Discrete Dynamics of Two-Dimensional Nonlinear Hybrid Automata

Lorenzo Sella and Pieter Collins

Centrum voor Wiskunde en Informatica,
Kruislaan 413, 1098 SJ Amsterdam, The Netherlands
{Lorenzo.Sella,Pieter.Collins}@cwi.nl

Abstract. In this paper, we develop an algorithm to compute under- and over-approximations to the discrete dynamics of a hybrid automaton. We represent the approximations to the dynamics as *sofic shifts*, which can be generated by a discrete automaton. We restrict to two-dimensional systems, since these give rise to one-dimensional return maps, which are significantly easier to study. Given generic non-degeneracy conditions, the under- and over-approximations computed by our algorithm converge to the discrete dynamics of the hybrid automaton. We apply the algorithms to two simple nonlinear hybrid systems, an affine switching system with hysteresis, and the singularly forced van der Pol oscillator.

Key words: hybrid automata, symbolic dynamics, interval computation, van der Pol equation

1 Introduction

Hybrid automata are dynamic systems which combine both discrete and continuous behaviour. Hybrid automata are frequently used to modelling systems in which dynamics occurs on different time scales, such as a slow-moving physical object controlled using a fast-switching digital controller. In many cases, the exact details of the continuous dynamics is relatively unimportant, and only the qualitative behaviour given by the discrete dynamics is of interest. It is therefore of interest to have numerical methods for computing approximations to the discrete dynamics of a hybrid automaton.

Existing work on finding discrete abstractions to hybrid automata has mostly focused on bisimulation by discrete automata [1]. However, since the class of systems admitting nontrivial finite bisimulations is highly restricted (generalisations of timed automata), this approach can only be used for simple classes of system. More complex classes of systems can only be studied by computing discrete abstractions which either simulate, or are simulated by, the exact discrete dynamics of the hybrid automaton. Sequences of discrete abstractions simulating the hybrid automaton were constructed in [2] for polynomial hybrid automata using first-order logic over the reals. Discrete abstractions simulating the hybrid automaton based on quantising the state space were given by [3] and

used for reachability analysis, and by [4] and used for supervisory control. Two-dimensional hybrid systems have been studied in [5], and the singular limit of the van der Pol oscillator was studied in [6, 7].

In this paper, we present a method for computing both over- and under-approximations to the discrete dynamics. In the language of symbolic dynamics, we construct sofic shifts over- and under approximating the exact shift of the hybrid automaton. In the language of transition systems, we construct discrete automata simulating of the hybrid automaton, and discrete automata simulated by the hybrid automaton. Our method is generically optimal in the sense that under certain nondegeneracy conditions the shift maps obtained converge to the exact discrete dynamics as the accuracy in increased. In the current work, we concentrate on two-dimensional hybrid systems, since these can be reduced to one-dimensional return maps which are easier to analyse.

The results of this paper extend the validity of existing methods by constructing under-approximations as well as over-approximations to the discrete dynamics, by applying to general nonlinear systems, and by being convergent for generic systems.

The paper is organised as follows. In Section 2, we give technical preliminaries on hybrid systems, shift spaces symbolic dynamics and interval methods which we need later. In Section 3, we present algorithms for computing over- and under-approximations to the symbolic dynamics and prove their convergence. In Section 4 we demonstrate the effectiveness of the method by computing the discrete dynamics of an affine switching hybrid automaton and the singular limit of the Van Der Pol oscillator. Finally, in Section 5 we give some conclusions and suggestions for further research.

The main mathematical techniques used in this article include interval analysis and validated solution of differential equations, one-dimensional dynamical systems and symbolic dynamics. Good references to these topics are include the books [8–13].

2   Theoretical Preliminaries

We now introduce the basic definitions, terminology and results on hybrid automata, symbolic dynamics and interval analysis which we will need. Throughout, we write $f :\subset X \to Y$ to denote a function from a subset of X to Y, and $f : X \rightrightarrows Y$ to denote a multivalued function from X to Y.

2.1   Hybrid Automata

A hybrid automaton is a dynamic system in which continuous-time evolution is interspersed with discrete jumps.

Definition 1 (Hybrid automaton). *A* hybrid automaton *is a tuple* $\mathscr{H} = (Q, E, X, G, \phi, r)$ *where* Q *is a finite set of* modes*,* E *is a finite set of* events*,* $X = \bigsqcup \{X_q \mid q \in Q\}$ *is the* state space*,* $G = \bigcup \{G_e \subset X \mid e \in E\}$ *is the* guard

set, $\phi :\subset X \times \mathbb{R} \to X$ *is the* continuous dynamic, *and* $r :\subset G \times E \to X$ *is the* reset map. *We write* $r_e : G_e \to X$ *for the reset map corresponding to event* e. *Typically,* ϕ_q *is defined by a differential equation* $\dot{x} = \chi_q(x)$ *for* $x \in X_q$.

An *execution* of a hybrid automaton is an alternating sequence of continuous and discrete transitions:

$$\xi :\ x_0 \xrightarrow{\phi_{t_0}} y_0 \xrightarrow{r_{e_0}} x_1 \xrightarrow{\phi_{t_1}} y_1 \xrightarrow{r_{e_1}} x_2 \cdots \tag{1}$$

Often the quantitative behaviour of a hybrid automaton is unimportant, and only the qualitative behaviour given by the sequence of discrete locations visited and the sequence of discrete events which occur.

Definition 2 (Discrete dynamics). *The* discrete dynamics *of a hybrid automaton* $\mathscr{H}$ *is the set of all transition sequences*

$$q_0 \xrightarrow{e_0} q_1 \xrightarrow{e_1} q_2 \cdots \tag{2}$$

such that there exists an execution ξ *given by* (1) *with* $y_i \in X_{q_i}$ *for all* i.

In order to compute the discrete dynamics we need only look at the state just before each transition, giving rise to the *return map*.

Definition 3 (Return map). *The* return map f *of a hybrid automaton* $\mathscr{H}$ *is the transition system* $f :\subset G \times E \rightrightarrows G$ *with* $f_e :\subset G_e \rightrightarrows G$ *given by*

$$f_e(x) := \{y \in G \mid \exists\, t \in \mathbb{R}^+ \ s.t.\ y = \phi_t(r_e(x))\}. \tag{3}$$

In other words, f_e is defined by the transitions $\phi_t \circ r_e$, where the continuous evolution ϕ_t proceeds until the state enters a guard set.

If we ignore the discrete-event labels, the return map is a dynamical system $f :\subset G \rightrightarrows G$ defined by

$$f(x) := \{y \in G \mid \exists\, e \in E,\ t \in \mathbb{R}^+ \text{ s.t. } y = \phi_t(r_e(x))\}. \tag{4}$$

Note that the return map need not be everywhere defined since a point need not have any further discrete transitions. As shown in Figure 1, the return map may also be discontinuous and multivalued if the initial state starts on the boundary of two guard sets (a) or the continuous evolution grazes the guard set and returns to the interior of the state set X before later hitting the guard set (b).

Typically, the return map is piecewise-continuous on the guard set. We therefore need to develop an algorithm for computing discrete dynamics for piecewise-continuous maps.

2.2   Shift spaces and finite automata

If A is a finite alphabet of symbols, recall that the sequence space A^ω is compact under the *product topology* defined by the metric $d(\vec{s}, \vec{t}) = 2^{-m}$ where $m = \min\{n \in \mathbb{N} \mid s_n \neq t_n\}$. In other words, two sequences are "close" if they agree on a long initial subword.
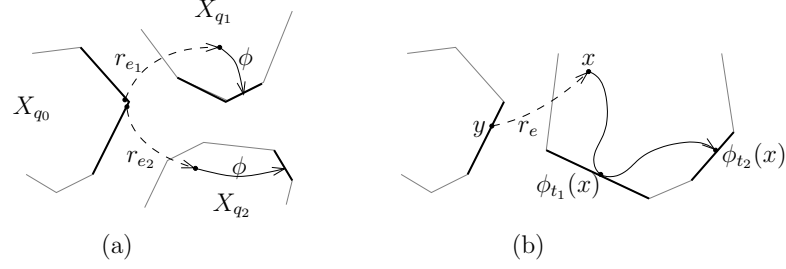
Fig. 1. Discontinuities of the return map. (a) Discontinuity at the boundary of a guard set. (b) Discontinuity caused by tangential contact with a guard set.

Definition 4 (Shifts). *Let A be a finite alphabet. The* shift map σ *on sequences A^ω is defined by $(\sigma\vec{s})_i = s_{i+1}$ for $i \in \mathbb{N}$. A* shift space *on A is a compact subset Σ of A^ω which is invariant under σ. A* shift *is the restriction of the shift map σ to a shift space Σ.*

A shift $\sigma|_\Sigma$ is a *subshift* of $\sigma|_{\widehat{\Sigma}}$ if $\Sigma \subset \widehat{\Sigma}$. Since shift spaces are compact subsets of a metric space, we can measure the difference between two shift spaces using the Hausdorff distance. If $\Sigma \subset \widehat{\Sigma}$, an alternative measure of the difference between Σ and $\widehat{\Sigma}$ is the difference in the topological entropies of $\sigma|_\Sigma$ and $\sigma|_{\widehat{\Sigma}}$.

A shift is *sofic* if it is generated by a finite automaton (as the sequence. Since the set of sofic shifts is dense in the space of all shifts on an alphabet A, sofic shifts are a convenient way of approximating arbitrary shifts.

2.3 Symbolic dynamics of piecewise-continuous maps

Symbolic dynamics is a powerful tool to analyse discrete-time dynamical systems. The basic idea is to compute the *itineraries* of orbits in terms of the regions of state space. The main complicating factor is that there is no nontrivial partition of a connected space M into compact pieces, so we instead use open sets whose closures cover the space.

Definition 5 (Topological partition). *A* topological partition *of a space M is a finite collection $\mathcal{P} = \{P_1, P_2, ..., P_n\}$ of mutually disjoint open sets such that $M = \bigcup_{i=1}^n \overline{P}_i$. The* boundary points *of $\mathcal{P}$ are elements of $\partial\mathcal{P} := \bigcup_{P \in \mathcal{P}} \partial P$.*

Given topological partitions $\mathcal{P}$ and $\mathcal{Q}$, we say that $\mathcal{P}$ is a refinement *of $\mathcal{Q}$ if for all $P \in \mathcal{P}$, there exists $Q \in \mathcal{Q}$ such that $P \subset Q$. The* join *of $\mathcal{P}$ and $\mathcal{Q}$ is defined by $\mathcal{P} \vee \mathcal{Q} = \{P \cap Q \mid P \in \mathcal{P}, \ Q \in \mathcal{Q} \text{ and } P \cap Q \neq \emptyset\}$.*

We shall consider piecewise-continuous functions defined as follows:

Definition 6 (Piecewise-continuous map). *Let $\mathcal{P}$ be a topological partition of M. A function $f : M \to M$ is $\mathcal{P}$-continuous if for all $P \in \mathcal{P}$, $f|_P$ is continuous and extends to a continuous function over $\overline{P}$. system*

We define $f_P^\circ = f|_P$, $\bar{f}_P$ to be the continuous extension of $f|_P$ to $\overline{P}$, $f^\circ :\subset X \to X$ by $f^\circ(x) = \bigcup_{P \in \mathcal{P}} f_p^\circ(x)$ and $\bar{f} = \bigcup_{P \in \mathcal{P}} \bar{f}_p(x)$.

We can use a topological partition to define an encoding of sequences in the space M.

Definition 7 (Itinerary). *Let $\mathcal{Q} = \{Q_s \mid s \in S\}$ be a topological partition of M, and $\vec{x} = (x_i)_{i \in \mathbb{N}}$ a sequence in M. A sequence $\vec{s}$ is a $\mathcal{Q}$-itinerary of $\vec{x}$ if $x_i \in Q_{s_i}$ for all $i \in \mathbb{N}$, and a $\overline{\mathcal{Q}}$-itinerary of $\vec{x}$ if $x_i \in \overline{Q}_{s_i}$ for all $i \in \mathbb{N}$.*

Given a topological partition, we can define the symbolic dynamics of a piecewise-continuous function f.

Definition 8 (Symbolic dynamics). *Let $\mathcal{Q} = \{Q_s \mid s \in S\}$ be a topological partition, $\mathcal{P}$ a refinement of $\mathcal{Q}$, and $f : M \to M$ a $\mathcal{P}$-continuous function.*

- *The* lower symbolic dynamics $\underline{\Sigma}(f)$ *of f is the closure of the set of all $\mathcal{Q}$-itineraries of orbits of f°.*
- *The* upper symbolic dynamics $\overline{\Sigma}(f)$ *of f is the closure of the set of all $\overline{\mathcal{Q}}$-itineraries of orbits of $\bar{f}$.*

The lower symbolic dynamics is a subshift of the upper symbolic dynamics. Intuitively, the lower symbolic dynamics consists of itineraries which are "robustly" present, and the upper symbolic dynamics excludes those itineraries which are "robustly" absent.

In order to prove that an itinerary $\vec{s}$ is not part of the upper shift, we use the following trivial result, which is valid in any dimension.

Proposition 1. *If there is an orbit $\vec{x}$ of $\bar{f}$ such that $x_i \in P_{s_i}$ for all i, then $\bar{f}(P_{s_i}) \cap P_{s_{i+1}} \neq \emptyset$ for all i.*

Symbolic dynamics for one-dimensional maps is substantially easier than in higher dimensions. The partition elements P are intervals, so can easily be represented by their boundary points. The symbolic dynamics can be computed using the kneading theory of [14] or by the following result.

Theorem 1. *Suppose $(R_0, R_1, \ldots)$ is a sequence of compact intervals such that f is continuous on each R_i and $f(R_i) \supset R_{i+1}$ for all i. Then there is an orbit $(x_0, x_1, \ldots)$ of f such that $x_i \in R_i$ for all i.*

For general piecewise-continuous functions, the lower symbolic dynamics may differ considerably from the upper symbolic dynamics. Under certain conditions, the lower symbolic dynamics $\underline{\Sigma}(f)$ and the upper symbolic dynamics $\overline{\Sigma}(f)$ coincide.

Theorem 2. *let $\mathcal{Q}$ be a partition of M, let $\mathcal{P}$ be a refinement of $\mathcal{Q}$, and let f be a $\mathcal{P}$-continuous map satisfying assumptions (A1-2) below. Then the lower symbolic dynamics $\underline{\Sigma}(f)$ equals the upper symbolic dynamics $\overline{\Sigma}(f)$.*

- **A1.** *The image under f° of every open subset of M contains an open set.*
- **A2.** *The forward orbits of all boundary points of $\mathcal{Q}$, all discontinuity points of f, and all critical points of f are disjoint from the boundary points of $\mathcal{Q}$.*

Assumption A1 is valid for any non-constant analytic map, and assumption A2 is valid for generic continuous functions.

2.4   Interval arithmetic

Since we typically cannot compute the return map of a hybrid system exactly, we resort to numerical approximation. In order to ensure that we can obtain rigorous conclusions from approximate numerics, we compute error bounds for all quantities. Hence a numerical approximation to a real number x is represented by an interval $\lfloor x \rceil = [\underline{x}, \overline{x}]$ such that $\underline{x} < x < \overline{x}$.

If f is a continuous function, then an *interval extension* $[f]$ of f is a function from intervals to intervals such that:

1. if $x \in \lfloor x \rceil$, then $f(x) \in [f](\lfloor x \rceil)$,
2. if $\lfloor x_2 \rceil \subset \lfloor x_1 \rceil$, then $[f](\lfloor x_2 \rceil) \subset \lfloor x_1 \rceil$, and
3. if $\lfloor y \rceil$ is any interval containing $y = f(x)$ in its interior, then there exists an interval $\lfloor x \rceil$ containing x such that $[f](\lfloor x \rceil) \subset \lfloor y \rceil$.

When computing an interval extension of f in practice, we set an a-priori numerical precision ϵ, and perform all computations to that precision. If the results are not sufficiently accurate, then we increase the precision and repeat the computations.

If f is n-times differentiable, we also assume that an interval extension is available for derivatives $f^{(i)}(x)$ for $i = 1, \ldots, n$. If f is piecewise-continuous, then an interval extension to f consists of:

1. interval approximations $\lfloor d_i \rceil$ to the discontinuity points d, and
2. an interval extension of $\tilde{f}_P$ over all continuous branches P.

If $[\underline{x}, \overline{x}]$ is an interval containing x, and $\lfloor y \rceil$ and $\lfloor z \rceil$ are intervals such that $[f](\lfloor y \rceil) < \underline{x}$ and $[f](\lfloor z \rceil) > \overline{x}$, then there is a point w in $\lfloor w \rceil = [\overline{y}, \underline{z}]$ such that $f(w) = x$. By using a modified version of the bisection algorithm, we can therefore compute preimages of points under f. We have the following result.

Lemma 1. *If f is strictly monotone on an interval P, then it is possible to compute an interval extension of $(f|_P)^{-1}$ from an interval extension of f.*

3   Algorithms for computing the discrete dynamics

In this section we present a numerical algorithm for computing the discrete dynamics of a two-dimensional hybrid automaton with one-dimensional guard sets, under assumptions which ensure that the return map is piecewise-continuous. We first briefly outline how to rigorously compute the return map and its derivatives arbitrarily accurately. We describe the algorithms to compute the symbolic dynamics of the return map, and finally state some convergence results.

3.1   Numerical computation of the return map

We now outline the numerical procedure to compute approximations to the return map. We consider hybrid automata such that:

B1. The state space X is a two-dimensional manifold with piecewise-differentiable boundary, and the guard set G is a subset of ∂X.

B2. The reset maps r_e are differentiable on their domains G_e.

B3. The continuous dynamics ϕ is given by a Lipschitz differential equation $\dot{x} = \chi(x)$.

We will also need a mild regularity assumption on the crossings of the continuous dynamics with the guard set. The following condition is sufficient:

B4. Whenever the continuous evolution $\xi(t)$ hits the guard set G, it either crosses G transversely, or touches G at a corner point or a quadratic tangency and continuous within X.

Under the assumptions $(B1 - 4)$, the return map is piecewise-differentiable, though may not be everywhere defined on G. Locally, we can represent the guard set G as $\{x \in X \mid g(x) = 0\}$ for some differentiable function g. The return map is then given by $y = f(x) = \phi_t(r(x))$ under the constraint $g(y) = 0$. Using a rigorous high order integration scheme, such as Lohner's method or Taylor methods [15–17],it is possible to compute $f(x)$ and $f'(x)$ to arbitrary accuracy away from corners of the guard set and grazing points of the flow.

We obtain a numerical discretisation of the return map f in terms of interval arithmetic. More specifically:

Theorem 3. *Let $\mathcal{H}$ be a hybrid system satisfying hypotheses (B1-4). Let f be the return map of $\mathcal{H}$. Then it is possible to compute an interval extension of f and its derivative f'.*

In the subsequent analysis, we shall only use information about the numerically-computable interval extension $[f]$, and not assume that we have an analytic description of the map itself.

3.2   Computing the discrete dynamics

In this section we show how to compute the symbolic dynamics of a map f which is piecewise continuous on branches $\mathcal{C}$ relative to a partition $\mathcal{Q}$. The basic strategy is outlined in the following procedure:

Algorithm 4 *Let $\mathcal{Q}$ be a partition of X, $\mathcal{E}$ be a topological partition of X and f be a piecewise-continuous function with nondegenerate critical points.*

1. *Fix a numerical precision ϵ and a maximum number of steps n.*
2. *Compute an approximate topological partition $\mathcal{C}$ refining $\mathcal{Q}$ and $\mathcal{E}$ such that f is continuous on each piece of $\mathcal{C}$*
3. *Refine the partition $\mathcal{C}$ to obtain an approximate topological partition $\mathcal{M}$ such that f is monotone on each partition element.*
4. *Refine the partition $\mathcal{M}$ by repeating the one of the following partitioning strategies at most n times to obtain a partition $\mathcal{R}$.*
   - *Forward refinement: Refine a partition $\mathcal{P}$ by introducing new partition boundary points at $f(p)$ for boundary points $p \in \partial\mathcal{P}$.*

- *Backward refinement: Refine a a partition $\mathcal{P}$ by introducing new partition boundary points at $f^{-1}(p)$ for $p \in \partial\mathcal{P}$*

  In either strategy, do not introduce any new points $\lfloor y \rceil$ which overlap existing boundary points $\lfloor p \rceil$.

5. *For each pair $R, R' \in \mathcal{R}$, compute whether $\bar{f}(\overline{R}) \cap \overline{R}' \neq \emptyset$, $f(R) \supset R'$ or $f(R) \subset R'$.*

6. *Define the under-approximation $\Lambda = \Lambda_{\epsilon,n}$ and the over-approximation $\Upsilon = \Upsilon_{\epsilon,n}$ to the symbolic dynamics to consist of all sequences $(Q_0, Q_1, \ldots)$ for which there exists $(R_0, R_1, \ldots) \in \mathcal{R}^\omega$ with $R_i \subset Q_i$ and*

   Λ: *$\exists k \in \mathbb{N} \cup \{\infty\}$ s.t. $f(R_i) \supset R_{i+1}$ $\forall i < k$, and $f(R_i) \subset R_{i+1}$ $\forall i \geq k$.*

   Υ: *$\bar{f}(\overline{R}) \cap \overline{R}' \neq \emptyset$ $\forall i \in \mathbb{N}$.*

The data type representing a boundary point p of a partition $\mathcal{P}$ has two fields, a `value` field which is an interval approximation $\lfloor p \rceil$ to p, and a `image` field which is a reference or pointer to the object representing $f(p)$. If p is a discontinuity point, then we store two image points, namely the image of p under both branches of $\bar{f}$ at p.

To compute the critical points, we need information on the derivative f. A point c is a critical point if $f'(c) = 0$, and the zeros of f' can easily be computed to arbitrary accuracy by a bisection strategy.

In certain degenerate cases, we may not be able to distinguish two discontinuity points of f, or a discontinuity point and a critical point. Although it is possible to handle these cases in a consistent way, in this paper we assume for simplicity that these cases do not arise.

The forward refinement of a partition $\mathcal{P}$ can be easily computed, since we need simply compute the images of all boundary points of $\mathcal{P}$. The boundary points of the backward refinement of $\mathcal{P}$ can be computed using Lemma 1. The main advantage of the forward refinement strategy are that better results can usually be obtained with fewer partition points, but convergence to the symbolic dynamics using forward refinement need not be monotone.

If $\mathcal{P}$ is a partition with monotone branches, then the n-step backward refinement of $\mathcal{P}$ consists of sets $P_{i_0, i_1, \ldots, i_{n-1}}$ defined recursively by $P_{i_0} \in \mathcal{P}$ and $P_{i_0, i_1, \ldots, i_{n-1}} = P_{i_0} \cap f^{-1}(P_{i_1, \ldots, i_{n-1}})$. For any $x \in P_{i_0, i_1, \ldots, i_{n-1}}$, we have $f^k(x) \in P_{i_k}$, so each partition element of an n-step backward refinement of $\mathcal{P}$ determines the first n elements of a $\mathcal{P}$-itinerary.

In order to determine whether $f(R) \supset R'$ or $\bar{f}(\overline{R})$ is disjoint from $\overline{R}'$, we need to know the relative ordering of the boundary points i.e. whether $r_i < r_j$, $r_i = r_j$ or $r_i > r_j$, and the image point r_k of r_i under f. As long as the interval approximations to the boundary points do not overlap, we can determine the relative ordering. Further, since when refining we compute each $\lfloor r_{i+1} \rceil$ as an image or preimage of some $p_i \in \lfloor r_i \rceil$, we have exact information about the images by construction.

3.3 Convergence to the discrete dynamics

We now describe how the sofic shifts computed by Algorithm 4 approximate the symbolic dynamics of f, and give sufficient conditions under which the approximations converge.

Theorem 5. *Let Λ and Υ be the shifts obtained by Algorithm 4. Then Λ is a subshift of $\underline{\Sigma}(f)$ and $\overline{\Sigma}(f)$ is a subshift of Υ.*

The following result shows that if the upper symbolic dynamics and the lower symbolic dynamics are equal, then the algorithm converges.

Theorem 6. *Suppose f is a piecewise-continuous map satisfying assumptions (A1-2). Then the under- and over-approximations $\Lambda_{\epsilon,n}$ and $\Upsilon_{\epsilon,n}$ to the discrete dynamics computed by Algorithm 4 using backwards refinement converge to the symbolic dynamics of f as the accuracy $1/\epsilon$ and maximum number of steps n increase.*

The proof is technical and will be published elsewhere. The main observation is that under condition A2, there is essentially no difference between an algorithm using interval arithmetic, and an algorithm using exact values. Under weaker assumptions, it is possible to prove that lower approximations $\Lambda_{\epsilon,n}$ and upper approximations $\Upsilon_{\epsilon,n}$ converge to the lower shift $\underline{\Sigma}$ and the upper shift $\overline{\Sigma}$.

3.4 Representing the symbolic dynamics by discrete automata

We can represent the discrete dynamics of the hybrid automaton by a discrete automaton whose states are the elements of $\mathcal{R}$, with two types of arrows. We draw a solid arrow $R \to R'$ if $f(R) \supset R$, and a dashed arrow $R - \to R'$ if merely $\bar{f}(\overline{R}) \cap \overline{R}' \neq \emptyset$. Further, if there is a sequence $(R_{s_0}, R_{s_0}, \ldots)$ such that $f(R_{s_i}) \subset R_{s_{i+1}}$ for all i, then we replace all dashed arrows $R_{s_i} - \to R_{s_{i+1}}$ with solid arrows $R_{s_i} \to R_{s_{i+1}}$. The shift Γ is generated by the solid arrows, and Υ is generated by both dashed and solid arrows.

We can label an arrow starting in R with the event e if $R \subset G_e$. If $\vec{s}$ is a sequence of symbols and there is a *solid* path:

$$R_{s_0} \xrightarrow{e_0} R_{s_1} \xrightarrow{e_1} R_{s_2} \cdots \tag{5}$$

then there must be an orbit $\vec{x} = (x_0, x_1, \ldots)$ of the return map f such that $x_i \in R_i$ for all i and $f_{e_i}(x_i) = x_{i+1}$. Hence there is an exectution of the hybrid automaton with itinerary:

$$q_0 \xrightarrow{e_0} q_1 \xrightarrow{e_1} q_2 \cdots \tag{6}$$

where $R_{s_i} \subset X_{q_i}$ for all i. From the construction of Υ and Proposition 1, we see that if there is an orbit of the return map with $x_i \in \overline{R}_i$ for all i, then there must be a *broken* path (which may consist of both dashed and solid arrows):

$$R_{s_0} - \xrightarrow{e_0} R_{s_1} \xrightarrow{e_1} R_{s_2} - \xrightarrow{e_k} R_{s_3} \cdots . \tag{7}$$

4   Case studies

In this section, we present two case studies: a simple hysteresis system and the singular limit of the van der Pol equation.

4.1   A hysteresis switching system

We now consider a piecewise-affine model of a system governed by hysteresis switching [18]. We let $H(x)$ be the hysteresis map $H(x) = 0$ for $x \leq 1$, the first mode, and $H(x) = 1$ for $x \geq 0$, the second mode, and consider the system:

$$\dot{x} = y + a_1 H(x/b) \tag{8}$$
$$\dot{y} = -x - 2\sigma y + a_2 H(x/b).$$

The return map is defined on the set $P = \{(x,y) \in \mathbb{R}^2 \mid x = 0, \ y > 0\}$ and maps a point $p \in P$ to a point $q \in P$ such that q belongs to the same orbit of p, and q is the next intersection after p in forward time of this orbit with P in the first mode. We have computed symbolic dynamics for the return map with parameter values $a_1 = -1$, $a_2 = -1$, $b = 0.3$ and $\sigma = -0.2$. The graph of the return map is shown in Fig. 2. We take an initial partition $\mathcal{Q}$ which are the
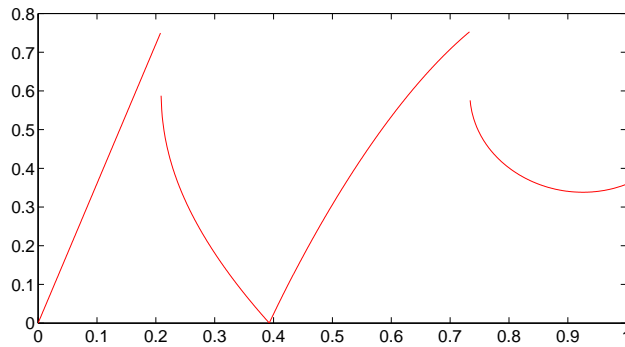

Fig. 2. The return map for the hysteresis system (9).

domains of the monotone branches. The partition elements are $Q_0 = [p_0, p_1]$, $Q_1 = [p_1, p_3]$, $Q_2 = [p_3, p_5]$, $Q_3 = [p_5, p_7]$ and $Q_4 = [p_7, p_8]$ where the boundary points are

$p_0 = 0.0, \ p_1 \approx 0.20894, \ p_3 \approx 0.39278, \ p_5 \approx 0.73329, \ p_7 \approx 0.92580, \ p_8 = 1.0.$

The associated symbolic dynamics is in Fig. 3(a). The two points of discontinuity are p_1 and p_5 and they can be proved to have the same left and right images. The partition after one iteration of forward refinements has the following additional
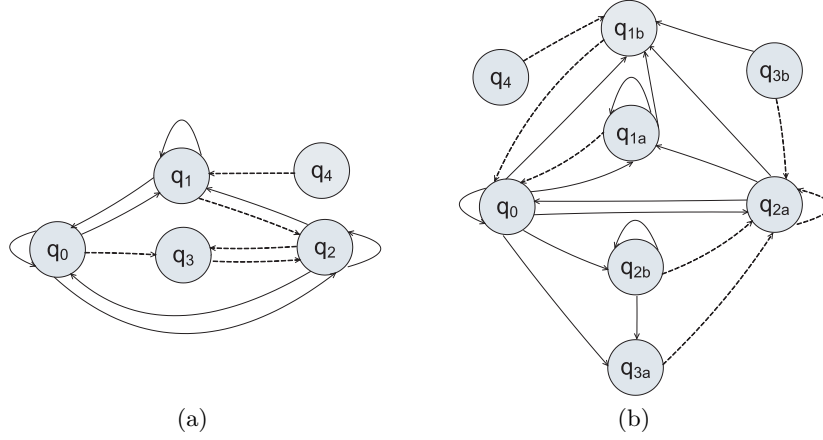
(a)   (b)

Fig. 3. Lower and upper approximation of symbolic dynamics for the hysteresis system (9) for (a) the initial partition and (b) the forward refinement of the initial partition

endpoints:

$$p_2 = f(p_7) \approx 0.33792, \ p_4 = f(p_1^+) = f(p_5^+) \approx 0.59890,$$
$$p_6 = f(p_1^-) = f(p_5^-) \approx 0.75340.$$

The symbolic dynamics generated by this partition is approximated by the graph in Fig. 3(b).

We notice that the lower approximation of the dynamics of the refined partition misses some sequences of the lower approximation of dynamics of the initial partition. This is due to the fact that although region $Q_1 = [p_1, p_3]$ covers $Q_0 = [p_0, p_1]$ under one iterate of the return map, neither of the subdivided regions $P_{1;0} = [p_1, p_2]$ and $P_{1;1} = [p_2, p_3]$ cover Q_0. Hence the convergence of the lower approximations to the symbolic dynamics computed using forward refinement is not monotone. With backward refinement the convergence can be shown to be monotone, but backward refinements have the disadvantage of being slower to compute than forward refinements.

The lower shift for the initial partition can be written as the regular expression

$$(q_0^* q_2)^* (q_0^\omega + q_1^\omega) + (q_0^* q_2 q_0)^* (q_1^\omega + q_2^\omega) + (q_0^* q_2)^\omega.$$

We can see for instance that the periodic sequence $(q_0 q_3 q_2 q_1)^\omega$ belongs to the upper shift but not to the lower shift. From the two shifts we can show that the topological entropy lies in the interval $[0.80958, 1.27020]$. The topological entropies obtained for further refinement are shown in the table :

steps	entropy
3	[0.97494,1.26249]
5	[1.02407,1.18582]
7	[1.04636,1.16493]
12	[1.06873,1.15087]

4.2   The van der Pol equation

The forced van der Pol equation is a nonlinear ordinary differential equation modeling oscillation in a vacuum tube triode circuit. Bifurcations in the singular limit of the forced van der Pol oscillator have been studied in [19]. In this paper we analyse the following version of the equation:

$$\ddot{x} + \mu(x^2 - 1)\dot{x} + x = a(x^2 - 1)\sin(2\pi\nu\tau) \tag{9}$$

in the singular limit as $\mu \to \infty$. To obtain a form more convenient for analysis, we rescale time $t = \tau/\mu$, introduce new parameters $\varepsilon = \frac{1}{\mu^2}$, $\omega = \nu\mu$ and $\theta = \omega t$, and define the new variable $y = \dot{x}/\mu^2 + x^3/3 - x$. We obtain the following autonomous system:

$$\varepsilon\dot{x} = y + x - x^3/3; \tag{10}$$
$$\dot{y} = -x + a(x^2 - 1)\sin(2\pi\theta); \tag{11}$$
$$\dot{\theta} = \omega.$$

The *fast subsystem* is defined by (10), since the dynamics of the fast variable x occurs on a time scale which is fast relative to the evolution of the *slow variables* y and θ.

   We see that on the *critical manifold* $y + x - x^3/3 = 0$ the system evolves on a time scale of order t. However, the critical manifold is unstable for the fast system if $|x| \le 1$, and that when this occurs, the value of x jumps instantaneously to one of the stable fixed points of (10).

   We can therefore view the singular limit as a hybrid system in which the continuous dynamics is given by the slow flow on the stable sheet of the critical manifold, and the reset map is given by the fast flow. By eliminating y, we obtain the following dynamics for the slow subsystem:

$$\dot{x} = -x + a(x^2 - 1)\sin(2\pi\theta) \tag{12}$$
$$\dot{\theta} = \omega(x^2 - 1)$$

The fast dynamics is described by the guard set and reset map

$$G = \{(x,\theta) \mid |x| = 1\}; \qquad r(x,\theta) = (-2\operatorname{sgn}(x), \theta). \tag{13}$$

In other words, when the guard condition $|x| = 1$ becomes satisfied the state jumps to $x = \mp 2$.

   Since the dynamics is symmetric under the transformation $T(x,\theta) = (-x, \theta + 1/2)$, we can post-compose the return map from the guard set $x = 1$ to the guard set $x = -1$ with T to obtain the *half return map* f taking $\{(r,\theta) \mid r = 1\}$ into itself. The graph of the half return map for parameter values $a = 5$ and $\omega = 3$ is shown in Fig. 4. We have computed the lower and upper symbolic dynamics with respect to the partition given by the continuous branches using forward refinement.
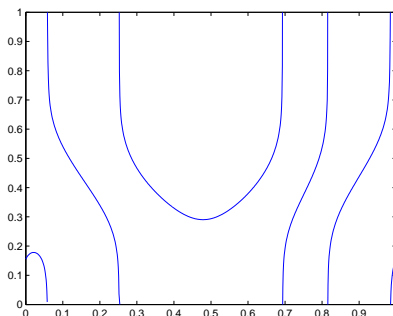
Fig. 4. The half return map for the singular limit of the forced van der Pol oscillator (12).

In the return map there are 5 discontinuity points:

$$p_2 \approx 0.05816, \ p_3 \approx 0.25226, \ p_5 \approx 0.69356, \ p_6 \approx 0.81553, \ p_7 \approx 0,98495.$$

and 2 critical points, a local maximum $p_1 \approx 0.02183$ and a local minimum $p_4 \approx 0.47872$. These points with the extremes of the interval $p_0 = 0$ and $p_8 = 1$ generate an initial partition of 9 pieces. After one forward iteration we obtain 11 pieces.

$$q_0 = f(p_0) \approx 0.15520, \ q_1 = f(p_1) \approx 0.17825, \ q_2 = f(p_4) \approx 0.29017,$$

The lower and upper discrete automata are not included for reasons of space. After one step of refinement, the discrete automaton representing the symbolic dynamics separate into two strongly connected components. Both the lower and upper shifts include the component with the highest entropy, while the lower shift does not include the smallest.

Therefore the topological entropy of the lower and upper shifts are equal and can be computed exactly yielding a numerical value of approximately 1.55705. From Fig. 4 we could already infer the entropy is at least $\log(3) \approx 1.09861$ because there are 3 continuous pieces of the partition which map the whole interval. From numerical computation we can deduce the existence of an attracting periodic orbit close to the local minimum. This let us infer the existence of a chaotic invariant Cantor set, every point non belonging to this set converges to the attracting periodic orbit.

5   Conclusion

In this paper, we have presented a method for computing the discrete dynamics of a hybrid automaton with a two-dimensional state space. We obtain sequences of sofic shifts which approximate the actual discrete dynamics from above and below. We have given nondegeneracy conditions under which the method is optimal, in the sense that the resulting shift spaces converge to the actual shift

space. In future work, we plan to remove the nondegeneracy conditions on the method, and extend the method to hybrid automata in higher dimensions.

References

1. Alur, R., Henzinger, T.A., Lafferriere, G., Pappas, G.J.: Discrete abstractions of hybrid systems. Proc. IEEE **88** (2000) 971–984
2. Tiwari, A., Khanna, G.: Series of abstractions for hybrid automata. In Tomlin, C.J., Greenstreet, M.R., eds.: Hybrid Systems: Computation and Control. Volume 2289 of Lecture Notes in Computer Science., Springer-Verlag (2002) 465–478
3. Lunze, J., Nixdorf, B.: Discrete reachability of hybrid systems. Internat. J. Control **76**(14) (2003) 1453–1468
4. Moor, T., Raisch, J., O'Young, S.: Discrete supervisory control of hybrid systems based on l-complete approximations. Discrete Event Dyn. Syst. **12**(1) (2002) 83–107 WODES '98 (Cagliari).
5. Guckenheimer, J., Johnson, S.: Planar hybrid systems. In: Hybrid systems, II (Ithaca, NY, 1994). Volume 999 of Lecture Notes in Comput. Sci. Springer, Berlin (1995) 202–225
6. Grasman, J., Nijmeijer, H., Veling, E.J.M.: Singular perturbations and a mapping on an interval for the forced van der Pol relaxation oscillator. Phys. D **13**(1-2) (1984) 195–210
7. Bold, K., Edwards, C., Guckenheimer, J., Guharay, S., Hoffman, K., Hubbard, J., Oliva, R., Weckesser, W.: The forced van der Pol equation. II. Canards in the reduced system. SIAM J. Appl. Dyn. Syst. **2**(4) (2003) 570–608 (electronic)
8. Moore, R.E.: Interval analysis. Prentice-Hall Inc., Englewood Cliffs, N.J. (1966)
9. Jaulin, L., Kieffer, M., Didrit, O., Walter, É.: Applied interval analysis. Springer-Verlag, London (2001)
10. Aberth, O.: Introduction to Precise Numerical Methods. Academic Press (2007)
11. Lind, D., Marcus, B.: An introduction to symbolic dynamics and coding. Cambridge University Press, Cambridge (1995)
12. Kitchens, B.P.: Symbolic dynamics. Universitext. Springer-Verlag, Berlin (1998) One-sided, two-sided and countable state Markov shifts.
13. Katok, A., Hasselblatt, B.: Introduction to the modern theory of dynamical systems. Volume 54 of Encyclopedia of Mathematics and its Applications. Cambridge University Press, Cambridge (1995)
14. Milnor, J., Thurston, W.: On iterated maps of the interval. In: Dynamical systems (College Park, MD, 1986–87). Volume 1342 of Lecture Notes in Math. Springer, Berlin (1988) 465–563
15. Berz, M., Makino, K.: Verified integration of ODEs and flows using differential algebraic methods on high-order Taylor models. Reliab. Comput. **4**(4) (1998) 361–369
16. Nedialkov, N.S., Jackson, K.R., Corliss, G.F.: Validated solutions of initial value problems for ordinary differential equations. Appl. Math. Comput. **105**(1) (1999) 21–68
17. Wilczak, D., Zgliczynski, P.: C^r Lohner algorithm. oai:arXiv.org:0704.0720 (2007)
18. Newcomb, R.W., El-Leithy, N.: Chaos generation using binary hysteresis. Circuits Systems Signal Process. **5**(3) (1986) 321–341
19. Guckenheimer, J., Hoffman, K., Weckesser, W.: The forced van der Pol equation. I. The slow flow and its bifurcations. SIAM J. Appl. Dyn. Syst. **2**(1) (2003) 1–35 (electronic)