Integral algebra for simulating dynamical systems with interval uncertainties

Luc Jaulin

Abstract

This paper presents an integral algebra and shows how it can be used to simulate a dynamical system with interval uncertainties. These uncertainties, can be either on the initial state vector, on the time-dependent inputs, or on the evolution function. Compared to other techniques used for the guaranteed integration of differential inclusion, the presented approach does not require the use of a fixed-point Picard operator. Two test-cases related to robotics are presented to illustrate the efficiency of the approach.

Index Terms

Differential inclusion, Integral algebra, Interval analysis, Interval integration, Reachability

I. Introduction

When dealing with non-linear dynamical systems such as mobile robots, we often need to guarantee that some properties are satisfied [14], mainly for security reasons. Different softwares or libraries have been designed for this purpose, see *e.g.,* Acumen [41], PHAVer [18], Ariadne [5], Codac [36], DynIbex [40], etc. A typical problem that is considered is the reachability analysis which asks for a proof that the system will never enter inside a forbidden region or that it will reach a target set. The guarantee can be obtained by using set computation [3, 21, 12, 11, 7, 2], invariant based approaches [19] or guaranteed integration methods [29, 34, 8, 33]. The goal of guaranteed integration is to find a tube envelope [26] enclosing all feasible trajectories of a dynamical system, assuming that the initial vector is known [23, 6]. It has been used to prove conjectures such as the existence of the Lorenz attractor [43], or that a given system is chaotic [20]. Interval integration methods have also been used for state estimation [25, 32, 1], localization [37, 22, 15, 44, 13] or SLAM [31].

The main default of guaranteed integration methods is that they are very sensitive to uncertainties. In the context of a badly known initial vector or when bounded errors exist in the evolution equation, the tube enclosing the trajectory is so large that no conclusion can be drawn. For engineering applications, it is fundamental to develop fast methods for interval simulation that are not too conservative, even if the class of systems to be considered is limited.

In this paper, we introduce an *integral algebra* and combine it with interval-based methods [24][30], in order to propagate uncertainties through differential inclusions [23]. More precisely, our goal is to compute an enclosure of the solution of a differential inclusion assuming that the initial state is inside a box that may be large and the input is inside an interval tube. Since the propagation of the uncertainties can only be causal (*i.e.*, forward in time), no constraint propagation techniques [9][42][10] could help for contraction.

Integral algebra is similar to differential algebra [35], used in control theory [17]. Now, whereas differential algebra introduces the time derivative as an operator, integral algebra introduces the time integral. From a control point of view, this integral operation seems useless and could be considered as limited compared to the derivative operator which benefits from systematic rules that can be applied to any expression. Now, we will see that the integral algebra will be convenient for interval prediction. Compared to other techniques used for the guaranteed integration of differential inclusions, the presented approach is fast, does not require the use of a fixed-point Picard operator and can deal with large uncertainties.

The paper is structured as follows. Section II provides basic notions on integral algebra and shows how it can be used for the prediction of a dynamical system with uncertainties as soon as they have an integral representation. Section III shows that linear systems have an integral representation. Two section are then devoted to applications. Section IV proposes to deal with the prediction of a car with a trailer. Section V presents the simulation of an hovercraft. For these test-cases all predictions are done with large interval uncertainties and I do not know any interval solver which would be able to get a non trivial enclosure for these test-cases. Section VI concludes the paper and proposes some perspectives.

II. Formalism

A. Integral algebra

In this section we introduce integral algebra. The presentation that we propose is voluntary similar to that of the differential algebra made in [35], or more recently in [17]. The main difference is that we use integrators instead of derivatives.

Definition 1. A *real integral ring* is a ring $(\mathcal{R},+,\cdot)$ equipped with the single integration $\int$ (with respect to a single variable denoted by t) such that

$$\begin{array}{ll} \text{(i)} & \mathbb{R}\subset\mathcal{R} \\ \text{(ii)} & \forall a\in\mathcal{R},\int a\in\mathcal{R} \end{array} \tag{1}$$

where $\mathbb{R}$ is the set of real numbers. The meaning of $\int$ is the primitive which cancels for $t=0$, *i.e.*,

$$\int a=\int_0^t a(\tau)d\tau. \tag{2}$$

Moreover, $\int$ satisfies the classical integral rules. For instance

$$\begin{array}{rcl} \int(a+b) & = & \int a+\int b \\ \int a\cdot\int b & = & \int(a\cdot\int b+\int a\cdot b) \end{array} \tag{3}$$

The second equation is a direct consequence of the Leibniz rule for derivatives. Indeed, since $\int$ cancels for $t=0$, we have

$$\begin{array}{rcl} \int a\cdot\int b & = & \int\left(\frac{d}{dt}\left(\int a\cdot\int b\right)\right) \\ & = & \int\left(a\cdot\int b+\int a\cdot b\right) \end{array} \tag{4}$$

An element of $\mathcal{R}$ is called *a signal*. A signal which is in $\mathbb{R}\subset\mathcal{R}$ is called a constant.

Example 2. Consider $\mathcal{R}_0$ the smallest real integral ring. We have

$$\begin{array}{ll} a=2\in\mathcal{R}_0 & \text{it is a constant} \\ b=2t\in\mathcal{R}_0 & \text{since }\; b=\int a \end{array} \tag{5}$$

Note that $\mathcal{R}_0$ contains polynomial in t and does not include periodic functions such as cos or sin. Even if $\cos(t)=\sum_{n=0}^{\infty}(-1)^n\frac{t^{2n}}{(2n)!}$, we cannot conclude that $\cos(t)$ belongs to $\mathcal{R}_0$. Indeed, in our definition of $\mathcal{R}_0$, only a finite number of additions is allowed.

An *integral ring extension* is $\mathcal{L}/\mathcal{R}$ is given by two integral rings $\mathcal{R},\mathcal{L}$ with $\mathcal{R}\subset\mathcal{L}$ such that the restriction of the operation of $\mathcal{L}$ to $\mathcal{R}$ coincides with the operations in $\mathcal{R}$.

A signal u of $\mathcal{L}$ is said to be *integral $\mathcal{R}$-algebraic independent* if all signals

$$u,\int u,\int^2 u,\int^3 u,\dots \tag{6}$$

are independent. Note that here, the multiple integral is denoted with an exponent, for instance $\int\int u=\int^2 u$.

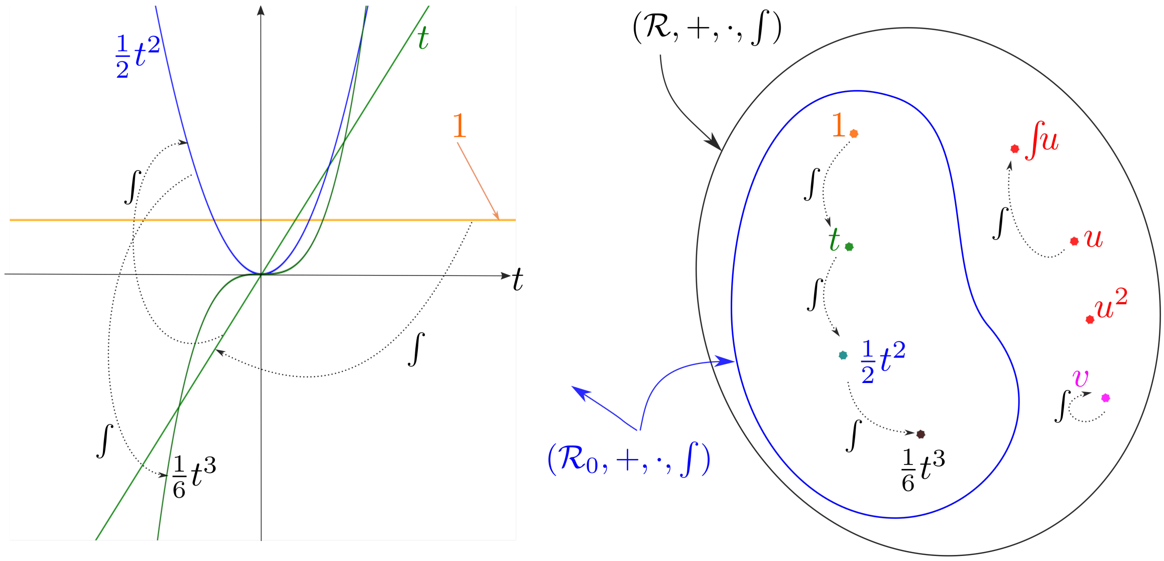

Fig. 1. Integral algebra $(\mathcal{R},+,\cdot,\int)$ contains signals that represent functions of t

As illustrated by Figure 1 the set $\mathcal{R}_0$ represents signals of $\mathcal{R}$ which are polynomial in t. The signal u is seen as an independent indeterminate which corresponds to some free signals which can be chosen arbitrarily. Similarly, $v \notin \mathcal{R}_0$, but since $v = \int v$, the signal v is integral $\mathcal{R}$-algebraic dependent.

Example 3. Assume that $\mathcal{L}_1 = \mathcal{R}_0 < u >$ is the smallest real integral ring which contains u, where u is integral $\mathcal{R}_0$-algebraic independent in $\mathcal{L}_1$. Assume that $\mathcal{L}_2$ is the smallest real integral ring which contains u where u satisfies $u = \int u$. We have $\mathcal{L}_2 \subset \mathcal{L}_1$. Indeed, since in $\mathcal{L}_2$, we have $u = \int u = \int^2 u, \dots$, the ring $\mathcal{L}_2$ can be generated by sums and multiplications of elements by $\mathcal{R}_0$ and u only. Whereas in $\mathcal{L}_1$, since $u, \int u, \int^2 u, \dots$ are independent, we need all integrals of u to build $\mathcal{L}_1$.

B. Enriched integral algebra

As presented previously, the integral algebra $(\mathcal{R},+,\cdot,\int)$ only allows a finite number of sums and multiplications. This limits the number of applications that could be treated. For instance, if u is a signal of $\mathcal{R}$, even if

$$\sin u = u - \frac{u^3}{3!} + \frac{u^5}{5!} - \frac{u^7}{7!} + \cdots, \tag{7}$$

we do not have $\sin u \in \mathcal{R}$. Now, the use of trigonometric functions is unavoidable as soon as we deal with robotics applications. Moreover using such trigonometric functions is not a problem for interval based methods. This is why we want to include them in the formalism. Extending our integral algebra to infinite sums is not easy and requires the introduction of a norm and some notions of topology which do not fit well with an algebraic world. A simple way to extend $(\mathcal{R},+,\cdot,\int)$ is to add a finite family $\mathcal{F}$ of real valued functions $f(x)$ that are defined for all $x \in \mathbb{R}$. We call the corresponding structure an *enriched integral algebra*. In what follows, we will always choose $\mathcal{F} = \{\exp, \sin, \cos\}$ but other functions could be added as soon as an interval counterpart exists. We will indicate that we use the enriched interval algebra using a bar decoration. For instance, we will write $(\overline{\mathcal{R}},+,\cdot,\int,\mathcal{F})$ or simply $\overline{\mathcal{R}}$ to denote the enriched version of the integral algebra $(\mathcal{R},+,\cdot,\int)$.

Example 4. If $(\overline{\mathcal{R}}, +, \cdot, \int, \mathcal{F})$ is an enriched integral algebra, we have

$$\begin{array}{rcl} \cos t & \in & \overline{\mathcal{R}} \\ \int \sin t^2 + 3 & \in & \overline{\mathcal{R}} \end{array} \tag{8}$$

Moreover if $u \in \overline{\mathcal{R}}$, we have

$$\begin{array}{rcl} \cos u & \in & \overline{\mathcal{R}} \\ \exp\left(\int\left(\sin \int^3 u\right)\right) & \in & \overline{\mathcal{R}} \end{array} \tag{9}$$

C. Integral dynamical system

Given an enriched integral algebra $\overline{\mathcal{R}}$. We denote by $\overline{\mathcal{R}} < u_1, u_2, \cdots >$, the enriched integral algebra generated by $\overline{\mathcal{R}}$ and by a finite set $\{u_1, u_2, \dots\}$ of indeterminates that are integral $\overline{\mathcal{R}}$-algebraic independent.

Example 5. Consider the enriched integral algebra $\overline{\mathcal{L}} = \overline{\mathcal{R}_0} < u >$ with $\mathcal{F} = \{\exp, \sin, \cos\}$. We have

$$\begin{array}{rcl} \cos t & \in & \overline{\mathcal{L}} \\ u + \int \sin u + 3 & \in & \overline{\mathcal{L}} \\ u + \int\left(\sin \int^3 u\right) + 3 & \in & \overline{\mathcal{L}} \end{array} \tag{10}$$

On the other hand,

$$\begin{array}{rcl} u(t-1) & \notin & \overline{\mathcal{L}} \\ \int_0^t w(t-\tau) \cdot u(\tau) d\tau & \notin & \overline{\mathcal{L}} \end{array} \tag{11}$$

where $w(t)$ is any given signal in $\overline{\mathcal{R}_0}$. This means that the delay or the convolution (in general) cannot be considered as a closed operations in $\overline{\mathcal{R}_0} < u >$. The main reason for this is that an infinite number of integral operators should be used to generate a delay or a convolution. This is indeed a consequence of the Fréchet's approximation theorem which states that a Volterra serie can be obtained for any time-invariant linear system. Note that for many convolutions, the kernel $w(t-\tau)$ is separable and in such a case, the convoluted signal may belong to $\overline{\mathcal{L}}$. For instance, if $w(t-\tau) = \sin(t-\tau)$, we have

$$\begin{array}{rcl} \int_0^t w(t-\tau) \cdot u(\tau) d\tau & = & \int_0^t \sin(t-\tau) \cdot u(\tau) d\tau \\ & = & \int_0^t (\sin t \cos\tau - \cos t \sin\tau) \cdot u(\tau) d\tau \\ & = & \sin t \left(\int_0^t \cos\tau \cdot u(\tau) d\tau\right) - \cos t \left(\int \sin\tau \cdot u(\tau) d\tau\right), \end{array} \tag{12}$$

which clearly belongs to $\overline{\mathcal{L}}$.

Definition 6. An *integral dynamical system* is defined as a finite subset $\{x_1, \dots, x_n\}$ of $\overline{\mathcal{R}_0} < u_1, \dots, u_m >$. The quantities $x_1, x_2, \dots$ are called the *state variables* and $u_1, u_2, \dots$ are called the inputs. In this definition, the variables $x_1, x_2, \dots$ do not need to be independent. This independency condition, which should be added for control purpose, will not be useful here.

Remark 7. Since $x_i \in \overline{\mathcal{R}_0} < u_1, \dots, u_m >$ for all i, there exists an expression which allows us to generate all state variables from the $u_j's$ using operators in $\{+, \cdot, \int, \sin, \cos, \exp\}$. This expression will be called an *integral representation*. It can be given under the form of a mathematical expression, an algorithm or a flow graph.

Consider a system of the form

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \\ \mathbf{x}(0) = \mathbf{x}_0 \end{cases} \tag{13}$$

Equivalently, (13) can be rewritten as

$$\mathbf{x}(t) = \mathbf{x}_0 + \int_{\tau=0}^{t} \mathbf{f}(\mathbf{x}(\tau), \mathbf{u}(\tau))d\tau. \tag{14}$$

The system (13) is an *integral dynamical system* if for all $i \in \{1, \dots, n\}$, $x_i \in \overline{\mathcal{R}_0} < u_1, \dots, u_m >$. It means that each x_i can be finitely generated from the set $\mathbb{R} \cup \{u_1, \dots, u_m\}$ by the operators $\{+, \cdot, \int, \sin, \cos, \exp\}$.

D. Interval extension of an integral dynamical system

A *tube* is a function which associates to any $t \in \mathbb{R}$ a subset of $\mathbb{R}^n$. In the case where these subsets are intervals or boxes, a tube can be represented in the computer by stepwise functions (see [4]) as illustrated in Figure 2.

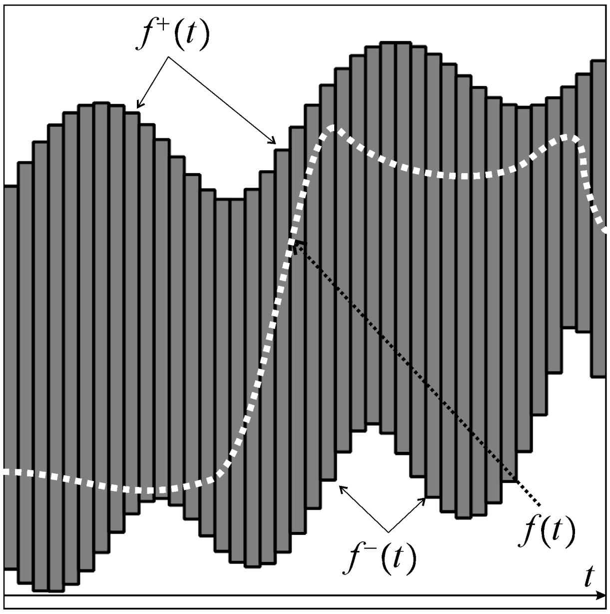

Fig. 2. In numerical computations, a tube $[f](t)$ can be approximated by a lower and an upper stepwise functions $f^-(t)$ and $f^+(t)$. The tube $[f](t)$ encloses an uncertain trajectory $f(t)$

It has been proved [30] for the integration, for the composition, and other operations (such as $+, -, /, \cdot$) that the fundamental inclusion property is satisfied. More precisely, for the integration, this inclusion property is

$$f(\cdot) \in [f](\cdot) \;\Rightarrow\; \forall t, \int_0^t f(\tau)\, d\tau \in \int_0^t [f](\tau)\, d\tau. \tag{15}$$

Consider an integral dynamical system $\{x_1 \dots, x_n\} \in \overline{\mathcal{R}_0} < u_1, \dots, u_m >$. For each x_i we can build an expression that involves the initial state variables $x_1(0), \dots, x_n(0)$ and the inputs $u_1, \dots, u_m$. An interval evaluation can be performed using the classical rules of interval arithmetic [30] and interval tube arithmetic [4][37]. Therefore, using the fundamental theorem of interval analysis, we get an interval extension which provides a guaranteed enclosure for the trajectories associated to each x_i. We do not give more details here, since the method to compute the tubes is well explained in the literature and is not needed to understand the contributions of the paper. Moreover, the principle of the approach will be illustrated through several test-cases in the following sections.

III. LINEAR SYSTEMS

A. Linear systems are integral dynamical systems

Proposition 8. *The system*

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{u} \quad (16)$$

is an integral dynamical system. An integral representation (see Figure 3) is

$$\mathbf{x}(t) \;=\; e^{\mathbf{A}t} \cdot \left(\mathbf{x}_0 + \int_0^t e^{-\mathbf{A}\tau}\mathbf{u}(\tau)d\tau\right) \quad (17)$$

Proof: The property is a direct consequence of the formula

$$\mathbf{x}(t) \;=\; e^{\mathbf{A}t} \cdot \mathbf{x}_0 + \int_0^t e^{\mathbf{A}\cdot(t-\tau)}\mathbf{u}(\tau)d\tau. \quad (18)$$

■

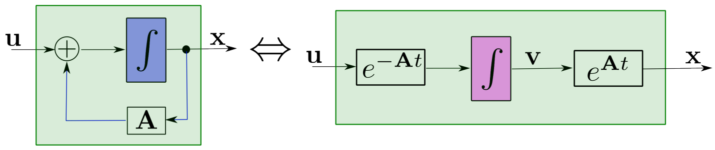

Fig. 3. Initial representation of a linear system and its integral representation

B. A two-dimensional example

Consider the system

$$\dot{\mathbf{x}} = \underbrace{\begin{pmatrix} \lambda & -\omega \\ \omega & \lambda \end{pmatrix}}_{\mathbf{A}} \mathbf{x} + \mathbf{u} \quad (19)$$

We have

$$e^{\mathbf{A}t} = e^{\lambda t} \cdot \begin{pmatrix} \cos\omega t & -\sin\omega t \\ \sin\omega t & \cos\omega t \end{pmatrix} \quad (20)$$

From (17), we get the following integral representation for (19):

$$\left\{ \begin{array}{lll} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} & = & e^{\lambda t} \cdot \begin{pmatrix} \cos\omega t & -\sin\omega t \\ \sin\omega t & \cos\omega t \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \\ \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} & = & \begin{pmatrix} x_1(0) \\ x_2(0) \end{pmatrix} + \int e^{-\lambda\tau} \cdot \begin{pmatrix} \cos\omega\tau & \sin\omega\tau \\ -\sin\omega\tau & \cos\omega\tau \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \end{array} \right. \quad (21)$$

As illustrated by Figure 3, the loop in the flow graph (blue) of the initial system does not exist anymore in the integral representation. This means x_1, x_2 are signals of $\overline{\mathcal{R}_0} < u_1, u_2 >$.

C. Illustration

If we take

$$\mathbf{u}(t) = \begin{pmatrix} u_1(t) \\ u_2(t) \end{pmatrix} \in \begin{pmatrix} [u_1](t) \\ [u_2](t) \end{pmatrix} = \begin{pmatrix} [-0.02, 0.02] \\ [-0.02, 0.02] \end{pmatrix} \quad (22)$$

for $t \in [0, 10]$, $\lambda = -0.1$ and $\omega = 2$, we get Figure 4 and 5 for two different initial boxes. In the (x_1, x_2)-space the initial state box is in red. The interval trajectories is obtained using the following interval computation

Input:			$[x_1](0), [x_2](0), [u_1](t), [u_2](t)$
Step 1	$\begin{pmatrix} [v_1](t) \\ [v_2](t) \end{pmatrix}$	$=$	$\begin{pmatrix} [x_1](0) \\ [x_2](0) \end{pmatrix} + \int_0^t e^{-\lambda\tau} \cdot \begin{pmatrix} \cos\omega\tau & \sin\omega\tau \\ -\sin\omega\tau & \cos\omega\tau \end{pmatrix} \begin{pmatrix} [u_1](\tau) \\ [u_2](\tau) \end{pmatrix}$
Step 2	$\begin{pmatrix} [x_1](t) \\ [x_2](t) \end{pmatrix}$	$=$	$e^{\lambda t} \cdot \begin{pmatrix} \cos\omega t & -\sin\omega t \\ \sin\omega t & \cos\omega t \end{pmatrix} \begin{pmatrix} [v_1](t) \\ [v_2](t) \end{pmatrix}$

The sampling time is taken as $dt = 0.01$. The computation time on a classical laptop is less than 0.04sec. In Figure 4, we observe that even with a small damping coefficient (here $\lambda = 0.1$), the approximation do not diverge, which is difficult to obtain with classical fixed point interval methods. In Figure 5, we have the initial red box which becomes the blue box for $t = 8$. This shows that the red box is a periodic positive invariant set [27]. Equivalently, all trajectories starting in the red box will enter the blue box for $t = 8$ and never leave the box painted gray for all $t \geq 0$. This property is not easy to prove with classical interval methods.

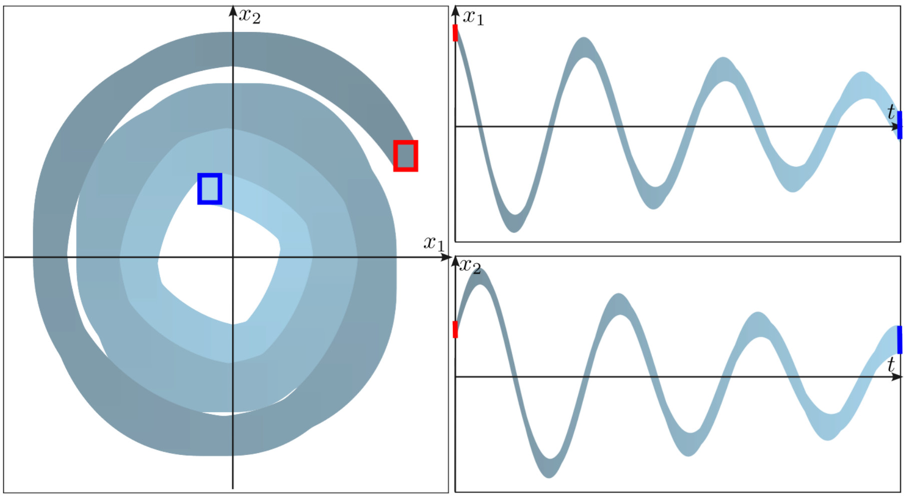

Fig. 4. Integral simulation of the linear system for $[\mathbf{x}](0) = [1.9, 2.1] \times [0.9, 1.1]$

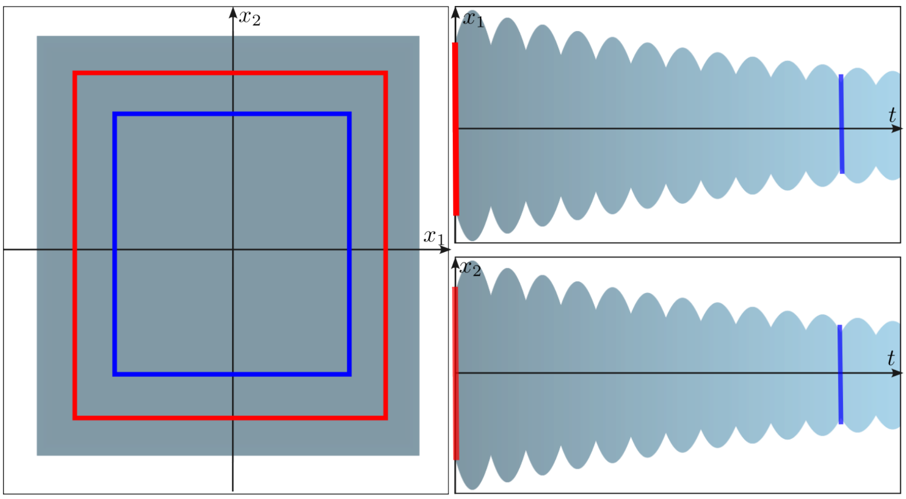

Fig. 5. Integral simulation of the linear system for $[\mathbf{x}](0) = [-2, 2] \times [-2, 2]$

IV. Car Trailer

A. Model

We consider the car trailer [38] of Figure 6. The pose of the front body corresponds to the vector (x_1, x_2, x_3). The heading of the second body is x_4. The speed of the front body is denoted by x_5.

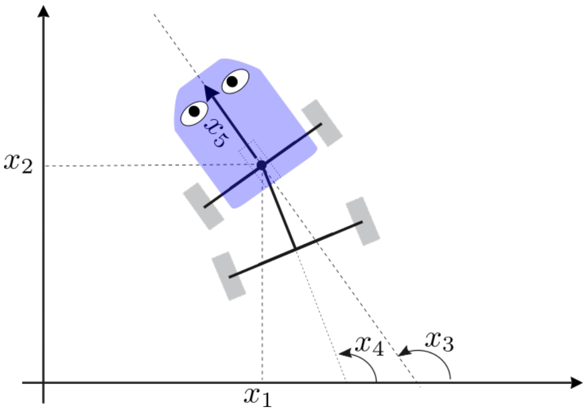

Fig. 6. The car-trailer has 5 state variables. It can change its heading and its speed via the two inputs u_1, u_2

The evolution is described by the state equations:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \end{pmatrix} = \begin{pmatrix} x_5 \cos x_3 \\ x_5 \sin x_3 \\ u_1 + x_5 \sin(x_3 - x_4) \\ x_5 \sin(x_3 - x_4) \\ u_2 \end{pmatrix} \tag{23}$$

As shown in Figure 7, there exists two loops and thus we cannot conclude that x_1, x_2, x_3, x_4 belong to $\mathcal{R}_0 < u_1, u_2 >$.

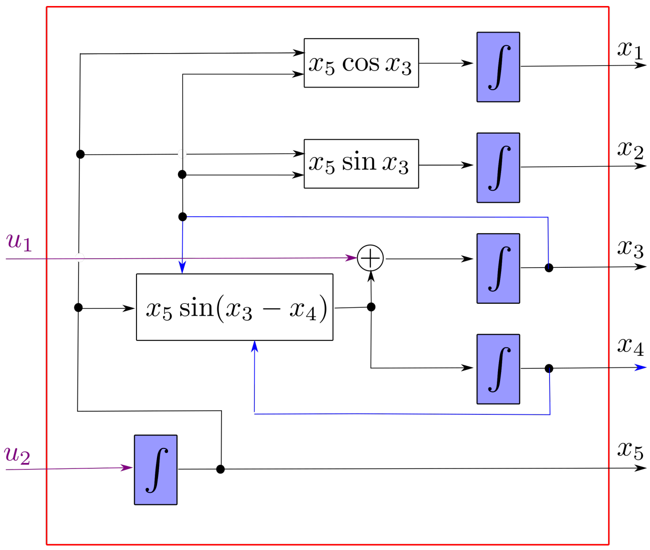

Fig. 7. Initial car-trailer flow graph

B. *Integral representation*

Proposition 9. *The car-trailer is an integral dynamical system. An integral representation of the car-trailer is*

$$\left\{\begin{array}{ccc} x_1 & = & x_1(0)+\int(x_5\cos x_3) \\ x_2 & = & x_2(0)+\int(x_5\sin x_3) \\ x_3 & = & x_4+v_1 \\ x_4 & = & x_4(0)+\int(x_5\sin v_1) \\ v_1 & = & v_1(0)+\int u_1 \\ x_5 & = & x_5(0)+\int u_2 \end{array}\right. \tag{24}$$

with

$$v_1(0)=x_3(0)-x_4(0). \tag{25}$$

This is illustrated by Figure 8. The four blue integrators correspond to the original state variable x_1, x_2, x_4, x_5. Now, the magenta integrator is an intermediate variable v_1 that have to be initialized following (25). Note that we do not have no more loops.

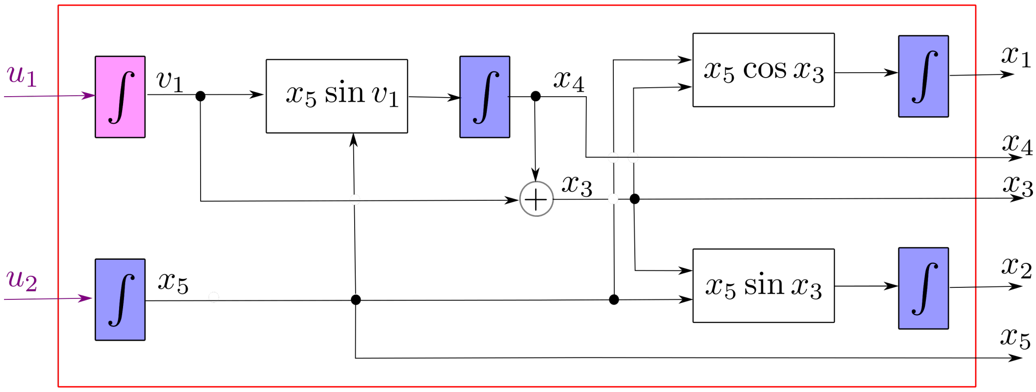

Fig. 8. Integral representation of the car-trailer

Proof: Since

$$\left\{\begin{pmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\\ \dot{x}_4\\ \dot{x}_5\end{pmatrix}=\begin{pmatrix} x_5\cos x_3\\ x_5\sin x_3\\ u_1+x_5\sin(x_3-x_4)\\ x_5\sin(x_3-x_4)\\ u_2\end{pmatrix}\right. \tag{26}$$

The two first equations of (24) are trivial. Set $v_1=x_3-x_4$. We get

$$\begin{pmatrix}\dot{v}_1\\ \dot{x}_4\\ \dot{x}_5\end{pmatrix}=\begin{pmatrix}\dot{x}_4-u_1+x_5\sin v_1\\ x_5\sin v_1\\ u_2\end{pmatrix} \tag{27}$$

i.e.,

$$\begin{pmatrix}\dot{v}_1\\ \dot{x}_4\\ \dot{x}_5\end{pmatrix}=\begin{pmatrix}u_1\\ x_5\sin v_1\\ u_2\end{pmatrix} \tag{28}$$

As a consequence

$$\begin{array}{rcl} x_3 & = & x_4 + v_1 \\ x_4 & = & x_4(0) + \int (x_5 \sin v_1) \\ v_1 & = & x_3(0) - x_4(0) + \int u_1 \\ x_5 & = & x_5(0) + \int u_2 \end{array} \quad (29)$$

To show that it corresponds indeed to an integral representation, we build the chain

$$\begin{array}{rcl} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} & \in & \overline{\mathcal{R}_0} < x_3, x_5 > \\ x_3 & \in & \overline{\mathcal{R}_0} < v_1, x_4 > \\ x_4 & \in & \overline{\mathcal{R}_0} < v_1, x_5 > \\ v_1 & \in & \overline{\mathcal{R}_0} < u_1 > \\ x_5 & \in & \overline{\mathcal{R}_0} < u_2 > \end{array} \quad (30)$$

By transitivity, we conclude

$$(x_1, x_2, x_3, x_4, x_5) \in \overline{\mathcal{R}_0} < u_1, u_2 > . \quad (31)$$

■

C. Illustration

Take

$$\mathbf{u}(t) = \mathbf{u}^*(t) = \begin{pmatrix} e^{-t} \\ e^{-t} \end{pmatrix} \text{ and } \mathbf{x}(0) = \mathbf{x}^*(0) = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1.5 \\ 1 \end{pmatrix}. \quad (32)$$

A Runge-Kutta integration of the state equations (23) yields the red trajectory of Figure 9 in the (x_1, x_2)-space for $t \in [0, 3]$. The initial robot ($t = 0$) is painted green. The final pose ($t = 3$) is painted yellow. Assume now, we want to enclose all trajectories for

$$\mathbf{u}(t) = \begin{pmatrix} u_1(t) \\ u_2(t) \end{pmatrix} \in \begin{pmatrix} [u_1](t) \\ [u_2](t) \end{pmatrix} = \mathbf{u}^*(t) + \begin{pmatrix} [-0.01, 0.01] \\ [-0.01, 0.01] \end{pmatrix} \quad (33)$$

and

$$\mathbf{x}(0) \in [\mathbf{x}](0) = \mathbf{x}^*(0) + \begin{pmatrix} [-0.001, 0.001] \\ [-0.001, 0.001] \\ [-0.2, 0.2] \\ [-0.01, 0.01] \\ [-0.001, 000.1] \end{pmatrix}. \quad (34)$$

The interval trajectory in the (x_1, x_2)-space is obtained using the following interval computation

$$\begin{array}{lccl} \text{Input :} & & & [x_1](0), [x_2](0), [x_3](0), [x_4](0), [x_5](0), [u_1](t), [u_2](t) \\ \text{Step 1} & [v_1](0) & = & [x_3](0) - [x_4](0). \\ \text{Step 2} & [v_1](t) & = & [v_1](0) + \int_0^t [u_1](\tau) d\tau \\ \text{Step 3} & [x_5](t) & = & [x_5](0) + \int_0^t [u_2](\tau) d\tau \\ \text{Step 4} & [x_4](t) & = & [x_4](0) + \int_0^t [x_5](\tau) \cdot \cos([v_1](\tau)) \cdot d\tau \\ \text{Step 5} & [x_3](t) & = & [x_4](t) + [v_1](t) \\ \text{Step 6} & \begin{pmatrix} [x_1](t) \\ [x_2](t) \end{pmatrix} & = & \begin{pmatrix} [x_1](0) \\ [x_2](0) \end{pmatrix} + \begin{pmatrix} \int_0^t [x_5](\tau) \cdot \cos([x_3](\tau)) \cdot d\tau \\ \int_0^t [x_5](\tau) \cdot \sin([x_3](\tau)) \cdot d\tau \end{pmatrix} \end{array} \quad (35)$$

We are able to enclose the corresponding interval trajectory with sampling time $dt = 0.01$. For a more accurate approximation, we bisected the initial box $[\mathbf{x}](0)$ with 80 subboxes. The superposition of all interval simulations in the (x_1, x_2) space is given by Figure 9. The computation time on a classical laptop is less than 1sec.

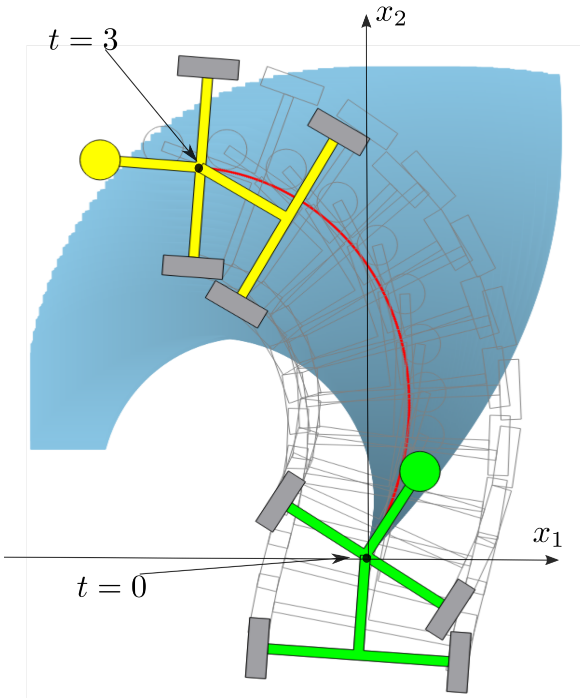

Fig. 9. Integral simulation of the car-trailer. The frame box is $[-5, 5] \times [-3, 6]$

V. HOVERCRAFT

A. Model

Consider the hovercraft, as shown on Figure 10. The state variables are the positions (x_1, x_2), the heading ψ, the speed variables (v_1, v_2) and the rotation rate ω.

The state equations are given by [16]

$$\left\{ \begin{array}{lcl} \dot{x}_1 & = & v_1 \cos \psi - v_2 \sin \psi \\ \dot{x}_2 & = & v_1 \sin \psi + v_2 \cos \psi \\ \dot{v}_1 & = & u_1 + \omega v_2 \\ \dot{v}_2 & = & -\omega v_1 \\ \dot{\psi} & = & \omega \\ \dot{\omega} & = & u_2 \end{array} \right. \tag{36}$$

As shown in Figure 11, the hovercraft has some loops (blue). Now, a decomposition into three blocks shows up a block in the middle with loops.

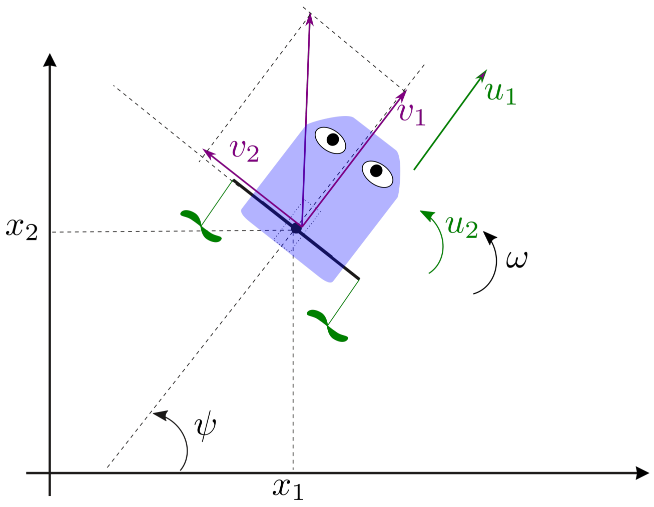

Fig. 10. The hovercraft has two propellers and can glide in all directions without any friction

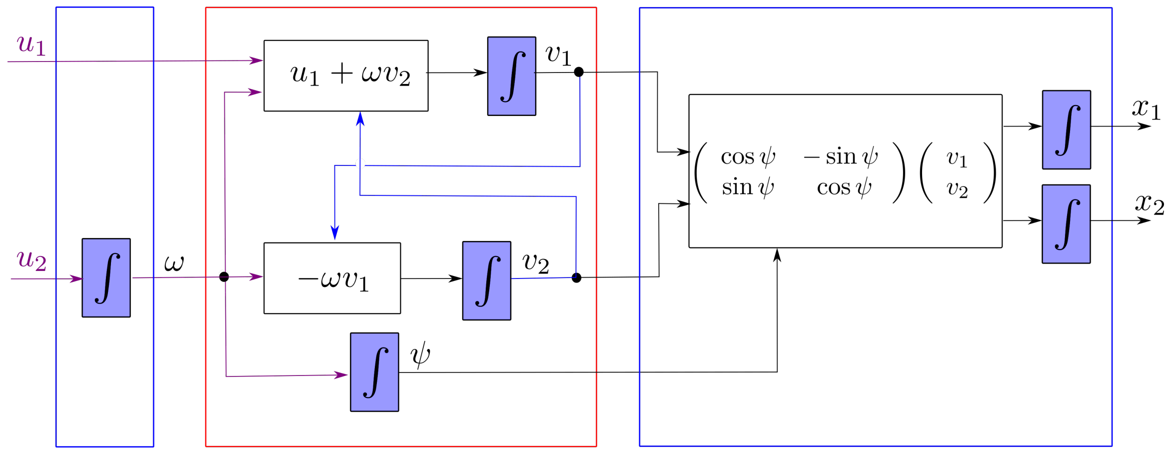

Fig. 11. Initial representation of the hovercraft

B. Integral representation

Proposition 10. *The hovercraft is an integral dynamical system. An integral representation is*

$$\left\{ \begin{array}{ccc} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} & = & \begin{pmatrix} x_1(0) \\ x_2(0) \end{pmatrix} + \begin{pmatrix} \int (\cos\psi \cdot v_1 - \sin\psi \cdot v_2) \\ \int (\sin\psi \cdot v_1 + \cos\psi \cdot v_2) \end{pmatrix} \\ \begin{pmatrix} v_1(t) \\ v_2(t) \end{pmatrix} & = & \begin{pmatrix} \cos\psi & \sin\psi \\ -\sin\psi & \cos\psi \end{pmatrix} \left(\begin{pmatrix} a_1(0) \\ a_2(0) \end{pmatrix} + \begin{pmatrix} \int (u_1 \cos\psi) \\ \int (u_1 \sin\psi) \end{pmatrix} \right) \\ \psi(t) & = & \psi(0) + \int \omega \\ \omega(t) & = & \omega(0) + \int u_2 \end{array} \right. \tag{37}$$

where

$$\begin{pmatrix} a_1(0) \\ a_2(0) \end{pmatrix} = \begin{pmatrix} \cos\psi(0) & -\sin\psi(0) \\ \sin\psi(0) & \cos\psi(0) \end{pmatrix} \begin{pmatrix} v_1(0) \\ v_2(0) \end{pmatrix}. \tag{38}$$

This is illustrated by Figure 12. The initial values are not represented. The integrators (blue) associated to x_1, x_2, ψ, ω should be initialized to $x_1(0), x_2(0), \psi(0), \omega(0)$, respectively. A special care should be taken for the integrator in magenta that are associated to the intermediate variables a_1, a_2 and not to the state variables v_1, v_2. The initialization should be performed as given by (38).

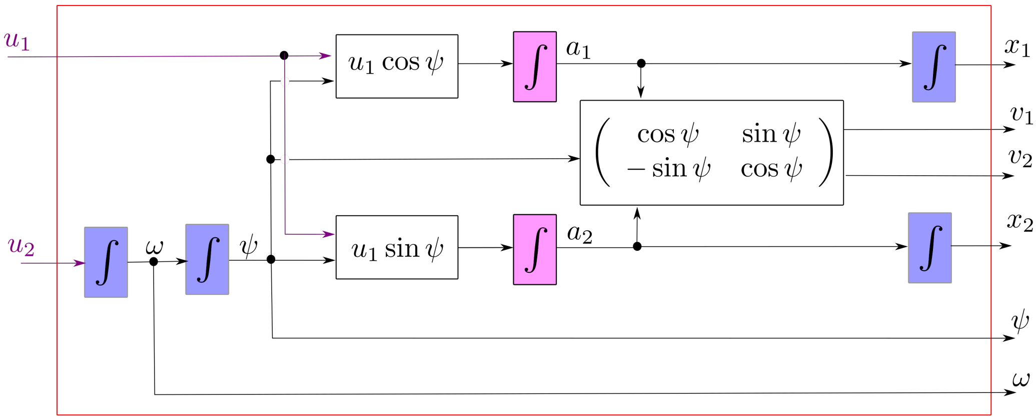

Fig. 12. Integral representation of the hovercraft

Proof: The state equations for our system are

$$\begin{array}{lccl} (i) & \dot{x}_1 & = & v_1 \cos\psi - v_2 \sin\psi \\ (ii) & \dot{x}_2 & = & v_1 \sin\psi + v_2 \cos\psi \\ (iii) & \dot{v}_1 & = & u_1 + \omega v_2 \\ (iv) & \dot{v}_2 & = & -\omega v_1 \\ (v) & \dot{\psi} & = & \omega \\ (vi) & \dot{\omega} & = & u_2 \end{array} \tag{39}$$

The two first equations *(i),(ii)* rewrite into the integral form

$$\left\{ \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_1(0) \\ x_2(0) \end{pmatrix} + \int \begin{pmatrix} \cos\psi & -\sin\psi \\ \sin\psi & \cos\psi \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \right. \tag{40}$$

We can easily check that Equations *(iii),(iv),(v)* (the red block in the figure) correspond to the following integral representation

$$\begin{array}{ccc} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} & = & \begin{pmatrix} \cos\psi & \sin\psi \\ -\sin\psi & \cos\psi \end{pmatrix} \left(\begin{pmatrix} a_1(0) \\ a_2(0) \end{pmatrix} + \begin{pmatrix} \int (u_1 \cos\psi) \\ \int (u_1 \sin\psi) \end{pmatrix} \right) \\ \psi & = & \psi(0) + \int \omega \end{array} \tag{41}$$

The integral form of Equation *(vi)* is

$$\omega = \omega(0) + \int u_2. \tag{42}$$

To show that it corresponds indeed to an integral representation, we build the chain

$$\left\{ \begin{array}{ccc} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} & = & \begin{pmatrix} x_1(0) \\ x_2(0) \end{pmatrix} + \begin{pmatrix} \int (\cos\psi \cdot v_1 - \sin\psi \cdot v_2) \\ \int (\sin\psi \cdot v_1 + \cos\psi \cdot v_2) \end{pmatrix} \\ \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} & = & \begin{pmatrix} \cos\psi & \sin\psi \\ -\sin\psi & \cos\psi \end{pmatrix} \left(\begin{pmatrix} a_1(0) \\ a_2(0) \end{pmatrix} + \begin{pmatrix} \int (u_1 \cos\psi) \\ \int (u_1 \sin\psi) \end{pmatrix} \right) \\ \psi & = & \psi(0) + \int \omega \\ \omega & = & \omega(0) + \int u_2 \end{array} \right. \tag{43}$$

$$\begin{array}{ccc} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} & \in & \overline{\mathcal{R}_0} < \psi, v_1, v_2 > \\ \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} & \in & \overline{\mathcal{R}_0} < \psi, u_1 > \\ \psi & \in & \overline{\mathcal{R}_0} < \omega > \\ \omega & \in & \overline{\mathcal{R}_0} < u_2 > \end{array} \tag{44}$$

By transitivity, we conclude

$$(x_1, x_2, v_1, v_2, \psi, \omega) \in \overline{\mathcal{R}_0} < u_1, u_2 > . \tag{45}$$

■

C. Illustration

Take

$$\mathbf{u}(t) = \mathbf{u}^*(t) = \begin{pmatrix} e^{-t} \\ e^{-t} \end{pmatrix} \text{ and } \mathbf{x}(0) = \mathbf{x}^*(0) = \begin{pmatrix} 0 \\ 0 \\ 2 \\ 0 \\ 1 \\ 0 \end{pmatrix}. \tag{46}$$

A Runge-Kutta integration with the state equations (36) yields the red trajectory of Figure 4 in the (x_1, x_2)-space for $t \in [0, 3]$. The initial hovercraft ($t = 0$) is painted green (with the two gray propellers). The final pose ($t = 3$) is painted yellow. Assume now, we want to enclose all trajectories for

$$\mathbf{u}(t) = \begin{pmatrix} u_1(t) \\ u_2(t) \end{pmatrix} \in \begin{pmatrix} [u_1](t) \\ [u_2](t) \end{pmatrix} = \mathbf{u}^*(t) + \begin{pmatrix} [-0.01, 0.01] \\ [-0.01, 0.01] \end{pmatrix} \tag{47}$$

$$\mathbf{x}(0) \in [\mathbf{x}](0) = \mathbf{x}^*(0) + \begin{pmatrix} [-0.001, 0.001] \\ [-0.001, 0.001] \\ [-0.001, 0.001] \\ [-0.001, 0.001] \\ [-0.2, 0.2] \\ [-0.001, 0.001] \end{pmatrix} \tag{48}$$

The interval trajectory in the (x_1, x_2)-space is obtained using the following interval computation

$$\begin{array}{lccl} \text{Input :} & & & [x_1](0), [x_2](0), [v_1](0), [v_2](0), [\psi](0), [\omega](0), [u_1](t), [u_2](t) \\ \text{Step 1} & \begin{pmatrix} [a_1](0) \\ [a_2](0) \end{pmatrix} & = & \begin{pmatrix} \cos([\psi](0)) & -\sin([\psi](0)) \\ \sin([\psi](0)) & \cos([\psi](0)) \end{pmatrix} \begin{pmatrix} [v_1](0) \\ [v_2](0) \end{pmatrix} \\ \text{Step 2} & \begin{pmatrix} [a_1](t) \\ [a_2](t) \end{pmatrix} & = & \begin{pmatrix} [a_1](0) \\ [a_2](0) \end{pmatrix} + \begin{pmatrix} \int_0^t [u_1](\tau) \cdot \cos([\psi](\tau)) \cdot d\tau \\ \int_0^t [u_1](\tau) \cdot \sin([\psi](\tau)) \cdot d\tau \end{pmatrix} \\ \text{Step 3} & [\omega](t) & = & [\omega](0) + \int_0^t [u_2](\tau) d\tau \\ \text{Step 4} & [\psi](t) & = & [\psi](0) + \int_0^t [\omega](\tau) d\tau \\ \text{Step 5} & \begin{pmatrix} [v_1](t) \\ [v_2](t) \end{pmatrix} & = & \begin{pmatrix} \cos([\psi](t)) & \sin([\psi](t)) \\ -\sin([\psi](t)) & \cos([\psi](t)) \end{pmatrix} \cdot \begin{pmatrix} [a_1](t) \\ [a_2](t) \end{pmatrix} \\ \text{Step 6} & \begin{pmatrix} [x_1](t) \\ [x_2](t) \end{pmatrix} & = & \begin{pmatrix} [x_1](0) \\ [x_2](0) \end{pmatrix} + \begin{pmatrix} \cos([\psi](t)) & -\sin([\psi](t)) \\ \sin([\psi](t)) & \cos([\psi](t)) \end{pmatrix} \begin{pmatrix} [v_1](t) \\ [v_2](t) \end{pmatrix} \end{array} \tag{49}$$

We are able to enclose the corresponding interval trajectory sampling time with $dt = 0.02$. For a more accurate approximation, we bisected the initial box $[\mathbf{x}](0)$ with 20 subboxes, the size of which is smaller than 0.01. The superposition of all interval simulations in the (x_1, x_2) space is given by Figure 13. The computation time on a classical laptop is less than 1sec.

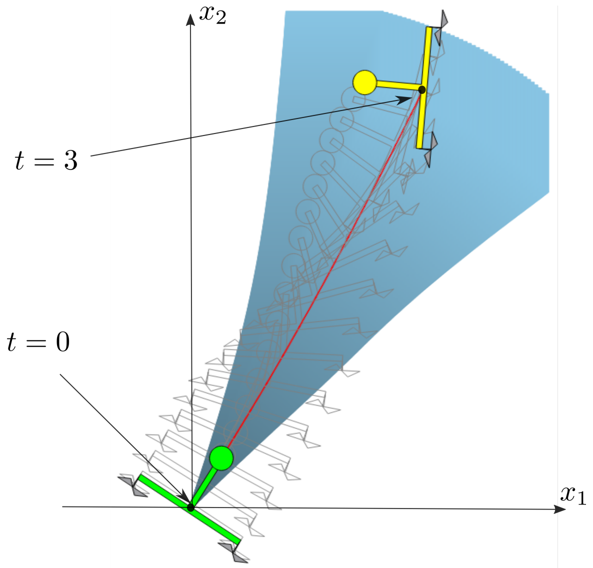

Fig. 13. Integral simulation of the hovercraft. The frame box is $[-2, 8] \times [-2, 9]$

VI. Conclusion

This paper has proposed an original approach to simulate a continuous-time dynamical system with interval uncertainties. The approach takes advantage of an integral algebra to reformulate the state equations of the system so that no loop occurs in the flow graph of the system. This allows an interval integration scheme which does not require any resolution, generally based on fixed point techniques. Our method *(i)* does not require an initial tube, *(ii)* can take into account large uncertainties, *(iii)* only require a basic interval arithmetic extended to tubes.

The main limitation of our approach is that a symbolic calculus has to be done to translate the state equations into an integral representation. Moreover, our method applies to a limited class of systems which includes linear time-invariant systems. This class could be extended by using operators such as delays or convolutions, whereas here, only $+, \cdot, \int$ was proposed. For the convolution, it is important to use a fast and accurate algorithm such as the interval fast convolution given in [28]. This algorithm uses the circular interval arithmetic [39] and provides an algorithm in $n \cdot \log n$ based on the Fast Fourier Transform.

The methodology developed in this paper has been illustrated by two test-cases taken from robotics: the car-trailer and the hovercraft. To my knowledge, no other existing techniques is able to compute a non-trivial guaranteed enclosure of the trajectories considered in the test-cases.

The implementation is done using the Codac library [36] and the source codes are available at

https://www.ensta-bretagne.fr/jaulin/integral.html

References

[1] F. Abdallah, A. Gning, and P. Bonnifait. Box particle filtering for nonlinear state estimation using interval analysis. *Automatica*, 44(3):807–815, 2008. I

[2] M. Althoff. An introduction to cora. *CPS Week*, pages 120–151, 2015. I

[3] E. Asarin, T. Dang, and A. Girard. Reachability analysis of nonlinear systems using conservative approximation. In *Hybrid Systems: Computation and Control*, pages 20–35. Springer Berlin Heidelberg, 2003. I

[4] F. Le Bars, J. Sliwka, O. Reynet, and L. Jaulin. State estimation with fleeting data. *Automatica*, 48(2):381–387, 2012. II-D, II-D

[5] L. Benvenuti, D. Bresolin, P. Collins, A. Ferrari, L. Geretti, and T. Villa. Ariadne: Dominance checking of nonlinear hybrid automata using reachability analysis. In A. Finkel, J. Leroux, and I. Potapov, editors, *Reachability Problems - 6th International Workshop, RP 2012, Bordeaux, France, September 17-19, 2012. Proceedings*, volume 7550 of *Lecture Notes in Computer Science*, pages 79–91. Springer, 2012. I

[6] M. Berz and K. Makino. Verified Integration of ODEs and Flows using Differential Algebraic Methods on High-Order Taylor Models. *Reliable Computing*, 4(3):361–369, 1998. I

[7] S. Bogomolov, M. Forets, G. Frehse, K. Potomkin, and C. Schilling. Juliareach: a toolbox for set-based reachability. In *HSCC 2019*, pages 39–44, 2009. I

[8] O. Bouissou and A. Chapoutot. An operational semantics for simulink's simulation engine. In *ACM SIGPLAN Notices*, 2012. I

[9] M. Cébério and L. Granvilliers. Solving nonlinear systems by constraint inversion and interval arithmetic. In John A. Campbell and Eugenio Roanes-Lozano, editors, *Artificial Intelligence and Symbolic Computation, International Conference AISC 2000 Madrid, Spain, July 17-19, 2000*, volume 1930 of *Lecture Notes in Computer Science*, pages 127–141. Springer, 2000. I

[10] G. Chabert and L. Jaulin. Contractor Programming. *Artificial Intelligence*, 173:1079–1100, 2009. I

[11] X. Chen, E. Abraham, and S. Sankaranarayanan. Flow*: An analyzer for non-linear hybrid system. In *CAV 2013*, pages 258–263, 2013. I

[12] P. Collins and A. Goldsztejn. The Reach-and-Evolve Algorithm for Reachability Analysis of Nonlinear Dynamical Systems. *Electronic Notes in Theoretical Computer Science*, (223):87–102, 2008. I

[13] J. Damers, L. Jaulin, and S. Rohou. Lie symmetries applied to interval integration. *Automatica*, 144, 2021. I

[14] D. Daney, N. Andreff, G. Chabert, and Y. Papegay. Interval Method for Calibration of Parallel Robots: Vision-based Experiments. *Mechanism and Machine Theory, Elsevier*, 41:926–944, 2006. I

[15] V. Drevelle and P. Bonnifait. Localization confidence domains via set inversion on short-term trajectory. *IEEE Transactions on Robotics*, 29(5):1244–1256, oct 2013. I

[16] I. Fantoni and R. Lozano. *Non-linear control for underactuated mechanical systems*. Springer-Verlag, 2001. V-A

[17] M. Fliess, J. Lévine, P. Martin, and P. Rouchon. Flatness and defect of non-linear systems: introductory theory and applications. *International Journal of Control*, (61):1327–1361, 1995. I, II-A

[18] G. Frehse. Phaver: Algorithmic verification of hybrid systems. *International Journal on Software Tools for Technology Transfer*, 10(3):23–48, 2008. I

[19] P.L. Garoche, T. Kahsai, and C. Tinelli. Incremental invariant generation using logic-based automatic abstract transformers. In *NASA Formal Methods Symposium*, pages 139–154. Springer, Berlin, Heidelberg, 2013. I

[20] A. Goldsztejn, W. Hayes, and P. Collins. Tinkerbell Is Chaotic. *SIAM Journal on Applied Dynamical Systems*, 10(4):1480–1501, 2011. I

[21] C. Le Guernic and A. Girard. Reachability analysis of linear systems using support functions. *Nonlinear Analysis*, 2009. I

[22] L. Jaulin. Nonlinear bounded-error state estimation of continuous-time systems. *Automatica*, 38:1079–1082, 2002. I

[23] T. Kapela and P. Zgliczynski. A Lohner-type algorithm for control systems and ordinary differential inclusions. *Discrete and Continuous Dynamical Systems*, 11(2):365–385, 2009. I

[24] B. Kearfott and V. Kreinovich. Applications of interval computations: An introduction. In *Applied*

Optimization, pages 1–22. Springer US, 1996. I

[25] M. Kieffer and E. Walter. Guaranteed Characterization of Exact non-Asymptotic Confidence Regions as Defined by LSCR and SPS. *Automatica*, 50(2):507–512, 2014. I

[26] W. Langson, I. Chryssochoos, S.V. Rakovic, and D.Q. Mayne. Robust model predictive control using tubes. *Automatica*, 40(1):125–133, 2004. I

[27] Y. Il Lee and B. Kouvaritakis. Constrained robust model predictive control based on periodic invariance. *Autom.*, 42(12):2175–2181, 2006. III-C

[28] G. Liu and V. Kreinovich. Fast convolution and fast fourier transform under interval and fuzzy uncertainty. *J. Comput. Syst. Sci.*, 76(1):63–76, 2010. VI

[29] R. Lohner. Enclosing the solutions of ordinary initial and boundary value problems. In E. Kaucher, U. Kulisch, and Ch. Ullrich, editors, *Computer Arithmetic: Scientific Computation and Programming Languages*, pages 255–286. BG Teubner, Stuttgart, Germany, 1987. I

[30] R. Moore. *Methods and Applications of Interval Analysis*. Society for Industrial and Applied Mathematics, jan 1979. I, II-D, II-D

[31] M. Mustafa, A. Stancu, N. Delanoue, and E. Codres. Guaranteed SLAM; An Interval Approach. *Robotics and Autonomous Systems*, 100:160–170, 2018. I

[32] T. Raissi, N. Ramdani, and Y. Candau. Set membership state and parameter estimation for systems described by nonlinear differential equations. *Automatica*, 40:1771–1777, 2004. I

[33] A. Rauh, E. Hofer, and E. Auer. Valencia-ivp: A comparison with other initial value problem solvers. In *International Symposium on Scientific Computing, Computer Arithmetic and Validated Numerics (SCAN 2006)*, 26-29 Sept. 2006, 2006. I

[34] N. Revol, K. Makino, and M. Berz. Taylor models and floating-point arithmetic: proof that arithmetic operations are validated in COSY. *Journal of Logic and Algebraic Programming*, 64:135–154, 2005. I

[35] J. Ritt. *Differential Algebra*. Amer. Math. Soc., New York, 1950. I, II-A

[36] S. Rohou. *Codac (Catalog Of Domains And Contractors), available at* `http://codac.io/`. Robex, Lab-STICC, ENSTA-Bretagne, 2021. I, VI

[37] S. Rohou, L. Jaulin, L. Mihaylova, F. Le Bars, and S. Veres. *Reliable Robot Localization*. Wiley, dec 2019. I, II-D

[38] P. Rouchon, M. Fliess, J. Lévine, and P. Martin. Flatness and motion planning: the car with n-trailers. *European Control Conference*, 01 1992. IV-A

[39] Siegfried M Rump. Fast and parallel interval arithmetic. *BIT Numerical Mathematics*, 39(3):534–554, 1999. VI

[40] J. Alexandre Dit Sandretto and A. Chapoutot. Dynibex: a differential constraint library for studying dynamical systems. In *Conference on Hybrid Systems: Computation and Control*, Vienne, Austria, 2016. I

[41] W. Taha and A. Duracz. Acumen: An open-source testbed for cyber-physical systems research. In *CYCLONE'15*, 2015. I

[42] G. Trombettoni and G. Chabert. Constructive Interval Disjunction. In *Proc. CP, Constraint Programming*, pages 635–650, LNCS 4741, 2007. I

[43] W. Tucker. The lorenz attractor exists. *Comptes Rendus de l'Académie des Sciences - Series I - Mathematics*, 328(12):1197–1202, jun 1999. I

[44] Zhan Wang, Alain Lambert, and Xun Zhang. Dynamic icsp graph optimization approach for car-like robot localization in outdoor environments. *Computers*, 8(63), 2019. I