Lie symmetries applied to interval integration

Julien Damers, Luc Jaulin, Simon Rohou

Lie symmetries applied to interval integration

Julien Damers [a], Luc Jaulin [a], Simon Rohou [a]

[a] *ENSTA Bretagne, Lab-STICC, UMR CNRS 6285, Brest, France*

Abstract

In this paper, we propose a new approach for improving significantly existing guaranteed integration methods for state equations with uncertain initial conditions. We first find a tube that encloses the solution of the differential equation assuming that the initial state is known. Then, using Lie symmetries, we inflate the tube in order to contain the uncertainty associated with the initial state. The method is shown to be efficient on examples coming from reachability analysis and robotics.

Key words: Interval analysis, guaranteed integration, IVP, Lie groups, symmetries, tubes.

1 Introduction

When dealing with non-linear dynamical systems such as mobile robots or cyber-physical systems, it is important to guarantee the compliance of some properties [14], for instance for security reasons. Different solvers have been designed for this purpose such as Acumen [52] or PHAVer [17]. For instance, we would like to guarantee that the system will not enter inside a forbidden region or that it will reach a target set. The guarantee can be obtained by using reachability analysis [3, 23, 13, 12, 5, 2], invariant based approaches [18] or guaranteed integration methods [33, 42, 7, 41]. The goal of guaranteed integration is to find a tube [32] enclosing all feasible trajectories of a system, assuming that the initial vector is known [30, 4]. It has been used to prove conjectures such as the existence of the Lorenz attractor [54], or that a given system is chaotic [20]. It has also been used for state estimation [31, 40, 1], localization [43, 26, 15, 55] or SLAM [36].

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. In this paper, we show for the first time that we may significantly reduce the pessimism of the integration with respect to some uncertainties using Lie groups and Lie symmetries.

Email addresses: julien.damers@ensta-bretagne.org (Julien Damers), lucjaulin@gmail.com (Luc Jaulin), simon.rohou@ensta-bretagne.fr (Simon Rohou).

A Lie group [48] is both an abstract group and a smooth n-dimensional manifold so that multiplication and inversion are both smooth. They have been introduced to model the continuous symmetries of differential equations [39] and are widely used for their resolution. The main idea behind Lie groups is to take advantage of the possible symmetries of the system in order to extend one solution of the problem to all other solutions [51, 50]. One can also note that in the context of control theory, symmetries have been used for stability analysis [46], observers design [6][24], navigation [11] and safety verification [49].

Our main contribution is to show that Lie symmetries can be combined with interval based methods [35], in order to propagate uncertainties [53] through differential equations. More precisely, our goal is to compute an enclosure of the solution of a differential equation assuming that the initial state is inside a box that may be large. We show that in this context, our method outperforms existing approaches. Some test-cases related to robotic applications illustrate the efficiency of this strategy.

The paper is structured as follows. Section 2 provides basic notions on symmetry groups with their interpretation in the context of differential equations. Section 3 defines Lie groups and shows how they can be used to compute one solution from another. Section 4 introduces our new guaranteed integration method that combines Lie symmetries with classical interval integration tools. The efficiency of the approach is shown on several examples. Section 5 proposes an application to the reachability analysis of a dynamical system. Section 6 concludes the paper and proposes some perspectives.

2 Symmetry groups

In this section, we recall some notions of symmetry groups and group actions. Most of these definitions are taken from the book of Olver [39]. They will be combined with interval integration techniques in Section 4 to reduce the propagation of uncertainties through a differential equation.

2.1 Group S_3

The main concepts behind Lie symmetries for differential equations can be understood from an example: the symmetric group S_3 consisting of all six permutations in a set $\mathbb{A} = \{a, b, c\}$.

$$\begin{aligned} S_3 &= \{\sigma_1, \ldots, \sigma_6\} \\ &= \{(a,b,c) \to (a,b,c);\ (a,b,c) \to (a,c,b); \\ &\quad (a,b,c) \to (b,a,c);\ (a,b,c) \to (c,b,a) \\ &\quad ,(a,b,c) \to (b,c,a);\ (a,b,c) \to (c,a,b)\}. \end{aligned} \tag{1}$$

A permutation $\sigma_i : \mathbb{A} \to \mathbb{A}$ is here denoted by extension. For instance, when we write $\sigma_6 : (a,b,c) \to (c,a,b)$, we mean that $\sigma_6(a) = c, \sigma_6(b) = b, \sigma_6(c) = a$. It is easy to check that $(S_3, \circ)$ is a group since: it is closed by the composition $\circ$, it satisfies the associative property, there is an identity element (here σ_1), and there is an inverse for each $\sigma \in S_3$ (e.g., $\sigma_5^{-1} = \sigma_6$). In this group, we can make some computations and solve equations. For instance, since

$$\begin{aligned} \sigma_6 \circ \sigma_2 \circ \sigma_6^{-1}(a,b,c) &= \sigma_6 \circ \sigma_2 (b,c,a) \\ = \quad \sigma_6(b,a,c) \quad &= \quad (c,b,a) \\ = \quad \sigma_4(a,b,c), & \end{aligned} \tag{2}$$

we have $\sigma_4 = \sigma_6 \circ \sigma_2 \circ \sigma_6^{-1}$.

Group action. Consider a set $\mathbb{F}$ of objects and the group $(G, \circ)$, with σ_1 as neutral element. Consider also a binary operator $\bullet$ from $G \times \mathbb{F}$ to $\mathbb{F}$. The structure $(G, \circ, \bullet)$ is a *left group action* on $\mathbb{F}$ if it satisfies the following axioms:

$$\begin{aligned} \forall f \in \mathbb{F}, \sigma_1 \bullet f = f \quad &\text{(identity)} \\ \forall \sigma_i \in G, \forall \sigma_j \in G & \\ (\sigma_i \circ \sigma_j) \bullet f = \sigma_i \bullet (\sigma_j \bullet f) \quad &\text{(compatibility)} \end{aligned} \tag{3}$$

If the compatibility condition is replaced by

$$(\sigma_i \circ \sigma_j) \bullet f = \sigma_j \bullet (\sigma_i \bullet f) \tag{4}$$

then $(G, \circ, \bullet)$ is a *right group action.*

In the context of S_3, the operator $\bullet$ may be seen as a composition $\circ$, but it is generally not the case. For instance, $\mathbb{F}$ can be a set of objects in a 3D word, and G may correspond to the set of Euclidean transformations (rotations, translations). In such a case, a typical action could be "rotate the object f", which is not a composition.

An illustration is given by Figure 1 where $\mathbb{F}$ is the set of all applications from $\mathbb{A} = \{a, b, c\}$ to $\mathbb{A}$. We define $f \in \mathbb{F}$ as the application $f(a) = a, f(b) = a, f(c) = b$. We have chosen to represent f under the form of a dynamical graph since later, it will correspond to a dynamical system. For instance, $f(c) = b$ means that if the system is at state c then at the next step, it will be at state b. Define the action of σ_i on f as

$$\sigma_i \bullet f = f \circ \sigma_i. \tag{5}$$

We can check that it satisfies the axioms of a right group action. Since the identity relation is trivial, it suffices to check the compatibility relation (4)

$$\begin{aligned} (\sigma_i \circ \sigma_j) \bullet f &\overset{(5)}{=} f \circ (\sigma_i \circ \sigma_j) \\ &= (f \circ \sigma_i) \circ \sigma_j \\ &\overset{(5)}{=} (\sigma_i \bullet f) \circ \sigma_j \\ &\overset{(5)}{=} \sigma_j \bullet (\sigma_i \bullet f) \end{aligned}$$

For instance, the action of σ_5 on f is $g = \sigma_5 \bullet f = f \circ \sigma_5$. Thus, $g(a) = \sigma_5 \bullet f(a) = f \circ \sigma_5(a) = f(c) = b$. We understand that the action of σ_5 transforms the dynamical system f into another one.

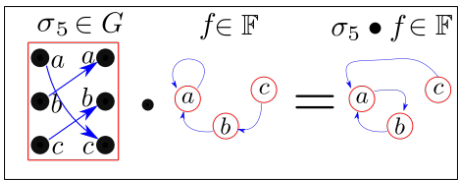

Fig. 1. A permutation σ_5 of S_3 acting on an object $f \in \mathbb{F}$.

Orbit. Consider a group G acting on a set $\mathbb{F}$. The orbit of an element x in $\mathbb{F}$ is $G(x) = \{g(x) | g \in G\}$.

Stabilizer. For f in $\mathbb{F}$, we define the stabilizer subgroup $\mathrm{Sym}_G(f)$ of G with respect to f as the set of all elements in G for which f remains unchanged:

$$\mathrm{Sym}_G(f) = \{\sigma \in G | \sigma \bullet f = f\}. \tag{6}$$

In our example we can check that:

$$\mathrm{Sym}_G(f) = \{\sigma_1, \sigma_3\}. \tag{7}$$

In group theory, the symmetry group of a geometric object f (*e.g.*, a cube or a cylinder) is the group of all trans-

formations for which the object is invariant. In the context of this paper, the object will be a vector field representing a state equation of a dynamical system. Transformations such as translations, rotations, scaling, ... may leave the vector field unchanged. These transformations form a symmetry group as explained in the following.

2.2 Application to differential equations

In the context of this paper, $\mathbb{A}$ corresponds to the state space $\mathbb{R}^n$ and $\mathbb{F}$ corresponds to the set of all state equations of the form $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$ where $\mathbf{x} \in \mathbb{R}^n$ [39]. In this paper, we assume that $\mathbf{f}$ is locally Lipschitz so that there exists a function $\mathbf{\Phi}_t : \mathbb{R}^n \to \mathbb{R}^n$, called the *flow*, which associates to all initial vectors $\mathbf{x}_0$ the solution $\mathbf{\Phi}_t(\mathbf{x}_0)$ of the state equation.

The stabilizers of $\mathbf{f} \in \mathbb{F}$ are the *symmetries* of the system. For ease of understanding, we will voluntarily use the same notations between the state equation $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$ and the vector field $\mathbf{f}$.

We denote by $\text{diff}(\mathbb{R}^n)$ the set of diffeomorphisms from $\mathbb{R}^n$ to $\mathbb{R}^n$.

Definition 1 *Consider a state equation $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$, $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{g} \in$ diff$(\mathbb{R}^n)$. We define the* action *of $\mathbf{g}$ as*

$$\mathbf{g} \bullet \mathbf{f} = \left(\frac{d\mathbf{g}}{d\mathbf{x}} \circ \mathbf{g}^{-1}\right) \cdot \left(\mathbf{f} \circ \mathbf{g}^{-1}\right). \tag{8}$$

It transforms the field $\mathbf{f}$ into another field.

Proposition 2 *Assume that $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$ and $\mathbf{y} = \mathbf{g}(\mathbf{x})$, where $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{g} \in$ diff$(\mathbb{R}^n)$. Then, we have $\dot{\mathbf{y}} = \mathbf{g} \bullet \mathbf{f}(\mathbf{y})$. Equivalently, the action of $\mathbf{g}$ generates from the system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$ the new system $\dot{\mathbf{y}} = (\mathbf{g} \bullet \mathbf{f})(\mathbf{y})$.*

Proof: We have the following first order approximation:

$$\begin{aligned}
&\mathbf{y}(t + dt) \\
&= \mathbf{g}(\mathbf{x}(t + dt)) \\
&= \mathbf{g}(\mathbf{x}(t)) + \tfrac{d\mathbf{g}}{d\mathbf{x}}(\mathbf{x}(t)) \cdot \tfrac{d\mathbf{x}}{dt}(t) \cdot dt + \mathbf{o}(dt) \\
&= \mathbf{g}(\mathbf{x}(t)) + \tfrac{d\mathbf{g}}{d\mathbf{x}}(\mathbf{x}(t)) \cdot \mathbf{f}(\mathbf{x}(t)) \cdot dt + \mathbf{o}(dt) \\
&= \mathbf{y}(t) + dt \cdot \tfrac{d\mathbf{g}}{d\mathbf{x}}\left(\mathbf{g}^{-1}(\mathbf{y}(t))\right) \cdot \mathbf{f}\left(\mathbf{g}^{-1}(\mathbf{y}(t))\right) + \mathbf{o}(dt) \\
&= \mathbf{y}(t) + dt \cdot (\mathbf{g} \bullet \mathbf{f})(\mathbf{y}(t)) + \mathbf{o}(dt)
\end{aligned}$$

Thus, due to the unicity of the first order Taylor expansion, we get $\dot{\mathbf{y}} = (\mathbf{g} \bullet \mathbf{f})(\mathbf{y})$.

Proposition 3 *The set (diff$(\mathbb{R}^n), \circ, \bullet$) equipped with the composition $\circ$ and the action $\bullet$ as defined by Equation (3) is a left group action.*

Proof: We first check that $(\text{diff}(\mathbb{R}^n), \circ, \bullet)$ is closed by the composition $\circ$: if $\mathbf{g}_1, \mathbf{g}_2 \in \text{diff}(\mathbb{R}^n)$ then $\mathbf{g}_1 \circ \mathbf{g}_2 \in \text{diff}(\mathbb{R}^n)$; We have the associativity $((\mathbf{g}_1 \circ \mathbf{g}_2) \circ \mathbf{g}_3 = \mathbf{g}_1 \circ (\mathbf{g}_2 \circ \mathbf{g}_3))$; there is an identity element $\mathbf{g}_0$ in $\text{diff}(\mathbb{R}^n)$; and for each $\mathbf{g}_1 \in \text{diff}(\mathbb{R}^n)$ there is $\mathbf{g}_2 \in \text{diff}(\mathbb{R}^n)$ such that $\mathbf{g}_1 \circ \mathbf{g}_2 = \mathbf{g}_0$. To check that it is a group action, we also need the identity and compatibility (see Equation (3)). The identity is trivial. Let us now check the compatibility. For this, we consider the state equation $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$.
(i) If $\mathbf{z} = \mathbf{h} \circ \mathbf{g}(\mathbf{x})$, and since $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$, we have from Proposition 2:

$$\dot{\mathbf{z}} = ((\mathbf{h} \circ \mathbf{g}) \bullet \mathbf{f})(\mathbf{z}).$$

(ii) Since $\mathbf{z} = \mathbf{h}(\mathbf{y})$ and since $\dot{\mathbf{y}} = (\mathbf{g} \bullet \mathbf{f})(\mathbf{y})$, we have from Proposition 2:

$$\dot{\mathbf{z}} = (\mathbf{h} \bullet (\mathbf{g} \bullet \mathbf{f}))(\mathbf{z}).$$

Thus we get the compatibility property

$$(\mathbf{h} \circ \mathbf{g}) \bullet \mathbf{f} = \mathbf{h} \bullet (\mathbf{g} \bullet \mathbf{f}).$$

Definition 4 *A transformation $\mathbf{g}$ is a* stabilizer *of $\mathbf{f}$ if $\mathbf{g} \bullet \mathbf{f} = \mathbf{f}$,* i.e. *if it satisfies the partial differential equation*

$$\mathbf{g} \bullet \mathbf{f} = \left(\frac{d\mathbf{g}}{d\mathbf{x}} \circ \mathbf{g}^{-1}\right) \cdot \left(\mathbf{f} \circ \mathbf{g}^{-1}\right) = \mathbf{f}. \tag{9}$$

Remark 5 *Equation* (9) *is equivalent to*

$$\left(\frac{d\mathbf{g}}{d\mathbf{x}}\right) \cdot \mathbf{f} = \mathbf{f} \circ \mathbf{g}. \tag{10}$$

When $\mathbf{g}$ is linear, we get $\mathbf{g} \circ \mathbf{f} = \mathbf{f} \circ \mathbf{g}$. This means that both functions $\mathbf{f}$ and $\mathbf{g}$ commute by composition. This is known as the equivariance *property [21].*

Proposition 6 *If $\mathbf{\Phi}_t : \mathbb{R}^n \to \mathbb{R}^n$ is the* flow *associated with the state equation $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$. We have:*

$$\mathbf{g} \bullet \mathbf{f} = \mathbf{f} \Leftrightarrow \mathbf{\Phi}_t \circ \mathbf{g} = \mathbf{g} \circ \mathbf{\Phi}_t. \tag{11}$$

Proof: *Take $\mathbf{y} = \mathbf{g}(\mathbf{x})$ and $\mathbf{x}_0 \in \mathbb{R}^n$. For the corresponding trajectory $\mathbf{x}(t) = \mathbf{\Phi}_t(\mathbf{x}_0)$, we have*

$$\begin{aligned}
&\mathbf{g} \circ \mathbf{\Phi}_t(\mathbf{x}_0) = \mathbf{\Phi}_t \circ \mathbf{g}(\mathbf{x}_0), \forall t \\
\Leftrightarrow\ &\mathbf{g}(\mathbf{x}(t)) = \mathbf{\Phi}_t(\mathbf{g}(\mathbf{x}_0)), \forall t \\
\Leftrightarrow\ &\mathbf{y}(t) = \mathbf{\Phi}_t(\mathbf{y}(0)), \forall t \\
\Leftrightarrow\ &\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y}) \\
\Leftrightarrow\ &\mathbf{g} \bullet \mathbf{f} = \mathbf{f}.
\end{aligned}$$

The last equivalence comes from Proposition 2 which states that $\dot{\mathbf{y}} = \mathbf{g} \bullet \mathbf{f}(\mathbf{y})$. *We have indeed*

$$\begin{cases} \dot{\mathbf{y}} = \mathbf{f}(\mathbf{y}) \\ \dot{\mathbf{y}} = \mathbf{g} \bullet \mathbf{f}(\mathbf{y}) \end{cases} \Rightarrow \mathbf{g} \bullet \mathbf{f} = \mathbf{f}$$

and reciprocally

$$\begin{cases} \mathbf{g} \bullet \mathbf{f} = \mathbf{f} \\ \dot{\mathbf{y}} = \mathbf{g} \bullet \mathbf{f}(\mathbf{y}) \end{cases} \Rightarrow \dot{\mathbf{y}} = \mathbf{f}(\mathbf{y}).$$

Remark 7 *Consider the solution* $t \mapsto \mathbf{\Phi}_t(\mathbf{x}_0)$ *of the state equation* $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$, *for* $\mathbf{x}(0) = \mathbf{x}_0$. *Set* $\mathbf{x}_1 = \mathbf{g}(\mathbf{x}_0)$, *where* $\mathbf{g}$ *is a stabilizer. From Proposition 6, we get:*

$$\mathbf{\Phi}_t(\mathbf{x}_1) = \mathbf{\Phi}_t \circ \mathbf{g}(\mathbf{x}_0) \overset{(11)}{=} \mathbf{g} \circ \mathbf{\Phi}_t(\mathbf{x}_0) \tag{12}$$

We thus get an expression of the solution of the state equation corresponding to the initial condition $\mathbf{x}(0) = \mathbf{x}_1$. If we have a family of stabilizers, we can generate a family of solutions from the unique solution $\mathbf{\Phi}_t(\mathbf{x}_0)$.

Example 8 *Since*

$$\mathbf{\Phi}_t \circ \mathbf{\Phi}_{t_1} = \mathbf{\Phi}_{t_1} \circ \mathbf{\Phi}_t = \mathbf{\Phi}_{t+t_1} \tag{13}$$

with $t_1 \in \mathbb{R}$, *then* $\mathbf{\Phi}_{t_1} \bullet \mathbf{f} = \mathbf{f}$, i.e., $\mathbf{\Phi}_{t_1}$ *is a stabilizer. The function* $\mathbf{\Phi}_{t_1}$ *is called the* one-parameter flow symmetry [39].

Proposition 9 *Consider a state equation* $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$ *and a stabilizer* $\mathbf{g}$. *In the new coordinate system* $\mathbf{y} = \mathbf{h}(\mathbf{x})$ *where* $\mathbf{h}$ *is bijective and smooth, the action* $\mathbf{h} \circ \mathbf{g} \circ \mathbf{h}^{-1}$ *is a stabilizer.*

Proof: In the $\mathbf{y}$ space, the flow is defined by:

$$\mathbf{\Psi}_t(\mathbf{y}) = \mathbf{h} \circ \mathbf{\Phi}_t \circ \mathbf{h}^{-1}(\mathbf{y}). \tag{14}$$

Since $\mathbf{g}$ is a stabilizer for the system, we have

$$\begin{aligned}
& \mathbf{\Phi}_t(\mathbf{x}) = \mathbf{g} \circ \mathbf{\Phi}_t \circ \mathbf{g}^{-1}(\mathbf{x}) \\
\Leftrightarrow\ & \mathbf{\Phi}_t \circ \mathbf{h}^{-1}(\mathbf{y}) = \mathbf{g} \circ \mathbf{\Phi}_t \circ \mathbf{g}^{-1} \circ \mathbf{h}^{-1}(\mathbf{y}) \\
\Leftrightarrow\ & \underbrace{\mathbf{h} \circ \mathbf{\Phi}_t \circ \mathbf{h}^{-1}(\mathbf{y})}_{\mathbf{\Psi}_t(\mathbf{y})} = \mathbf{h} \circ \mathbf{g} \circ \underbrace{\mathbf{\Phi}_t}_{\mathbf{h}^{-1} \circ \mathbf{\Psi}_t \circ \mathbf{h}} \circ \mathbf{g}^{-1} \circ \mathbf{h}^{-1}(\mathbf{y}) \\
\Leftrightarrow\ & \mathbf{\Psi}_t(\mathbf{y}) = \left(\mathbf{h} \circ \mathbf{g} \circ \mathbf{h}^{-1}\right) \circ \mathbf{\Psi}_t \circ \left(\mathbf{h} \circ \mathbf{g}^{-1} \circ \mathbf{h}^{-1}\right)(\mathbf{y}) \\
\Leftrightarrow\ & \mathbf{\Psi}_t \circ \left(\mathbf{h} \circ \mathbf{g} \circ \mathbf{h}^{-1}\right)(\mathbf{y}) = \left(\mathbf{h} \circ \mathbf{g} \circ \mathbf{h}^{-1}\right) \circ \mathbf{\Psi}_t.
\end{aligned}$$

As a consequence, $\mathbf{h} \circ \mathbf{g} \circ \mathbf{h}^{-1}$ is a stabilizer for the system in the $\mathbf{y}$ coordinates.

Proposition 10 *The set of all stabilizers of* $\mathbf{f}$ *is a group with respect to the composition* $\circ$. *It is called the* symmetry group *of* $\mathbf{f}$ *and is denoted by Sym*$(\mathbf{f})$.

Proof: To check the group property, we take $\mathbf{g}_1, \mathbf{g}_2$ in $\mathrm{Sym}(\mathbf{f})$. We have

$$\begin{aligned}
(\mathbf{g}_1 \circ \mathbf{g}_2) \circ \mathbf{\Phi}_t &= \mathbf{g}_1 \circ (\mathbf{g}_2 \circ \mathbf{\Phi}_t) \overset{(11)}{=} \mathbf{g}_1 \circ (\mathbf{\Phi}_t \circ \mathbf{g}_2) \\
&= (\mathbf{g}_1 \circ \mathbf{\Phi}_t) \circ \mathbf{g}_2 \overset{(11)}{=} (\mathbf{\Phi}_t \circ \mathbf{g}_1) \circ \mathbf{g}_2 \\
&= \mathbf{\Phi}_t \circ (\mathbf{g}_1 \circ \mathbf{g}_2)
\end{aligned}$$

Thus $\mathbf{g}_1 \circ \mathbf{g}_2 \in \mathrm{Sym}(\mathbf{f})$. The other properties to be checked to make $\mathrm{Sym}(\mathbf{f})$ a group are trivial. See [51] for a detailed proof.

2.3 Test case 1

Consider the system

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \mathbf{f}(\mathbf{x}) = \begin{pmatrix} 1 \\ -x_2 \end{pmatrix}. \tag{15}$$

As illustrated by Figure 2, the mirror symmetry $\mathbf{g}_1$ with respect to the axis Ox_1 is a stabilizer since its action does not change the field. The horizontal translation $\mathbf{g}_2$ is also a stabilizer. Since for all t_1, $\mathbf{\Phi}_{t_1} \circ \mathbf{\Phi}_t = \mathbf{\Phi}_t \circ \mathbf{\Phi}_{t_1}$ we get that $\mathbf{\Phi}_{t_1}$ is also a stabilizer. Therefore, $\mathbf{g}_1$, $\mathbf{g}_2$, $\mathbf{\Phi}_{t_1}$ all belong to $\mathrm{Sym}(\mathbf{f})$. In the figure, $\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}, \mathbf{e}, \mathbf{f}, \mathbf{g}, \mathbf{h}$ are points of the state space that are fixed to illustrate how we can move along any trajectory.

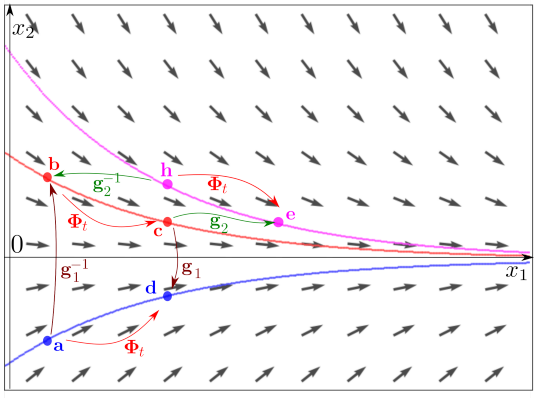

Fig. 2. The system has a x_2 symmetry translation and an Ox_1 mirror-symmetry.

From this example, we see the main idea we want to develop. For the system, if we know one trajectory (called the *reference*, red in the picture), we may generate other trajectories by the application of symmetries. For instance if we want to compute $\mathbf{\Phi}_t(\mathbf{a})$, we apply the formula $\mathbf{\Phi}_t(\mathbf{a}) = \mathbf{g}_1 \circ \mathbf{\Phi}_t \circ \mathbf{g}_1^{-1}(\mathbf{a})$, *i.e.*, we first move to

the reference using $\mathbf{g}_1^{-1}$ to get $\mathbf{b}$, then we move forward on the reference to get $\mathbf{c}$, and we come back to the right state with $\mathbf{g}_1$ to finally get $\mathbf{d} = \mathbf{\Phi}_t(\mathbf{a})$.

In a similar way, since $\mathbf{\Phi}_t(\mathbf{h}) = \mathbf{g}_2 \circ \mathbf{\Phi}_t \circ \mathbf{g}_2^{-1}(\mathbf{h})$, we can compute $\mathbf{e} = \mathbf{\Phi}_t(\mathbf{h})$ using the knowledge of the reference only (red) and the symmetry $\mathbf{g}_2$.

As we will see on the following section, in some cases, it is possible to get a symmetry $\mathbf{g}_1$ to move from the initial state onto the reference. If we assume that we have computed accurately the solution $\mathbf{\Phi}_t$, for a specific initial state to get the reference, then, we will be able to move precisely forward and backward in time everywhere in the state space.

3 Lie groups of symmetries

3.1 Definition

Consider a state equation $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$ and a manifold $\mathbb{P}$. A Lie group $G_{\mathbf{p}}$ of symmetries is a family of diffeomorphisms $\mathbf{g}_{\mathbf{p}} \in \text{diff}(\mathbb{R}^n)$ parameterized by $\mathbf{p} \in \mathbb{P}$ such that

- $G_{\mathbf{p}}$ is a Lie group with respect to the composition $\circ$,
- $\forall \mathbf{p} \in \mathbb{P}, \mathbf{g}_{\mathbf{p}} \bullet \mathbf{f} = \mathbf{f}$.

Lie symmetries are usually found from the physics and the intuition we have on the system. Nevertheless, for many systems such as the chaotic Lorenz attractor, such symmetries probably will not exist [54]. Indeed, for such chaotic system, we believe that there is no diffeomorphism that transforms one trajectory into any other.

3.2 Transport function

Consider a Lie group of symmetries $G_{\mathbf{p}}$. We assume that the group action $G_{\mathbf{p}}$ is *transitive, i.e.,* it has only one orbit [39]. In this case, there exists a function $\mathbf{h} : \mathbb{R}^n \times \mathbb{R}^n \mapsto \mathbb{P}$, named *transport function,* such that $\mathbf{h}(\mathbf{x}, \mathbf{a})$ corresponds to the displacement $\mathbf{p}$ to be chosen so that the point $\mathbf{a}$ is moved to $\mathbf{x}$ by $\mathbf{g}_{\mathbf{p}}$, *i.e.*,

$$\mathbf{g}_{\mathbf{h}(\mathbf{x},\mathbf{a})}(\mathbf{a}) = \mathbf{x}. \tag{16}$$

Note that such a transport function is not necessary unique.

The notion of transport function is related to the *moving frame* [6] with some differences: here, we assume that we have one orbit which allows us to avoid the introduction of the notion of cross section. Moreover, our transport function $\mathbf{h}$ has two arguments instead of one which makes simpler the transport from one point to any other point of the state space. Finding an expression for $\mathbf{h}(\mathbf{x}, \mathbf{a})$ can be done using symbolic method as for instance by solving the normalization equations as explained in [38] page 163.

We will show that it has a fundamental role for a combination with interval methods for guaranteed integration. In what follows, we will assume that we have a closed form for $\mathbf{h}(\mathbf{x}, \mathbf{a})$. In practice, it can be derived from the symmetries of the problem as illustrated by the following examples.

3.3 Example 1 (continued)

System (15) (from the previous Test case 1) has the following Lie symmetry:

$$\mathbf{g}_{\mathbf{p}} : \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \to \begin{pmatrix} x_1 + p_1 \\ p_2 x_2 \end{pmatrix}, \quad \mathbf{p} \in \mathbb{R}^2. \tag{17}$$

This can be found by the geometry of the vector field. To check this, we need to prove that $\forall \mathbf{p}, \mathbf{g}_{\mathbf{p}} \bullet \mathbf{f} = \mathbf{f}$. We have

$$\begin{aligned}
\mathbf{g}_{\mathbf{p}} \bullet \mathbf{f}(\mathbf{x}) &\overset{(8)}{=} \left(\left(\frac{d\mathbf{g}_{\mathbf{p}}}{d\mathbf{x}} \circ \mathbf{g}_{\mathbf{p}}^{-1} \right) \cdot \left(\mathbf{f} \circ \mathbf{g}_{\mathbf{p}}^{-1} \right) \right)(\mathbf{x}) \\
&= \left(\left(\frac{d\mathbf{g}_{\mathbf{p}}}{d\mathbf{x}} \right) \cdot \mathbf{f} \right) \circ \mathbf{g}_{\mathbf{p}}^{-1}(\mathbf{x}) \\
&= \left(\begin{pmatrix} 1 & 0 \\ 0 & p_2 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ -x_2 \end{pmatrix} \right) \circ \begin{pmatrix} x_1 - p_1 \\ \frac{x_2}{p_2} \end{pmatrix}. \\
&= \begin{pmatrix} 1 \\ -p_2 x_2 \end{pmatrix} \circ \begin{pmatrix} x_1 - p_1 \\ \frac{x_2}{p_2} \end{pmatrix} \\
&= \begin{pmatrix} 1 \\ -x_2 \end{pmatrix} = \mathbf{f}(\mathbf{x})
\end{aligned}$$

The transport function is obtained as follows:

$$\begin{aligned}
\mathbf{g}_{\mathbf{p}}(\mathbf{a}) = \mathbf{x} &\Leftrightarrow \begin{pmatrix} a_1 + p_1 \\ p_2 a_2 \end{pmatrix} = \mathbf{x} \\
&\Leftrightarrow \mathbf{p} = \begin{pmatrix} x_1 - a_1 \\ x_2/a_2 \end{pmatrix} = \mathbf{h}(\mathbf{x}, \mathbf{a}).
\end{aligned} \tag{18}$$

3.4 Test case 2

Consider the system:

$$\begin{cases} \dot{x}_1 = 1 \\ \dot{x}_2 = \sin x_1 \end{cases}. \tag{19}$$

This system is invariant by a translation with respect to the direction x_2. Equivalently, if we move the related vector-field with respect to the x_2-axis, the field does not change. The set of all x_2-translations is a Lie group $\mathcal{L}$ which is diffeomorphic to $(\mathbb{R}, +)$. Furthermore, if we

perform a translation $p_2 = \pm k2\pi$ along x_1, the field does not change neither. However, p_2 belongs to a discrete set and not to a continuous manifold as required to have a Lie symmetry.

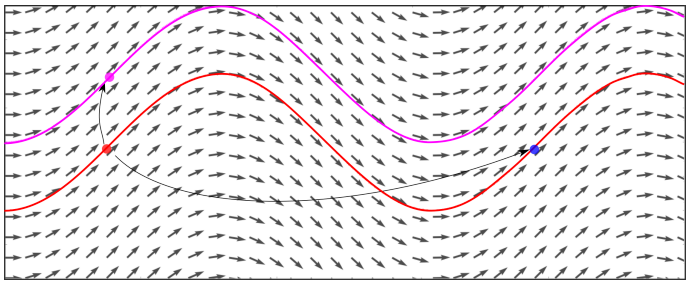

Fig. 3. We switch from one trajectory to another by any translation along x_2 or any 2π translation along x_1.

Proposition 11 *The following transformations*

$$\mathbf{g_p}(\mathbf{x}) = \begin{pmatrix} 0 \\ p_1 \end{pmatrix} + \mathbf{\Phi}_{p_2}(\mathbf{x}) \tag{20}$$

with $\mathbf{p} = (p_1, p_2)$ *are Lie symmetries for System (19).*

This proposition illustrates that the flow itself $\mathbf{\Phi}_t(\mathbf{x})$ can be used as a symmetry (see Example 8).

Proof: If we define

$$\mathbf{g}^1_{p_1} : \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \to \begin{pmatrix} x_1 \\ p_1 + x_2 \end{pmatrix}, p_1 \in \mathbb{R}, \tag{21}$$

and

$$\mathbf{g}^2_{p_2} : \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \to \mathbf{\Phi}_{p_2}(\mathbf{x}), p_2 \in \mathbb{R}, \tag{22}$$

then we have $\mathbf{g_p} = \mathbf{g}^1_{p_1} \circ \mathbf{g}^2_{p_2}$. We thus have to check that both $\mathbf{g}^1_{p_1}$ and $\mathbf{g}^2_{p_2}$ are stabilizers (See (i), (ii) below).

(i) From the equivariance property (see Remark 5), for $\mathbf{g}^1_{p_1}$ which is linear, we get:

$$\underbrace{\mathbf{I}_{2\times 2}}_{\frac{d\mathbf{g}^1_{p_1}}{d\mathbf{x}}(\mathbf{x})} \cdot \underbrace{\begin{pmatrix} 1 \\ \sin x_1 \end{pmatrix}}_{\mathbf{f}(\mathbf{x})} = \underbrace{\begin{pmatrix} 1 \\ \sin x_1 \end{pmatrix}}_{\mathbf{f}(\mathbf{x})} \circ \underbrace{\begin{pmatrix} x_1 \\ p_1 + x_2 \end{pmatrix}}_{\mathbf{g}^1_{p_1}(\mathbf{x})}$$

which is true.

(ii) From Example 8, we know that $\mathbf{g}^2_{p_2}$ is a stabilizer.

Transport function. The transport function (depicted in Figure 4) is obtained as follows:

$$\begin{aligned}
\mathbf{g_p}(\mathbf{a}) = \mathbf{x} &\Leftrightarrow \begin{pmatrix} 0 \\ p_1 \end{pmatrix} + \mathbf{\Phi}_{p_2}(\mathbf{a}) = \mathbf{x} \\
&\Leftrightarrow \begin{pmatrix} 0 \\ p_1 \end{pmatrix} + \begin{pmatrix} \phi_{p_2,1}(\mathbf{a}) \\ \phi_{p_2,2}(\mathbf{a}) \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \\
&\Leftrightarrow \begin{pmatrix} 0 \\ p_1 \end{pmatrix} + \begin{pmatrix} a_1 + p_2 \\ \phi_{p_2,2}(\mathbf{a}) \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \\
&\Leftrightarrow \begin{pmatrix} p_2 \\ p_1 + \phi_{x_1-a_1,2}(\mathbf{a}) \end{pmatrix} = \begin{pmatrix} x_1 - a_1 \\ x_2 \end{pmatrix} \\
&\Leftrightarrow \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} = \underbrace{\begin{pmatrix} x_2 - \phi_{x_1-a_1,2}(\mathbf{a}) \\ x_1 - a_1 \end{pmatrix}}_{\mathbf{h}(\mathbf{x},\mathbf{a})}
\end{aligned} \tag{23}$$

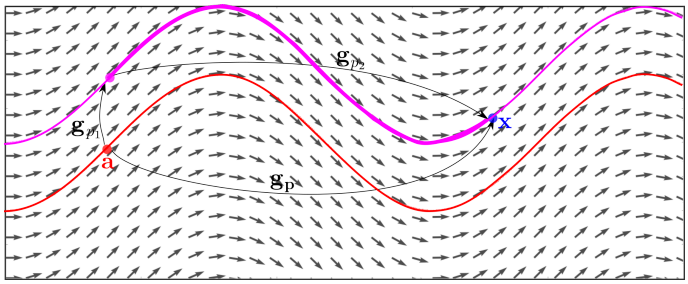

Fig. 4. From the transport function, we find the right $\mathbf{p}$ to move from $\mathbf{a}$ to $\mathbf{x}$ by using $\mathbf{g_p}$.

3.5 Test-case 3

We now consider a test-case taken from [25], Chapter 1, defined by:

$$\begin{cases} \dot{x}_1 = -x_1^3 - x_1 x_2^2 + x_1 - x_2 \\ \dot{x}_2 = -x_2^3 - x_1^2 x_2 + x_1 + x_2 \end{cases}. \tag{24}$$

Proposition 12 *The following transformations*

$$\mathbf{g_p}(\mathbf{x}) = \frac{1}{\sqrt{p_2 + (x_1^2 + x_2^2)(1 - p_2)}} \cdot \mathbf{R}_{p_1} \cdot \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \tag{25}$$

with

$$\mathbf{R}_{p_1} = \begin{pmatrix} \cos p_1 & -\sin p_1 \\ \sin p_1 & \cos p_1 \end{pmatrix} \tag{26}$$

and $\mathbf{p} = (p_1, p_2)$ *are Lie symmetries for System (24).*

Proof: We have $\mathbf{g_p} = \mathbf{g}_1 \circ \mathbf{g}_2$, where

$$\begin{aligned} \mathbf{g}_1 &: \mathbf{x} \to \mathbf{R}_{p_1} \cdot \mathbf{x} \\ \mathbf{g}_2 &: \mathbf{x} \to \frac{1}{\sqrt{p_2 + \left(x_1^2 + x_2^2\right)(1-p_2)}} \cdot \mathbf{x} \end{aligned} \tag{27}$$

Therefore, we need to show that for all $\mathbf{p}$, both $\mathbf{g}_1$ and $\mathbf{g}_2$ are stabilizers. Let us rewrite into polar coordinates. We have

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = r \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix}$$
$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} -r\sin\theta & \cos\theta \\ r\cos\theta & \sin\theta \end{pmatrix} \begin{pmatrix} \dot{\theta} \\ \dot{r} \end{pmatrix}$$

Therefore

$$\begin{aligned} \begin{pmatrix} \dot{\theta} \\ \dot{r} \end{pmatrix} &= \begin{pmatrix} -r\sin\theta & \cos\theta \\ r\cos\theta & \sin\theta \end{pmatrix}^{-1} \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} \\ &= \begin{pmatrix} -\frac{\sin\theta}{r} & \frac{\cos\theta}{r} \\ \cos\theta & \sin\theta \end{pmatrix} \cdot \begin{pmatrix} -x_1^3 - x_1 x_2^2 + x_1 - x_2 \\ -x_2^3 - x_1^2 x_2 + x_1 + x_2 \end{pmatrix} \\ &= \begin{pmatrix} -\frac{\sin\theta}{r} & \frac{\cos\theta}{r} \\ \cos\theta & \sin\theta \end{pmatrix} \cdot \\ &\quad \begin{pmatrix} -r^3\cos^3\theta - r^3\cos\theta\sin^2\theta + r\cos\theta - r\sin\theta \\ -r^3\sin^3\theta - r^3\cos^2\theta\sin\theta + r\cos\theta + r\sin\theta \end{pmatrix} \\ &= \begin{pmatrix} 1 \\ -r^3 + r \end{pmatrix} \end{aligned}$$

Since these two equations for the evolution θ and r are decoupled, we need two find a stabilizer for $\dot{\theta} = 1$ and for $\dot{r} = -r\left(r^2 - 1\right)$ separately. A stabilizer for $\dot{\theta} = 1$ is trivially a translation of the form $g_1(\theta) = \theta + p_1$. Let us find stabilizer $g_2(r)$ for $\dot{r} = -r\left(r^2 - 1\right)$. From Equation (10), we get the equivariance condition:

$$\begin{aligned} \tfrac{dg_2}{dr}(r) \cdot r\left(r^2 - 1\right) &= \left(r\left(r^2 - 1\right)\right) \circ g_2(r) \\ &= g_2(r)\left(g_2^2(r) - 1\right). \end{aligned} \tag{28}$$

Or equivalently

$$\frac{dg_2}{dr}(r) = \frac{g_2(r)\left(g_2^2(r) - 1\right)}{r\left(r^2 - 1\right)} \tag{29}$$

which is a Bernoulli equation. The solution has the form

$$g_2(r) = \frac{r}{\sqrt{p_2 + r^2(1 - p_2)}} \tag{30}$$

where p_2 is a parameter. As a consequence, the polar coordinate transformation

$$\begin{pmatrix} \theta \\ r \end{pmatrix} \overset{\mathbf{g}}{\to} \begin{pmatrix} \theta + p_1 \\ \frac{r}{\sqrt{p_2 + r^2(1-p_2)}} \end{pmatrix} \tag{31}$$

is a stabilizer. To get the stabilizer in the Cartesian coordinates, we apply Proposition 9 with

$$\mathbf{h}\begin{pmatrix} \theta \\ r \end{pmatrix} = r\begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix}.$$

And so the stabilizer is

$$\begin{aligned} &\mathbf{h} \circ \mathbf{g} \circ \mathbf{h}^{-1} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \\ &= \mathbf{h} \circ \mathbf{g} \begin{pmatrix} \operatorname{atan2}(x_2, x_1) \\ \sqrt{x_1^2 + x_2^2} \end{pmatrix} \\ &= \mathbf{h} \circ \begin{pmatrix} \operatorname{atan2}(x_2, x_1) + p_1 \\ \frac{\sqrt{x_1^2 + x_2^2}}{\sqrt{p_2 + \left(x_1^2 + x_2^2\right)(1-p_2)}} \end{pmatrix} \\ &= \frac{\sqrt{x_1^2 + x_2^2}}{\sqrt{p_2 + \left(x_1^2 + x_2^2\right)(1-p_2)}} \begin{pmatrix} \cos\left(\operatorname{atan2}(x_2, x_1) + p_1\right) \\ \sin\left(\operatorname{atan2}(x_2, x_1) + p_1\right) \end{pmatrix} \\ &= \frac{1}{\sqrt{p_2 + \left(x_1^2 + x_2^2\right)(1-p_2)}} \cdot \mathbf{R}_{p_1} \cdot \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}. \end{aligned} \tag{32}$$

Figure 5 shows the vector field associated with our system. In the approach, that we will use later for our guaranteed integration, we first need to compute numerically one solution: the reference, painted red. On this reference we are able to apply $\mathbf{\Phi}_t$, the flow $\mathbf{\Phi}$ for a time t. To apply $\mathbf{\Phi}_t$ somewhere else, we apply a transformation $\mathbf{g}^{-1}$ to move on the reference, then we apply $\mathbf{\Phi}_t$ along this reference, and we finally move back using $\mathbf{g}$. This operation corresponding to Equation (11) is represented by the two green paths. These paths $(\mathbf{a}_i, \mathbf{b}_i, \mathbf{c}_i, \mathbf{d}_i)$, $i \in \{1, 2\}$ allow us to move from $\mathbf{a}_i$ to $\mathbf{d}_i = \mathbf{\Phi}_t(\mathbf{a}_i)$ using the symmetry $\mathbf{g}_i$ and the flow $\mathbf{c}_i = \mathbf{\Phi}_t(\mathbf{b}_i)$. Here we have $n = 2$ different strategies to move from any point to the reference trajectory (*i.e.*, either we rotate by the angle p_1 or we inflate by p_2), whereas $n - 1$ are sufficient in general, since we can always benefit of the one-parameter flow symmetry (see Example 8).

Transport function. We rewrite $\mathbf{g_p}(\mathbf{a}) = \mathbf{x}$ into the form $\mathbf{p} = \mathbf{h}(\mathbf{x}, \mathbf{a})$. We have

$$\begin{aligned} &\mathbf{g_p}(\mathbf{a}) = \mathbf{x} \\ \Leftrightarrow\ &\frac{1}{\sqrt{p_2 + \|\mathbf{a}\|^2(1-p_2)}} \cdot \mathbf{R}_{p_1} \cdot \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}. \end{aligned} \tag{33}$$

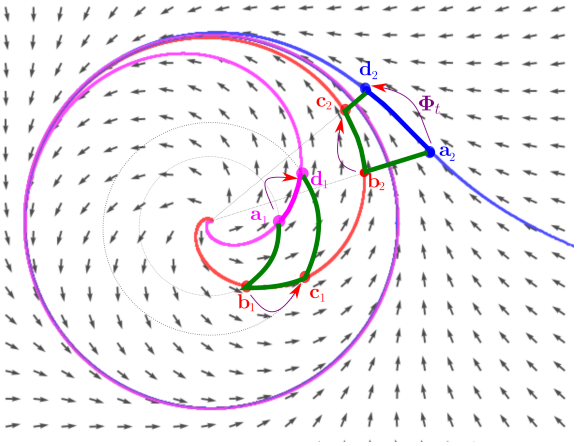

Fig. 5. Illustration of Test-case 3. The symmetry group transforms one trajectory into another.

Thus

$$p_1 = \text{atan2}(a_1x_2 - a_2x_1, a_1x_1 + a_2x_2).$$

Moreover

$$\frac{1}{p_2 + \|\mathbf{a}\|^2(1-p_2)}\|\mathbf{a}\|^2 = \|\mathbf{x}\|^2$$

i.e.

$$p_2 = \frac{\|\mathbf{a}\|^2 - \|\mathbf{x}\|^2\|\mathbf{a}\|^2}{\|\mathbf{x}\|^2 - \|\mathbf{x}\|^2\|\mathbf{a}\|^2}. \tag{34}$$

Finally

$$\mathbf{h}(\mathbf{x},\mathbf{a}) = \begin{pmatrix} \text{atan2}(a_1x_2 - a_2x_1, a_1x_1 + a_2x_2) \\ \frac{1}{1-\|\mathbf{a}\|^2}\left(\frac{\|\mathbf{a}\|^2}{\|\mathbf{x}\|^2} - \|\mathbf{a}\|^2\right) \end{pmatrix}. \tag{35}$$

3.6 Test-case 4

We now consider a robotic test-case (see e.g., [16] or [27], Chapter 4) where the system is the Dubins car defined as:

$$\begin{cases} \dot{x}_1 &= u_1 \cdot \cos x_3 \\ \dot{x}_2 &= u_1 \cdot \sin x_3 \\ \dot{x}_3 &= u_2 \end{cases} \tag{36}$$

where u_1, u_2 are time dependent. To avoid the time dependency in $\mathbf{u}$, we rewrite the system into

$$\begin{cases} \dot{x}_1 &= u_1(x_4) \cdot \cos x_3 \\ \dot{x}_2 &= u_1(x_4) \cdot \sin x_3 \\ \dot{x}_3 &= u_2(x_4) \\ \dot{x}_4 &= 1 \end{cases} \tag{37}$$

where x_4 is the clock variable.

Proposition 13 *The following transformations*

$$\mathbf{g_p}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} + \mathbf{R}_{p_3} \cdot \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \\ x_3 + p_3 \\ x_4 \end{pmatrix} \circ \mathbf{\Phi}_{p_4}(\mathbf{x}) \tag{38}$$

where $\mathbf{R}_{p_3}$ is the rotation matrix previously given in Equation (26) are Lie symmetries for System (37).

It corresponds to two transformations: the translation related to (p_1, p_2) and the rotation linked to p_3. The symmetry $\mathbf{g_p}$ corresponds to the direct Euclidean group $SE(2)$, well known in geometry.

Proof: If we define

$$\mathbf{g}^1_\mathbf{p}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} p_1 + x_1 \\ p_2 + x_2 \\ x_3 \\ x_4 \end{pmatrix}, \tag{39}$$

$$\mathbf{g}^2_\mathbf{p}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} \mathbf{R}_{p_3} \cdot \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \\ x_3 + p_3 \\ x_4 \end{pmatrix} \tag{40}$$

and $\mathbf{g}^3_\mathbf{p} = \mathbf{\Phi}_{p_4}(\mathbf{x})$ then we have $\mathbf{g_p} = \mathbf{g}^1_\mathbf{p} \circ \mathbf{g}^2_\mathbf{p} \circ \mathbf{g}^3_\mathbf{p}$. We thus have to check that $\mathbf{g}^1_\mathbf{p}$, $\mathbf{g}^2_\mathbf{p}$ and $\mathbf{g}^3_\mathbf{p}$ are stabilizers (see (i), (ii), (iii) below).

(i) We check Equation (9) for $\mathbf{g}^1_\mathbf{p}$:

$$\begin{aligned} &\underbrace{\mathbf{I}_{4\times 4}}_{\frac{d\mathbf{g}^1_\mathbf{p}}{d\mathbf{x}}(\mathbf{x})} \cdot \underbrace{\begin{pmatrix} u_1(x_4)\cdot\cos x_3 \\ u_1(x_4)\cdot\sin x_3 \\ u_2(x_4) \\ x_4 \end{pmatrix}}_{\mathbf{f}(\mathbf{x})} \\ = &\underbrace{\begin{pmatrix} u_1(x_4)\cdot\cos x_3 \\ u_1(x_4)\cdot\sin x_3 \\ u_2(x_4) \\ x_4 \end{pmatrix}}_{\mathbf{f}(\mathbf{x})} \circ \underbrace{\begin{pmatrix} p_1 + x_1 \\ p_2 + x_2 \\ x_3 \\ x_4 \end{pmatrix}}_{\mathbf{g}^1_\mathbf{p}(\mathbf{x})} \end{aligned} \tag{41}$$

which is true.

(ii) We check Equation (9) for $\mathbf{g}_{\mathbf{p}}^2$:

$$\underbrace{\begin{pmatrix} \mathbf{R}_{p_3} & \mathbf{0}_{2\times 2} \\ \mathbf{0}_{2\times 2} & \mathbf{I}_{2\times 2} \end{pmatrix}}_{\frac{d\mathbf{g}_{\mathbf{p}}^2}{d\mathbf{x}}(\mathbf{x})} \cdot \underbrace{\begin{pmatrix} u_1(x_4)\cdot\cos x_3 \\ u_1(x_4)\cdot\sin x_3 \\ u_2(x_4) \\ 1 \end{pmatrix}}_{\mathbf{f}(\mathbf{x})}$$
$$= \underbrace{\begin{pmatrix} u_1(x_4)\cdot\cos x_3 \\ u_1(x_4)\cdot\sin x_3 \\ u_2(x_4) \\ 1 \end{pmatrix}}_{\mathbf{f}(\mathbf{x})} \circ \underbrace{\begin{pmatrix} \cos p_3\cdot x_1 - \sin p_3\cdot x_2 \\ \sin p_3\cdot x_1 + \cos p_3\cdot x_2 \\ x_3 + p_3 \\ x_4 \end{pmatrix}}_{\mathbf{g}_{\mathbf{p}}^2(\mathbf{x})}$$

i.e.,

$$\begin{pmatrix} \mathbf{R}_{p_3} & \mathbf{0}_{2\times 2} \\ \mathbf{0}_{2\times 2} & \mathbf{I}_{2\times 2} \end{pmatrix} \cdot \begin{pmatrix} u_1(x_4)\cdot\cos x_3 \\ u_1(x_4)\cdot\sin x_3 \\ u_2(x_4) \\ 1 \end{pmatrix} = \begin{pmatrix} u_1(x_4)\cdot\cos(x_3+p_3) \\ u_1(x_4)\cdot\sin(x_3+p_3) \\ u_2(x_4) \\ 1 \end{pmatrix} \quad (42)$$

which is true.

(iii) From Example 8, we know that $\mathbf{g}_{\mathbf{p}}^3$ is a stabilizer.

Transport function. To get the transport function, we rewrite $\mathbf{g}_{\mathbf{p}}(\mathbf{a}) = \mathbf{x}$ into the form $\mathbf{p} = \mathbf{h}(\mathbf{x}, \mathbf{a})$:

$$\mathbf{g}_{\mathbf{p}}(\mathbf{a}) = \mathbf{x}$$
$$\overset{(38)}{\Leftrightarrow} \begin{cases} \begin{pmatrix} \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} + \mathbf{R}_{p_3}\cdot\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \\ y_3 + p_3 \\ y_4 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} \\ \mathbf{y} = \mathbf{\Phi}_{p_4}(\mathbf{a}) \end{cases} \quad (43)$$

Now, we have $x_4 = y_4 = \phi_{p_4,4}(\mathbf{a}) = a_4 + p_4$, due to the fact that x_4 corresponds to the time and thus $p_4 = x_4 - a_4$. We deduce that $\mathbf{g}_{\mathbf{p}}(\mathbf{a}) = \mathbf{x}$ is equivalent to

$$\begin{pmatrix} \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} + \mathbf{R}_{p_3}\cdot\begin{pmatrix} \phi_{p_4,1}(\mathbf{a}) \\ \phi_{p_4,2}(\mathbf{a}) \end{pmatrix} \\ p_3 \\ p_4 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 - \phi_{p_4,3}(\mathbf{a}) \\ x_4 - a_4 \end{pmatrix}$$

i.e.,

$$\begin{pmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{pmatrix} = \mathbf{x} - \underbrace{\begin{pmatrix} \mathbf{R}_{x_3-\phi_{x_4-a_4,3}(\mathbf{a})}\cdot\begin{pmatrix} \phi_{x_4-a_4,1}(\mathbf{a}) \\ \phi_{x_4-a_4,2}(\mathbf{a}) \end{pmatrix} \\ \phi_{x_4-a_4,3}(\mathbf{a}) \\ a_4 \end{pmatrix}}_{=\mathbf{h}(\mathbf{x},\mathbf{a})} \quad (44)$$

4 Integration method

The tools presented so far can be used in the context of guaranteed integration. We will see how the use of Lie symmetries can enhance classical methods of interval integration. For all test-cases presented below, we used a Processor Intel I7 8650U @ 1.90GHz.

4.1 Problem

Consider the system

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) \quad (45)$$

where the initial vector $\mathbf{x}_0$ is known to be inside the box $[\mathbf{x}_0]$. Define by $\mathbb{X}_t$ the set of all states $\mathbf{x}$ at time t consistent with the initial box $[\mathbf{x}_0]$ and with Equation (45). In this section, we want to characterize the sets $\mathbb{X}_t$ for $t \in \mathbb{T}$. The set $\mathbb{T}$ may be discrete, $\mathbb{T} = \{t_1, t_2, \ldots, t_m\}$ or an interval $\mathbb{T} = [0, t_{\max}]$.

Proposition 14 *If $\mathbf{\Phi}_t$ is the flow associated to $\mathbf{f}$, then we have*

$$\mathbb{X}_t = \mathbf{\Phi}_{-t}^{-1}([\mathbf{x}_0]). \quad (46)$$

Proof: We have

$$\begin{aligned} \mathbf{x} \in \mathbb{X}_t &\Leftrightarrow \exists \mathbf{x}_0 \in [\mathbf{x}_0],\ \mathbf{x} = \mathbf{\Phi}_t(\mathbf{x}_0) \\ &\Leftrightarrow \exists \mathbf{x}_0 \in [\mathbf{x}_0],\ \mathbf{x}_0 = \mathbf{\Phi}_{-t}(\mathbf{x}) \\ &\Leftrightarrow \mathbf{\Phi}_{-t}(\mathbf{x}) \in [\mathbf{x}_0] \\ &\Leftrightarrow \mathbf{x} \in \mathbf{\Phi}_{-t}^{-1}([\mathbf{x}_0]) \end{aligned} \quad (47)$$

which concludes the proof.

As a consequence, characterizing $\mathbb{X}_t$ is a set inversion problem. If we have an inclusion function for $\mathbf{\Phi}_{-t}^{-1}$ when t is set, then we can get an inner and an outer approximation for the sets $\mathbb{X}_t$, $t \in \mathbb{T}$, using a set inversion algorithm such as SIVIA [29]. This formulation allows us to get an inner approximation in a much simpler way than the approach proposed in [22].

4.2 Getting an inclusion function for the flow

Let $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$ be a state equation and $\mathbf{\Phi}_t$ the corresponding flow. Assume that the initial vector $\mathbf{x}(0)$ is inside the box $[\mathbf{x}_0]$. Let $[t]$ be an interval containing the current time t. An interval integrator is an algorithm which computes in a finite number of steps, a box $[\mathbf{y}]$ such that

$$\forall t \in [t], \forall \mathbf{x}_0 \in [\mathbf{x}_0], \mathbf{\Phi}_t(\mathbf{x}_0) \in [\mathbf{y}]. \tag{48}$$

There exist in the literature several algorithms which provide such interval integrators such as [10] or [56]. Moreover, except for atypical situations (such as when $\mathbf{f}$ is not locally Lipschitz or $\mathbf{f}$ is non computable), these integrators are efficient and accurate, *i.e.*, when the diameters of $[t]$ and $[\mathbf{x}](0)$ are infinitely small, the diameter of the box $[\mathbf{y}]$ is also infinitely small. The enclosure is guaranteed even with a calculus made with floating point numbers. This guarantee is made possible thanks to interval computation and the Picard operator. One of the main efficient interval integrators are based on the Lohner algorithm [33].

In this section, we assume that we have an accurate interval enclosure $[\mathbf{a}](t)$ for a reference $\mathbf{a}(t) = \mathbf{\Phi}_t(\mathbf{a}_0)$ with $\mathbf{a}(0) = \mathbf{a}_0$. The reference satisfies Equation (45). Getting such an accurate tube $[\mathbf{a}](t)$ for the reference can be done for most dynamical systems since no uncertainty should be considered and since the initial point is known.

The following proposition will make it possible to get an efficient inclusion function for the flow.

Proposition 15 *If $\mathbf{h}(\mathbf{x}, \mathbf{a})$ is a transport function for $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$, then we have*

$$\mathbf{\Phi}_t(\mathbf{x}) = \mathbf{g}_{\mathbf{h}(\mathbf{x},\mathbf{a}_0)} \circ \mathbf{a}(t). \tag{49}$$

Proof: Since $\mathbf{g}_\mathbf{p}$ is a symmetry, we have for all $\mathbf{p}$,

$$\mathbf{\Phi}_t(\mathbf{x}) = \mathbf{g}_\mathbf{p} \circ \mathbf{\Phi}_t \circ \mathbf{g}_\mathbf{p}^{-1}(\mathbf{x}). \tag{50}$$

Taking $\mathbf{p} = \mathbf{h}(\mathbf{x}, \mathbf{a}_0)$ we get

$$\begin{aligned} \mathbf{\Phi}_t(\mathbf{x}) &= \mathbf{g}_{\mathbf{h}(\mathbf{x},\mathbf{a}_0)} \circ \mathbf{\Phi}_t \circ \mathbf{g}_{\mathbf{h}(\mathbf{x},\mathbf{a}_0)}^{-1}(\mathbf{x}) \\ &\overset{(16)}{=} \mathbf{g}_{\mathbf{h}(\mathbf{x},\mathbf{a}_0)} \circ \mathbf{\Phi}_t(\mathbf{a}_0) \\ &= \mathbf{g}_{\mathbf{h}(\mathbf{x},\mathbf{a}_0)} \circ \mathbf{a}(t). \end{aligned} \tag{51}$$

Corollary 16 *An inclusion function for $\mathbf{\Phi}_t(\mathbf{x})$ is thus*

$$[\mathbf{\Phi}]_{[t]}([\mathbf{x}]) = [\mathbf{g}]_{[\mathbf{h}]([\mathbf{x}],\mathbf{a}_0)}([\mathbf{a}]([t])) \tag{52}$$

where $[\mathbf{g}]$, $[\mathbf{h}]$ are inclusion functions for $\mathbf{g}$, $\mathbf{h}$ and $[\mathbf{a}](t)$ encloses the reference $\mathbf{a}(t)$. The point $\mathbf{a}_0$ is exactly known.

Proof: This claim is a consequence of Proposition 15 and of the fundamental theorem of interval arithmetic [34].

Figure 6 provides an illustration of the interval inclusion function $[\mathbf{\Phi}]_{[t]}([\mathbf{x}])$ which computes a box $[\mathbf{y}]$ enclosing $\mathbf{y} = \mathbf{\Phi}_t(\mathbf{x}), \mathbf{x} \in [\mathbf{x}], t \in [t]$. Note that we have a closed form expression for $\mathbf{g}_\mathbf{p}$ and for $\mathbf{h}$, and a thin enclosure of the reference $t \mapsto \mathbf{a}(t)$ under the form of a tube [43].

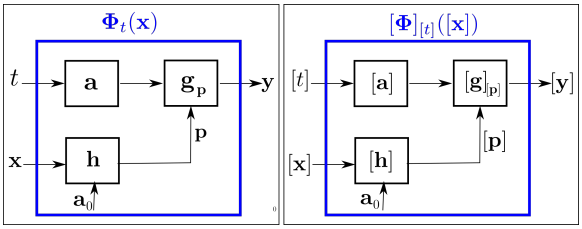

Fig. 6. Left: Construction of the function $\mathbf{\Phi}_t(\mathbf{x}) = \mathbf{g}_{\mathbf{h}(\mathbf{x},\mathbf{a}_0)}(\mathbf{a}(t))$; Right: its interval counterpart $[\mathbf{\Phi}]_{[t]}([\mathbf{x}]) = [\mathbf{g}]_{[\mathbf{h}]([\mathbf{x}],\mathbf{a}_0)}([\mathbf{a}]([t]))$

4.3 Method

Given a box $[\mathbf{x}](0)$ containing the initial state $\mathbf{x}_0$, we want to characterize the set of all feasible trajectories. To get the set $\mathbb{X}_{t_i}$ of all feasible states at time t_i, we propose the following method:

- Step 1. Define a reference $\mathbf{a}(t)$ and enclose it in a thin tube $[\mathbf{a}](t)$. The initial vector $\mathbf{a}_0$ is exactly known.
- Step 2. Find a Lie group of symmetries $G_\mathbf{p}$ and give an expression for the transport function $\mathbf{h}(\mathbf{x}, \mathbf{a})$.
- Step 3. Solve the set inversion problem of Equation (46) with SIVIA, using the inclusion of Equation (52).

For the implementation, we used Tubex [44, 43]. We first generate a precise tube $[\mathbf{a}](t)$ for $\mathbf{a}(t)$ by using CAPD [56], that is based on the Lohner algorithm [33].

We will now illustrate the approach on four examples. For each of these examples, we will compute the set

$$\bigcup_{t\in\mathbb{T}} \mathbb{X}_t = \bigcup_{t\in\mathbb{T}} \mathbf{\Phi}_{-t}^{-1}([\mathbf{x}_0]) \tag{53}$$

where $\mathbf{\Phi}_t(\mathbf{x}) = \mathbf{g}_{\mathbf{h}(\mathbf{x},\mathbf{a}_0)} \circ \mathbf{a}(t)$, and where $\mathbb{T}$ is either

- a discrete set $\mathbb{T} = \{1, \ldots, m\}$, which allows us to draw different non-overlapping sets $\mathbb{X}_{t_i}$ on the same picture;
- an interval $\mathbb{T} = [0, t_{\max}]$, in order to approximate forward reach sets.

Note that the union appearing in Equation (53), as well as other set-theoretical operators (such as the projection), can easily be done using separator algebra [28], also based on interval analysis.

4.4 Test-case 1 (continued)

This is the continuation of the example considered in Subsection 2.3.

- Step 1. An analytical expression for a reference $\mathbf{a}(t)$ can be obtained for Equation (15). We take the initial condition $\mathbf{a}_0 = (0, 1)^\intercal$ and we get:
$$\mathbf{a}(t) = \begin{pmatrix} t \\ e^{-t} \end{pmatrix}. \tag{54}$$
- Step 2. The transport function is given by Equation (18).
- Step 3. From Proposition 15, we get
$$\begin{aligned} \mathbf{\Phi}_t(\mathbf{x}) &= \mathbf{g}_{\mathbf{h}(\mathbf{x},\mathbf{a}_0)} \circ \mathbf{a}(t) \\ &= \mathbf{g}_{(x_1,x_2)} \circ \begin{pmatrix} t \\ e^{-t} \end{pmatrix} \\ &= \begin{pmatrix} t + x_1 \\ x_2 \cdot e^{-t} \end{pmatrix}. \end{aligned} \tag{55}$$

For $[\mathbf{x}_0] = [0, 1] \times [2, 3]$ and for $t \in \mathbb{T} = \{1, 2, 3, 4, 5, 6\}$, SIVIA yields Figure 7 in 1.4 sec. For $\mathbb{T} = [0, 6]$, it generates Figure 8 in 4.6 sec. Both represent the set $\bigcup_{t\in\mathbb{T}} \mathbb{X}_t$. The green box represents the initial box $[\mathbf{x}_0]$. The magenta boxes are all inside the solution set and the blue boxes are all outside. The black trajectory corresponds to an accurate enclosure $t \mapsto [\mathbf{a}](t)$ for the reference $t \mapsto \mathbf{a}(t)$.

4.5 Test-case 2 (continued)

This is the continuation of the example considered in Subsection 3.4.

- Step 1. The reference $t \mapsto \mathbf{a}(t)$ is obtained analytically. For $\mathbf{a}_0 = (0, 0)^\intercal$, we have:
$$\begin{cases} a_1(t) = t \\ a_2(t) = 1 - \cos t \end{cases} \tag{56}$$
- Step 2. The transport function is given by Equation (23).

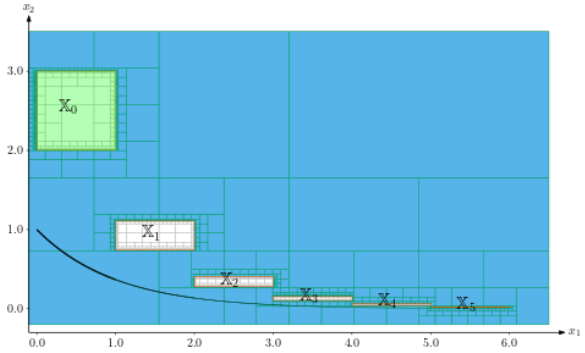

Fig. 7. Result of Test-case 1 for a discrete time set: $\bigcup_{t\in\{0,1,2,3\}} \mathbb{X}_t$.

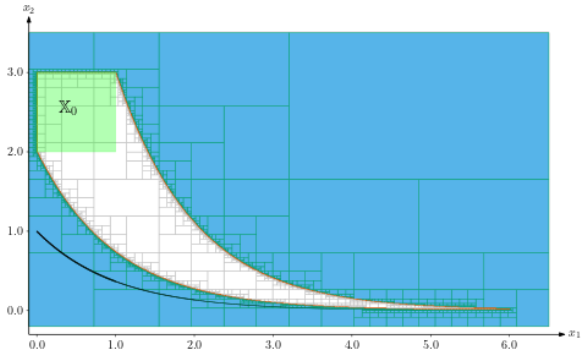

Fig. 8. Result of Test-case 1 for a continuous time set: $\bigcup_{t\in[0,5]} \mathbb{X}_t$

- Step 3. Since $\mathbf{\Phi}_t(\mathbf{0}) = \mathbf{a}(t)$, we have $\mathbf{\Phi}_{x_1}(\mathbf{0}) = \mathbf{a}(x_1)$. Thus
$$\mathbf{h}(\mathbf{x}, \mathbf{0}) \overset{(23)}{=} \begin{pmatrix} x_2 - \phi_{x_1,2}(\mathbf{0}) \\ x_1 \end{pmatrix} = \begin{pmatrix} x_2 - a_2(x_1) \\ x_1 \end{pmatrix}.$$

Moreover,

$$\mathbf{g}_\mathbf{p}(\mathbf{x}) = \begin{pmatrix} 0 \\ p_1 \end{pmatrix} + \mathbf{\Phi}_{p_2}(\mathbf{x}). \tag{57}$$

Therefore:

$$\begin{aligned} \mathbf{g}_\mathbf{p} \circ \mathbf{a}(t) &\overset{(20)}{=} \begin{pmatrix} 0 \\ p_1 \end{pmatrix} + \mathbf{\Phi}_{p_2}(\mathbf{a}(t)) \\ &= \begin{pmatrix} 0 \\ p_1 \end{pmatrix} + \mathbf{a}(t + p_2) \\ &= \begin{pmatrix} a_1(t + p_2) \\ p_1 + a_2(t + p_2) \end{pmatrix}, \end{aligned} \tag{58}$$

thus:

$$\begin{aligned}\mathbf{\Phi}_t(\mathbf{x}) &= \mathbf{g}_{\mathbf{h}(\mathbf{x},\mathbf{0})} \circ \mathbf{a}(t) \\ &= \begin{pmatrix} a_1(t+x_1) \\ x_2 - a_2(x_1) + a_2(t+x_1) \end{pmatrix} \\ &= \begin{pmatrix} t + x_1 \\ x_2 + \cos x_1 - \cos(t+x_1) \end{pmatrix}. \end{aligned} \tag{59}$$

Figure 9 shows the set of feasible states at times $t \in \{2,4,6,8\}$ and Figure 10 shows the forward reach set for $t \in [0,8]$. The initial box is $[\mathbf{x}_0] = [-0.5, 0.5]^2$ (green painted). The computing time is 10.6 seconds.

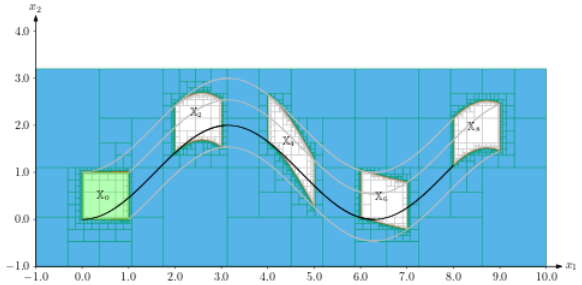

Fig. 9. Result of Test-case 2 for a discrete time set: $\bigcup_{t\in\{2,4,6,8\}} \mathbb{X}_t$

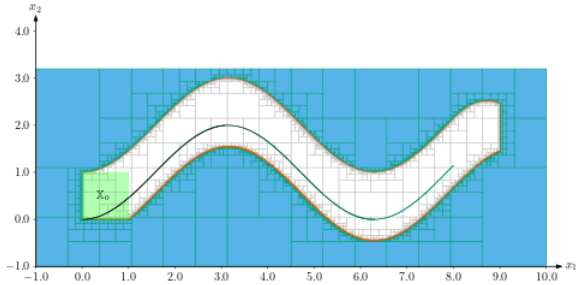

Fig. 10. Result of Test-case 2 for a continuous time set: $\bigcup_{t\in[0,8]} \mathbb{X}_t$

4.6 Test-case 3 (continued)

This is the continuation of the example considered in Subsection 3.5.

- Step 1. The reference $t \mapsto \mathbf{a}(t)$ is obtained by integrating System (24) from the initial vector $\mathbf{a}_0 = (1/2, 0)^\intercal$.
- Step 2. The transport function is given by Equation (35).
- Step 3. We have

$$\mathbf{\Phi}_t(\mathbf{x}) = \mathbf{g}_{\mathbf{h}(\mathbf{x},\mathbf{a}_0)} \circ \mathbf{a}(t)$$

If we develop this expression using Proposition 15 we get

$$\mathbf{\Phi}_t(\mathbf{x}) = \frac{\begin{pmatrix} x_1 & -x_2 \\ x_2 & x_1 \end{pmatrix} \begin{pmatrix} a_1^0 & a_2^0 \\ -a_2^0 & a_1^0 \end{pmatrix} \cdot \mathbf{a}(t)}{\|\mathbf{a}_0\|^2 \sqrt{\frac{1-\|\mathbf{a}(t)\|^2}{1-\|\mathbf{a}_0\|^2} + \left(\frac{\|\mathbf{a}(t)\|^2}{\|\mathbf{a}_0\|^2} - \frac{1-\|\mathbf{a}(t)\|^2}{1-\|\mathbf{a}_0\|^2}\right)\|\mathbf{x}\|^2}}$$

i.e., for $\mathbf{a}_0 = (1/2, 0)^\intercal$:

$$\mathbf{\Phi}_t(\mathbf{x}) = \frac{\sqrt{3}\begin{pmatrix} x_1 & -x_2 \\ x_2 & x_1 \end{pmatrix} \cdot \mathbf{a}(t)}{\sqrt{1 - \|\mathbf{a}(t)\|^2 + (4\|\mathbf{a}(t)\|^2 - 1)\|\mathbf{x}\|^2}}$$

To illustrate the integration, we have taken the example already considered by [13] where the initial set is a disk with center $(1.5, 1.5)$ and radius 0.2. Figure 11 shows the set of feasible states at times $t \in \{0,1,2,3,4,5\}$ and Figure 12 shows the forward reach set for $t \in [0,6]$. Again, the magenta boxes are all inside the solution set and the blue are all outside. The black trajectory corresponds to the reference $t \mapsto \mathbf{a}(t)$. The computing time is less than 5sec for the discrete time case and 130 seconds for continuous time case. Compare to the results given in [13] our results are much more accurate, we have both an inner and an outer approximation (and not only an outer approximation), and the computing time is less (130sec instead of 250 sec).

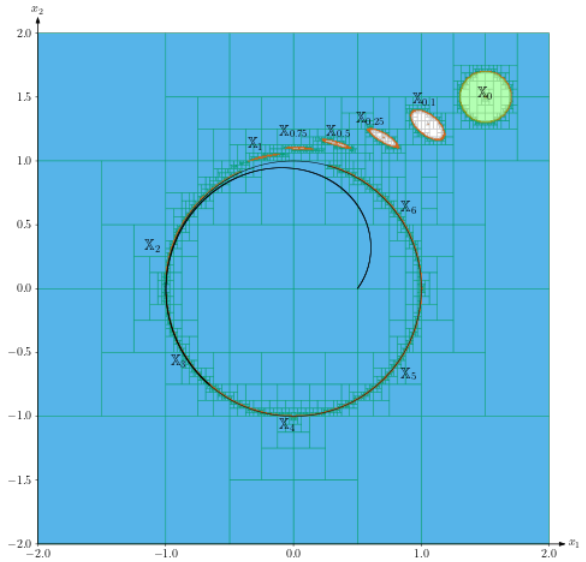

Fig. 11. Result of Test-case 3 for a discrete time set: $\bigcup_{t\in\{0,5\}} \mathbb{X}_t$

4.7 Test-case 4 (continued)

This is the continuation of the example considered in Subsection 3.6.

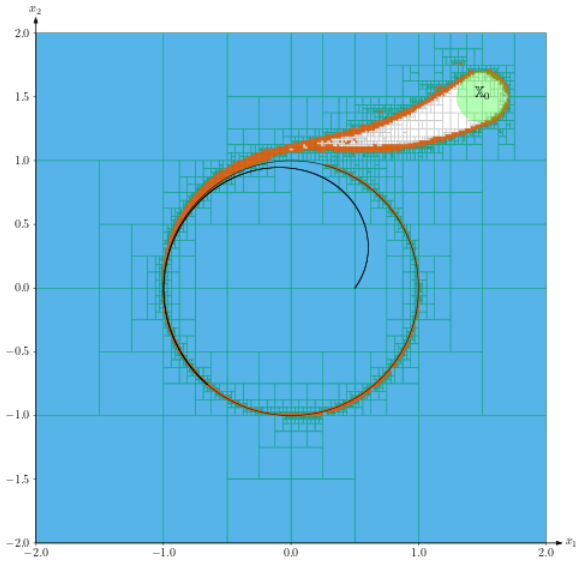

Fig. 12. Result of Test-case 3 for a continuous time set: $\bigcup_{t\in[0,6]} \mathbb{X}_t$

- Step 1. The reference $t \mapsto \mathbf{a}(t)$ is obtained by integrating System (37) from $\mathbf{a}_0 = (0,0,0,0)^\intercal$.
- Step 2. The transport function is given by Equation (44).
- Step 3. Since $\mathbf{\Phi}_t(\mathbf{0}) = \mathbf{a}(t)$, we have $\mathbf{\Phi}_{x_4}(\mathbf{0}) = \mathbf{a}(x_4)$. Thus:

$$\mathbf{h}(\mathbf{x},\mathbf{0}) \overset{(44)}{=} \mathbf{x} - \begin{pmatrix} \mathbf{R}_{x_3-a_3(x_4)} \cdot \begin{pmatrix} a_1(x_4) \\ a_2(x_4) \end{pmatrix} \\ a_3(x_4) \\ 0 \end{pmatrix}. \quad (60)$$

Moreover,

$$\mathbf{g_p} \circ \mathbf{a}(t) \overset{(38)}{=} \begin{pmatrix} \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} + \mathbf{R}_{p_3} \cdot \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \\ x_3 + p_3 \\ x_4 \end{pmatrix} \circ \underbrace{\mathbf{\Phi}_{p_4}(\mathbf{a}(t))}_{=\mathbf{a}(t+p_4)}$$
$$= \begin{pmatrix} \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} + \mathbf{R}_{p_3} \cdot \begin{pmatrix} a_1(t+p_4) \\ a_2(t+p_4) \end{pmatrix} \\ a_3(t+p_4) + p_3 \\ a_4(t+p_4) \end{pmatrix}.$$

Thus:

$$\begin{aligned} & \mathbf{\Phi}_t(\mathbf{x}) \\ = \; & \mathbf{g}_{\mathbf{h}(\mathbf{x},\mathbf{0})} \circ \mathbf{a}(t) \\ = \; & \begin{pmatrix} \begin{pmatrix} h_1(\mathbf{x},\mathbf{0}) \\ h_2(\mathbf{x},\mathbf{0}) \end{pmatrix} + \mathbf{R}_{h_3(\mathbf{x},\mathbf{0})} \cdot \begin{pmatrix} a_1(t+h_4(\mathbf{x},\mathbf{0})) \\ a_2(t+h_4(\mathbf{x},\mathbf{0})) \end{pmatrix} \\ a_3(t+h_4(\mathbf{x},\mathbf{0})) + h_3(\mathbf{x},\mathbf{0}) \\ a_4(t+h_4(\mathbf{x},\mathbf{0})) \end{pmatrix} \\ = \; & \begin{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \mathbf{R}_{x_3-a_3(x_4)} \cdot \begin{pmatrix} a_1(t+x_4) - a_1(x_4) \\ a_2(t+x_4) - a_2(x_4) \end{pmatrix} \\ x_3 + a_3(t+x_4) - a_3(x_4) \\ a_4(t+x_4) \end{pmatrix}. \end{aligned}$$

Figures 13 shows the projection of the set of feasible states at times $t \in \{1,2,\ldots,14\}$ on the (x_1,x_2) space. Figure 14 depicts the projection of the forward reach set for $t \in [0,14]$. The initial box is $[\mathbf{x}_0] = [-0.1,0.1]^2 \times [-0.4,0.4] \times [0,0]$ (green painted). The magenta boxes are proved to be all inside the projection while the blue ones are all outside. The thin trajectory corresponds to the reference $t \mapsto \mathbf{a}(t)$. The computing time is less than 750 sec.

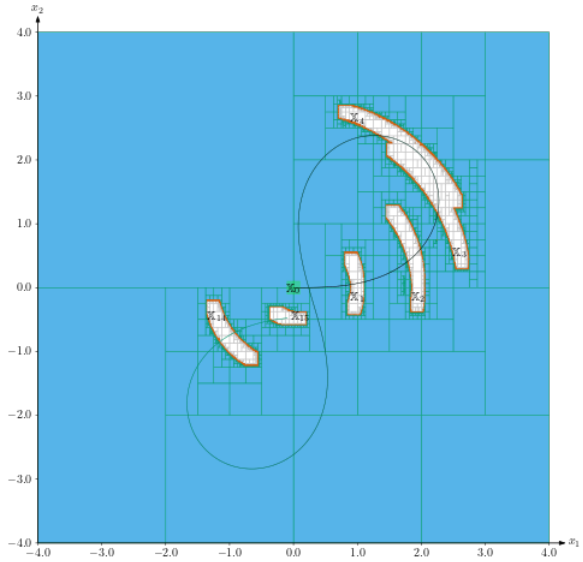

Fig. 13. Result of Test-case 4 for a discrete time set: Proj $_{(x_1,x_2)} \bigcup_{t\in\{0,14\}} \mathbb{X}_t$.

In this section, we have provided four test-cases to illustrate our approach. The first two examples can be

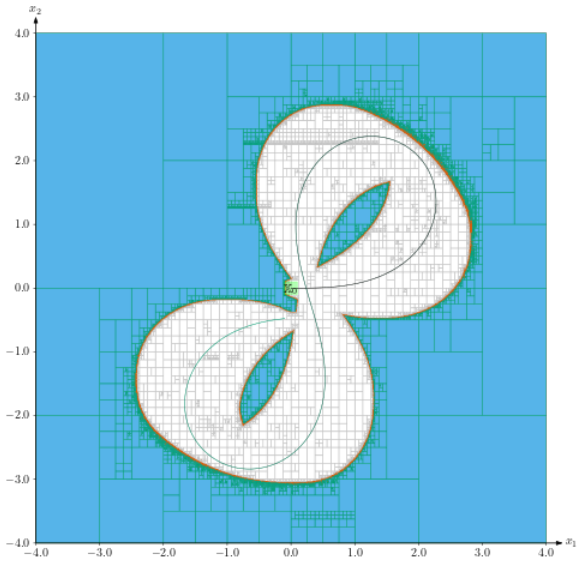

Fig. 14. Result of Test-case 4 for a continuous time set: Proj $\bigcup_{(x_1,x_2)t\in[0,14]} \mathbb{X}_t$.

solved analytically and are given for ease of understanding. Examples 3 and 4 are more complex and we used the interval integration solver CAPD [56], but we could have used V-Node [37] or DynIbex [47] instead. Due to the initial uncertainty, the fast explosion of the size of the tubes makes these solvers inefficient, when they are not combined with Lie symmetries as proposed here.

Remark. Some details related to the computation of all test-cases are given by the following table.

Test-case	Comp. time	Bisections	Accuracy
1 (continuous)	229ms	523	0.1
1 (discrete)	70ms	435	0.1
2 (continuous)	502ms	1288	0.1
2 (discrete)	106ms	797	0.1
3 (continuous)	130s	3112	0.01
3 (discrete)	5s	1807	0.01
4 (continuous)	682s	21435	0.01
4 (discrete)	11.9s	8677	0.01

5 Reachability

In this section, we illustrate how to use our interval integration in order to solve reachability problems [5][2][12]. Previously, we have have computed an inner approximation and an outer approximation of the reachable set and bisections on the state space was needed. Without such bisections, a tube similar to that represented on Figure 15 is obtained is few seconds. The tube goes from yellow for $t = 0$ to magenta.

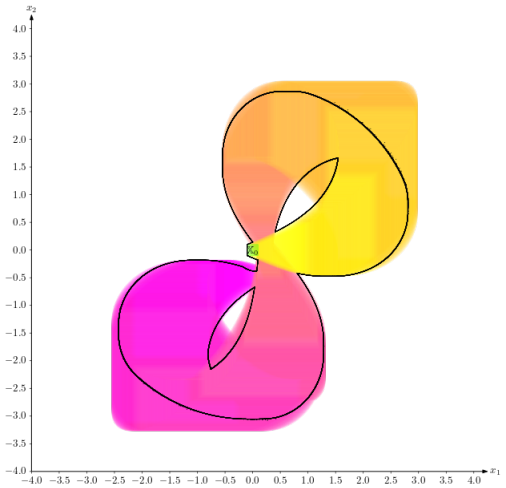

Fig. 15. Tube obtained from the interval integration of the small green box

Now, often in the context of reachability, we want to check that the system will reach a given zone or that system will never enter in some forbidden region $\mathcal{R}$. The reachability problem is transformed into a constraint satisfaction problem (CSP). The unsatisfiability of the CSP should be equivalent than checking the reachability problem.

To illustrate the approach, we consider one more Test-case 4. Figure 16 represents the area $\mathcal{G}$ that we want to reach (green disk with center $(0.1, 1)$ and radius $\sqrt{0.75}$) and the forbidden zone $\mathcal{R}$ (red disk with center $(1.2, 1.3)$ and radius , rayon 0.1). The black curve corresponds to the boundary of the set represented on Figure 14. Three constraints are involved

$$\begin{aligned} &\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}),\ \mathbf{x}(0) \in [\mathbf{x}_0] \quad (i)\\ &\forall t \in [0, 15], \mathbf{x}(t) \notin \mathcal{G} \quad (ii)\\ &\exists t \in [0, 15], \mathbf{x}(t) \in \mathcal{R} \quad (iii) \end{aligned}$$

- *(i)* claims that the system is initialized in the box $[\mathbf{x}_0]$ and that it satisfies the required state equation
- *(ii)* claims that the trajectory avoids the area to be reached
- *(iii)* claims that the trajectory enters the forbidden region

Using a contractor based approach, we get that there we have no solution for Constraints (i) and (ii). We thus conclude that $\mathcal{G}$ will always be reached for some $t \in [0, 15]$.

Equivalently, Constraints (i) and (iii) are unsatisfiable simultaneously. We conclude that trajectories will never enter inside the forbidden region.

For both CSP, no bisection on the state space is required. The contractor based approach require to consider each constraint separately and contract the tubes accordingly. The contraction process has to be done on all constraints of the CSP until the empty solution set is obtained. In the process our symmetry-based approach is used to contract the tube for $[\mathbf{x}](\cdot)$ with respect to Constraint *(i)*. The contraction had to be performed in a forward and a backward manner.

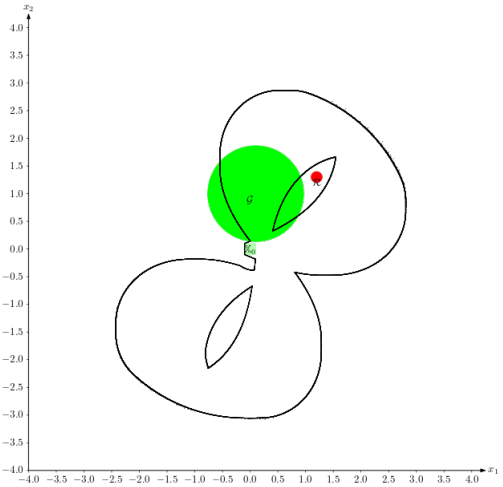

Fig. 16. We want to show that the green disk will be reached and that the small red disk will be avoided

6 Conclusion

This paper has presented a new approach for guaranteed interval integration of a state equation with uncertain initial conditions in a context of reachability analysis. The approach uses Lie groups of symmetries, which has never been done in the context of guaranteed integration. It allows us to extend existing methods to cases of large uncertainties. Compared to existing approaches, our method behaves very well when the initial state is strongly uncertain and when symmetries of the problem are known.

In a near future, we would like to explore the following perspectives.

- In the paper, the symmetries have been found by hand for each example. Symbolic methods [8] would probably be useful to find them automatically, at least for a large class of mobile robotic problems.
- When no symmetry exists in the formulation of the problem, our method does not apply. Now, the nonlinear invariance condition (9) could be replaced by equivalent but simpler linear conditions reflecting a form of infinitesimal invariance [39]. This infinitesimal formulation could be useful to deal with systems for which the symmetries exist but are not known. The combination with interval integration methods remains to be studied.
- In the paper, we have studied a forward propagation of the uncertainty. It seems straightforward to generalize the approach to allow both forward and backward propagation, as needed when we solve state estimation problems [19]. For such problems, existing methods perform the interval integration many times in both time directions. With our approach, the integration has to be done only once: for the reference $\mathbf{a}(t)$. Hence, we can expect efficient observers in a bounded-error context.
- Our Lie group based approach has been proposed in the context of interval analysis and bounded-error paradigm. In a probabilistic world, we often use a particle filter. Our strategy could probably be used to avoid redundant integrations for each particle. In this way, the integration is factorized once for all particles. We could then obtain much more efficient and accurate particle filters.

Available libraries

The Codac library [45] has been used during this work. It is a C++/Python library providing interval tools for constraint programming over reals, trajectories and sets. It has many applications in state estimation or robot localization. This framework is compatible with IBEX [9]: a C++ library for system solving and global optimization based on interval arithmetic and constraint programming, see http://www.ibex-lib.org. The code associated to all examples of this paper is given at: https://codac.io/lie-symmetries.

Acknowledgments

This work has been funded by Kopadia, a French company specialized in underwater systems engineering and operations. It has also been partially supported by the French Agence National de la Recherche (ANR) [grant number ANR-16-CE33-0024].

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.
[2] M. Althoff. An introduction to cora. *CPS Week*, pages 120–151, 2015.
[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.
[4] 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.

[5] 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.

[6] S. Bonnabel, P. Martin, and P. Rouchon. Symmetry-preserving observers. *IEEE Transactions on Automatic Control*, 53(11), 2008.

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

[8] J. Carminati and K. Vu. Symbolic computation and differential equations: Lie symmetries. *Journal of Symbolic Computation*, 29(1):95–116, 2018.

[9] G. Chabert. *IBEX 2.0, available at* , `http://www.emn.fr/z-info/ibex/`. Ecole des mines de Nantes, 2013.

[10] A. Chapoutot, J. Alexandre Dit Sandretto, and O. Mullier. Validated Explicit and Implicit Runge-Kutta Methods. In *Summer Workshop on Interval Methods*, June 2015.

[11] P. Chauchat, A. Barrau, and S. Bonnabel. Invariant smoothing on lie groups. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2018*, 2018.

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

[13] 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.

[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.

[15] V. Drevelle and P. Bonnifait. Localization confidence domains via set inversion on short-term trajectory. *IEEE Transactions on Robotics*, 2013.

[16] L. E. Dubins. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, 79(3):497–516, 1957.

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

[18] 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.

[19] A. Gning and P. Bonnifait. Constraints propagation techniques on intervals for a guaranteed localization using redundant data. *Automatica*, 42(7):1167–1175, 2006.

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

[21] M. Golubitsky and I. Stewart. *The symmetry perspective: from equilibrium to chaos in phase space and physical space*. Birkhauser, Berlin, 2003.

[22] E. Goubault, O. Mullier, S. Putot, and M. Kieffer. Inner approximated reachability analysis. In *Proceedings of the 17th international conference on Hybrid systems: computation and control, HSCC'14*, pages 163–172, Berlin, Germany, 2014.

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

[24] T. Hamel and R. Mahony. Attitude estimation on so(3) based on direct inertial measurements. In *Int. Conf. Robot. Automat. (ICRA 06)*, pages 2170–2175, 2006.

[25] P. E. Hydon. *Symmetry methods for differential equations, A beginner's guide*. Cambridge Texts in Applied Mathematics, Cambridge, 2000.

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

[27] L. Jaulin. *Mobile Robotics*. ISTE editions, 2015.

[28] L. Jaulin and B. Desrochers. Introduction to the algebra of separators with application to path planning. *Engineering Applications of Artificial Intelligence*, 33:141–147, 2014.

[29] L. Jaulin and E. Walter. Set inversion via interval analysis for nonlinear bounded-error estimation. *Automatica*, 29(4):1053–1064, 1993.

[30] 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.

[31] 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.

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

[33] 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.

[34] R. E. Moore. *Interval Analysis*. Prentice-Hall, Englewood Cliffs, NJ, 1966.

[35] R. E. Moore. *Methods and Applications of Interval Analysis*. SIAM, Philadelphia, PA, 1979.

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

[37] N. Nedialkov, K. Jackson, and G. Corliss. Validated solutions of initial value problems for ordinary differential equations. *Applied Mathematics and Computation*, 105(1):21–68, October 1999.

[38] P.J. Olver. *Classical Invariant Theory*. Graduate Texts in Mathematics. Univ. Press, Cambridge, U.K., 1999.

[39] P.J. Olver. *Applications of Lie Groups to Differential Equations*. Graduate Texts in Mathematics. Springer New York, 2012.

[40] 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.

[41] 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.

[42] 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.

[43] S. Rohou, L. Jaulin, L. Mihaylova, F. Le Bars, and S. Veres. *Reliable robot localization*. ISTE Group, 2019.

[44] S. Rohou, L. Jaulin, M. Mihaylova, F. Le Bars, and S. Veres. Guaranteed Computation of Robots Trajectories. *Robotics and Autonomous Systems*, 93:76–84, 2017.

[45] Simon Rohou, Benoit Desrochers, et al. The Codac library – Constraint-programming for robotics, 2022. http://codac.io.

[46] G. Russo and J.J. Slotine. Symmetries, stability, and control in nonlinear systems and networks. *Physical Review E*, 84(4), 2011.

[47] 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.

[48] J.P. Serre. *Lie Algebras and Lie Groups*. Springer, 1965.

[49] H. Sibai, N. Mokhlesi, and S. Mitra. Using symmetry transformations in equivariant dynamical systems for their safety verification. *Automated Technology for Verification and Analysis*, 2019.

[50] J. Starrett. Solving differential equations by symmetry groups. *The American Mathematical Monthly*, 114(9):778–792, 2007.

[51] R. Steinhou. *The Truth About Lie Symmetries: Solving Differential Equations With Symmetry Methods*. PhD dissertation, Senior Independent Study Theses, 2013.

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

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

[54] W. Tucker. The Lorenz Attractor Exists. *Comptes Rendus de l'Académie des Sciences*, 328(12):1197–1202, 1999.

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

[56] D. Wilczak and P. Zgliczynski. Cr-Lohner algorithm. *Schedae Informaticae*, 20:9–46, 2011.