mathbf{x}_v, \mathbf{g}_c(\mathbf{x}_c) + \mathbf{e}_u) \\ \mathbf{f}_c(\mathbf{x}_c, \mathbf{g}_v(\mathbf{x}_v) + \mathbf{e}_y) \end{pmatrix}}_{\mathbf{f}(\mathbf{x},\mathbf{u})} \tag{58}$$

where $\mathbf{x} = (\mathbf{x}_v, \mathbf{x}_c)$ is the extended state vector, $\mathbf{u} = (\mathbf{e}_u, \mathbf{e}_y)$ is the extended state input and $\mathbf{f} = (\mathbf{f}_v, \mathbf{f}_c)$ is the extended evolution function.

We want to show that the vehicle with its controller will enter a given zone $\mathbb{A}$ while avoiding a zone $\mathbb{B}$. This corresponds to a reachability problem for which we can use the preconditioned IDS.

5.2 Dubins car

The vehicle we consider is a Dubins car described by the state equations

$$\begin{array}{ll} (i) & \left\{ \begin{array}{rcl} \dot{x}_{v1} & = & x_{v4} \cos x_{v3} \\ \dot{x}_{v2} & = & x_{v4} \sin x_{v3} \\ \dot{x}_{v3} & = & u_{v1} + e_{u1} \\ \dot{x}_{v4} & = & u_{v2} + e_{u2} \end{array} \right. \\ (ii) & \left\{ \begin{array}{rcl} y_{v1} & = & x_{v3} + e_{y1} \\ y_{v2} & = & x_{v4} + e_{y2} \end{array} \right. \end{array} \tag{59}$$

where (i) is the evolution equation and (ii) is the output equation. The position is (x_{v1}, v_{v2}), the heading is x_{v3} and the speed is x_{v4}. The vehicle only measures its heading and its speed. At time $t = 0$, the state $\mathbf{x}_v$ is known to be inside a box $[\mathbf{x}_v](0)$. The noise of the actuators e_{u1}, e_{u2} are unknown and are assumed to be inside known intervals. The noise of the sensors e_{y1}, e_{y2} are also unknown and assumed to be inside known intervals.

5.3 Controller

We want to find a controller such that at time t_1 the position is inside the set $\mathbb{A} \subset \mathbb{R}^2$ and such that for all t, the position is never inside the set $\mathbb{B} \subset \mathbb{R}^2$. To find such a controller, we use a feedback linearization method. For this purpose, we ask the robot to follow a target point $\mathbf{c}(t)$ chosen such that (see Figure 11):

- $\mathbf{c}(0)$ is approximately the position of the car at $t = 0$,
- at time t_1 the point $\mathbf{c}$ is deep inside $\mathbb{A}$,
- $\mathbf{c}(t)$ clearly avoids $\mathbb{B}$.

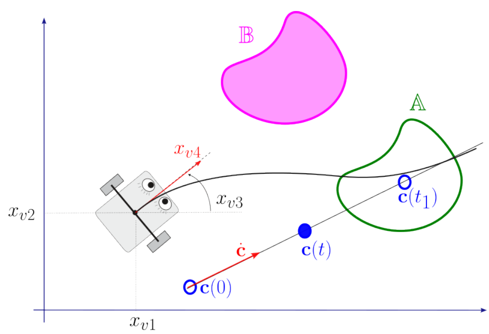

Figure 11: The Dubins car, has to reach $\mathbb{A}$ at time t_1 and to avoid $\mathbb{B}$

We get the following state feedback controller

$$\mathbf{u}_v = \begin{pmatrix} -\left(c_1(t) - x_{v1}\right)\frac{\sin x_{v3}}{x_{v4}} + \left(c_2(t) - x_{v2}\right)\frac{\cos x_{v3}}{x_4} \\ \left(c_1(t) - x_{v1}\right)\cos x_{v3} + \left(c_2(t) - x_{v2}\right)\sin x_{v3} - 2x_{v4} \end{pmatrix}. \quad (60)$$

Now, this controller cannot be implemented in this form since the position x_{v1}, x_{v2} are not measured. We need to use a predictor. It can be taken as (see the first two equation of 59)

$$\begin{array}{rcl} \dot{x}_{c1} & = & y_{v2} \cos y_{v1} \\ \dot{x}_{c2} & = & y_{v2} \sin y_{v1}. \end{array} \quad (61)$$

This prediction only uses a measure y_{v1} of the heading and a measure y_{v2} of the speed. This poor estimation can now be used by the controller (11), as illustrated by Figure 12.

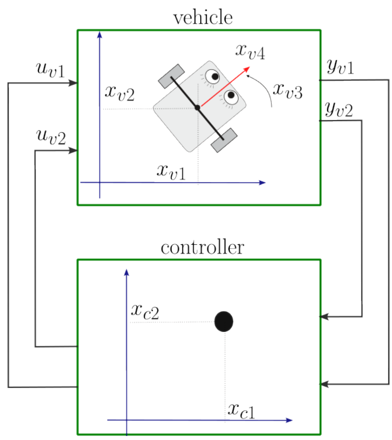

Figure 12: The Dubins car, has two outputs (heading and speed) used by the controller

The state equations of the controller are now

$$(iii) \quad \left\{ \begin{array}{lcl} \dot{x}_{c1} & = & y_{v2}\cos y_{v1} \\ \dot{x}_{c2} & = & y_{v2}\sin y_{v1} \end{array} \right.$$

$$(iv) \quad \left\{ \begin{array}{lcl} u_{v1} & = & -(c_1(t)-x_{c1})\frac{\sin y_{v1}}{y_{v2}} + (c_2(t)-x_{c2})\frac{\cos y_{v1}}{y_{v2}} \\ u_{v2} & = & (c_1(t)-x_{c1})\cos y_{v1} + (c_2(t)-x_{c2})\sin y_{v1} - 2y_{v2}. \end{array} \right.$$

5.4 Closed loop system

If we take the notation $\mathbf{x} = (\mathbf{x}_v, \mathbf{x}_c)$ and $\mathbf{u} = (\mathbf{e}_u, \mathbf{e}_y)$, the state equations of the closed loop system, including both the vehicle and the controller, are

$$\begin{array}{lcl} \dot{x}_1 & = & x_4\cos x_3 \\ \dot{x}_2 & = & x_4\sin x_3 \\ \dot{x}_3 & = & -(c_1(t)-x_5)\frac{\sin(x_3+u_3)}{x_4+u_4} + (c_2(t)-x_6)\frac{\cos(x_3+u_3)}{x_4+u_4} + u_1 \\ \dot{x}_4 & = & (c_1(t)-x_5)\cos(x_3+u_3) + (c_2(t)-x_6)\sin(x_3+u_3) - 2(x_4+u_4) + u_2 \\ \dot{x}_5 & = & (x_4+u_4)\cos(x_3+u_3) \\ \dot{x}_6 & = & (x_4+u_4)\sin(x_3+u_3). \end{array}$$

5.5 Reachability using the conditioned IDS

Assume that the initial state and the input $\mathbf{u}$ satisfy;

$$\begin{array}{lcl} \mathbf{x}(0) & \in & \underbrace{[-\bar{\varepsilon},\bar{\varepsilon}]\times[-\bar{\varepsilon},\bar{\varepsilon}]\times[-2,2]\times[1-\bar{\varepsilon},1+\bar{\varepsilon}]\times[0,0]\times[0,0]}_{=[\mathbf{x}](0)} \\ \mathbf{u}(t) & \in & \underbrace{[-\bar{\varepsilon}^2,\bar{\varepsilon}^2]\times[-\bar{\varepsilon}^2,\bar{\varepsilon}^2]\times[-\bar{\varepsilon}^2,\bar{\varepsilon}^2]\times[-\bar{\varepsilon}^2,\bar{\varepsilon}^2]}_{=[\mathbf{u}]},\forall t \end{array}$$

where $\bar{\varepsilon} = 10^{-3}$. For the target point, take:

$$\mathbf{c}_i(t) = \begin{pmatrix} 1.8 + 0.9 \cdot t \\ 0.4 + 0.2 \cdot t \end{pmatrix}. \tag{62}$$

The Muller integration of the IDS presented in Section 3 generates Figure 13. The blue set encloses all feasible positions for $t \in [0, 4]$. The red zones contain the position of the car for $t \in \{1, 2, 3\}$. The black curves represents a feasible trajectory.

We were able to prove that the position of the car has reached the set $\mathbb{A} = [2, 3] \times [0.2, 0.8]$ at time $t = 3$. Moreover, for all $t \in [0, 4]$ we proved that the car has never entered inside the forbidden zone $\mathbb{B} = [1, 2] \times [0.8, 1]$.

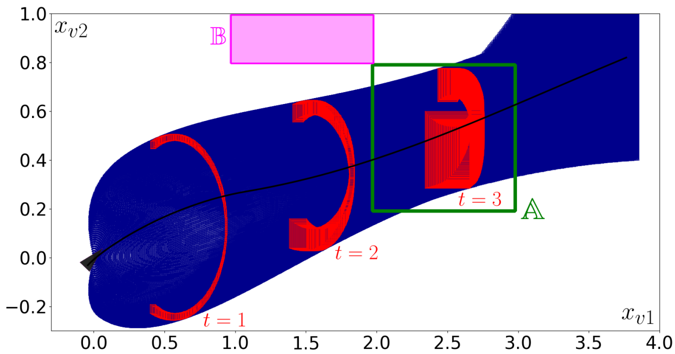

Figure 13: At time $t = 3$, the car reaches $\mathbb{A}$ and always avoid $\mathbb{B}$

With the same accuracy and using a conditioner, as presented in Section 4, the predictor yields Figure 3.

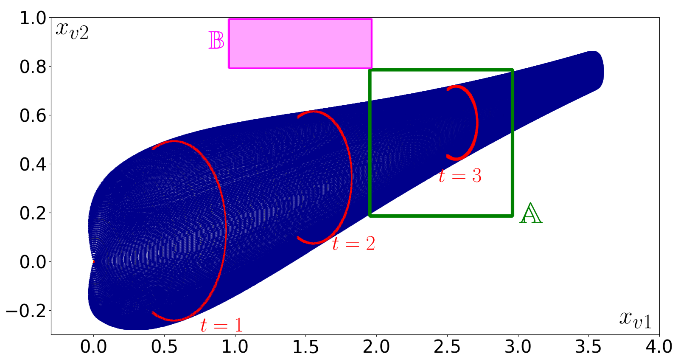

Figure 14: At time $t = 3$, the car has reached $\mathbb{A}$ always avoiding $\mathbb{B}$

For both, the results have been obtained by bisecting the initial box into 40 small boxes. For both, the resulting computing time was less than 500 sec.

Figure 15 shows the superposition of many feasible trajectories obtained by sampling. The pessimism (distance between the yellow and the blue approximations) can be considered as low.

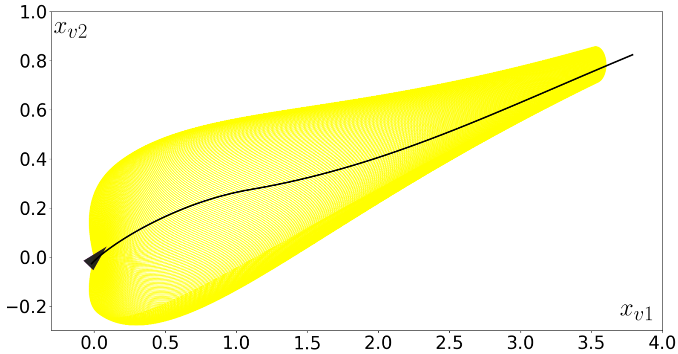

Figure 15: Many trajectories generated to get an inner approximation of the set of reachable positions

6 Conclusion

This paper has proposed a method to reformulate a reachability problem into an ordinary differential equation (ODE) using the Muller theorem. Solving numerically this ODE yields a interval tube which contains all feasible trajectories. One important intuition of using the Muller transform is to take into account the bijective property of the flow function. Indeed, if we know that a function $\mathbf{f} : \mathbb{R}^n \mapsto \mathbb{R}^n$ is a bijection, computing the range of $\mathbf{f}([\mathbf{x}])$, for a box $[\mathbf{x}]$, amounts to compute the range of $\mathbf{f}(\partial[\mathbf{x}])$, where $\partial[\mathbf{x}]$ is the boundary of $[\mathbf{x}]$.

Although our approach suffers from a wrapping effect, due to the interval approximation, we have shown that the approximation is asymptotically minimal if the right conditioner is chosen. This means that when the uncertainties are infinitesimal ($O(\varepsilon)$ for the initial state and $O(\varepsilon^2)$ for the state noise) then almost no overestimation is introduced (in $O(\varepsilon^2)$).

The implementation is done using the Codac library [24] and the source codes are available at https://webperso.ensta.fr/jaulin/muller.html.

References

[1] Althoff, M. An Introduction to CORA. *CPS Week*, pages 120–151, 2015. URL: https://mediatum.ub.tum.de/doc/1280439/document.pdf.

[2] Arnold, V. *Geometrical Methods In The Theory Of Ordinary Differential Equations*. Springer-Verlag, 1988. DOI: 10.1007/978-3-662-11832-0.

[3] Arnold, V. I. *Ordinary Differential Equations*. Springer, 2 edition, 1992. URL: https://link.springer.com/book/9783540345633.

[4] Asarin, E., Dang, T., and Girard, A. Reachability analysis of nonlinear systems using conservative approximation. In Maler, O. and Pnueli, A., editors, *Hybrid Systems: Computation and Control*, Volume 2623 of *Lecture Notes in Computer Science*, pages 20–35. Springer, 2003. DOI: 10.1007/3-540-36580-X_5.

[5] Aubin, J. P. *Viability Theory*. Birkhäuser Boston, 2009. DOI: 10.1007/978-0-8176-4910-4.

[6] Collins, P. and Goldsztejn, A. The reach-and-evolve algorithm for reachability analysis of nonlinear dynamical systems. *Electronic Notes in Theoretical Computer Science*, 223:87–102, 2008. DOI: 10.1016/j.entcs.2008.12.033.

[7] Coogan, S. Mixed monotonicity for reachability and safety in dynamical systems. In *Proceedings of the 59th IEEE Conference on Decision and Control*, pages 5074–5085. IEEE, 2020. DOI: 10.1109/CDC42340.2020.9304391.

[8] Cyranka, J., Islam, A., Byrne, G., Jones, P., Smolka, S., and Grosu, R. Lagrangian reachabilty. In *Proceedings of the 29th International Conference on Computer Aided Verification, Part I*, Volume 10426 of *Lecture Notes in Computer Science*, pages 379–400. Springer, 2017. DOI: 10.1007/978-3-319-63387-9_19.

[9] Frehse, G. PHAVer: Algorithmic verification of hybrid systems. *International Journal on Software Tools for Technology Transfer*, 10(3):23–48, 2008. DOI: 10.1007/978-3-540-31954-2_17.

[10] Geretti, L., dit Sandretto, J. A., Althoff, M., Benet, L., Chapoutot, A., Chen, X., Collins, P., Forets, M., D.Freire, Immler, F., Kochdumper, N., Sanders, D., and Schilling, C. ARCH-COMP20 category report: Continuous and hybrid systems with nonlinear dynamics. In *Proceedings of the 7th International Workshop on Applied Verification of Continuous and Hybrid Systems*, pages 49–21, Berlin, Germany, 2020. DOI: 10.29007/zkf6.

[11] Goubault, E., Mullier, O., Putot, S., and Kieffer, M. Inner approximated reachability analysis. In *Proceedings of the 17th International Conference on Hybrid Systems: Computation and Control*, pages 163–172. ACM, 2014. DOI: 10.1145/2562059.2562113.

[12] Gouzé, J. L., Rapaport, A., and Hadj-Sadok, M. Z. Interval observers for uncertain biological systems. *Ecological Modelling*, 133(1):45–56, 2000. DOI: 10.1016/S0304-3800(00)00279-9.

[13] Jaulin, L. Asymptotically minimal interval contractors based on the centered form. *Acta Cybernetica*, 26(4):933–954, 2024. DOI: 10.14232/actacyb.306222.

[14] Kapela, T., Mrozek, M., Wilczak, D., and Zgliczynski, P. CAPD::DynSys: A flexible C++ toolbox for rigorous numerical analysis of dynamical systems. *Communications in Nonlinear Science and Numerical Simulation*, 101:105578, 2021. DOI: 10.1016/j.cnsns.2020.105578.

[15] Kieffer, M. and Walter, E. Guaranteed nonlinear state estimation for continuous-time dynamical models from discrete-time measurements. In *Proceedings of the 17th IFAC World Congress*, Seoul, Republic of Korea, 2006. DOI: 10.3182/20060709-3-KR-2910.00020.

[16] Kurzhanski, A. and Varaiya, P. Ellipsoidal techniques for reachability under state constraints. *SIAM Journal on Control and Optimization*, 45(4):1369–1394, 2006. DOI: 10.1137/S0363012903437605.

[17] Le Bars, F., Sliwka, J., Jaulin, L., and Reynet, O. Set-membership state estimation with fleeting data. *Automatica*, 48(2):381–387, 2012. DOI: 10.1016/j.automatica.2011.11.004.

[18] Le Guernic, C. and Girard, A. Reachability analysis of linear systems using support functions. *Nonlinear Analysis: Hybrid Systems*, 4(2):250–262, 2010. DOI: 10.1016/j.nahs.2009.03.003.

[19] Moore, R. *Methods and Applications of Interval Analysis*. Society for Industrial and Applied Mathematics, 1979. DOI: 10.1137/1.9781611970906.

[20] Müller, M. Über das Fundamentaltheorem in der Theorie der gewöhnlichen Differentialgleichungen. *Mathematische Zeitschrift*, 26:619–645, 1927. DOI: 10.1007/BF01475477.

[21] Nedialkov, N. S., Jackson, K. R., and Corliss, G. F. Validated solutions of initial value problems for ordinary differential equations. *Applied Mathematics and Computation*, 105(1):21–68, 1999. DOI: 10.1016/S0096-3003(99)00004-8.

[22] Raissi, T., Efimov, D., and Zolghadri, A. Interval state estimation for a class of nonlinear systems. *IEEE Transactions on Automatic Control*, 57(1):260–265, 2012. DOI: 10.1109/TAC.2011.2164730.

[23] Rauh, A., Westphal, R., and Aschemann, H. Verified simulation of control systems with interval parameters using an exponential state enclosure technique. In *Proceedings of the 18th International Conference on Methods and Models in Automation and Robotics*, pages 241–246, 2013. DOI: 10.1109/MMAR.2013.6669913.

[24] Rohou, S., Desrochers, B., and Le Bars, F. The codac library. *Acta Cybernetica*, 26(4):871–887, 2024. DOI: 10.14232/actacyb.302772.

[25] Rohou, S., Jaulin, L., Mihaylova, L., Le Bars, F., and Veres, S. *Reliable Robot Localization*. Wiley, 2019. DOI: 10.1002/9781119680970.

[26] Scott, J. K. and Barton, P. I. Bounds on the reachable sets of nonlinear control systems. *Automatica*, 49(1):93–100, 2013. DOI: 10.1016/j.automatica.2012.09.020.

[27] Taha, W. et al. Acumen: An open-source testbed for cyber-physical systems research. In *Proceedings of the Conference on CYber physiCaL systems, iOt and sensors Networks*, Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, pages 118–130, Cham, 2015. Springer International Publishing. DOI: 10.1007/978-3-319-47063-4_11.

[28] Wan, J. *Computationally reliable approaches of contractive model predictive control for discrete-time systems*. PhD dissertation, Universitat de Girona, Girona, Spain, 2007. URL: https://www.tdx.cat/handle/10803/7740#page=1.

[29] Xue, B., Mosaad, P. N., Franzle, M., Chen, M., Li, Y., and Zhan, N. Safe over- and under-approximation of reachable sets for delay differential equations. In *Proceedings of the 15th International Conference on Formal Modeling and Analysis of Timed Systems*, Volume 10419 of *Lecture Notes in Computer Science*, pages 281–299. Springer, 2017. DOI: 10.1007/978-3-319-65765-3_16.