Interval Reachability Analysis using Second-Order Sensitivity ⋆

Pierre-Jean Meyer * **Murat Arcak** *

* *Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, USA,* `{pjmeyer, arcak}@berkeley.edu`

Abstract: We propose a new approach to compute an interval over-approximation of the finite time reachable set for a large class of nonlinear systems. This approach relies on the notions of sensitivity matrices, which are the partial derivatives representing the variations of the system trajectories in response to variations of the initial states. Using interval arithmetics, we first over-approximate the possible values of the second-order sensitivity at the final time of the reachability problem. Then we exploit these bounds and the evaluation of the first-order sensitivity matrices at a few sampled initial states to obtain an over-approximation of the first-order sensitivity, which is in turn used to over-approximate the reachable set of the initial system. Unlike existing methods relying only on the first-order sensitivity matrix, this new approach provides guaranteed over-approximations of the first-order sensitivity and can also provide such over-approximations with an arbitrary precision by increasing the number of samples.

Keywords: Reachability analysis, mixed-monotonicity, sensitivity, interval.

1. INTRODUCTION

Reachability analysis is the problem of evaluating the set of all the successor states that can be reached in finite time by a system starting from a given set of initial states (Blanchini and Miani, 2008). Since the reachable set can rarely be computed exactly, we often rely on methods to over-approximate this set. In the literature, we primarily find two classes of reachability approaches. The first class considers complex and flexible set representations, such as zonotopes (Althoff, 2015), zonotope bundles (Althoff and Krogh, 2011) ellipsoids (Kurzhanskiy and Varaiya, 2007), support functions (Girard and Le Guernic, 2008), paving of intervals (Jaulin, 2001). Their main focus is to over-approximate the reachable set as tightly as possible, which is particularly interesting to solve simple verification problems such as those with safety or reachability specifications where the obtained over-approximation is immediately checked against a set of unsafe or target states.

The second class considers a simpler set representation in the form of (multi-dimensional) intervals, using methods based on differential inequalities (Scott and Barton, 2013), Taylor models (Chen et al., 2012), growth bounds (Reissig et al., 2016) or monotonicity (Meyer et al., 2019). Due to the simpler set representation, these methods tend to offer better efficiency and scalability at the cost of the accuracy of the over-approximations, and are thus particularly used in the field of abstraction-based control synthesis (see e.g. Moor and Raisch, 2002; Coogan and Arcak, 2015; Reissig et al., 2016; Meyer and Dimarogonas, 2019) where the number of reachable set over-approximations required for the creation of an abstraction grows exponentially in the dimension of the state space.

In the subset of monotonicity-based interval reachability approach, the simplest method, used in Moor and Raisch (2002), relies directly on a monotonicity property (Angeli and Sontag, 2003) and guarantees that an interval over-approximation of the reachable set can be computed by evaluating the successors of only two vertices of the interval of initial states. A generalization of this property called mixed-monotonicity was then introduced and used for reachability analysis in Coogan and Arcak (2015), where an auxiliary monotone system can be created by decomposing the initial system into its increasing and decreasing components. A further generalization of mixed-monotonicity to any system with a bounded Jacobian matrix was recently proposed in Yang et al. (2019) and used for reachability analysis in Meyer and Dimarogonas (2019). Finally, another interval reachability method inspired by the notion of mixed-monotonicity and applicable to continuous-time nonlinear systems was proposed in Meyer et al. (2018), where bounds on the sensitivity matrix (the partial derivative describing the influence of initial conditions on the successor states) are used to compute an over-approximation interval of the reachable set.

While Meyer et al. (2018) considers two approaches to evaluate these sensitivity bounds, both have shortcomings: one provides very conservative bounds by applying the interval arithmetics results from Althoff et al. (2007), the other only computes empirical bounds through a time-consuming sampling procedure which is not guaranteed to result in an over-approximation of the sensitivity values. In this paper, we propose a novel and more flexible algorithm to obtain sensitivity bounds by combining the advantages of these two approaches while overcoming their main

⋆ This work was supported in part by the U.S. National Science Foundation grant ECCS-1906164, the U.S. Air Force Office of Scientific Research grant FA9550-18-1-0253 and the ONR grant N00014-18-1-2209.

drawbacks. In addition to the first-order sensitivity matrix used above, the proposed approach also relies on the second-order sensitivity in the following 3-step procedure:

- first over-approximate the reachable tube (over the whole time range) for the first-order sensitivity matrix using interval arithmetics,
- next use these bounds to over-approximate the reachable set (at the final time only) for the second-order sensitivity using interval arithmetics,
- finally combine the second-order sensitivity bounds with the numerical evaluation of the first-order sensitivity on some sampled initial states to obtain an over-approximation of the reachable set of the first-order sensitivity.

This result has two major advantages. Compared to the purely empirical sampling approach from Meyer et al. (2018), the proposed algorithm is sound since for any number of samples we are guaranteed to over-approximate the set of first-order sensitivity values. Compared to the one-step interval arithmetics method from Meyer et al. (2018), which is conservative, we can now obtain arbitrarily tight bounds of the first-order sensitivity by increasing the number of samples. Indeed, the sampling in our third step can be used to tune the desired tradeoff between the computational complexity and the conservativeness of the over-approximation. Compared to methods relying on Taylor models such as Chen et al. (2012) which usually require a decomposition of the time range to reduce the accumulation of errors, the proposed approach relying on mixed-monotonicity does not have this problem and all over-approximations can be computed in a single time step.

The paper is structured as follows. In Section 2, we provide the notations and mathematical preliminaries that are used throughout this paper. The considered reachability problem for a nonlinear system is defined in Section 3. In Section 4, we provide the definitions and equations describing the first-order and second-order sensitivity matrices. Section 5 presents the overall algorithm to solve the reachability problem. Finally, the proposed approach and its advantages compared to Meyer et al. (2018) are illustrated on a numerical example in Section 6.

2. PRELIMINARIES

2.1 Notations

Let $\mathbb{R}$ and $\mathbb{N}$ be the sets of reals and positive integers, respectively. Let $I_n \in \mathbb{R}^{n\times n}$ and $\mathbf{0}_{n\times p}, \mathbf{1}_{n\times p} \in \mathbb{R}^{n\times p}$ denote the identity matrix of dimension n and the $n \times p$ matrices filled with zeros and ones, respectively. Given two matrices $A \in \mathbb{R}^{n\times p}$ and $B \in \mathbb{R}^{q\times r}$, we denote their matrix product (if $p = q$) as $A * B = AB \in \mathbb{R}^{n\times r}$ and their Kronecker product as $A \otimes B \in \mathbb{R}^{nq\times pr}$.

Let $\mathcal{I} \subseteq 2^{\mathbb{R}}$ be the set of closed real intervals, i.e., for all $X \in \mathcal{I}$, there exist $\underline{x}, \overline{x} \in \mathbb{R}$ such that $X = [\underline{x}, \overline{x}] = \{x \in \mathbb{R} \mid \underline{x} \leq x \leq \overline{x}\} \subseteq \mathbb{R}$. $\mathcal{I}^n$ and $\mathcal{I}^{n\times p}$ then represent the sets of interval vectors in $\mathbb{R}^n$ and interval matrices in $\mathbb{R}^{n\times p}$, respectively. Given two interval matrices $[\underline{A}, \overline{A}], [\underline{B}, \overline{B}] \in \mathcal{I}^{n\times p}$, their sum is: $[\underline{A}, \overline{A}] + [\underline{B}, \overline{B}] = [\underline{A} + \underline{B}, \overline{A} + \overline{B}]$. From Jaulin (2001), the product of two scalar intervals is defined as

$$[\underline{a}, \overline{a}] * [\underline{b}, \overline{b}] = [\min(\underline{a}\underline{b}, \underline{a}\overline{b}, \overline{a}\underline{b}, \overline{a}\overline{b}), \max(\underline{a}\underline{b}, \underline{a}\overline{b}, \overline{a}\underline{b}, \overline{a}\overline{b})] \in \mathcal{I}.$$

For $[\underline{A}, \overline{A}] \in \mathcal{I}^{n\times p}$ and $[\underline{B}, \overline{B}] \in \mathcal{I}^{p\times q}$, the product $[\underline{C}, \overline{C}] = [\underline{A}, \overline{A}] * [\underline{B}, \overline{B}] \in \mathcal{I}^{n\times q}$ is defined elementwise such that

$$[\underline{C}_{ij}, \overline{C}_{ij}] = \sum_{k=1}^{p} [\underline{A}_{ik}, \overline{A}_{ik}] * [\underline{B}_{kj}, \overline{B}_{kj}] \in \mathcal{I},$$

and the product of a scalar interval with a matrix interval is defined as $[\underline{C}, \overline{C}] = [\underline{a}, \overline{a}] * [\underline{B}, \overline{B}] \in \mathcal{I}^{p\times q}$ with

$$[\underline{C}_{ij}, \overline{C}_{ij}] = [\underline{a}, \overline{a}] * [\underline{B}_{ij}, \overline{B}_{ij}] \in \mathcal{I}.$$

For $[\underline{A}, \overline{A}] \in \mathcal{I}^{n\times p}$ and $[\underline{B}, \overline{B}] \in \mathcal{I}^{q\times r}$, the interval Kronecker product $[\underline{C}, \overline{C}] = [\underline{A}, \overline{A}] \otimes [\underline{B}, \overline{B}] \in \mathcal{I}^{nq\times pr}$ is defined as a $n \times p$ block interval matrix with (i, j) block

$$[\underline{C}_{ij}, \overline{C}_{ij}] = [\underline{A}_{ij}, \overline{A}_{ij}] * [\underline{B}, \overline{B}] \in \mathcal{I}^{q\times r}.$$

2.2 Functional matrices

In this section, we provide definitions and results on the manipulation of functional matrices used throughout the paper. We first introduce the differential operator D for a scalar differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ to be:

$$Df(x) = \begin{pmatrix} \frac{\partial f(x)}{\partial x_1} & \cdots & \frac{\partial f(x)}{\partial x_n} \end{pmatrix}.$$

Then for a functional matrix $A : \mathbb{R}^n \to \mathbb{R}^{p\times q}$, its differential $DA(x) \in \mathbb{R}^{p\times nq}$ is the $p \times q$ block matrix where each element $A_{ij}(x) \in \mathbb{R}$ of $A(x) \in \mathbb{R}^{p\times q}$ is replaced by the row vector of its differential $DA_{ij}(x) \in \mathbb{R}^{1\times n}$:

$$DA(x) = \begin{pmatrix} DA_{11}(x) & \cdots & DA_{1q}(x) \\ \vdots & \ddots & \vdots \\ DA_{p1}(x) & \cdots & DA_{pq}(x) \end{pmatrix} \tag{1}$$

$$= \begin{pmatrix} \frac{\partial A_{11}(x)}{\partial x_1} \cdots \frac{\partial A_{11}(x)}{\partial x_n} & \cdots & \frac{\partial A_{1q}(x)}{\partial x_1} \cdots \frac{\partial A_{1q}(x)}{\partial x_n} \\ \vdots \qquad\quad \vdots & \vdots & \vdots \qquad\quad \vdots \\ \frac{\partial A_{p1}(x)}{\partial x_1} \cdots \frac{\partial A_{p1}(x)}{\partial x_n} & \cdots & \frac{\partial A_{pq}(x)}{\partial x_1} \cdots \frac{\partial A_{pq}(x)}{\partial x_n} \end{pmatrix}.$$

This notation ensures that we only work with 2-dimensional matrices, instead of matrices with more than two dimensions for which cumbersome matrix product definitions would need to be introduced.

For a time-varying functional matrix $A : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^{p\times q}$, its time derivative is denoted with a dot

$$\dot{A}(t, x) = \frac{\partial A(t, x)}{\partial t},$$

and we keep the notation $DA(t, x)$ as in (1) to denote its derivative with respect to the second variable $x \in \mathbb{R}^n$.

For the product of two functional matrices, its differential is obtained as in the following result from (Cheng et al., 2012, Corollary 18.1).

Lemma 1. (Product rule). Given $A : \mathbb{R}^n \to \mathbb{R}^{p\times q}$, $B : \mathbb{R}^n \to \mathbb{R}^{q\times r}$, we have $D(A(x)B(x)) \in \mathbb{R}^{p\times nr}$ given by

$$D(A(x)B(x)) = DA(x) * (B(x) \otimes I_n) + A(x) * DB(x).$$

Next, we introduce the chain rule for the composition of a functional vector and functional matrix.

Lemma 2. (Chain rule). Given $A : \mathbb{R}^m \to \mathbb{R}^{p\times q}$ and $b : \mathbb{R}^n \to \mathbb{R}^m$, we have $D(A(b(x))) \in \mathbb{R}^{p\times nq}$ given by

$$D(A(b(x))) = DA(y)|_{y=b(x)} * (I_q \otimes Db(x)).$$

The proof of Lemma 2 is straightforward and omitted.

2.3 Reachability analysis of interval affine systems

The method presented in this paper partly relies on results from Althoff et al. (2007) which use interval arithmetics to over-approximate the reachable set and reachable tube of affine interval systems. These results are summarized in this section for self-containment of the paper.

Consider an affine interval system of the form

$$\dot{z} \in \mathcal{A}z + \mathcal{B}, \tag{2}$$

with state $z \in \mathbb{R}^{p\times q}$ and interval matrices $\mathcal{A} = [\underline{A}, \overline{A}] \in \mathcal{I}^{p\times p}$ and $\mathcal{B} = [\underline{B}, \overline{B}] \in \mathcal{I}^{p\times q}$. Given an interval matrix of initial states $Z_0 = [\underline{z_0}, \overline{z_0}] \in \mathcal{I}^{p\times q}$ and a time step $\tau > 0$, we denote the reachable set of (2) as $z(\tau, Z_0) \subseteq \mathbb{R}^{p\times q}$ and its reachable tube as $z([0,\tau], Z_0) = \bigcup_{t\in[0,\tau]} z(t, Z_0) \subseteq \mathbb{R}^{p\times q}$.

The results from Althoff et al. (2007) rely on Taylor series truncated at an order $r \in \mathbb{N}$ which needs to satisfy $r > \|\mathcal{A}\|_\infty \tau - 2$, where the infinity norm of the interval matrix is defined by $\|\mathcal{A}\|_\infty = \|\max(|\underline{A}|, |\overline{A}|)\|_\infty$ using componentwise absolute value and max operators. Then we introduce

$$\begin{aligned}
C(\tau) &= [-\mathbf{1}_{p\times p}, \mathbf{1}_{p\times p}] * \frac{(\|\mathcal{A}\|_\infty \tau)^{r+1}}{(r+1)!} \frac{r+2}{r+2-\|\mathcal{A}\|_\infty \tau}, \\
D(\tau) &= \sum_{i=0}^{r} \frac{(\mathcal{A}\tau)^i}{i!} + C(\tau), \\
E(\tau) &= \sum_{i=0}^{r} \frac{\mathcal{A}^i \tau^{i+1}}{(i+1)!} + C(\tau)\tau, \\
F(\tau) &= \left[\sum_{i=2}^{r} \left(i^{\frac{-i}{i-1}} - i^{\frac{-1}{i-1}}\right) \frac{(\mathcal{A}\tau)^i}{i!}, \mathbf{0}_{p\times p}\right] + C(\tau),
\end{aligned}$$

where all sums and products of interval matrices follow the definitions in Section 2.1. We also define the interval hull of two interval matrices $[\underline{a}, \overline{a}], [\underline{b}, \overline{b}] \in \mathcal{I}^{p\times q}$ as $H([\underline{a}, \overline{a}], [\underline{b}, \overline{b}]) = [\min(\underline{a}, \underline{b}), \max(\overline{a}, \overline{b})]$ using the componentwise min and max operators.

Lemma 3. (Althoff et al. (2007)). The reachable set of (2) at time $\tau \geq 0$ is over-approximated by an interval in $\mathcal{I}^{p\times q}$ as follows:

$$z(\tau, Z_0) \subseteq D(\tau)Z_0 + E(\tau)\mathcal{B}. \tag{3}$$

If in addition we have $\mathcal{B} = \{\mathbf{0}_{p\times q}\}$, then the reachable tube of (2) over time range $[0,\tau]$ is over-approximated by an interval in $\mathcal{I}^{p\times q}$ as follows:

$$z([0,\tau], Z_0) \subseteq H(Z_0, D(\tau)Z_0) + F(\tau)Z_0. \tag{4}$$

3. PROBLEM FORMULATION

We consider a continuous-time, time-varying system

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

with state $x \in \mathbb{R}^n$ and vector field $f : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^n$ assumed to be twice differentiable in the state. Note that a system $\dot{x} = f(t, x, p)$ with constant but uncertain parameters $p \in \mathbb{R}^q$ can be written as in (5) by considering p as states whose dynamics are $\dot{p} = 0$. We denote as $\Phi(t; t_0, x_0) \in \mathbb{R}^n$ the state reached by (5) at time $t \geq t_0$ from initial state x_0. In this paper, our goal is to compute an interval over-approximation of the finite-time reachable set of (5) as defined below.

Problem 4. Given a time range $[t_0, t_f] \in \mathcal{I}$ and an interval of initial states $X_0 = [\underline{x}, \overline{x}] \in \mathcal{I}^n$, find an interval in $\mathcal{I}^n$ over-approximating the reachable set of system (5) defined as

$$R(t_f; t_0, X_0) = \{\Phi(t_f; t_0, x_0) \mid x_0 \in X_0\}.$$

To solve Problem 4 with the method presented in Section 5, we assume that bounds on both the first-order and second-order Jacobian matrices of (5) are provided by the user. These two Jacobian matrices are defined below using the differential operator D of the vector field $f(t, x)$ with respect to state x as introduced in Section 2.2:

$$\begin{aligned}
J^x(t, x) &= Df(t, x) \in \mathbb{R}^{n\times n}, \\
J^{xx}(t, x) &= DJ^x(t, x) \in \mathbb{R}^{n\times n^2}.
\end{aligned}$$

Then our main assumption is formulated as follows, using the known time range $[t_0, t_f]$ from Problem 4:

Assumption 5. Given an invariant state space $X \subseteq \mathbb{R}^n$ for system (5), there exist $[\underline{J^x}, \overline{J^x}] \in \mathcal{I}^{n\times n}$ and $[\underline{J^{xx}}, \overline{J^{xx}}] \in \mathcal{I}^{n\times n^2}$ such that for all $t \in [t_0, t_f]$ and $x \in X$ we have $J^x(t, x) \in [\underline{J^x}, \overline{J^x}]$ and $J^{xx}(t, x) \in [\underline{J^{xx}}, \overline{J^{xx}}]$.

4. SENSITIVITY EQUATIONS

The method presented in Section 5 to solve Problem 4 relies on the definition of the sensitivity matrices of system (5) representing the differential influence of the initial conditions on the successor $\Phi(t; t_0, x_0)$ at time t. Similarly to the definition of the Jacobian matrices above, we use D to denote the differential operator of the trajectory $\Phi(t; t_0, x_0)$ with respect to initial state x_0. Then the first-order and second-order sensitivity matrices are defined as:

$$S^x(t; t_0, x_0) = D\Phi(t; t_0, x_0) \in \mathbb{R}^{n\times n}, \tag{6}$$

$$S^{xx}(t; t_0, x_0) = DS^x(t; t_0, x_0) \in \mathbb{R}^{n\times n^2}. \tag{7}$$

Both sensitivity matrices defined in (6) and (7) can also be described by the time-varying affine systems below.

Proposition 6. Using the short-hand notations $S^x := S^x(t; t_0, x_0)$, $S^{xx} := S^{xx}(t; t_0, x_0)$, $J^x := J^x(t, \Phi(t; t_0, x_0))$ and $J^{xx} := J^{xx}(t, \Phi(t; t_0, x_0))$, the sensitivity matrices defined in (6) and (7) follow:

$$\dot{S}^x = J^x * S^x, \tag{8}$$

$$\dot{S}^{xx} = J^x * S^{xx} + J^{xx} * (S^x \otimes S^x), \tag{9}$$

with $S^x(t_0; t_0, x_0) = I_n$ and $S^{xx}(t_0; t_0, x_0) = \mathbf{0}_{n\times n^2}$.

Proof. System (8) is obtained as in Donzé and Maler (2007) by applying the chain rule to the vector field f:

$$\begin{aligned}
\dot{S}^x(t; t_0, x_0) &= D\dot{\Phi}(t; t_0, x_0) \\
&= Df(t, \Phi(t; t_0, x_0)) \\
&= Df(t, y)|_{y=\Phi(t;t_0,x_0)} * D\Phi(t; t_0, x_0) \\
&= J^x(t, \Phi(t; t_0, x_0)) * S^x(t; t_0, x_0).
\end{aligned}$$

Since $S^{xx} = DS^x$ from (7), system (9) is obtained by differentiating (8) and then applying the product rule and the chain rule from Lemmas 1 and 2, respectively:

$$
\begin{aligned}
\dot{S}^{xx}(t;t_0,x_0) =& D\dot{S}^{x}(t;t_0,x_0) \\
=& DJ^x(t,\Phi(t;t_0,x_0)) * (S^x(t;t_0,x_0) \otimes I_n) \\
&+ J^x(t,\Phi(t;t_0,x_0)) * DS^x(t;t_0,x_0) \\
=& J^{xx}(t,\Phi(t;t_0,x_0)) * (I_n \otimes S^x(t;t_0,x_0)) \\
&* (S^x(t;t_0,x_0) \otimes I_n) \\
&+ J^x(t,\Phi(t;t_0,x_0)) * S^{xx}(t;t_0,x_0).
\end{aligned}
$$

Finally, $(I_n \otimes S^x)(S^x \otimes I_n) = S^x \otimes S^x$ is a property of the Kronecker product. The initial conditions are immediately obtained by using $\Phi(t_0;t_0,x_0) = x_0$ in (6) and (7). $\square$

5. REACHABILITY ALGORITHM

The proposed approach to solve Problem 4 is summarized in Algorithm 1 and Figure 1. Below, we briefly explain this algorithm by going backwards from step 4 to step 1.

The end goal in step 4 is to over-approximate the reachable set of the nonlinear system (5) using the recent reachability method in Meyer et al. (2018) that relies on interval bounds on the reachable set of the first-order sensitivity $S^x(t_f;t_0,X_0)$. The method in Meyer et al. (2018) uses either conservative bounds from a direct application of Lemma 3 or empirical bounds from a sampling procedure. In contrast, here we derive guaranteed bounds on S^x in step 3 by combining bounds on the reachable set of the second-order sensitivity $S^{xx}(t_f;t_0,X_0)$ with the numerical evaluation of S^x at time t_f on a finite set of sampled initial states. The resulting bounds on $S^x(t_f;t_0,X_0)$ can be made arbitrarily tight by increasing the number of samples.

The bounds on S^{xx} are computed in step 2 by applying (3) in Lemma 3 to (9), which requires the knowledge of bounds of both Jacobian matrices (from Assumption 5) and on the reachable tube of the first-order sensitivity $S^x([t_0,t_f];t_0,X_0)$. This reachable tube of S^x is over-approximated in step 1 by applying (4) in Lemma 3 to (8), which requires bounds on J^x taken from Assumption 5.

These steps are detailed in the following subsections. A further discussion for using the first three steps instead of directly over-approximating $S^x(t_f;t_0,X_0)$ with Lemma 3 as in Meyer et al. (2018) is given in Section 5.4.

Input: Reachability problem for (5): t_0, t_f, $X_0 = [\underline{x},\overline{x}]$
Data: Jacobian bounds $[\underline{J^x},\overline{J^x}]$, $[\underline{J^{xx}},\overline{J^{xx}}]$
Step 1: Apply (4) to (8) and obtain an interval over-approximation of $S^x([t_0,t_f];t_0,X_0)$
Step 2: Apply (3) to (9) and obtain an interval over-approximation of $S^{xx}(t_f;t_0,X_0)$
Step 3: Obtain an interval over-approximation of $S^x(t_f;t_0,X_0)$ from the bounds on S^{xx} and the evaluation of $S^x(t_f;t_0,x_0)$ on a finite subset of X_0
Step 4: Obtain an interval over-approximation of $R(t_f;t_0,X_0)$ as in Meyer et al. (2018) using the bounds on S^x
Output: Interval solving Problem 4
Algorithm 1: Reachability analysis of system (5).

5.1 Interval arithmetics on the sensitivity systems

For the first step of Algorithm 1, we first need to rewrite the time-varying linear system of the first-order sensitivity

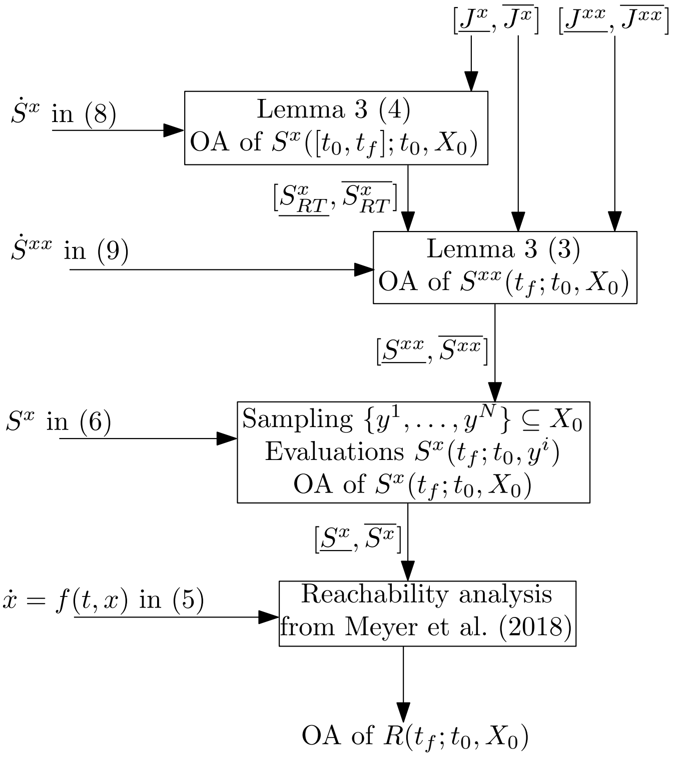

Fig. 1. Sketch of the 4-step reachability procedure in Algorithm 1 where "OA" stands "over-approximation". For each box, top arrows are the input requirements, side arrows are the equations used and bottom arrows are the output results.

(8) into a linear interval system similarly to (2). This is done using the bounds on J^x from Assumption 5:

$$\dot{S}^x(t;t_0,x_0) \in [\underline{J^x},\overline{J^x}] * S^x(t;t_0,x_0). \tag{10}$$

Then, applying (4) in Lemma 3 with $\mathcal{A} = [\underline{J^x},\overline{J^x}]$, $\mathcal{B} = \{\mathbf{0}_{n\times n}\}$ and $Z_0 = \{I_n\}$ leads to an over-approximation of the reachable tube $S^x([t_0,t_f];t_0,[\underline{x},\overline{x}]) \subseteq [\underline{S^x_{RT}},\overline{S^x_{RT}}] \in \mathcal{I}^{n\times n}$ defined as:

$$[\underline{S^x_{RT}},\overline{S^x_{RT}}] = H(\{I_n\},D(t_f-t_0)) + F(t_f-t_0).$$

For the second step in Algorithm 1, we use the bounds $[\underline{S^x_{RT}},\overline{S^x_{RT}}]$ obtained in the previous step alongside the Jacobian bounds from Assumption 5 to rewrite the time-varying affine system of the second-order sensitivity (9) into an affine interval system as in (2) with $\mathcal{A} = [\underline{J^x},\overline{J^x}]$, $\mathcal{B} = [\underline{J^{xx}},\overline{J^{xx}}] * ([\underline{S^x_{RT}},\overline{S^x_{RT}}] \otimes [\underline{S^x_{RT}},\overline{S^x_{RT}}])$ and the initial condition $Z_0 = \{\mathbf{0}_{n\times n^2}\}$ from Proposition 6. This leads to an over-approximation of the reachable set $S^{xx}(t_f;t_0,[\underline{x},\overline{x}]) \subseteq [\underline{S^{xx}},\overline{S^{xx}}] \in \mathcal{I}^{n\times n^2}$ defined as:

$$[\underline{S^{xx}},\overline{S^{xx}}] = E(t_f-t_0)\mathcal{B}.$$

Remark 7. Although step 2 only focuses on S^{xx} at time t_f, the interval matrix $\mathcal{B}$ in (2) used in Lemma 3 needs to bound the values of $J^{xx} * (S^x \otimes S^x)$ from system (9) for all time in $[t_0,t_f]$. This is why step 1 considers the whole reachable tube of S^x instead of only the reachable set.

5.2 Sampling for the first-order sensitivity

Step 3 of Algorithm 1 relies on the evaluation of the first-order sensitivity for some sampled initial states. Let

$\{y^1, \dots, y^N\} = Y \subseteq [\underline{x}, \overline{x}]$ be a finite set of N samples in the interval of initial states $[\underline{x}, \overline{x}]$. Similarly to (Tempo et al., 2012, Section 7.4.4), we define below the dispersion of this set of samples, where the infinity norm of a state $x \in \mathbb{R}^n$ is defined as $\|x\|_\infty = \max_{i \in \{1,\dots,n\}} |x_i|$.

Definition 8. Given a finite set $Y \subseteq [\underline{x}, \overline{x}]$, the dispersion of Y in $[\underline{x}, \overline{x}]$ is defined as:

$$d(Y) = \sup_{x \in [\underline{x}, \overline{x}]} \min_{y \in Y} \|x - y\|_\infty \in \mathbb{R}.$$

Smaller values of $d(Y)$ imply that the sample states in Y are well scattered in the interval $[\underline{x}, \overline{x}]$. After evaluating the first-order sensitivity $S^x(t_f; t_0, y^i)$ at time t_f for each of these sampled states through numerical integration of (6) or (8), we can derive guaranteed bounds on the set $S^x(t_f; t_0, [\underline{x}, \overline{x}])$ as follows.

Theorem 9. Given bounds on the second-order sensitivity $S^{xx}(t_f; t_0, [\underline{x}, \overline{x}]) \subseteq [\underline{S^{xx}}, \overline{S^{xx}}] \in \mathcal{I}^{n \times n^2}$ and a finite set $Y \subseteq [\underline{x}, \overline{x}]$ of sampled initial states, define $M \in \mathbb{R}^{n \times n}$ as

$$M = \max\left(|\underline{S^{xx}}|, |\overline{S^{xx}}|\right) * (I_n \otimes (\mathbf{1}_n * d(Y))),$$

using componentwise absolute value and max operators. Then the set of first-order sensitivity values at time t_f is over-approximated as $S^x(t_f; t_0, [\underline{x}, \overline{x}]) \subseteq [\underline{S^x}, \overline{S^x}] \in \mathcal{I}^{n \times n}$ with, for all $i, j \in \{1, \dots, n\}$:

$$\overline{S^x}_{ij} = \max_{y \in Y}\left(S^x_{ij}(t_f; t_0, y)\right) + M_{ij},$$
$$\underline{S^x}_{ij} = \min_{y \in Y}\left(S^x_{ij}(t_f; t_0, y)\right) - M_{ij}.$$

Proof. Taking any $x, y \in [\underline{x}, \overline{x}]$, we define the straight line between x and y as $\gamma : [0,1] \to \mathbb{R}^n$ with $\gamma(\lambda) = y + \lambda(x - y)$. Then for all $i, j \in \{1, \dots, n\}$, the fundamental theorem of calculus applied to S^x_{ij} along γ gives:

$$S^x_{ij}(t_f; t_0, x) - S^x_{ij}(t_f; t_0, y) = \int_0^1 DS^x_{ij}(t_f; t_0, \gamma(\lambda)) * (x - y) d\lambda.$$

From (1), we know that $DS^x_{ij}(t_f; t_0, \gamma(\lambda)) \in \mathbb{R}^{1 \times n}$ are the elements of $S^{xx}(t_f; t_0, \gamma(\lambda))$ in row i and from column $1 + (j-1)n$ to column jn. By definition of the dispersion, for any initial state $x \in [\underline{x}, \overline{x}]$, there exists $y \in Y$ such that $\|x - y\|_\infty \leq d(Y)$. Then, for any such (x, y) pair, the distance of their first-order sensitivity S^x_{ij} can be bounded as follows:

$$\begin{aligned}
&\left|S^x_{ij}(t_f; t_0, x) - S^x_{ij}(t_f; t_0, y)\right| \\
&\leq \int_0^1 \sum_{k=1}^n \left|S^{xx}_{i,k+(j-1)n}(t_f; t_0, \gamma(\lambda)) * (x_k - y_k)\right| d\lambda \\
&\leq \sum_{k=1}^n \max_{x \in [\underline{x}, \overline{x}]} \left|S^{xx}_{i,k+(j-1)n}(t_f; t_0, x)\right| * d(Y).
\end{aligned}$$

Since $\max_{x \in [\underline{x}, \overline{x}]} \left|S^{xx}_{i,k+(j-1)n}(t_f; t_0, x)\right|$ is equal to element $(i, k + (j-1)n)$ of matrix $\max\left(|\underline{S^{xx}}|, |\overline{S^{xx}}|\right)$, we then have

$$\left|S^x_{ij}(t_f; t_0, x) - S^x_{ij}(t_f; t_0, y)\right| \leq M_{ij}.$$

The theorem statement is finally obtained by bounding $S^x_{ij}(t_f; t_0, y)$ by its extremal values over the set $y \in Y$. □

The over-approximation interval $[\underline{S^x}, \overline{S^x}]$ in Theorem 9 thus corresponds to the interval hull of the sampled sensitivity evaluations $\{S^x(t_f; t_0, y) | y \in Y\}$ dilated by M. Although this result is valid for any non-empty set $Y \subseteq [\underline{x}, \overline{x}]$ of sampled initial states, the value of the dispersion as in Definition 8 can be challenging to compute or to upper-bound for any system with more than one state dimension ($n > 1$). Below, we give a result adapted from Tempo et al. (2012) stating that this dispersion can be exactly computed for a sampling set defined as a uniform grid.

Lemma 10. Let Y be defined as a uniform grid in $[\underline{x}, \overline{x}]$ with $a \in \mathbb{N}$ elements per dimension (i.e. containing $N = a^n$ sample states) and such that on each dimension $i \in \{1, \dots, n\}$ the samples are separated by $\frac{\overline{x}_i - \underline{x}_i}{a}$ and the first sample is shifted of $\frac{\overline{x}_i - \underline{x}_i}{2a}$ from $\underline{x}_i$. Then the dispersion of Y is given by:

$$d(Y) = \frac{\|\overline{x} - \underline{x}\|_\infty}{2a}.$$

From the definition of M in Theorem 9, we can see that the size of the obtained bounds on the first-order sensitivity S^x grows with the dispersion of the sampling set Y. As a consequence, the set Y can be used to tune the tradeoff between reducing the conservativeness of the sensitivity bounds $[\underline{S^x}, \overline{S^x}]$ and limiting the computation time (related to the number of samples). If computation capabilities were unlimited, Theorem 9 could then provide interval bounds of the first-order sensitivity values with arbitrary precision, as formulated below.

Proposition 11. If the sample number grows to infinity $N \to \infty$, we can design the sampling set Y such that $[\underline{S^x}, \overline{S^x}]$ from Theorem 9 converge to the unique tight interval over-approximation of the set $S^x(t_f; t_0, [\underline{x}, \overline{x}])$.

Proof. To ensure that we obtain $\lim_{N \to \infty} d(Y) = 0$, we need to pick the set Y such that the the whole interval $[\underline{x}, \overline{x}]$ is sampled (instead of just sampling a subset). The uniform grid in Lemma 10 satisfies this property since we have $\lim_{a \to \infty} d(Y) = 0$. This leads to $M \to \mathbf{0}_{n \times n}$ and Theorem 9 then states that each element of the bounds $\underline{S^x}$ and $\overline{S^x}$ is obtained from the sensitivity evaluation $S^x(t_f; t_0, y)$ for a state $y \in Y \subseteq [\underline{x}, \overline{x}]$. This implies that any interval strictly contained in $[\underline{S^x}, \overline{S^x}]$ cannot contain the whole set $S^x(t_f; t_0, [\underline{x}, \overline{x}])$, which is the definition of tightness of an over-approximation. □

5.3 Reachability analysis of the initial system

This section corresponds to step 4 of Algorithm 1 in which we apply the method for reachability analysis introduced in Meyer et al. (2018). This reachability result is summarized below for self-containment of this paper.

Let $S^{x*} \in \mathbb{R}^{n \times n}$ denote the center of $[\underline{S^x}, \overline{S^x}]$ and define the decomposition function $g : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ whose i^{th} component with $i \in \{1, \dots, n\}$ is

$$g_i(t_0, x, y) = \Phi_i(t_f; t_0, z^i) + \alpha^i(x - y), \tag{11}$$

where the state $z^i = [z^i_1; \dots; z^i_n] \in \mathbb{R}^n$ and row vector $\alpha^i = [\alpha^i_1, \dots, \alpha^i_n] \in \mathbb{R}^{1 \times n}$ are such that for all $j \in \{1, \dots, n\}$,

$$(z^i_j, \alpha^i_j) = \begin{cases} (x_j, \max(0, -\underline{S^x}_{ij})) & \text{if } S^{x*}_{ij} \geq 0, \\ (y_j, \max(0, \overline{S^x}_{ij})) & \text{if } S^{x*}_{ij} < 0. \end{cases} \tag{12}$$

Then an over-approximation of the reachable set of (5) is obtained by computing only two evaluations of the decomposition function g.

Lemma 12. (Meyer et al. (2018)). Given bounds on the first-order sensitivity $S^x(t_f;t_0,[\underline{x},\overline{x}]) \subseteq [\underline{S^x},\overline{S^x}] \in \mathcal{I}^{n\times n}$ and the definitions in (11)-(12), an over-approximation of the reachable set of (5) is given by:

$$R(t_f;t_0,[\underline{x},\overline{x}]) \subseteq [g(t_0,\underline{x},\overline{x}), g(t_0,\overline{x},\underline{x})].$$

Although the option for an arbitrary precision on $[\underline{S^x},\overline{S^x}]$ from Proposition 11 does not transfer to the over-approximation of $R(t_f;t_0,[\underline{x},\overline{x}])$, the following remark highlights under which conditions the result in Lemma 12 provides a tight over-approximation.

Remark 13. (Meyer et al. (2018)). If each element of the sensitivity bounds $[\underline{S^x},\overline{S^x}]$ is sign-stable (i.e. for all $i,j \in \{1,\dots,n\}$, either $\underline{S^x}_{ij} \geq 0$ or $\overline{S^x}_{ij} \leq 0$), then the interval defined in Lemma 12 is the unique tight over-approximating interval of the reachable set $R(t_f;t_0,[\underline{x},\overline{x}])$.

5.4 Comparison to Meyer et al. (2018)

Two alternatives for the computation of bounds $[\underline{S^x},\overline{S^x}]$ on the first-order sensitivity $S^x(t_f;t_0,[\underline{x},\overline{x}])$ were initially introduced in Meyer et al. (2018), both with their own shortcomings. To highlight the novelties and advantages of the approach proposed in this paper, we briefly describe below these two alternatives and compare them to steps 1-3 from Algorithm 1. The main points of comparison of these three approaches are summarized in Table 1.

The first method in Meyer et al. (2018) relies on replacing steps 1-3 from Algorithm 1 by a single step where we over-approximate the reachable set of the first-order sensitivity directly. To do this, we apply the interval arithmetics result from (3) in Lemma 3 to the linear interval system (10) of the first-order sensitivity. This results in the following over-approximation:

$$S^x(t_f;t_0,[\underline{x},\overline{x}]) \subseteq D(t_f - t_0). \tag{13}$$

Similarly to Algorithm 1, this provides a guaranteed over-approximation of the possible values taken by the first-order sensitivity. The computation time is very short in most cases, but the obtained over-approximation tends to be overly conservative due to being directly influenced in (13) by the (possibly large) first-order Jacobian bounds from Assumption 5.

The second alternative is simulation-based and has two steps: sampling and falsification. The sampling step is done similarly to Section 5.2, where we pick a finite sampling set $Y \subseteq [\underline{x},\overline{x}]$, evaluate the first-order sensitivity $S^x(t_f;t_0,y)$ for all $y \in Y$ through numerical integration of (6) or (8) and then define approximate bounds $[\underline{S^x},\overline{S^x}]$ as

$$[\underline{S^x}_{ij},\overline{S^x}_{ij}] = \left[\min_{y\in Y}\left(S^x_{ij}(t_f;t_0,y)\right), \max_{y\in Y}\left(S^x_{ij}(t_f;t_0,y)\right)\right].$$

Then for all $i,j \in \{1,\dots,n\}$, the falsification step runs an optimization problem to find other initial states $x \in [\underline{x},\overline{x}]$ whose sensitivity evaluation does not belong to the current bounds ($S^x(t_f;t_0,x) \notin [\underline{S^x},\overline{S^x}]$). If such state is found, the bounds are enlarged accordingly and the falsification step is repeated until we stop finding states falsifying the current bounds. Since this is a simulation-based approach, it tends to give very accurate approximation of the actual set of first-order sensitivity values $S^x(t_f;t_0,[\underline{x},\overline{x}])$, and it requires no assumption on system (5) or its Jacobian matrices. On the other hand, both sampling and falsification steps are computationally expensive (with an exponential growth in the state dimension n) and since the falsification step can only deal with local minima, the obtained bounds are not guaranteed to be a true over-approximation of $S^x(t_f;t_0,[\underline{x},\overline{x}])$.

	IA	SF	Algorithm 1
Guarantees	**yes**	no	**yes**
Conservativeness	large	**small**	**tunable**
Computation time	**small**	large	**tunable**
Assumptions	$[\underline{J^x},\overline{J^x}]$	**none**	$[\underline{J^x},\overline{J^x}]$, $[\underline{J^{xx}},\overline{J^{xx}}]$

Table 1. Comparison of the properties of the over-approximation of $S^x(t_f;t_0,[\underline{x},\overline{x}])$ for three methods: one based on interval arithmetics (IA) from Meyer et al. (2018), one based on sampling and falsification (SF) from Meyer et al. (2018), and the one from Algorithm 1.

In comparison, the method presented in this paper to over-approximate the reachable set of the first-order sensitivity (steps 1-3 from Algorithm 1) aims to combine the advantages of both above approaches while eliminating their shortcomings. As in the interval arithmetics alternative, we obtain a guaranteed over-approximation of $S^x(t_f;t_0,[\underline{x},\overline{x}])$, which can be computed very quickly if we pick a small sampling set Y in step 3. As in the sampling and falsification approach, we can choose to obtain an arbitrarily close over-approximation (as highlighted in Proposition 11) by increasing the number of samples in Y. The main drawback of this approach is that it requires the user to provide bounds for both the first-order and the second-order Jacobian matrices as in Assumption 5.

6. NUMERICAL ILLUSTRATION

In this section, we illustrate the approach in Algorithm 1 and the alternative methods from Meyer et al. (2018) on a numerical example and highlight the elements of comparison discussed in Section 5.4. We consider the continuous-time uncertain unicycle model described as:

$$\dot{x} = \begin{pmatrix} v\cos(x_3) + x_4 \\ v\sin(x_3) + x_5 \\ \omega + x_6 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \tag{14}$$

where $[x_1;x_2]$ is the 2D position of the unicycle, x_3 is its orientation, $[x_4;x_5;x_6]$ are constant uncertain parameters in the dynamics of the first three states, $v = 0.25$ is the controlled forward velocity and $\omega = 0.3$ is the controlled angular velocity. Using the conservative bounds $\cos(x_3), \sin(x_3) \in [-1,1]$, global Jacobian bounds of (14) satisfying Assumption 5 are obtained by taking $[\underline{J^x}_{1,4},\overline{J^x}_{1,4}] = [\underline{J^x}_{2,5},\overline{J^x}_{2,5}] = [\underline{J^x}_{3,6},\overline{J^x}_{3,6}] = \{1\}$, $[\underline{J^x}_{1,3},\overline{J^x}_{1,3}] = [\underline{J^x}_{2,3},\overline{J^x}_{2,3}] = [\underline{J^{xx}}_{1,15},\overline{J^x}_{1,15}] = [\underline{J^{xx}}_{2,15},\overline{J^x}_{2,15}] = [-v,v]$ and $[\underline{J^x}_{ij},\overline{J^x}_{ij}] = [\underline{J^{xx}}_{ij},\overline{J^x}_{ij}] = \{0\}$ for all other elements.

Taking the initial time $t_0 = 0$, we want to evaluate the reachable set of (14) at time $t_f = 10$ for the following interval of initial conditions: $X_0 = [0,1] \times [0,1] \times [\frac{\pi}{8},\frac{2\pi}{8}] \times$

$[-0.05, 0.05] \times [-0.05, 0.05] \times [-0.03, 0.03]$. This reachability problem is solved in five ways described below.

- We first apply Algorithm 1 three times using a uniform grid sampling as in Lemma 10 with an increasing number of samples per dimension of the state space $a \in \{1, 2, 3\}$ (leading to a total number of sample points of $N = a^6 \in \{1, 64, 729\}$). In Figures 2 and 3, these results are plotted in dashed red, dot-dashed blue and plain green, respectively.
- Next we use the one-step interval arithmetics ("IA" in Table 2) approach from Meyer et al. (2018) described in Section 5.4, plotted in dotted purple.
- Finally we apply the sampling and falsification ("SF" in Table 2) approach from Meyer et al. (2018) described in Section 5.4 using $N = 64$ samples, plotted in dashed orange.

The computation times for each of the four steps in Algorithm 1 (or alternatively, for obtaining bounds on $S^x(t_f; t_0, X_0)$ in both methods from Meyer et al. (2018)) are reported in Table 2. The obtained bounds on $S^x_{1,3}$ and $S^x_{2,3}$ for step 3 are plotted in Figure 2 and the final reachability analysis (step 4) on states x_1 and x_2 is shown in Figure 3. In both figures, the cloud of black dots represents the numerical integration of (6) and (14), respectively, for 500 random samples in X_0.

From Table 2, we first note that the computation of the final reachable set (step 4) is very fast and identical for all method since this step is oblivious to the way the sensitivity bounds $[\underline{S^x}, \overline{S^x}]$ are obtained. As expected, the three steps relying on the interval arithmetics results from Lemma 3 (steps 1 and 2 in Algorithm 1 and step 3 in method "IA") are also achieved quickly. The sampling computations in step 3 of Algorithm 1 naturally grows with the number of samples. For the sampling and falsification approach from Meyer et al. (2018), the sampling time is identical to the one in the second call of Algorithm 1 (due to having the same number of samples $N = 64$), but then the total computation time is increased by the 2 iterations of the falsification procedure used to improve the estimated bounds on S^x. Such expansion of the bounds is not required in Algorithm 1 since from Theorem 9, step 3 is already guaranteed to over-approximate $S^x(t_f; t_0, X_0)$.

In Figure 2, we can first note that, as hinted in Proposition 11, the bounds on the first-order sensitivity obtained in Algorithm 1 shrink as we increase the number of samples. As mentioned in Section 5.4 and Table 1, we can see that the one-step interval arithmetics method from Meyer et al. (2018) gives very conservative bounds on S^x (similar in size to Algorithm 1 with a single sample point). While the sampling and falsification method from Meyer et al. (2018) gives the closest approximation of $S^x(t_f; t_0, X_0)$, the obtained bounds are not actually an over-approximation of this set (despite the 2 iterations of falsification), which means that applying step 4 with such bounds is not sound for the reachability analysis of (14).

Finally, we can combine Figure 3 and Table 2 to conclude on the ability of Algorithm 1 to tune to our needs the tradeoff between computation time and conservativeness. The sampling and falsification approach from Meyer et al. (2018) is discarded from this discussion as we already

	Algorithm 1			IA	SF
Samples N	1	64	729	-	64
$[\underline{S^x_{RT}}, \overline{S^x_{RT}}]$	0.72			-	-
$[\underline{S^{xx}}, \overline{S^{xx}}]$	0.87			-	-
$[\underline{S^x}, \overline{S^x}]$	0.35	3.2	36	0.44	$3.1 + 4.2$
$R(t_f; t_0, X_0)$	0.07				

Table 2. Time comparison (in seconds) of the steps for reachability analysis in Algorithm 1 with three different sampling grids, and in both methods from Meyer et al. (2018) using a single step interval arithmetics (IA) or sampling and falsification (SF).

showed above that it is unreliable when we want guaranteed over-approximations. When computation time is our main concern, we can take $N = 1$ in Algorithm 1 to obtain results comparable to the one-step "IA" method from Meyer et al. (2018), in terms of both conservativeness and low computation time. In particular, although the computation time of the interval arithmetics steps 1-2 would slightly increase with higher state dimension n, the computational complexity of steps 3-4 is constant (i.e. independent of the state dimension) when we take $N = 1$. On the other hand, if more computational power is available, increasing the number of samples tightens the over-approximation and in this example, we can see in Figure 3 that both $N = 64$ and $N = 729$ give tighter bounds than the method from Meyer et al. (2018).

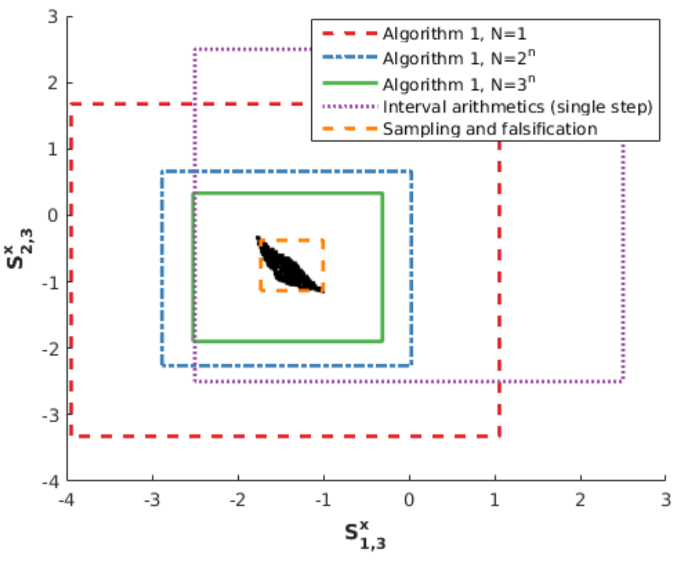

Fig. 2. Comparison of over-approximations of the first-order sensitivity components $S^x_{1,3}$ and $S^x_{2,3}$ at time t_f.

7. CONCLUSION

This paper provides a new reachability analysis relying on the first-order and second-order sensitivity matrices of a continuous-time nonlinear system. The proposed algorithm first uses interval arithmetics to over-approximate the reachable tube of the first-order sensitivity, then the reachable set of the second-order sensitivity. The obtained bounds are then combined with a sampling procedure on

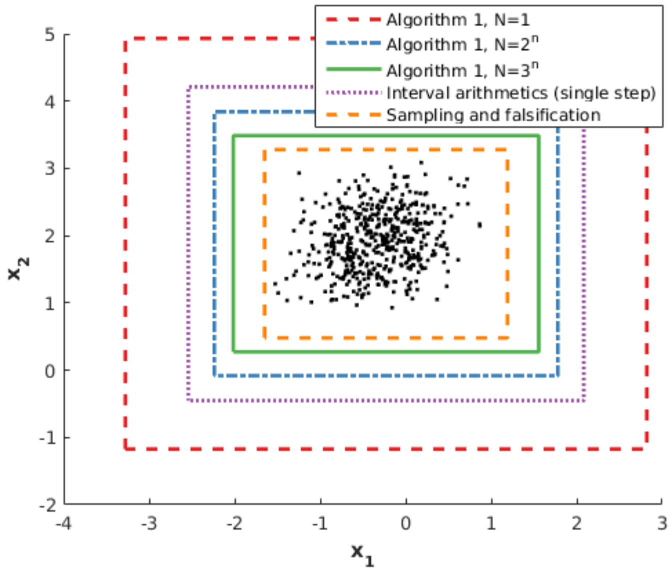

Fig. 3. Comparison of over-approximations of the reachable set of (14) at time t_f for states x_1 and x_2.

the first-order sensitivity matrix to obtain a guaranteed over-approximation of its reachable set, which is in turn used to over-approximate the reachable set of the initial system. Although in the general case, the proposed method has an exponential complexity in the state dimension due to the gridded sampling, its main strength is its flexibility allowing the user to tune the desired tradeoff between conservativeness and computational cost. Indeed within the same method, we can either pick a single sample point to obtain a more conservative result but with a very low complexity when computational power is limited, or increase the size of the sampling set to tighten the over-approximation if more computational power is available.

Current efforts are focused on the integration of this new reachability algorithm within the recently published toolbox TIRA (Meyer et al., 2019) which gathers several other interval reachability methods. Future work will aim to propose more efficient sampling criteria guided by the obtained bounds on the second-order sensitivity to tighten the over-approximations at a lesser computational cost compared to the current uniform gridding.

REFERENCES

Althoff, M. (2015). An introduction to CORA 2015. In *ARCH@ CPSWeek*, 120–151.

Althoff, M. and Krogh, B.H. (2011). Zonotope bundles for the efficient computation of reachable sets. In *50th IEEE Conference on Decision and Control and European Control Conference*, 6814–6821.

Althoff, M., Stursberg, O., and Buss, M. (2007). Reachability analysis of linear systems with uncertain parameters and inputs. In *46th IEEE Conference on Decision and Control*, 726–732.

Angeli, D. and Sontag, E.D. (2003). Monotone control systems. *IEEE Transactions on Automatic Control*, 48(10), 1684–1698.

Blanchini, F. and Miani, S. (2008). *Set-theoretic methods in control.* Springer.

Chen, X., Abraham, E., and Sankaranarayanan, S. (2012). Taylor model flowpipe construction for non-linear hybrid systems. In *IEEE 33rd Real-Time Systems Symposium*, 183–192.

Cheng, D., Qi, H., and Zhao, Y. (2012). *An introduction to semi-tensor product of matrices and its applications.* World Scientific.

Coogan, S. and Arcak, M. (2015). Efficient finite abstraction of mixed monotone systems. In *18th International Conference on Hybrid Systems: Computation and Control*, 58–67.

Donzé, A. and Maler, O. (2007). Systematic simulation using sensitivity analysis. In *International Workshop on Hybrid Systems: Computation and Control*, 174–189.

Girard, A. and Le Guernic, C. (2008). Efficient reachability analysis for linear systems using support functions. *IFAC Proceedings Volumes*, 41(2), 8966–8971.

Jaulin, L. (2001). *Applied interval analysis: with examples in parameter and state estimation, robust control and robotics*, volume 1. Springer Science & Business Media.

Kurzhanskiy, A.A. and Varaiya, P. (2007). Ellipsoidal techniques for reachability analysis of discrete-time linear systems. *IEEE Transactions on Automatic Control*, 52(1), 26–38.

Meyer, P.J., Coogan, S., and Arcak, M. (2018). Sampled-data reachability analysis using sensitivity and mixed-monotonicity. *IEEE Control Systems Letters*, 2(4), 761–766.

Meyer, P.J., Devonport, A., and Arcak, M. (2019). TIRA: Toolbox for interval reachability analysis. In 22^{nd} *ACM International Conference on Hybrid Systems: Computation and Control*, 224–229.

Meyer, P.J. and Dimarogonas, D.V. (2019). Hierarchical decomposition of LTL synthesis problem for nonlinear control systems. *IEEE Transactions on Automatic Control*, 64(11), 4676–4683.

Moor, T. and Raisch, J. (2002). Abstraction based supervisory controller synthesis for high order monotone continuous systems. In *Modelling, Analysis, and Design of Hybrid Systems*, 247–265.

Reissig, G., Weber, A., and Rungger, M. (2016). Feedback refinement relations for the synthesis of symbolic controllers. *IEEE Transactions on Automatic Control*, 62(4), 1781–1796.

Scott, J.K. and Barton, P.I. (2013). Bounds on the reachable sets of nonlinear control systems. *Automatica*, 49(1), 93–100.

Tempo, R., Calafiore, G., and Dabbene, F. (2012). *Randomized algorithms for analysis and control of uncertain systems: with applications.* Springer Science & Business Media.

Yang, L., Mickelin, O., and Ozay, N. (2019). On sufficient conditions for mixed monotonicity. *IEEE Transactions on Automatic Control.*