Experimental Evaluation of Methods of Reachable Set Computation in Context of Symbolic Controller Synthesis

A DISSERTATION

Submitted in partial fulfillment of the

requirements for the award of the degree

of

MASTER OF TECHNOLOGY

in

DEPARTMENT OF ELECTRICAL ENGINEERING

(With specialization in Systems and Control)

By

Mahendra Singh Tomar

DEPARTMENT OF ELECTRICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY ROORKEE

ROORKEE - 247 667 (INDIA)

MAY, 2018

CANDIDATE'S DECLARATION

I hereby declare that this thesis report entitled **Experimental Evaluation of Methods of Reachable Set Computation in Context of Symbolic Controller Synthesis**, submitted to the Department of Electrical Engineering, Indian Institute of Technology, Roorkee, India, in partial fulfillment of the requirements for the award of the Degree of Master of Technology in Electrical Engineering with specialization in Systems and Control is an authentic record of the work carried out by me during the period from September 2017 to April 2018 under the supervision of **Dr. Matthias Rungger, Dr. Majid Zamani, (TU Munich) and Dr. G. N. Pillai, IIT Roorkee**. The matter presented in this thesis report has not been submitted by me for the award of any other degree of this institute or any other institute.

Date: 16-05-2018
Place: Roorkee

Mahendra Singh Tomar
(Enr. N0. 16530009)

CERTIFICATE

This is to certify that the above statement made by the candidate is true to the best of my knowledge and belief.

Dr. G. N. PILLAI
Professor
Department of Electrical Engineering
Indian Institute of Technology, Roorkee.

ABSTRACT

The design of hardware and software for safety critical applications often involves use of formal verification techniques to ensure correctness of implementation. If specification fails to satisfy then the cycle of design and verification needs to be repeated. Lengthy and costly design process can be avoided by the use of automated controller synthesis techniques which can provide correct-by-construction controller that enforces the desired formal specification on the given system. Use of synthesis techniques with continuous systems require construction of a symbolic model which in turn require computation of reachable sets. The set of states that can be attained by a system for given values of initial state, input and time horizon is referred to as reachable state set.

As reachable sets are infinite objects their exact computation is difficult, thus their overapproximations are usually computed whose accuracy affects the computation time and size of the synthesized controller. The choice of set representation also plays an important role in accuracy of approximation and time requirement. We used zonotopes which are centrally symmetric convex polytopes and are closed under linear transformation and Minkowski sum.

After a short introduction to the process of symbolic model construction we discuss three methods of reachable set computation for nonlinear control systems. These are then used together with SCOTS, which is a tool for symbolic controller synthesis, to synthesize controllers for reachability and invariance specifications on four examples.

Acknowledgements

This is an opportunity to express my sense of gratitude for the support and encouragement received from my Indian supervisor Dr. G. N. Pillai, IIT Roorkee. Under his kind guidance I got the opportunity to do part of the master thesis at Hybrid Control Systems group, TU Munich. I am thankful to Dr. Majid Zamani for his kindness and interest in the progress of the work. The time and knowledge shared by Dr. Matthias Rungger together with his patience towards my pace of learning have been crucial to this work. Further I would like to thank Pushpak Jagtap for the helpful discussions both academic and otherwise.

Mahendra Singh Tomar

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Symbolic models (also referred to as discrete abstraction) allow to use automated synthesis techniques to obtain correct by construction controller that enforces some given specification, say expressed in linear temporal logic (LTL), on a system [1–4]. The designed controller does not need any separate verification phase, as there comes a formal guarantee of satisfaction of the specification. Synthesis refers to the generation of system description based on desired behavior specified in some formal language. The synthesis process merges the design and verification steps which are otherwise two distinct steps in the usual design-verify methodology involving multiple verification rounds. This helps to control both time and cost.

Symbolic model as a finite state machine can be obtained by use of reachable state sets which refer to the set of states that can be attained by a system from it's given initial state set under a given set of input over a given time. Consider the Van der Pol oscillator dynamics taken from [5]

$$\begin{aligned} \dot{x_1} &= x_2 \\ \dot{x_2} &= x_2 - x_1 - x_1^2 x_2 \end{aligned} \tag{1.1}$$

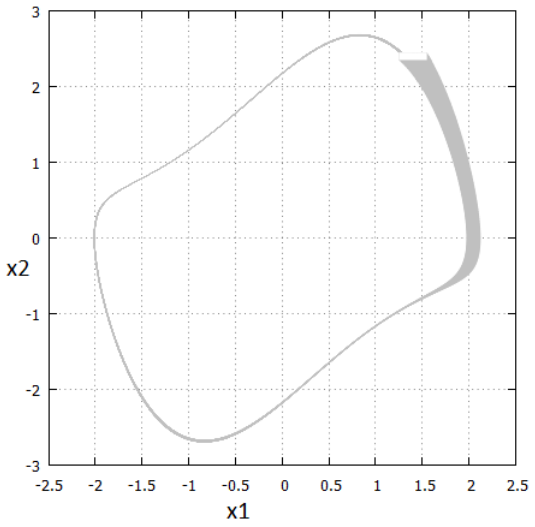

FIGURE 1.1: An overapproximation of reachable set for Van der Pol oscillator.

with initial state $X_0 = [1.25, 1.55] \times [2.35, 2.45]$ and time duration 0 to 7. An overapproximation of its reachable set is shown in grey in Figure 1.1.

As reachable sets are infinite objects their exact computation is difficult, thus their overapproximations are usually computed. Overapproximation of reachable set of nonlinear systems to any desired accuracy is investigated in [6]. Two lines of research can be seen in approximate reachability computation. One concerns the computation working directly with the original dynamics of the system [7, 8], while in the second reachable sets are computed using abstracted models which are simplified representation of the original system [9, 10]. To get abstracted model of nonlinear system the state space may be partitioned to get local abstractions. Abstraction to piecewise linear systems together with fixed structure partitioning of state space is developed in [11]. This is also referred to as hybridization, as each area of linearization has it's own particular dynamics. A disadvantage of fixed partitioning is the exponential increase in the number of partitions with the number of states. This can be reduced by the use of on-the-fly partitioning [12] where partitions are resorted to only when required.

Use of Taylor model [13] (polynomial function together with an interval remainder) to compute reachable set for nonlinear systems is discussed in [14]. Flow* [15] which is a tool for verification of cyber physical systems is based on Taylor models. For application of zonotopes and support functions as geometric structures for set representation in reachability analysis of linear systems refer respectively

[16], [17]. The choice of set representation plays an important role in accuracy of overapproximation and time requirement. For example, we can get arbitrary accuracy in approximation of reachable set for linear systems by use of general polytopes. But their Minkowski sum results in polytopes of greater complexity. Set representation as ellipsoids or parallelotopes are not closed under Minkowski sum and thus further approximations are required. Error due to these sequential approximations keep on adding up and result in greater conservativeness of the final set. Such an increase in error is called wrapping effect [18]. Use of zonotopes is advantageous as they are closed under linear transformation and Minkowski sum, and these operations can be easily and efficiently computed.

The aim of this work was to implement and compare some of the methods of computation of reachable set in context of symbolic controller synthesis. For this we made use of SCOTS [19] which is a software tool for controller synthesis of systems with nonlinear dynamics. As input, the tool requires the system description in the form of differential equation, together with discretization parameters for computation of symbolic model which is related under feedback refinement relation (FRR) with the given system. FRR ensures that the controller designed for symbolic model also enforces the given specification on the original system.

With better accuracy in approximation of reachable set we obtain lower number of transitions in the symbolic model which results in faster controller synthesis. It also result in larger controller domain i.e., we have control input available for a larger area of state space.

SCOTS currently uses the method of growth bound to compute reachable set. The method is compared with two other methods, both based on conservative linearization, one of them implemented in the toolbox CORA [20] that involves reachability algorithms, and the other in SpaceEx [21] which is a verification platform for hybrid systems.

Among numerous other applications reachable sets are also used in verification of safety properties [5] of nonlinear and hybrid control systems. The set of states which are undesired and to be avoided are referred to as unsafe set. If intersection

of overapproximation of reachable set with the unsafe set is empty then safety is ensured.

Chapter 2 present some basic terms and notations involved. In Chapter 3 we present a simplified version of the process in which symbolic models are constructed in SCOTS. In Chapter 4 three methods for reachable set computation of nonlinear control systems are discussed. Chapter 5 present the results and conclusion.

Chapter 2

Notations and Preliminaries

2.1 Notations

- Interval vectors and matrices denoted by Zapf Chancery letters ($\mathcal{A}, \mathcal{B}, \ldots$). By an interval matrix we mean a matrix with interval components.
- Zonotopes denoted by raised $\mathcal{Z}$ ($A^{\mathcal{Z}}, B^{\mathcal{Z}}, \ldots$)

$\mathbb{N}$	set of natural numbers {0, 1, 2 ... }
$\mathbb{R}$	set of real valued numbers
$\mathbb{R}_+$	set of non-negative real valued numbers
$\mathbb{IR}$	set of real valued intervals
$\mathbb{IR}^{n\times m}$	set of $n \times m$ real interval matrices

2.2 Interval Computations

For two intervals $a = [\underline{a}, \overline{a}] \in \mathbb{IR}$ and $b = [\underline{b}, \overline{b}] \in \mathbb{IR}$, the addition and multiplication operations are defined as

$$\begin{aligned} a + b &= [\underline{a} + \underline{b}, \overline{a} + \overline{b}] \\ ab &= [min\{\underline{ab}, \underline{a}\overline{b}, \overline{a}\underline{b}, \overline{ab}\}, max\{\underline{ab}, \underline{a}\overline{b}, \overline{a}\underline{b}, \overline{ab}\}] \end{aligned} \tag{2.1}$$

2.3 Matrix Exponential

The matrix exponential e^{At} can be overapproximated by $e_p^{At} \in \mathbb{IR}^{n\times n}$ using the first p terms of the Taylor series and an interval bound for the remainder [22]. For $A \in \mathbb{R}^{n\times n}$, and some p satisfying $\epsilon < 1$

$$e^{At} \in e_p^{At} = \sum_{i=0}^{p} \frac{(At)^i}{i!} + \mathcal{E}(t), \tag{2.2}$$

where

$$\begin{aligned} \mathcal{E}(t) &= \frac{(\|A\|_\infty t)^{p+1}}{(p+1)!}\frac{1}{1-\epsilon}[-\mathbf{1}, \mathbf{1}], \\ \epsilon &= \frac{\|A\|_\infty t}{p+2} < 1 \end{aligned} \tag{2.3}$$

$\mathbf{1} \in \mathbb{R}^{n\times n}$ is a matrix of ones, $[-\mathbf{1}, \mathbf{1}] \in \mathbb{IR}^{n\times n}$

2.4 Zonotopes

We begin with definitions of some geometric structures which are helpful to visualize zonotopes.

Definition 2.1. (Hyperplane). A hyperplane H(f,d) is a set that can be represented as

$$H(f, d) = \left\{x \in \mathbb{R}^n : f^T x = d\right\} \tag{2.4}$$

where $f \in \mathbb{R}^n, d \in \mathbb{R}$.

Definition 2.2. (Half-space). A hyperplane divides the Euclidean space into two open half-spaces.

$$\mathrm{Hs}(f, d) = \left\{x \in \mathbb{R}^n : f^T x < d\right\} \tag{2.5}$$

Definition 2.3. (Polyhedra). A convex polyhedra $P(F, d)$ is the region formed by intersection of a finite set of half-spaces [23]

$$P(F, D) = \{x \in \mathbb{R}^n : F_i x \leq d_i, i = 1, \ldots, m\} \tag{2.6}$$

where F_i is the i^{th} row of the matrix $F \in \mathbb{R}^{m\times n}$ and $D = [d_1 \dots d_m]^T$.

Polytopes are bounded polyhedra i.e., bounded intersection of finite number of half-spaces. Zonotopes are centrally symmetric convex polytopes, which can be represented by a center and a set of generators.

Definition 2.4. (Zonotope). A zonotope $Z^{\mathcal{Z}}$ is a set such that:

$$Z^{\mathcal{Z}} = \left\{ x \in \mathbb{R}^n : x = c + \sum_{i=1}^{q} \beta^{(i)} g^{(i)}, -1 < \beta^{(i)} < 1 \right\} \tag{2.7}$$

$$= (c, g^{(1)}, \dots, g^{(q)}) \tag{2.8}$$

where $c \in \mathbb{R}^n$ is the center, q is the number of generators, and $g^{(i)} \in \mathbb{R}^n$ is the i^{th} generator.

Number of vertices $\leq 2\sum_{i=0}^{n-1} \binom{q-1}{i}$
Number of facets $\leq 2\binom{q}{n-1}$

- **A 2D example:** Consider a two dimensional zonotope having three generators.

$$Z = \left(\underbrace{\begin{bmatrix} 1 \\ 2 \end{bmatrix}}_{c}, \underbrace{\begin{bmatrix} 0.5 \\ 0 \end{bmatrix}}_{g^{(1)}}, \underbrace{\begin{bmatrix} 0 \\ 0.5 \end{bmatrix}}_{g^{(2)}}, \underbrace{\begin{bmatrix} 0.7 \\ 0.7 \end{bmatrix}}_{g^{(3)}} \right)$$

  Figure 2.1(a) shows the zonotope which is a line segment obtained when only $g^{(1)}$ is considered. It is as if the centre is traversed in both directions parallel to the vector $g^{(1)}$ by a distance equal to its magnitude. When $g^{(2)}$ is included its direction and magnitude decide the direction along which the line segment is to be traversed and by what amount. Result is shown in Figure 2.1. The complete zonotope is shown in Figure 2.1(d).

- Absolute value of zonotope $Z^{\mathcal{Z}} = (c, g^{(1)}, \dots, g^{(q)})$ is computed as

$$\left|Z^{\mathcal{Z}}\right| = |c| + \sum_{i=1}^{q} \left|g^{(i)}\right| \tag{2.9}$$

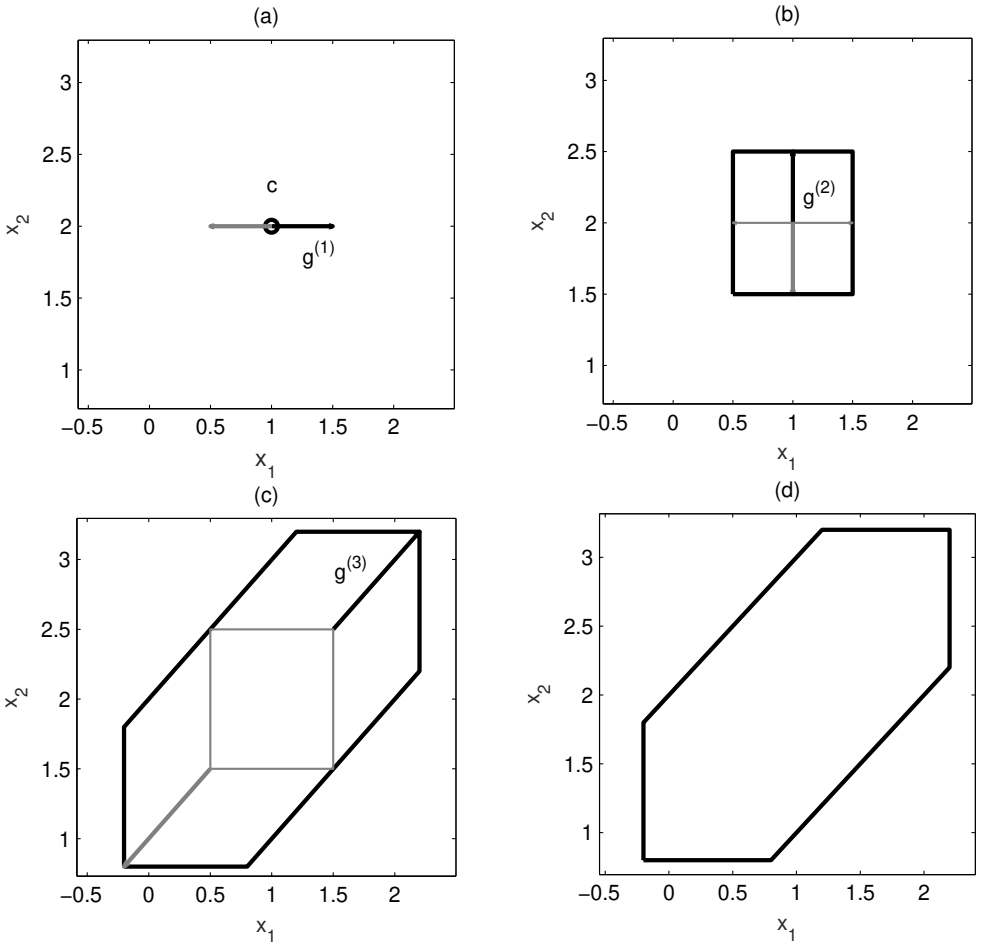

FIGURE 2.1: Construction of a 2D zonotope with 3 generators.

2.4.1 Minkowski sum

Minkowski sum of two sets $A, B \subset \mathbb{R}^n$ is

$$A + B = \{a + b : a \in A, b \in B\} \tag{2.10}$$

Zonotopes are closed under Minkowski sum, i.e. the result is also a zonotope. For two zonotopes $Z_a^{\mathcal{Z}} = (c_a, g_a^{(1)}, \ldots, g_a^{(q)}), \quad Z_b^{\mathcal{Z}} = (c_b, g_b^{(1)}, \ldots, g_b^{(m)})$ the Minkowski sum can be obtained by addition of centers and concatenation of the generators,

$$Z_a^{\mathcal{Z}} + Z_b^{\mathcal{Z}} = \left(c_a + c_b, g_a^{(1)}, \ldots, g_a^{(q)}, g_b^{(1)}, \ldots, g_b^{(m)}\right) \tag{2.11}$$

2.4.2 Order of Zonotope

Given a zonotope $Z^{\mathcal{Z}} = (c, g^{(1)}, \ldots, g^{(q)}) \subseteq \mathbb{R}^n$, having q generators, its order is defined to be $\frac{q}{n}$. Sequence of operations such as Minkowski addition keep on increasing the order of the resulting zonotope. With this, memory requirement and computation time may also increase. Thus arises the need to control the order of the involved zonotopes.

Zonotope order reduction:
For a zonotope $Z^{\mathcal{Z}} = (c, g^{(1)}, \ldots, g^{(q)})$ in $\mathbb{R}^n$, if its order is to be reduced to m, where $\frac{q}{n} > m$, $m \in \{1, 2, \ldots, \}$ then $p = q - (n(m-1))$ generators are replaced by n generators such that the origin centered zonotope constituted by the former group of generators is included in that by the latter [16]. Result will be a zonotope with mn generators and thus of order m. To select the group of generators that is to be replaced, difference of infinity norm from the unit norm for each generator is sorted

$$\left\|g^{(1)}\right\|_1 - \left\|g^{(1)}\right\|_\infty \leq \ldots \leq \left\|g^{(q)}\right\|_1 - \left\|g^{(q)}\right\|_\infty$$

Then the first p generators are selected as they are similar to vectors which are parallel to one of the axes and thus the origin centered zonotope formed by them is well overapproximated by an interval hull. If $h^{(1)}, \ldots, h^{(p)}$ are the selected generators then the group is replaced by the generators of the zonotope representation of the interval hull of the zonotope $(0, h^{(1)}, \ldots, h^{(p)})$.

2.4.3 Matrix multiplication

Zonotopes are closed under linear transformation. For a zonotope $Z^{\mathcal{Z}} \subset \mathbb{R}^n$ the linear map described by a matrix $L \in \mathbb{R}^{m \times n}$ results in the zonotope

$$\begin{aligned} LZ^{\mathcal{Z}} &= \{x \in \mathbb{R}^m : x = Lc + \sum_{i=1}^{q} \beta^{(i)} L g^{(i)}, -1 \leq \beta^{(i)} \leq 1\} \\ &= (Lc, Lg^{(1)}, \ldots, Lg^{(q)}) \end{aligned} \tag{2.12}$$

2.4.4 Interval matrix multiplication

Let M be an interval matrix which can be expressed as $\mathcal{M} = \widetilde{M} + [-\hat{M}, \hat{M}]$ where $[-\hat{M}, \hat{M}]$ is a symmetric interval matrix, $\widetilde{M}, \hat{M} \in \mathbb{R}^{n\times n}$. A zonotope over-approximation of the multiplication of $\mathcal{M}$ with $Z^{\mathcal{Z}}$ can be obtained as [24]

$$\begin{aligned} \mathcal{M}Z^{\mathcal{Z}} &= \left(\widetilde{M}c, \widetilde{M}g^{(1)}, \ldots, \widetilde{M}g^{(q)}, v^{(1)}, \ldots, v^{(n)}\right) \\ v_j^{(i)} &= \begin{cases} 0, & i \neq j \\ \hat{M}_j \left(|c| + \sum_{k=1}^{q} |g|^{(k)}\right), & i = j \end{cases} \end{aligned} \tag{2.13}$$

2.4.5 Convex hull

The convex hull of two sets $A, B \in \mathbb{R}^n$ is the smallest convex set that contains both of them. In general, the convex hull of two zonotopes is no more a zonotope. The zonotope which over-approximates the convex hull of $Z_a^{\mathcal{Z}} = (c_a, g^{(1)}, \ldots, g^{(q)})$ and $Z_b^{\mathcal{Z}} = (c_b, f^{(1)}, \ldots, f^{(m)})$, $m \geq q$ can be obtained as [16]

$$\begin{aligned} \mathrm{CH}(Z_a^{\mathcal{Z}}, Z_b^{\mathcal{Z}}) = &\frac{1}{2}\Big(c_a + c_b, g^{(1)} + f^{(1)}, \ldots, g^{(q)} + f^{(q)}, \\ &c_a - c_b, g^{(1)} - f^{(1)}, \ldots, g^{(q)} - f^{(q)}, \\ &2f^{(q+1)}, \ldots, 2f^{(m)}\Big) \end{aligned} \tag{2.14}$$

An example of zonotope overapproximation of convex hull of two 2-dimensional zonotopes is given in Figure 2.2.

2.4.6 Interval hull

Interval hull, i.e. the axis-aligned smallest enclosing box, of a zonotope $Z^{\mathcal{Z}} \subset \mathbb{R}^n$ can be obtained as

$$\mathrm{IH}(Z^{\mathcal{Z}}) = \left[\underline{\eta}, \overline{\eta}\right] \in \mathbb{IR}^n \tag{2.15}$$

where, $\underline{\eta}, \overline{\eta} \in \mathbb{R}^n, \underline{\eta} = c - \sum_{i=1}^{q} \left|g^{(i)}\right|, \overline{\eta} = c + \sum_{i=1}^{q} \left|g^{(i)}\right|$, absolute values taken element wise. The interval $\left[\underline{\eta}, \overline{\eta}\right]$ can be represented as a zonotope $\eta^{\mathcal{Z}} = (c_\eta, g_\eta^{(1)}, \ldots, g_\eta^{(q)})$,

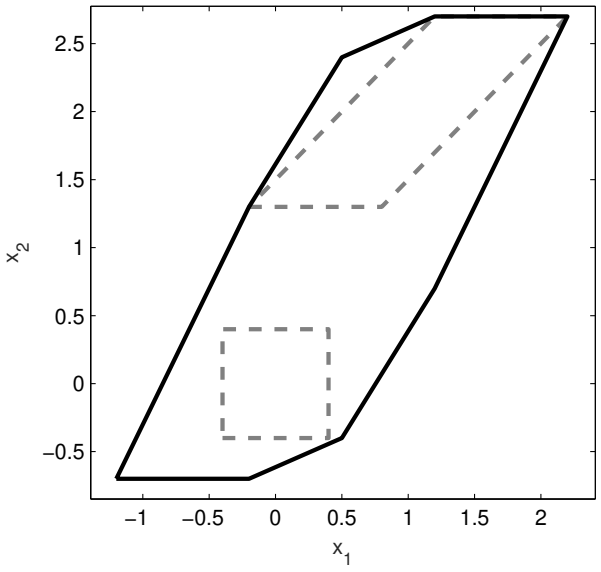

FIGURE 2.2: Convex hull of two zonotopes.

where $c_\eta = 0.5(\underline{\eta} + \overline{\eta})$, and

$$g_{\eta j}^{(i)} = \begin{cases} 0, & i \neq j \\ 0.5(\overline{\eta}_j - \underline{\eta}_j), & i = j \end{cases}, i \in \{1, \ldots, q\}, j \in \{1, \ldots, n\}.$$

Each generator of $\eta^{\overline{z}}$ has only one non zero element.

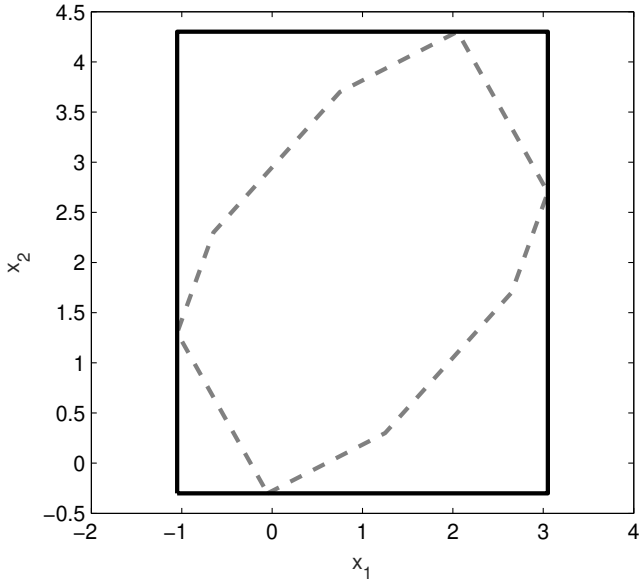

FIGURE 2.3: Interval Hull of a two dimensional zonotope.

2.4.7 Quadratic Map

For matrices $Q^{(i)} \in \mathbb{R}^{n\times n}$, $(i = 1, \ldots, n)$ and a zonotope $Z^{\mathcal{Z}} = (c, g^{(1)}, \ldots, g^{(q)})$, a zonotope overapproximation of the set

$$Z_Q = \left\{\phi \mid \phi_i = x^T Q^{(i)} x, x \in Z^{\mathcal{Z}}\right\}$$

can be obtained as [25]

$$\text{quad}(Q, Z^{\mathcal{Z}}) = \left(d, h^{(1)}, \ldots, h^{(m)}\right) \tag{2.16}$$

where $m = \begin{pmatrix} q+2 \\ 2 \end{pmatrix} - 1$,

$$\begin{aligned}
&d_i = c^T Q^{(i)} c + 0.5 \sum_{s=1}^{q} g^{(s)^T} Q^{(i)} g^{(s)} \\
&j = 1, \ldots, q: \quad h_i^{(j)} = c^T Q^{(i)} g^{(j)} + g^{(j)^T} Q^{(i)} c \\
&j = 1, \ldots, q: \quad h_i^{(q+j)} = 0.5 g^{(j)^T} Q^{(i)} g^{(j)} \\
&l = \sum_{j=1}^{q-1} \sum_{k=j+1}^{q} 1 : h_i^{(2q+l)} = g^{(j)^T} Q^{(i)} g^{(k)} + g^{(k)^T} Q^{(i)} g^{(j)}
\end{aligned}$$

Proof.

$$Z_Q = \left\{\phi \mid \phi_i = x^T Q^{(i)} x, x \in Z^{\mathcal{Z}}\right\}$$

As x is any point inside $Z^{\mathcal{Z}} = \left\{c + \sum_{j=1}^{q} \beta^{(j)} g^{(j)}, -1 \leq \beta^{(j)} \leq 1\right\}$, it can be substituted as

$$= \left\{\phi \mid \phi_i = \left(c + \sum_{j=1}^{q} \beta^{(j)} g^{(j)}\right)^T Q^{(i)} \left(c + \sum_{j=1}^{q} \beta^{(j)} g^{(j)}\right), -1 \leq \beta^{(j)} \leq 1\right\}$$

on rearranging we get

$$
\begin{aligned}
Z_Q = \Bigg\{ & \phi | \phi_i = \underbrace{c^T Q^{(i)} c + \sum_{j=1}^{q} 0.5 g^{(j)^T} Q^{(i)} g^{(j)}}_{d_i} \\
& + \sum_{j=1}^{q} \beta^{(j)} \underbrace{\left(c^T Q^{(i)} g^{(j)} + g^{(j)^T} Q^{(i)} c \right)}_{h_i^{(j)}} \\
& + \sum_{j=1}^{q} (2(\beta^{(j)})^2 - 1) \underbrace{0.5 g^{(j)^T} Q^{(i)} g^{(j)}}_{h_i^{(q+j)}} \\
& + \sum_{j=1}^{q-1} \sum_{k=j+1}^{q} \beta_j \beta_k \underbrace{\left(g^{(j)^T} Q^{(i)} g^{(k)} + g^{(k)^T} Q^{(i)} g^{(j)} \right)}_{h_i^{(2q+l)}}, \beta_i \in [-1, 1] \Bigg\} \\
& \subseteq \left(d, h^{(1)}, \ldots, h^{(m)} \right)
\end{aligned}
$$

Resulting zonotope is an overapproximation as $\beta^{(j)} \in [-1, 1]$ which if replaced by the interval appears multiple times in the same expression. □

Chapter 3

Construction of Abstraction

3.1 Systems and relations

Definition 3.1. (Nonlinear control system). A nonlinear control system is a tuple $\Sigma = (\mathbb{R}^n, U, \mathcal{U}, f)$, where $\mathbb{R}^n$ is the state space, $U \subseteq \mathbb{R}^m$ is a bounded input set, $\mathcal{U}$ is a subset of set of all functions of time from R_0^+ to U, and f is a locally Lipschitz continuous function from $\mathbb{R}^n \times U$ to $\mathbb{R}^n$.

The trajectory ξ is said to be a solution of Σ if there exists $\upsilon \in \mathcal{U}$ satisfying:

$$\dot{\xi}(t) = f(\xi(t), \upsilon(t)), \tag{3.1}$$

for any $t \in \mathbb{R}_0^+$. Existence and uniqueness of the solution ξ is ensured by the assumption of locally Lipschitz continuity. The value of the solution at time t under the input signal υ and starting from initial condition x is represented by $\xi_{x,\upsilon}(t)$.

Next we define transition system which will be used to represent both the sampled nonlinear control system and the symbolic model.

Definition 3.2. (Transition system). A transition systems is a tuple $S = (X, X_0, U, \longrightarrow)$ where X is a set of states, $X_0 \subseteq X$ is a set of initial states, U is a set of inputs, $\longrightarrow \subseteq X \times U \times X$ is a transition relation.

Definition 3.3. (Sampled control system as a transition system). For sampling time $\tau \in \mathbb{R}^+$, the sampled nonlinear control system is a tuple $S_\tau(\Sigma) = (X_\tau, X_{\tau 0}, U_\tau, \longrightarrow_\tau)$, where $X_\tau = \mathbb{R}^n, X_{\tau 0} \subseteq X_\tau, U_\tau = U$, the transition relation is defined as

$$x_\tau \xrightarrow{u}_\tau x'_\tau \text{ iff } x'_\tau = \xi_{x_\tau,u}(\tau),$$

where the input signal $\upsilon(t) = u \in U_\tau$ for $t \in [k\tau, (k+1)\tau]$, i.e. $\upsilon(t)$ is piecewise constant.

In rest of the work we will consider only piecewise constant input signal, therefore we will use the notation $\xi_{x,u}(t)$ to represent the value of solution at time t starting from initial condition x under a input signal kept fixed at u.

Only the states reached at instants $k\tau, k \in \mathbb{N}$ are related under the transition relation $\longrightarrow_\tau$.

Definition 3.4. (Feedback refinement relation, FRR). Consider two transition systems $S_1 = (X_1, X_{10}, U_1, \longrightarrow_1)$ and $S_2 = (X_2, X_{20}, U_2, \longrightarrow_2)$ having $U_2 \subseteq U_1$. A strict relation $Q \subseteq X_1 \times X_2$ is a feedback refinement relation from S_1 to S_2 if following holds for every pair $(x_1, x_2) \in Q$:

(i) $U_2(x_2) \subseteq U_1(x_1)$,

(ii) $u \in U_2(x_2) \Rightarrow Q(Post_u(x_1)) \subseteq Post_u(x_2)$,

and the feedback refinement relation from S_1 to S_2 is denoted by $S_1 \preceq_Q S_2$. $Post_u(x)$ is the set of all u-successors of state x.

3.2 Symbolic model

Definition 3.5. (Symbolic model). For state space quantisation $\eta \in \mathbb{R}^n$ and input space quantisation $\mu \in \mathbb{R}^m$, a symbolic model of $S_\tau(\Sigma)$ is given by a tuple

$S_q(\Sigma) = (X_q, X_{q0}, U_q, \longrightarrow_q)$, where $X_q = [\mathbb{R}^n]_\eta$ is the quantised state space, $X_{q0} \subseteq X_q$, $U_q = [U]_\mu$ is the quantised input space, x_q and $x'_q \in X_q$, $x_q \xrightarrow{u}_q x'_q$ iff $x'_q \cap R_\tau(x_q, u) \neq \emptyset$. $R_\tau(x_q, u) = \bigcup_{x \in x_q} \xi_{x,u}(\tau)$ is the reachable set at time τ starting from initial state x_q moving along system trajectory under input u.

Each element of the quantised state space X_q and the quantised input space U_q is a hyper-interval whose size is determined by the vector η and the vector μ respectively. Thus we have a grid of cells in both the spaces. Centres of the cells in the input space will be considered as the available inputs.

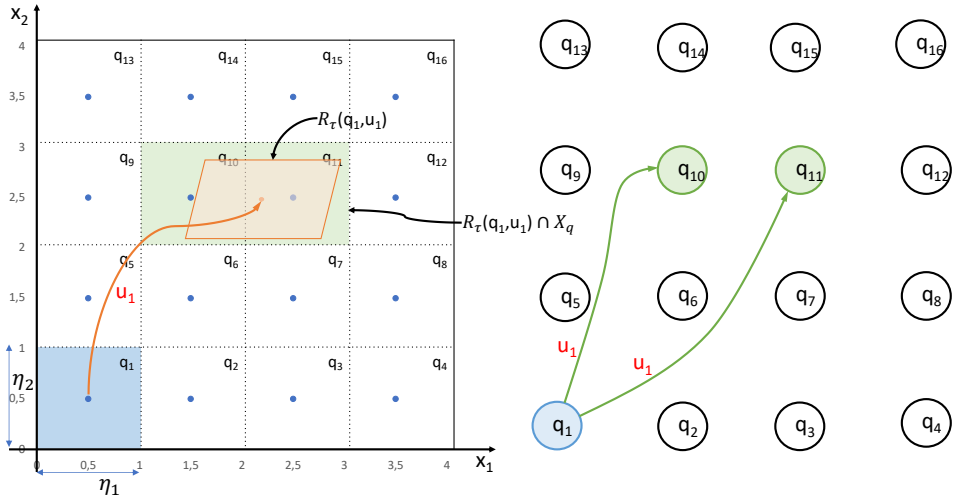

FIGURE 3.1: Construction of symbolic model (a)

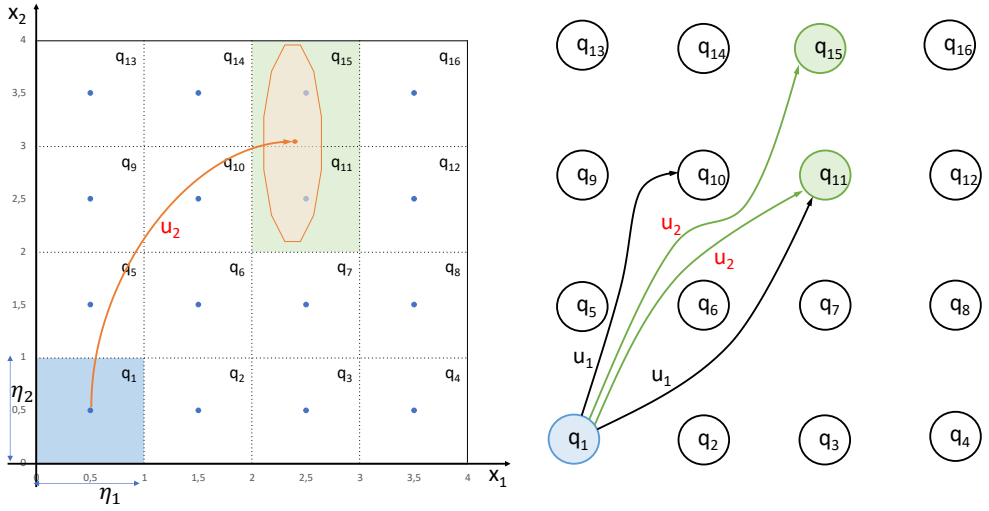

FIGURE 3.2: Construction of symbolic model (b)

Figure 3.1 to 3.4 depicts a simplified construction of symbolic model of a two dimensional system with one dimensional input and two partitions in the input

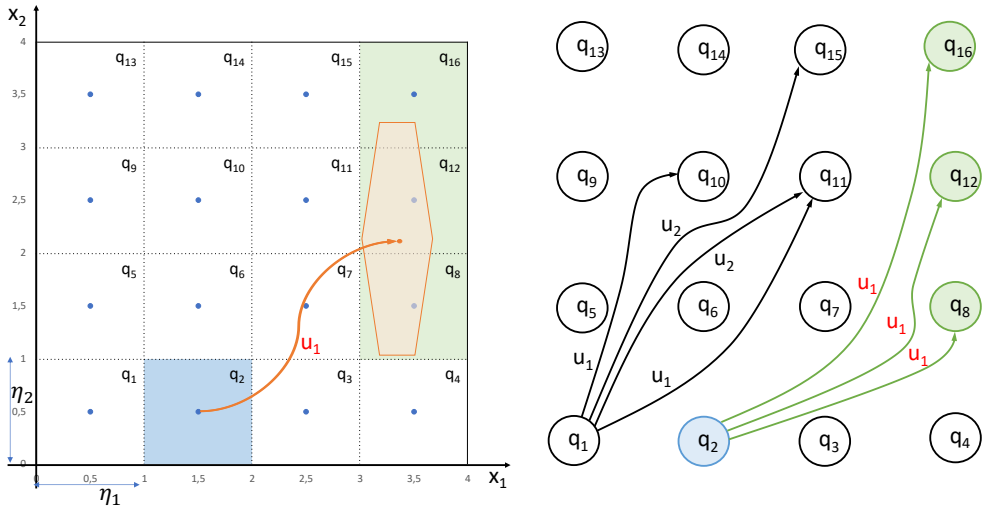

FIGURE 3.3: Construction of symbolic model (c)

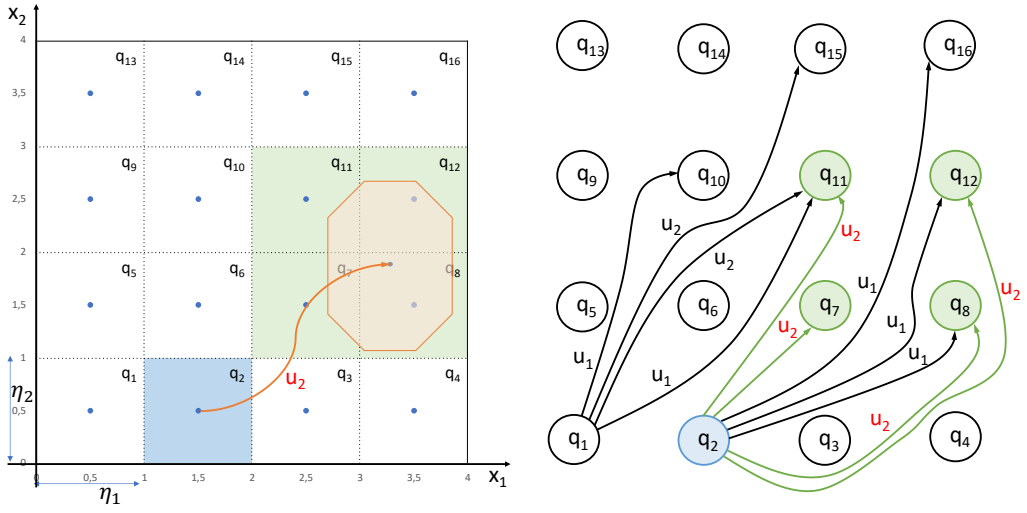

FIGURE 3.4: Construction of symbolic model (d)

space. For partition q_1 in state space, the reachable set under input u_1 at time τ, $R_\tau(q_1, u_1)$ together with its intersection with the state space grid is shown in Figure 3.1. In the symbolic model, all the cells which intersect the reachable set have a transition from the state q_1. The process is repeated for every combination of cells from the state and the input space.

Theorem 3.6. *If $S_q(\Sigma)$ is a symbolic model of $S_\tau(\Sigma)$ then $S_\tau(\Sigma) \preceq_Q S_q(\Sigma)$.*

The theorem says that for a nonlinear control system Σ the sampled system $S_\tau(\Sigma)$ as per the definition 3.3 and the symbolic model $S_q(\Sigma)$ as per the definition 3.5 are related under the feedback refinement relation.

To construct symbolic model of a continuous time nonlinear system S, we first obtain sampled system S_τ. Sampling time should not be extremely small, as otherwise it may result in self-loops in the finite state model which may lead to an empty controller. If sampling time is too large then it may adversely affect the accuracy of approximation of reachable sets. For symbolic model the state space and the input space are quantized. Quantisation parameter if too small will result in very large number of cell partitions which will increase the computation time. Symbolic model obtained as per the definition 3.5 is related under FRR with the sampled system. Here the set membership relation acts as the FRR. Now a controller C_2 is synthesized that enforces the given specification on the symbolic model. Usually this controller needs to be refined to another controller C_1 such that C_1 enforces the specification on the original system. The refinement step may introduce dynamics into the controller which means that the controller itself will contain a symbolic model, thus its complexity rises. With the use of FRR the set membership relation acts as a static quantizer.

Physical system $\longrightarrow$ sampled system and symbolic model related under feedback refinement relation $\longrightarrow$ controller for the symbolic model such that specification enforced $\longrightarrow$ this controller together with a quantizer will also enforce the specification on the physical system.

Chapter 4

Reachable set over approximation

In this chapter, we will consider three methods to compute over-approximation of reachable set.

4.1 Growth bound

Consider a nonlinear system $\Sigma = (\mathbb{R}^n, U, \mathcal{U}, f)$ as per Definition (3.1). Let $U' \subseteq U$, $K \subseteq K' \subseteq \mathbb{R}^n$ where K' is convex. K' called a priori enclosure is such that for any $\upsilon(t) = u \in U'$, $x \in K$, $t \in [0, \tau]$, we have $\xi_{x,u}(t) \in K'$, i.e. all trajectories of duration τ starting from K under input u lie within K'.

Definition 4.1. A function $\beta : \mathbb{R}^n_+ \times U' \to \mathbb{R}^n_+$ is a growth bound [2] of Σ defined with respect to τ, K and U', and can be given as

$$\beta(r, u) = e^{L(u)\tau} r \tag{4.1}$$

where $L : U' \to \mathbb{R}^{n \times n}$ is obtained by upper bounding the Jacobian of f over K' ,

$$\forall x \in K', \quad L_{i,j}(u) \geq \begin{cases} \frac{\partial f_i(\xi,u)}{\partial \xi_j}, & i = j \\ \left| \frac{\partial f_i(\xi,u)}{\partial \xi_j} \right|, & otherwise \end{cases} \tag{4.2}$$

An over approximation of the reachable state set of Σ, at time τ, with initial states within the hyper interval $K = [\underline{a}, \overline{a}]$, under input $u \in U'$ can be obtained using the growth bound as

$$R_\tau([\underline{a}, \overline{a}]) = \xi_{c,u}(\tau) + [-r', r'] \tag{4.3}$$

where $c = 0.5 * (\underline{a} + \overline{a})$ and $r = 0.5 * (\overline{a} - \underline{a})$ are the center and the radius of the initial state set, K' can be obtained either by manually supplying the matrix L or by the use of Algorithm 1, and $r' = e^{L(u)\tau} r$.

Algorithm 1 Computation of a priori enclosure

Require: f, X_0 as hyperinterval, u, τ, θ_m, θ_{pr}, X_{safe}

1: $\tau_{sub} = \tau$,
2: $K_1 = X_0$
3: $\tau_{division} = \frac{\tau}{\tau_{sub}}$
4: **for** $i = 1$ to $\tau_{division}$ **do**
5: find M_1 such that $|f(x, u)| \leq M_1, \forall x \in K_1$
6: $c =$ centre and $r =$ radius of K_1 i.e., $K_1 = c + [-r, r]$
7: $K_2 = c + [-(r + M_1\tau_{sub}), (r + M_1\tau_{sub})]$
8: find M_2 such that $|f(x, u)| \leq M_2, \forall x \in K_2$
9: **while** $(\max(M_2 - M_1) > \theta_m)$ **do**
10: $M_1 = M_2$
11: $K_2 = c + [-(r + M_1\tau_{sub}), (r + M_1\tau_{sub})]$
12: update M_2 such that $|f(x, u)| \leq M_2, \forall x \in K_2$
13: **if** $\max(\text{radius}(K_2) - \theta_{pr} * \text{radius}(X_{safe})) > 0$ **then**
14: $\tau_{sub} = \tau_{sub} * 0.5$
15: go to step 2
16: **end if**
17: **end while**
18: $K_1 = K_2$
19: **end for**
20: $K' = K_2$

In Algorithm 1, user need to supply the parameters θ_m and θ_{pr}. Sufficiently small vaue for θ_m (say 10^{-4}) will enable to identify the step when the maximum element

of the vector $(M_2 - M_1)$ has become smaller than θ_m, and thus M_2 can be considered to have stabilized. X_{safe} represents the hyperinterval in state space inside which the system trajectories are expected to stay for all considered time. θ_{pr} (say 100) is used to compare radius of the enclosure computed till any step with the radius of the safe set X_{safe}. If the former becomes larger than θ_{pr} times the later, then the time step is halved and we begin again with $K_1 = X_0$.

4.2 Method 2

Objective is to compute the reachable set in one sampling tme for dynamics in (3.1) for a given value of input which is kept fixed over the duration.

4.2.1 Overview

The method involves computation of reachable set of a nonlinear system through linearization utilising zonotopes for set representation [12]. For time $t \in [0, \tau]$ the nonlinear system $\dot{\xi}(t) = f(\xi(t), u)$ is first linearized into a system of form $\dot{\xi}(t) \in f_{lin}(\xi(t), u) = \bar{f} + A\Delta\xi(t) + \mathcal{L}$ where $\bar{f} = f(\bar{x}, u), \Delta\xi(t) = \xi(t) - \bar{x}, \bar{x}$ is the linearization point, $\mathcal{L}$ is the set of possible linearization errors such that $f(\xi(t), u) \in f_{lin}(\xi(t), u)$ for the considered time. As we are considering piecewise constant input signal so u stays fixed for each sampling period τ. Assuming some small value for $\mathcal{L}$ denoted by $\bar{\mathcal{L}}$ the reachable set for duration τ, $R_{[0,\tau]}(X_0, u)$ for given initial state set X_0 is computed using the linearized system f_{lin}. Now overapproximation of linearization error $\hat{\mathcal{L}}$ is computed over the set $R_{[0,\tau]}(X_0, u)$. If $\hat{\mathcal{L}} \not\subseteq \bar{\mathcal{L}}$ i.e. the computed error is not a subset of the applied error, then the value of assumption for $\bar{\mathcal{L}}$ needs to be updated to a larger value and the procedure is repeated until $\hat{\mathcal{L}} \subseteq \bar{\mathcal{L}}$. Using the accepted $\hat{\mathcal{L}}$ the reachable set at time τ, $R_\tau(X_0, u)$ can be obtained.

4.2.2 Linearization

For sampling time $[0, \tau]$ we assume the input u to be constant, then $f(\xi(t), u)$ can be represented as $f(\xi(t))$. The dynamics of a nonlinear system as given in (3.1) can be expressed using Taylor series expansion about the point $\bar{x}$ as

$$\begin{aligned}\dot{\xi}_i(t) &= f_i(\xi(t)) \\ &= f_i(\bar{x}) + \frac{\partial f_i(\xi)}{\partial \xi}\bigg|_{\xi(t)=\bar{x}} (\xi(t) - \bar{x}) + \frac{1}{2}(\xi(t) - \bar{x})^T \frac{\partial^2 f_i(\xi)}{\partial \xi^2}\bigg|_{\xi(t)=\bar{x}} (\xi(t) - \bar{x}) + \dots\end{aligned} \tag{4.4}$$

First order Taylor series and it's Lagrange remainder $\mathcal{L}$ can be used to overapproximate the infinite series (4.4) as [26]

$$\dot{\xi}_i(t) \in f_i(\bar{x}) + \frac{\partial f_i(\xi)}{\partial \xi}\bigg|_{\xi(t)=\bar{x}} (\xi(t) - \bar{x}) + \underbrace{\frac{1}{2}(\xi(t) - \bar{x})^T \frac{\partial^2 f_i(\zeta(\xi(t), \bar{x}))}{\partial \xi^2}(\xi(t) - \bar{x})}_{\text{Lagrange remainder, } \mathcal{L}_i}, \tag{4.5}$$

where ζ lies between $\xi(t)$ and $\bar{x}$, $\zeta \in \{\bar{x} + \alpha(\xi(t) - \bar{x}) | \alpha \in [0, 1]\}$ and $\mathcal{L}_i$ is the i^{th} element of the interval vector $\mathcal{L} \in \mathbb{IR}^n$.

Thus we have

$$\dot{\xi}(t) \in \bar{f} + A\Delta\xi(t) + \mathcal{L}, \tag{4.6}$$

where $\bar{f} = f(\bar{x})$, $A = \frac{\partial f(\xi)}{\partial \xi}\Big|_{\xi(t)=\bar{x}}$ and $\Delta\xi(t) = \xi(t) - \bar{x}$.

The expression in (4.6) can be written as

$$\dot{\xi}(t) = A\Delta\xi(t) + u, \text{where } u \in \bar{f} + \mathcal{L} \tag{4.7}$$

For $\xi(t), \bar{x} \in Z^{\mathcal{Z}} = (c, g^{(1)}, \dots, g^{(q)})$, where $Z^{\mathcal{Z}}$ is the reachable set for duration τ, we have $\zeta \in Z^{\mathcal{Z}}$.

Now, for $\sigma = \xi(t) - \bar{x}$, $\Delta Z^{\mathcal{Z}} = Z^{\mathcal{Z}} - \bar{x}$ and $H^{(i)}(\zeta) = \frac{\partial^2 f_i(\zeta)}{\partial \xi^2}$ we can write

$$\mathcal{L}_i = \left\{ \frac{1}{2} \sigma^T H^{(i)}(\zeta) \sigma \mid \zeta \in Z^{\mathcal{Z}}, \sigma \in \Delta Z^{\mathcal{Z}} \right\} \tag{4.8}$$

4.2.2.1 Overapproximation of $\mathcal{L}$

An overapproximation of the absolute value of the Lagrange remainder $\mathcal{L}$ can be obtained as [12]

$$\begin{aligned} |\mathcal{L}_i| &\subseteq [0, \hat{\mathcal{L}}_i], \\ \text{where } \hat{\mathcal{L}}_i &= \frac{1}{2} \gamma^T \max_{\zeta \in Z^{\mathcal{Z}}} \left(\left| H^{(i)}(\zeta) \right| \right) \gamma, \\ \gamma &= |c - \bar{x}| + \sum_{i=1}^{q} |g^{(i)}|, \end{aligned} \tag{4.9}$$

the max-operator and the absolute values taken elementwise.

Proof. From (4.8) we have

$$\begin{aligned} \mathcal{L}_i &= \left\{ \frac{1}{2} \sigma^T H^{(i)}(\zeta) \sigma \,\middle|\, \zeta \in Z^{\mathcal{Z}},\ \sigma \in \Delta Z^{\mathcal{Z}} \right\} \\ |\mathcal{L}_i| &= \left\{ \frac{1}{2} |\sigma^T H^{(i)}(\zeta) \sigma| \,\middle|\, \zeta \in Z^{\mathcal{Z}},\ \sigma \in \Delta Z^{\mathcal{Z}} \right\} \\ &\subseteq \frac{1}{2} \left[0, \max \left(|\sigma^T H^{(i)}(\zeta) \sigma| \right) \right],\ \zeta \in Z^{\mathcal{Z}}, \sigma \in \Delta Z^{\mathcal{Z}} \\ &\subseteq \frac{1}{2} \left[0, \max_{\sigma \in \Delta Z^{\mathcal{Z}}} (|\sigma|)^T \max_{\zeta \in Z^{\mathcal{Z}}} (|H^{(i)}(\zeta)|) \max_{\sigma \in \Delta Z^{\mathcal{Z}}} (|\sigma|) \right] \end{aligned}$$

As

$$\Delta Z^{\mathcal{Z}} = \left\{ x : x = (c - \bar{x}) + \sum_{i=1}^{q} \beta^{(i)} g^{(i)}, -1 \leq \beta^{(i)} \leq 1 \right\}$$

we have

$$\max_{\sigma \in \Delta Z^{\mathcal{Z}}} (|\sigma|) = |c - \bar{x}| + \sum_{i=1}^{q} |g^{(i)}|$$

Thus we get (4.9) □

As is evident from (4.9) γ is minimum for $\bar{x} = c$, but c is not known beforehand. To keep linearization error low, the linearization point $\bar{x}$ is selected closer to the

center of the reachable set at time τ [25],

$$\bar{x} = c_0 + \frac{\tau}{2} f(c_0) \tag{4.10}$$

where $Z_0^{\mathcal{Z}} = (c_0, g_0^{(1)}, \ldots, g_0^{(q)})$ is the initial state set.

A less conservative overapproximation of $\mathcal{L}$: [25]

Let $H_{j,k}^{(i)}(\zeta)$ represent the element in row j and column k of the matrix $H^{(i)}(\zeta)$ and $\mathcal{H}_{j,k}^{(i)} = \left\{ H_{j,k}^{(i)}(\zeta) \mid \zeta \in \mathrm{IH}(Z^{\mathcal{Z}}) \right\}$. Then $\mathcal{H}^{(i)} \in \mathbb{IR}^n$ is an interval matrix which can be split as $\mathcal{H}^{(i)} = H_c^{(i)} + \left[-H_\Delta^{(i)}, H_\Delta^{(i)} \right]$. From (4.8) we can write

$$\begin{aligned}
\mathcal{L}_i &\subseteq \left\{ \frac{1}{2} \sigma^T \mathcal{H}^{(i)} \sigma \mid \sigma \in \Delta Z^{\mathcal{Z}} \right\} \\
&= \left\{ \frac{1}{2} \sigma^T \left(H_c^{(i)} + \left[-H_\Delta^{(i)}, H_\Delta^{(i)} \right] \right) \sigma \mid \sigma \in \Delta Z^{\mathcal{Z}} \right\} \\
&= \frac{1}{2} \left\{ \sigma^T H_c^{(i)} \sigma + \sigma^T \left[-H_\Delta^{(i)}, H_\Delta^{(i)} \right] \sigma \mid \sigma \in \Delta Z^{\mathcal{Z}} \right\} \\
\mathcal{L}_i \subseteq L^{\mathcal{Z}} &= \frac{1}{2} \left(\mathrm{quad}(H_c^{(i)}, \Delta Z^{\mathcal{Z}}) + [-\eta, \eta] \right) \\
\text{where } \eta &= \left| \Delta Z^{\mathcal{Z}} \right|^T H_\Delta^{(i)} \left| \Delta Z^{\mathcal{Z}} \right|
\end{aligned} \tag{4.11}$$

4.2.3 Reachable set of linear system

Consider a linear system of form

$$\dot{\xi}(t) = A\xi(t) + u, \tag{4.12}$$

with initial state $x \in X_0 \subset \mathbb{R}^n$ and input $u \in U \subset \mathbb{R}^n$. Making use of superposition principle the solution of (4.12) can be written as

$$\xi_{x,u}(t) = \xi_{x,0}(t) + \xi_{0,u}(t) \tag{4.13}$$

where the homogeneous part $\xi_{x,0}(t)$ is the solution obtained by considering input to be zero, and the inhomogeneous part $\xi_{0,u}(t)$ is the solution obtained by considering the initial state to be zero. $R_t(X_0, 0)$ and $R_t(0, U)$ are used to refer to the reachable set corresponding to the homogeneous solution and the inhomogeneous solution

respectively. Then the reachable set of (4.12) can be written as $R_t(X_0, U) = \{\xi_{x,0}(t) + \xi_{0,u}(t) | \xi_{x,0}(t) \in R_t(X_0, 0), \xi_{0,u}(t) \in R_t(0, U), u \in U\}$.

4.2.3.1 Homogeneous solution

The solution of $\dot{\xi}(t) = A\xi(t)$ at time τ can be written as $\xi_{x,0}(\tau) = e^{A\tau}x$. The corresponding reachable set is $R_\tau(X_0, 0) = e^{A\tau}X_0$. When X_0 is a zonotope and the matrix exponential is computed as per (2.2), the set $R_t(X_0, 0)$ also comes out to be a zonotope. Given points x and $\xi_{x,0}(\tau)$, the set of points on the straight line connecting the two is $\{x + \frac{t}{\tau}(e^{A\tau}x - x) | t \in [0, \tau]\}$. We can then write

$$\xi_{x,0}(t) \in \left(x + \frac{t}{\tau}\left(e^{A\tau}x - x\right) + \mathcal{F}x \right), t \in [0, \tau] \tag{4.14}$$

where $\mathcal{F} \in \mathbb{IR}^{n \times n}$ called the correction matrix is such that addition of $\mathcal{F}x$ to the line segment ensures satisfaction of (4.14).

$$\mathcal{F} = \sum_{i=2}^{p} \left[\left(i^{\frac{-i}{i-1}} - i^{\frac{-1}{i-1}} \right) \tau^i, 0 \right] \frac{A^i}{i!} + \mathcal{E}(\tau) \tag{4.15}$$

For computation of $\mathcal{F}$ see Appendix A.1. Substituting the set of initial states X_0 in (4.14) we obtain

$$\begin{aligned} R_{[0,\tau]}(X_0, 0) &= \left\{ X_0 + \frac{t}{\tau}\left(e^{A\tau}X_0 - X_0\right) + \mathcal{F}X_0 | t \in [0, \tau] \right\} \\ \text{or, } R_{[0,\tau]}(X_0, 0) &= \mathrm{CH}(X_0, e^{A\tau}X_0) + \mathcal{F}X_0 \end{aligned} \tag{4.16}$$

where $\mathrm{CH}(A, B) = \{a + \beta(b - a) | a \in A, b \in B, \beta \in [0, 1]\}$ gives the convex hull of the two sets A and B, and it's overapproximation can be computed as per (2.14) for the case of zonotopes.

4.2.3.2 Inhomogeneous solution

For zero initial state and constant input u during the period $[0, \tau]$, the solution of (4.12) is

$$\begin{aligned} \xi_{0,u}(\tau) &= e^{A\tau} \int_0^\tau e^{-At} u dt \\ &= \int_0^\tau e^{A(\tau-t)} u dt \\ &= \int_0^\tau e^{A(\tau-t)} dt(u) \end{aligned} \tag{4.17}$$

Substitute, $s = \tau - t$

$$\xi_{0,u}(\tau) = \int_0^\tau e^{As} ds(u) \tag{4.18}$$

$$= A^{-1}(e^{A\tau} - I)u \tag{4.19}$$

Substituting for the matrix exponential from (2.2) in (4.18) we get

$$\begin{aligned} \xi_{0,u}(\tau) &\in \int_0^\tau \left(\sum_{i=0}^{p} \frac{(As)^i}{i!} + \mathcal{E}(s) \right) ds(u) \\ &= \int_0^\tau \left(\sum_{i=0}^{p} \frac{(As)^i}{i!} ds \right) u + \int_0^\tau \left(\mathcal{E}(s) ds \right)(u) \\ &= \left(\sum_{i=0}^{p} \frac{A^i \tau^{i+1}}{(i+1)!} \right) u + \int_0^\tau \left(\mathcal{E}(s) ds \right)(u) \end{aligned} \tag{4.20}$$

From(2.3) we have

$$\begin{aligned} \mathcal{E}(t) &= \frac{(\|A\|_\infty t)^{p+1}}{(p+1)!} \frac{1}{1-\epsilon} [-\mathbf{1}, \mathbf{1}] \\ &= \phi(t)[-\mathbf{1}, \mathbf{1}] \end{aligned}$$

where $\phi(t) = \frac{(\|A\|_\infty t)^{p+1}}{(p+1)!} \frac{1}{1-\epsilon} \in \mathbb{R}^+$ is monotone in t, thus

$$\int_0^\tau \phi(t) dt < \phi(\tau)\tau$$

Therefore

$$\begin{aligned}\int_0^\tau \mathcal{E}(s)ds &= \int_0^\tau \phi(s)[-\mathbf{1},\mathbf{1}]ds \\ &= \int_0^\tau \phi(s)ds[-\mathbf{1},\mathbf{1}] \\ &\subset \phi(\tau)\tau[-\mathbf{1},\mathbf{1}] \\ &= \mathcal{E}(\tau)\tau\end{aligned} \tag{4.21}$$

Combining (4.20) and above equation we get

$$\xi_{0,u}(\tau) \in \left(\sum_{i=0}^{p} \frac{A^i\tau^{i+1}}{(i+1)!}\right) u + \mathcal{E}(\tau)\tau u \tag{4.22}$$

Then an overapproximation of the exact reachable set for the system (4.12) with zero initial state and input set U is

$$R_\tau(0,U) = \left(\sum_{i=0}^{p} \frac{A^i\tau^{i+1}}{(i+1)!}\right) U + \mathcal{E}(\tau)\tau U \tag{4.23}$$

For the reachable set $R_{[0,\tau]}(0,U)$ corresponding to the inhomogeneous part we need to distinguish between the two cases: origin is contained in U, and not contained in U. For this we will make use of the relation $R_{\tau+\Delta t}(0,U) = e^{A\tau}R_{\Delta t}(0,U)+R_\tau(0,U)$ which can be obtained as shown below.

For $\Delta t > 0$ with (4.17) we have

$$\begin{aligned}\xi_{0,u}(\tau) &= \int_0^\tau e^{A(\tau-s)}ds.u \\ \xi_{0,u}(\tau+\Delta t) &= \int_0^{\tau+\Delta t} e^{A(\tau+\Delta t-s)}ds.u \\ &= \int_0^{\Delta t} e^{A(\tau+\Delta t-s)}ds.u + \int_{\Delta t}^{\Delta t+\tau} e^{A(\tau+\Delta t-s)}ds.u\end{aligned}$$

Substitute $(\Delta t - s) = -m$ in the second integral term to get

$$\begin{aligned}\xi_{0,u}(\tau+\Delta t) &= e^{A\tau}\int_0^{\Delta t} e^{A(\Delta t-s)}ds.u + \int_0^\tau e^{A(\tau-m)}dm.u \\ &= e^{A\tau}\xi_{0,u}(\Delta t) + \xi_{0,u}(\tau)\end{aligned} \tag{4.24}$$

Thus in terms of reachable set we get

$$R_{\tau+\Delta t}(0,U) = e^{A\tau}R_{\Delta t}(0,U) + R_{\tau}(0,U) \tag{4.25}$$

The set U contains origin

Now consider two sets $A, B \subset \mathbb{R}^n$, such that B contains the origin ($0 \in B$) then A is contained in the Minkowski sum of the sets (i.e. $A \subseteq A + B$). From (4.23) we can see that if set U contains origin, then the set $R_\tau(0,U)$ also contains origin and so does the set $e^{A\tau}R_{\Delta t}(0,U)$. Thus

$$R_{\tau}(0,U) \subseteq R_{\tau+\Delta t}(0,U) \tag{4.26}$$

in words, the set $R_\tau(0,U)$ contains the reachable sets of previous points in time. Therefore

$$R_{[0,\tau]}(0,U) = R_{\tau}(0,U) \tag{4.27}$$

The set U doesn't contain origin

When the set U doesn't contain origin, we can express it as $U = \tilde{u} + \tilde{U}$, such that $0 \in \tilde{U} \subset \mathbb{R}^m$ and $\tilde{u} \in \mathbb{R}^m$. The reachable set due to $\tilde{U}$ can be computed as shown in the preceding discussion. The reachable set due to $\tilde{u}$ can be written from (4.19) as $\xi_{0,\tilde{u}}(\tau) = A^{-1}(e^{A\tau} - I)\tilde{u}$. The set of points on the straight line joining 0 and $\xi_{0,\tilde{u}}(\tau)$ is $\left\{0 + \frac{t}{\tau}A^{-1}(e^{A\tau} - I)\tilde{u} | t \in [0,\tau]\right\}$. This set can be expanded so as to contain $\xi_{0,\tilde{u}}(t)$,

$$\xi_{0,\tilde{u}}(t) \in \left(0 + \frac{t}{\tau}A^{-1}(e^{A\tau} - I)\tilde{u} + \tilde{\mathcal{F}}\tilde{u}\right), t \in [0,\tau] \tag{4.28}$$

The input correction matrix $\tilde{\mathcal{F}}$ is such that (4.28) is satisfied.

$$\tilde{\mathcal{F}} = \sum_{i=2}^{p} \left[\left(i^{\frac{-i}{i-1}} - i^{\frac{-1}{i-1}}\right)\tau^i, 0\right] \frac{A^{i-1}}{i!} + \mathcal{E}(\tau)/\|A\|_\infty \tag{4.29}$$

See Appendix A.2 for derivation of $\tilde{\mathcal{F}}$. From (4.28), (4.14) and (4.16) we can write

$$R_{[0,\tau]}(X_0,0) + R_{[0,\tau]}(0,\tilde{u}) = \mathrm{CH}\left(X_0, e^{A\tau}X_0 + \xi_{0,\tilde{u}}(\tau)\right) + \mathcal{F}X_0 + \tilde{\mathcal{F}}\tilde{u} \tag{4.30}$$

Thus

$$R_{[0,\tau]}(X_0, U) = R_{[0,\tau]}(X_0, 0) + R_{[0,\tau]}(0, \tilde{u}) + R_{[0,\tau]}(0, \tilde{U}) \tag{4.31}$$

where $R_{[0,\tau]}(0, \tilde{U})$ can be computed using (4.27) and (4.23).

Algorithm 2 Reachable set computation using Method 2

Require: f, $X_0 = (c, g^{(1)}, \ldots, g^{(q)}) \subseteq \mathbb{R}^n$, u, τ

1: $\bar{x} = c + 0.5\tau f(c,u)$; $\bar{f} = f(\bar{x},u)$; $A = \left.\frac{\partial f(x,u)}{\partial x}\right|_{\bar{x}}$; $p = 5$; $\epsilon = \frac{\|A\|_\infty \tau}{p+2}$

2: **while** $(\epsilon \geq 1)$ **do**

3: $\quad p = p+1, \epsilon = \frac{\|A\|_\infty \tau}{p+2}$

4: **end while**

5: $E_{bound} = \frac{(\|A\|_\infty \tau)^{p+1}}{(p+1)!}\frac{1}{1-\epsilon}$

6: **while** $(E_{bound} \geq 10^{-3})$ **do**

7: $\quad$ p = p+1, $E_{bound} = \frac{(\|A\|_\infty \tau)^{p+1}}{(p+1)!}\frac{1}{1-\epsilon}$

8: **end while**

9: $E(\tau) = E_{bound}[-\mathbf{1}, \mathbf{1}]$

10: $e_p^{A\tau} = \sum_{i=0}^{p} \frac{(A\tau)^i}{i!} + E(\tau)$

11: $F = \sum_{i=2}^{p}[(i^{\frac{-i}{i-1}} - i^{\frac{-1}{i-1}}), 0]\frac{(\tau A)^i}{i!} + E(\tau)$

12: $\widetilde{F} = \sum_{i=2}^{p}[(i^{\frac{-i}{i-1}} - i^{\frac{-1}{i-1}}), 0]\frac{\tau^i A^{i-1}}{i!} + \frac{E(\tau)}{\|A\|_\infty}$

13: $\Delta X_0 = X_0 - \bar{x}$

14: $R_{\bar{f}} = \left[\sum_{i=0}^{p} \frac{A^i \tau^{i+1}}{(i+1)!} + \tau E(\tau)\right] \bar{f}$

15: $R_1 = \left(CH(\Delta X_0, e_p^{A\tau}\Delta X_0 + R_{\bar{f}}) + F\Delta X_0\right) + \bar{x}$

16: $\bar{L} = 0, chk = 2$

17: **while** $(chk \geq 1)$ **do**

18: $\quad R_{err} = \left[\sum_{i=0}^{p} \frac{A^i \tau^{i+1}}{(i+1)!} + \tau E(\tau)\right] [-\bar{L}, \bar{L}]$

19: $\quad$ **if** $(0 \notin \bar{f} + [-\bar{L}, \bar{L}])$ **then**

20: $\quad\quad R_2 = R_1 + \widetilde{F}\bar{f}$

21: $\quad$ **else**

22: $\quad\quad R_2 = R_1$

23: $\quad$ **end if**

24: $\quad R_{[0,\tau]} = R_2 + R_{err}$

25: $\quad$ Compute $\hat{L}$ over $R_{[0,\tau]}$

26: $\quad chk = \max_i(\hat{L}_i / \bar{L}_i)$

27: $\quad \bar{L} = 1.1\hat{L}$

28: **end while**

29: $R_{err} = \left[\sum_{i=0}^{p} \frac{A^i \tau^{i+1}}{(i+1)!} + \tau E(\tau)\right] [-\hat{L}, \hat{L}]$

30: $R_\tau(X_0) = \left(e_p^{A\tau}\Delta X_0 + R_{\bar{f}} + R_{err}\right) + \bar{x}$

In Algorithm 2, $\hat{L}$ can be computed by either of (4.9) or (4.11). With (4.11), $\hat{L}$ obtained by taking interval hull of the computed zonotope $L^{\mathcal{Z}}$, also in computation of R_{err} we use $L^{\mathcal{Z}}$ instead of $[-\hat{L}, \hat{L}]$.

4.3 Method 3

The method in [17] gives an over approximation of exact reachable set of linear systems together with the upper bound on the Hausdorf distance between the exact reachable set $R^e_{[0,\tau]}(X_0, U)$ and it's overapproximation $R_{[0,\tau]}(X_0, U)$. Let us define the Hausdorf distance between two compact sets $A, B \subseteq \mathbb{R}^n$ as

$$d_H(A, B) = max\left(\sup_{x\in A}\inf_{x'\in B}\|x - x'\|, \sup_{x'\in B}\inf_{x\in A}\|x - x'\|\right) \tag{4.32}$$

When the sets are equal $A = B$, we have $d_H(A, B) = 0$.
We again consider the linear system in (4.12). Given the set of initial states X_0 and the set of input U, let us denote

$$R_{X_0} = \max_{x\in X_0}\|x\|, D_{X_0} = \max_{x,y\in X_0}\|x - y\|, R_U = \max_{u\in U}\|u\|,$$

where $\|.\|$ represents the infinity norm. From [17] we have an over approximation of the reachable set over duration $[0, \tau]$ for states starting in X_0 under input $u \in U$ to be

$$R_{[0,\tau]}(X_0, U) = \mathrm{CH}\left(X_0, e^{\tau A}X_0 + \tau U + \alpha_\tau \square B\right), \tag{4.33}$$

such that,

$$d_H\left(R^e_{[0,\tau]}(X_0, U), R_{[0,\tau]}(X_0, U)\right) \leq \frac{1}{4}\left(e^{\tau\|A\|} - 1\right)D_{X_0} + 2\alpha_\tau$$

where $\alpha_\tau = (e^{\tau\|A\|} - 1 - \tau\|A\|)(R_{X_0} + \frac{R_U}{\|A\|})$ and $\square B$ represents the ball in infinity norm with origin as center and unity radius. Further an overapproximation of the exact reachable set at time τ, $R^e_\tau(X_0, U)$ can be obtained as

$$R_\tau(X_0, U) = e^{\tau A}X_0 + \tau U + \beta_\tau \square B, \tag{4.34}$$

such that,

$$d_H\left(R^e_\tau(X_0,U), R_\tau(X_0,U)\right) \leq 2\beta_\tau$$

where $\beta_\tau = (e^{\tau\|A\|} - 1 - \tau\|A\|)\frac{R_U}{\|A\|}$

Algorithm 3 Reachable set computation using Method 3

Require: f, $X_0 = (c, g^{(1)}, ..., g^{(q)}) \subseteq \mathbb{R}^n$, u, τ

1: $\bar{x} = c + 0.5\tau f(c,u)$; $\bar{f} = f(\bar{x},u)$; $A = \left.\frac{\partial f(x,u)}{\partial x}\right|_{\bar{x}}$; $\Delta X_0 = X_0 - \bar{x}$

2: $R_{\Delta X_0} = \max_{x\in\Delta X_0}\|x\|_\infty$

3: $\bar{L} = 0, chk = 2$

4: **while** $(chk \geq 1)$ **do**

5: $\quad V = \bar{f} + [-\bar{L}, \bar{L}]$

6: $\quad R_V = \max_{v\in V}\|v\|_\infty$

7: $\quad \alpha_\tau = \left(e^{\tau\|A\|_\infty} - 1 - \tau\|A\|_\infty\right)\left(R_{\Delta X_0} + \frac{R_V}{\|A\|_\infty}\right)$

8: $\quad R_{[0,\tau]} = CH\left(\Delta X_0, e^{\tau A}\Delta X_0 + \tau V + \alpha_\tau B\right) + \bar{x}$

9: $\quad$ Compute $\hat{L}$ over $R_{[0,\tau]}$

10: $\quad chk = \max_i(\hat{L}_i/\bar{L}_i)$

11: $\quad \bar{L} = 1.1\hat{L}$

12: **end while**

13: $V = \bar{f} + [-\hat{L}, \hat{L}]$

14: $R_V = \max_{v\in V}\|v\|_\infty$

15: $\beta_\tau = \left(e^{\tau\|A\|_\infty} - 1 - \tau\|A\|_\infty\right)\frac{R_V}{\|A\|_\infty}$

16: $R_\tau(X_0) = \left(e^{\tau A}\Delta X_0 + \tau V + \beta_\tau B\right) + \bar{x}$

4.4 Implementation

The procedure to compute reachable set using either method 2 or 3 is implemented in C++ and is interfaced with SCOTS which is a tool for synthesis of symbolic controller for nonlinear systems. SCOTS implementation uses the growth bound method which gives relatively more conservative overapproximation of reachable set than the Method 2.

4.4.1 Installation

The code can be downloaded from `https://github.com/mahendrasinghtomar/ReachableSetZonotope`. Replace the files 'Abstraction.hh' and 'SymbolicModel.hh' in the 'src' folder of SCOTS with the downloaded ones. Additional C++ libraries needed:

1. Eigen a C++ library for linear algebra. `[http://eigen.tuxfamily.org/index.php?title=Main_Page]`
2. vnodelp for interval arithmatic. `[http://www.cas.mcmaster.ca/~nedialk/vnodelp/]`
3. FADBAD++ for automatic differentiation. Comes bundled with vnodelp or can be downloaded from `[http://www.fadbad.com/fadbad.html]`
4. gnuplot for plotting sets `[http://www.gnuplot.info/]`
5. The header file "gnuplot-iostream.h" as an interface to send data from C++ to gnuplot. It can be downloaded from `http://www.stahlke.org/dan/gnuplot-iostream`

4.4.2 Usage

'Example_2D_cartpole.cc' to serve as a template example file. System dynamics to be written inside the function 'func_Lj_system'.

Chapter 5

Examples and Conclusion

Computation of all the examples carried out on Core i7 3.5GHz processor.

5.1 Example 1

Consider a system with the dynamics:

$$\begin{aligned} \dot{x}_1 &= x_2 - 0.1x_1 \\ \dot{x}_2 &= ux_1 \end{aligned} \tag{5.1}$$

We compare it's reachable set $R_\tau(X_0, u)$ for $X_0 = ([-49, -50]^T, diag(0.25, 0.25)), u = -1$ from the three methods for three different sampling times in Figure 5.1. As we can observe, among the three we obtain the tightest overapproximation from Method 2, while the most conservative from Method 3. The extent of conservativeness of the Method 3 keep on increasing with sampling time.

Next we design controller for reachability specification using the first two methods with SCOTS. Consider state space $X = [-50, 50] \times [-50, 50]$, state space quantisation $\eta_{ss} = [0.5; 0.5]$, input space $U = [-2, 2]$, input space quantisation $\eta_{is} = 1$, target set $= [40, 49.5] \times [40, 49.5]$, and sampling time $\tau = 0.75$. Table 5.1 gives

TABLE 5.1: Example 1, $\tau = 0.75$

	Growth bound manual L	Method 2 $\hat{L}$ by (4.9)
Number of transitions	1589727	944747
Controller domain size	11007	40009
Computation time	0.6287 sec	191.71 sec

the number of transitions in the constructed discrete abstraction, the obtained controller domain size and the computation time. The controller domain size refer to the number of states in X for which we have available control inputs which can drive the system such that the specification is realized. Table 5.2 lists the data obtained for a sampling time of $\tau = 0.1$. The first data column is for the case when the matrix L in the growth bound method is to be supplied manually by the user to SCOTS. The second data column is for the case when L is computed internally by first obtaining the a priori enclosure using the Algorithm 1.

During the construction of discrete abstraction if the overapproximation of reachable set is tighter i.e. smaller in size, then the number of state space grid cells with which it intersects may come out to be smaller. This will result in lesser nondeterminism and hence lower number of transitions in the discrete abstraction.

Transitions corresponding to those reachable sets are discarded which happen to intersect with the avoid set in specification, goes outside the safe set or the allowed state space. When the overapproximation is tighter we have less number of such occurrences, and thus less combinations discarded of state space cell and input.

Figure 5.2 represents in red color the target area, in grey the states for which we have a control input available in the designed controller and in white the states for which we didn't obtain any control input. The black curved line represents a simulated trajectory.

Figure 5.3 depicts the reachable sets obtained for $X_0 = ([-49, -50]^T, diag(0.25, 0.25))$ and $\tau = 0.75$ for three different inputs u= -2, -1 and 1 respectively. The black lines coming into the set give a part of the trajectory of the system

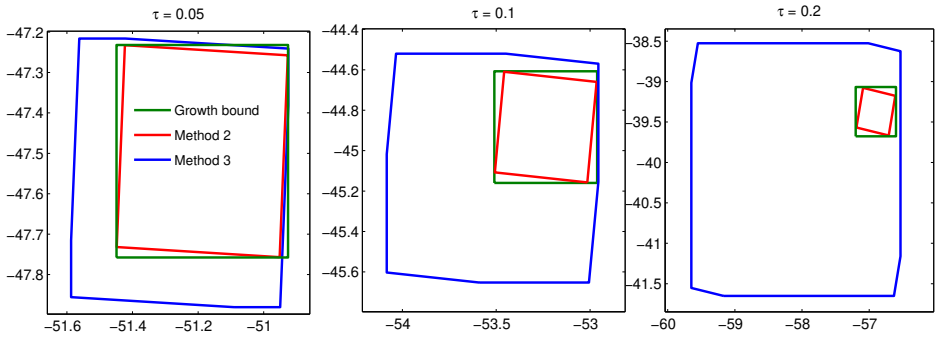

FIGURE 5.1: Example 1: Reachable set from the three methods for different sampling times

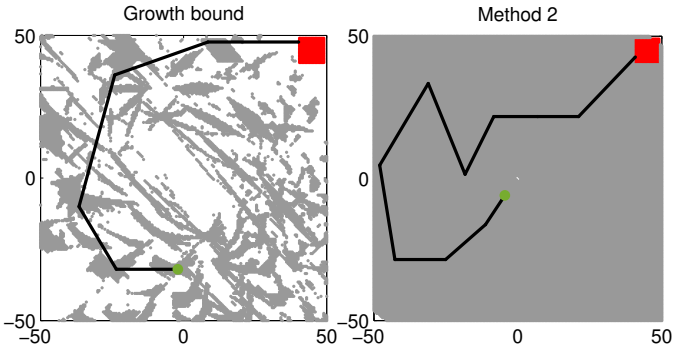

FIGURE 5.2: Example1: Domain of reachability controller

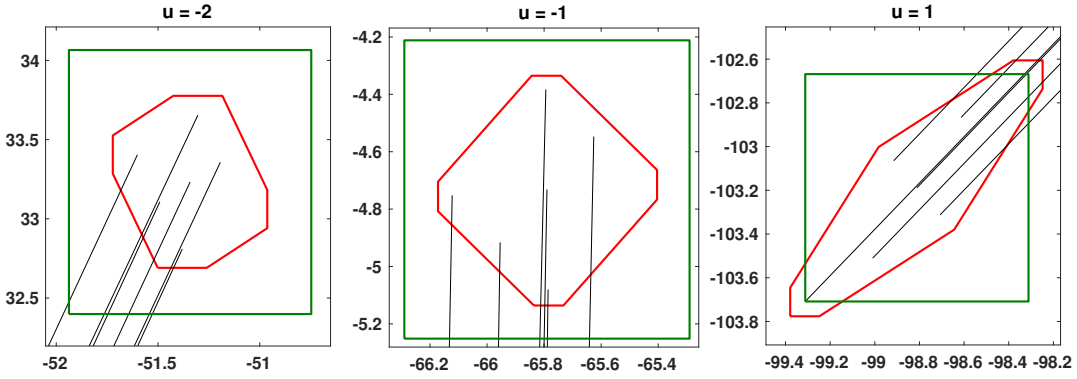

FIGURE 5.3: Example1: Reachable sets using growth bound (in green) and the method 2 (in red) for three different inputs.

5.2 Example (Cartpole)

Invariance specification. Consider a two dimensional model of pendulum on a cart [27]

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= -\omega^2 \left(sin(x_1) + u_1 cos(x_1) \right) - 2\gamma x_2, \end{aligned} \tag{5.2}$$

TABLE 5.2: Example 1, $\tau = 0.1$

	Growth bound manual L	Growth bound automatic L	Method 2 $\hat{L}$ by (4.9)	Method 2 $\hat{L}$ by (4.11)
Number of transitions	932125	932125	927045	927045
Controller domain size	38528	38528	38532	38532
Computation time	0.57 sec	1.77 sec	41 sec	173.56 sec

TABLE 5.3: Example: cartpole

	Growth bound manual L	Growth bound automatic L	Method 2 $\hat{L}$ by (4.9)	Method 2 $\hat{L}$ by (4.11)
Number of transitions	388066	361072	352952	351674
Controller domain size	830	836	836	837
Computation time	0.33 sec	1.43 sec	20 sec	116.54 sec

where $\omega = 1$ and $\gamma = 0.0125$. For an invariance problem with $X = [0.5\pi, 1.5\pi] \times [-1, 1]$, $\eta_{ss} = [0.05; 0.1], U = [-3, 3], \eta_{is} = 0.1, \tau = 0.25$ and safe set $= X$, the obtained controller data is shown in Table 5.3

5.3 Example (Vehicle)

Path planning for an autonomous vehicle. Consider the dynamics [2]

$$\begin{aligned} \dot{x}_1 &= u_1 \frac{\cos(\alpha + x_3)}{\cos(\alpha)}, \\ \dot{x}_2 &= u_1 \frac{\sin(\alpha + x_3)}{\cos(\alpha)}, \\ \dot{x}_3 &= u_1 \tan(u_2), \end{aligned} \tag{5.3}$$

where $\alpha = tan^{-1}(\tan(u_2)/2)$, (x_1, x_2) is the position and x_3 is the orientation of the vehicle in the 2-dimensional plane. The control inputs u_1 and u_2 are the rear wheel velocity and the steering angle. We designed controller using SCOTS for

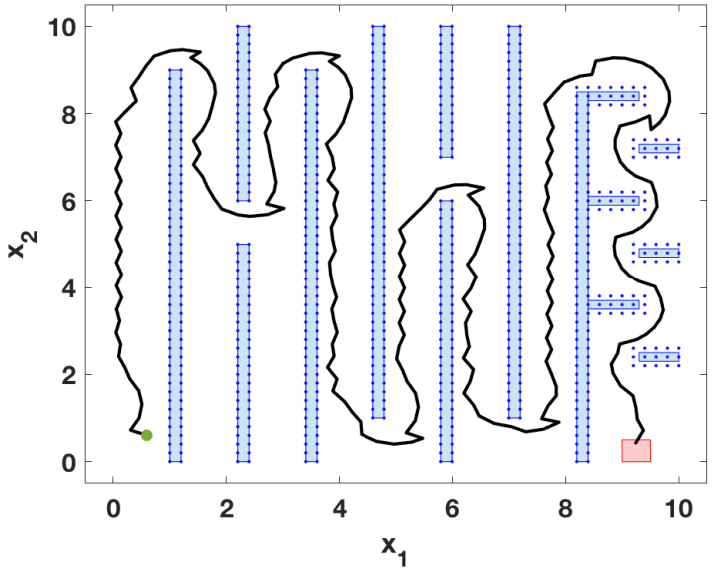

FIGURE 5.4: Output trajectory for reach and avoid specification

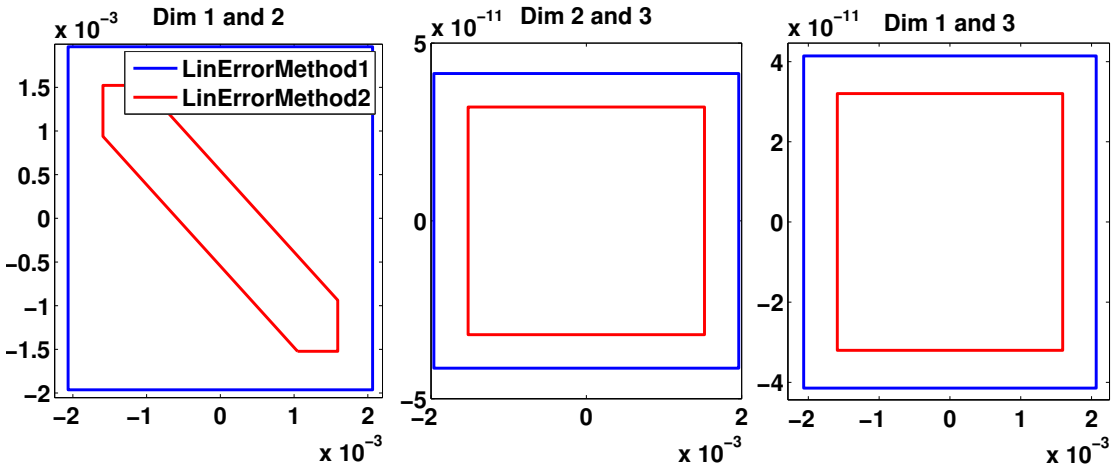

FIGURE 5.5: 2D projection of the computed linearisation error for vehicle example

reachability specification where the target set is $= [9, 9.5] \times [0, 0.5]$, $\tau = 0.3$, $X = [0, 10] \times [0, 10] \times [-3.5, 3.5]$, $\eta_{ss} = [0.2; 0.2; 0.2]$, $\mathrm{U} = [-1, 1] \times [-1, 1]$, $\eta_{is} = [0.3; 0.3]$.

Figure 5.4 shows the two dimensional state space. Rectangular boxes in blue color represent the obstacles while the red rectangle represents the target set. Green dot is the starting position and the black curved line is a simulated trajectory using the designed controller.

Figure 5.5 shows the projection on two dimensional plane of the computed linearisation error by the two methods for $\tau = 0.3$, $X_0 = [0.8, 1.2] \times [0.8, 1.2] \times$

TABLE 5.4: Example: vehicle

	Growth bound manual L	Growth bound automatic L	Method 2 $\hat{L}$ by (4.9)	Method 2 $\hat{L}$ by (4.11)
Number of transitions	35772302	34740770	34764178	34697234
Controller domain size	48018	48314	48354	48354
Computation time	20.79 sec	71.72 sec	20.25 min	2.6 hr

$[1.8, 2.2], u = [0.6; 0.6]$. LinErrorMethod1 and LinErrorMethod2 represent the error computed by (4.9) and (4.11) and their computation times are 0.0045 sec and 0.0111 sec respectively.

5.4 Example (Aircraft)

An aircraft landing maneuver. Consider the aircraft dynamics taken from [2]

$$\begin{aligned}
\dot{x}_1 &= \frac{1}{m}(u_1 \cos(u_2) - D(u_2, x_1) - mg \sin(x_2)),\\
\dot{x}_2 &= \frac{1}{mx_1}(u_1 \sin(u_2) + L(u_2, x_1) - mg \cos(x_2)),\\
\dot{x}_3 &= x_1 \sin(x_2),
\end{aligned} \tag{5.4}$$

where

$$\begin{aligned}
D(u_2, x_1) &= (2.7 + 3.08(1.25 + 4.2u_2)^2)x_1^2,\\
L(u_2, x_1) &= (68.6(1.25 + 4.2u_2))x_1^2
\end{aligned}$$

$m = 60 \times 10^3$, $g = 9.81$. $x1, x2$ and $x3$ refer to the velocity, the flight path angle and the altitude of the aircraft respectively. The inputs u_1 and u_2 are the thrust of the engines and the angle of attack. When objective is to control the aircraft from some height to close to the ground we can consider it as a reachability problem. For sampling time $\tau = 0.25$, target set $= [63, 75] \times [-\pi/60, 0] \times [0, 2.5]$
$X = [58, 83] \times [-\pi/60, 0] \times [0, 56]$, $\eta_{ss} = [0.05; 0.0008; 0.17]$,
$U = [0, 32000] \times [0, 2\pi/45]$, $\eta_{is} = [32000; 0.02]$, the designed controllers have domain sizes as shown in Table 5.5.

TABLE 5.5: Example: aircraft

	Growth bound manual L	Growth bound automatic L	Method 2 $\hat{L}$ by (4.9)
Number of transitions	2315456528	2313163518	2318432048
Controller domain size	2870811	2876307	2878810
Computation time	8.84 min	1 hr	76.3 hr

5.5 Conclusion

As we can see from the examples the accuracy of approximation of reachable set affects the size of the symbolic model and of the obtained controller domain. Reachable sets from the Method-2 are clearly least conservative. Although arbitrarily close [17] overapproximation can be obtained by decreasing the time step size by Method-3 for linear systems, our implementation which is for only one time step and based on linearization is highly dependent on the sampling time. With increasing sampling time the conservativeness of the reachable set keep on rising for the Method-3, and so it is not used for symbolic controller synthesis. Large values of computation time for Method-2 may be ascribed to inefficiency in code implementation. For example for calculation of derivatives automatic differentiation is used which needs to be computed at every single iteration. Use of symbolic differentiation which needs computation only once and value substitution in all other subsequent steps can result in significant reduction in computation time.

Appendix A

A.1 Computation of the correction matrix $\mathcal{F}$

(see [24]). From (4.14)

$$\xi_{x,0}(t) \in \left(x + \frac{t}{\tau}\left(e^{A\tau}x - x\right) + \mathcal{F}x \right), t \in [0, \tau]$$

where $\xi_{x,0}(t) = e^{At}x$. When (2.2) used as matrix exponential, we get

$$e_p^{At}x \subseteq \left(x + \frac{t}{\tau}\left(e_p^{A\tau}x - x\right) + \mathcal{F}x \right), t \in [0, \tau]$$

$$\text{Therefore, } \mathcal{F} \supseteq e_p^{At} - I - \frac{t}{\tau}(e_p^{A\tau} - I), \forall t \in [0, \tau]$$

Now,

$$\begin{aligned} e_p^{At} - I - \frac{t}{\tau}(e_p^{A\tau} - I) &= \sum_{i=0}^{p} \frac{(At)^i}{i!} + \mathcal{E}(t) - I - \frac{t}{\tau}\left(\sum_{i=0}^{p} \frac{(A\tau)^i}{i!} + \mathcal{E}(\tau) - I \right) \\ &= \sum_{i=2}^{p} (t^i - t\tau^{i-1}) \frac{1}{i!} A^i + \mathcal{E}(t) - \frac{t}{\tau}\mathcal{E}(\tau) \end{aligned} \tag{A.1}$$

The second derivative of $(t^i - t\tau^{i-1})$ is positive for $t \in (0, \tau]$, therefore it has a local minimum which can obtained by equating its first derivative to zero:

$$\frac{d}{dt}(t^i - t\tau^{i-1}) = 0$$

$$it_{min}^{i-1} - \tau^{i-1} = 0$$

$$t_{min} = i^{-\frac{1}{i-1}} r$$

As $(t^i - t\tau^{i-1}) = 0$ for $t = 0$ and $t = \tau$, therefore we have

$$(t^i - t\tau^{i-1}) \in \left[(i^{\frac{-i}{i-1}} - i^{\frac{-1}{i-1}})r^i, 0\right], \forall t \in [0, \tau] \tag{A.2}$$

Next, bounds for $\mathcal{E}(t) - \frac{t}{\tau}\mathcal{E}(\tau)$ are computed. From(2.3) we have

$$\begin{aligned}\mathcal{E}(t) &= \frac{(\|A\|_\infty t)^{p+1}}{(p+1)!}\frac{1}{1-\epsilon}[-\mathbf{1}, \mathbf{1}] \\ &= \phi(t)[-\mathbf{1}, \mathbf{1}]\end{aligned}$$

where $\phi(t) = \frac{(\|A\|_\infty t)^{p+1}}{(p+1)!}\frac{1}{1-\epsilon} \in \mathbb{R}^+$ is monotonically increasing in t. So we have $\left|\left(\phi(t) - \frac{t}{\tau}\phi(\tau)\right)\right| \leq \phi(\tau), \forall t \in [0, \tau]$ Thus,

$$\begin{aligned}\mathcal{E}(t) - \frac{t}{\tau}\mathcal{E}(\tau) &= \left(\phi(t) - \frac{t}{\tau}\phi(\tau)\right)[-1, 1] \\ &\subseteq \phi(\tau)[-1, 1] \\ &= \mathcal{E}(\tau)\end{aligned} \tag{A.3}$$

From (A.1), (A.2) and (A.3) we can write

$$\mathcal{F} = \sum_{i=2}^{p}\left[\left(i^{\frac{-i}{i-1}} - i^{\frac{-1}{i-1}}\right)\tau^i, 0\right]\frac{A^i}{i!} + \mathcal{E}(\tau) \tag{A.4}$$

A.2 Computation of the input correction matrix $\tilde{\mathcal{F}}$

From (4.28) and (2.2) we have

$$A^{-1}(e_p^{At} - I)\tilde{u} \subseteq \left(0 + \frac{t}{\tau}A^{-1}(e^{A\tau} - I)\tilde{u} + \tilde{\mathcal{F}}\tilde{u}\right), t \in [0, \tau]$$

Therefore

$$\begin{aligned}\tilde{\mathcal{F}} &\supseteq A^{-1}(e_p^{At} - I) - \frac{t}{\tau}A^{-1}(e^{A\tau} - I) \\ &= A^{-1}\left((e_p^{At} - I) - \frac{t}{\tau}(e^{A\tau} - I)\right) \\ \tilde{\mathcal{F}} &= A^{-1}\mathcal{F}\end{aligned}$$

$$= A^{-1}\left(\sum_{i=2}^{p}\left[\left(i^{\frac{-i}{i-1}} - i^{\frac{-1}{i-1}}\right)\tau^i, 0\right]\frac{A^i}{i!} + \mathcal{E}(\tau)\right)$$

$$= \sum_{i=2}^{p}\left[\left(i^{\frac{-i}{i-1}} - i^{\frac{-1}{i-1}}\right)\tau^i, 0\right]\frac{A^{i-1}}{i!} + \mathcal{E}(\tau)/\|A\|_\infty$$

Here, $A^{-1}\mathcal{E} = \mathcal{E}/\|A\|_\infty$ as infinity norm is involved in the computation of $\mathcal{E}$ (see [28]).

Bibliography

[1] Paulo Tabuada. *Verification and control of hybrid systems: a symbolic approach*. Springer Science & Business Media, 2009.

[2] Gunther Reissig, Alexander Weber, and Matthias Rungger. Feedback refinement relations for the synthesis of symbolic controllers. *IEEE Transactions on Automatic Control*, 62(4):1781–1796, 2017.

[3] Paulo Tabuada. An approximate simulation approach to symbolic control. *IEEE Transactions on Automatic Control*, 53(6):1406–1418, 2008.

[4] Antoine Girard. Low-complexity quantized switching controllers using approximate bisimulation. *Nonlinear Analysis: Hybrid Systems*, 10:34–44, 2013.

[5] Matthias Althoff, Stanley Bak, Dario Cattaruzza, Xin Chen, Goran Frehse, Rajarshi Ray, and Stefan Schupp. Arch-comp17 category report: Continuous and hybrid systems with linear continuous dynamics. In *4th Applied Verification for Continuous and Hybrid Systems Workshop (ARCH)*, 2017.

[6] Matthias Rungger and Majid Zamani. Accurate reachability analysis of uncertain nonlinear systems. In *Proceedings of the 21st International Conference on Hybrid Systems: Computation and Control (part of CPS Week)*, pages 61–70. ACM, 2018.

[7] Thao Dang. Approximate reachability computation for polynomial systems. In *International Workshop on Hybrid Systems: Computation and Control*, pages 138–152. Springer, 2006.

[8] Claire J Tomlin, Ian Mitchell, Alexandre M Bayen, and Meeko Oishi. Computational techniques for the verification of hybrid systems. *Proceedings of the IEEE*, 91(7):986–1001, 2003.

[9] Rajeev Alur, Thao Dang, and Franjo Ivančić. Progress on reachability analysis of hybrid systems using predicate abstraction. In *International Workshop on Hybrid Systems: Computation and Control*, pages 4–19. Springer, 2003.

[10] Ashish Tiwari and Gaurav Khanna. Series of abstractions for hybrid automata. In *International Workshop on Hybrid Systems: Computation and Control*, pages 465–478. Springer, 2002.

[11] Eugene Asarin, Thao Dang, and Antoine Girard. Reachability analysis of nonlinear systems using conservative approximation. In *International Workshop on Hybrid Systems: Computation and Control*, pages 20–35. Springer, 2003.

[12] Matthias Althoff, Olaf Stursberg, and Martin Buss. Reachability analysis of nonlinear systems with uncertain parameters using conservative linearization. In *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, pages 4042–4048. IEEE, 2008.

[13] Martin Berz and Kyoko Makino. Verified integration of odes and flows using differential algebraic methods on high-order taylor models. *Reliable Computing*, 4(4):361–369, 1998.

[14] Sanja Živanović and Pieter Collins. Numerical solutions to noisy systems. In *Decision and Control (CDC), 2010 49th IEEE Conference on*, pages 798–803. IEEE, 2010.

[15] Xin Chen, Erika Ábrahám, and Sriram Sankaranarayanan. Flow*: An analyzer for non-linear hybrid systems. In *International Conference on Computer Aided Verification*, pages 258–263. Springer, 2013.

[16] Antoine Girard. Reachability of uncertain linear systems using zonotopes. In *International Workshop on Hybrid Systems: Computation and Control*, pages 291–305. Springer, 2005.

[17] Colas Le Guernic and Antoine Girard. Reachability analysis of linear systems using support functions. *Nonlinear Analysis: Hybrid Systems*, 4(2):250–262, 2010.

[18] Wolfgang Kühn. Rigorously computed orbits of dynamical systems without the wrapping effect. *Computing*, 61(1):47–67, 1998.

[19] Matthias Rungger and Majid Zamani. Scots: A tool for the synthesis of symbolic controllers. In *Proceedings of the 19th International Conference on Hybrid Systems: Computation and Control*, pages 99–104. ACM, 2016.

[20] Matthias Althoff. An introduction to cora 2015. In *ARCH@ CPSWeek*, pages 120–151, 2015.

[21] Goran Frehse, Colas Le Guernic, Alexandre Donzé, Scott Cotton, Rajarshi Ray, Olivier Lebeltel, Rodolfo Ripado, Antoine Girard, Thao Dang, and Oded Maler. Spaceex: Scalable verification of hybrid systems. In *International Conference on Computer Aided Verification*, pages 379–395. Springer, 2011.

[22] ML Liou. A novel method of evaluating transient response. *Proceedings of the IEEE*, 54(1):20–23, 1966.

[23] Hoai-Nam Nguyen. Constrained control of uncertain, time-varying, discrete-time systems. *An Interpolation-Based Approach (Cham: Springer)*, 2014.

[24] Matthias Althoff, Olaf Stursberg, and Martin Buss. Reachability analysis of linear systems with uncertain parameters and inputs. In *Decision and Control, 2007 46th IEEE Conference on*, pages 726–732. IEEE, 2007.

[25] Matthias Althoff and Bruce H Krogh. Reachability analysis of nonlinear differential-algebraic systems. *IEEE Transactions on Automatic Control*, 59 (2):371–383, 2014.

[26] Martin Berz and Georg Hoffstätter. Computation and application of taylor polynomials with interval remainder bounds. *Reliable Computing*, 4(1):83–97, 1998.

[27] Gunther Reißig. Abstraction based solution of complex attainability problems for decomposable continuous plants. In *Decision and Control (CDC), 2010 49th IEEE Conference on*, pages 5911–5917. IEEE, 2010.

[28] Matthias Althoff. *Reachability analysis and its application to the safety assessment of autonomous cars.* PhD thesis, Technische Universität München, 2010.