Computation of symbolic dynamics of low-dimensional maps

Thomas Stieltjes Institute
for Mathematics

NWO
Netherlands Organisation for Scientific Research

VRIJE UNIVERSITEIT

Computation of symbolic dynamics of low-dimensional maps

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad Doctor aan
de Vrije Universiteit Amsterdam,
op gezag van de rector magnificus
prof.dr. L.M. Bouter,
in het openbaar te verdedigen
ten overstaan van de promotiecommissie
van de faculteit der Exacte Wetenschappen
op dinsdag 1 december 2009 om 10.45 uur
in de aula van de universiteit,
De Boelelaan 1105

door

Lorenzo Sella

geboren te Verona, Italië

promotor:	prof.dr.ir. J.H. van Schuppen
copromotor:	dr. P.J. Collins

"To see a world in a grain of sand,
And a heaven in a wild flower,
Hold infinity in the palm of your hand,
And eternity in an hour."

William Blake

Contents

Introduction

The field of discrete and continuous dynamical systems is one of the fields of mathematics most applied to modern sciences since the age of Newton. Since an early stage the study of these systems showed to be frequently challenging because of the complexity of their solutions. An emblematic example is the three body problem which was considered an intractbale problem by Isaac Newton and only two centuries later Henry Poincaré provided a first comprehensive insight of the problem.

Many dynamical systems are described by equations which are not solvable analytically and not reliably solvable numerically. In the late 19th century Poincaré developed a new qualitative approach to study complex dynamical systems which frequently were not investigable by other means. His work was one of the first important step into the study of chaotic systems.

Another pioneer in this field was Edward Lorenz. In the 1950's and 1960's this mathematician and meteorologist studied a simplified version of Navier-Stokes equations modelling the weather. By analyzing some numerical solutions of this equations he showed this system has high sensitivity respect to the initial conditions, a phenomenon called the butterfly effect.

Chaos is a property of dynamical system which has been characterised in many ways, as we mention above sensitivity to initial conditions, the presence of a periodic orbit whose period is three (more generallly an orbit whose period is not a power of two as was showed by Sharkovsky), positive topological entropy of a dynamical system over a compact space which is the measure of the complexity of the dynamic of the system.

Chaotic systems are extremely common in nature. Besides the Lorenz system other two very well studied systems discovered respectively by Duffing and Van der Pol in the first part of the 20th century, describe a mechanical system and electric circuit respectively. Frequently the same system can model physical systems of different disciplines; for instance, the system of van Der Pol is also used in neurology and seismology.

At roughly the same time Hedlund and Morse investigated geodesics on surfaces of negative curvature by infinite sequences of symbols. In their work

they studied sequence spaces with the shift transformation as dynamical system. Their work was the foundation of a new subject which they called symbolic dynamics. This subject revealed to be very useful to study dynamical systems. The key feature of symbolic dynamics is that represents a dynamical systems by a semiconjugate system over sequences of symbols, the itineraries of orbits, where the symbols corresponds to the elements of a partition of a state space. The symbolic system is much simpler to study than the original system, especially for determining qualitative properties and computing the topological entropy. Frequently the partition of a system modelling an applied science system can be chosen in such away that the sequences of the symbols have a quite relevant physical meaning.

For uniformly hyperbolic discrete-time systems there exists a partition of the state space such that the image of each element of the partition under the map of the system is the union of one or more elements of the partitions and there is a conjugacy between symbolic sequences - in terms of elements of the partition - and orbits of the systems. This type of partition is called a Markov partition. It is straightforward to express the symbolic dynamics of the system in terms of Markov partition as the infinite sequences associated to the infinite paths over a graph which represents the mapping of the system over the partition. In the general case the Markov partition may not exist, but it is still possible to choose a partition which induces a set of symbolic sequences conjugate to the orbits of the systems. The description of these sequences is not simple as in the uniformly hyperbolic case, it requires some strategy which approximate the symbolic dynamics by computing a superset and a subset of it.

Important work about how to perform this computation already exists in the literature, we will refer in details to all this work later on in the appropriate chapters, here we just mention the main contributions:

- covering relations [1],
- kneading theory for continuous maps in one dimension [2, 3, 4],
- tangle theory for continuous maps in two dimensions [5, 6, 7],
- braids theory for periodic orbit of surfaces diffeomorphism [8, 9, 10, 11],
- Conley index for higher dimension [12, 13].

The goals of the project related to this thesis was to find and to implement new techniques for computing symbolic dynamics of discontinuous maps in one and higher dimension as well as from these results to find an approach

to compute the discrete dynamics of hybrid systems, a class of system which combines continuous and discrete dynamics. The work in this thesis provide algorithms for computation of symbolic dynamics for piecewise-continuous maps in one dimension and piecewise-affine maps in two and higher dimension. These algorithms allow to study directly discrete-time systems, and indirectly continuous and hybrid system via the return map, the extension to discontinuous maps is essential for the study of the last class of systems, because the return maps of hybrid systems are in general discontinuous.

The one-dimensional case is used as a strategy based on covering relations combine with kneading theory. The study of symbolic dynamics in two and higher dimensions is based on the Conley index, an index defined in homology theory. In order to perform the computation of the index we developed a package which implement a calculus of non convex polyhedra which is the a type of object we used to model partitions, and homology objects.

Outline

The remaining part of this thesis is organized as follows:

In Chapter 1 we are going to give formal definitions of all concepts which are essential for this work some of which has already been mentioned above. We start by presenting shift theory and symbolic dynamics which are the mathematical framework this work is based on. We then present symbolic dynamics in one dimension and we describe kneading theory. Afterwards we present symbolic dynamics in two dimensions and we describe a homoclinic tangle, a structure derived from intersection of stable and unstable manifolds of a fixed point and the Conley index, an index defined in algebraic homology which is useful for proving existence of invariant set and consequently finding symbolic dynamics. At the end we discuss the way we deal with numerical error by interval arithmetic.

In Chapter 2 we will describe techniques we have developed for the computation of symbolic dynamics in one dimension for piecewise continuous maps. This strategy is based on covering relations. A second strategy combines covering relation with kneading theory. We apply our results to a number of examples, unimodal and bimodal maps, return map of hysteretic hybrid systems and the singular limit of the Van der Pol equation

In Chapter 3 we will describe techniques we have developed for the computation of symbolic dynamics in two dimensions for piecewise affine maps. We develop an algorithm for the computation of the tangle in order to apply the tangle algorithms as benchmark for our algorithms. We describe a calculus of non convex polyhedra which we need to develop our algorithm for symbolic dynamics based on the decomposition of the Conley Index. We

apply the results to a number of case studies, continuous and discontinuous Lozi-like maps and to a return maps of an hybrid system.

In Chapter 4 we show that the discrete dynamics of hybrid system is uncomputable by showing a sequence of hybrid systems whose limit of the discrete dynamics is not the discrete dynamics of the limit.

Chapter 1

Theoretical preliminaries

In this chapter we present definitions and results of topics which are relevant for this thesis. Section 1.1 contains a description of symbolic dynamics and shift space theory, which is the mathematical framework where symbolic dynamics is defined. Section 1.2 presents definitions and results about symbolic dynamics in one dimension, which includes the kneading theory of Milnor and Thurston. Section 1.3 presents main results about symbolic dynamics in two dimensions, which includes homoclinic tangle by Rom-Kedar, Collins and Conley Index decomposition by Szymzac which in principle can be adopted for any dimension. In section 1.4 we finally discuss some issues and solutions about how to deal with numerical errors and numerical approximation in the computation of the symbolic dynamics.

1.1 Shifts and symbolic dynamics

1.1.1 Shift space theory

Shift space theory deals with infinite sequences $\vec{s} = (s_0, s_1, s_2, \ldots)$ of symbols over a finite alphabet $A = \{a_0, ..., a_{n-1}\}$, the sequence space $A^{\mathbb{N}}$ 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.

Definition 1.1.1 (Shift map)**.** *The* shift map σ *on sequences* A^ω *is defined by* $(\sigma\vec{s})_i = s_{i+1}\ \forall i \in \mathbb{N}$.

Definition 1.1.2 (Shift space)**.** *A* shift space *on* A *is a compact subset* Σ *of* $A^{\mathbb{N}}$ *which is invariant under* Σ*. A shift* Σ *is a* subshift *of* $\widehat{\Sigma}$ *if* $\Sigma \subset \widehat{\Sigma}$*. If* Σ *is a subshift of* $\widehat{\Sigma}$*, then the shift map on* $\widehat{\Sigma}$ *simulates the shift map on* Σ*.*

Since shift spaces are compact subsets of a metric space, we can measure the difference between two shift spaces using the Hausdorff distance. If Σ is a subshift of $\widehat{\Sigma}$, an alternative measure of the difference between a shift and a subshift is the difference in entropy h defined as follows:

Definition 1.1.3 (Entropy of a shift)**.** *Let Σ be a shift space then the entropy of Σ is*

$$h(\Sigma) = \lim_{n\to\infty} \frac{1}{n} \log |B_n(\Sigma)|$$

where $B_n(\Sigma)$ denotes the set of subwords of length n of sequences of Σ.

We must note that Hausdorff distance and entropy provide informations substantially different, in facts the first is a measure about transient behaviour while the second about the recurrent behaviour. Some shifts might be very close in Hausdorff distance but distant in entropy.

A natural way of generating shift spaces is by finite automata. A *discrete automaton* is a tuple $\mathscr{A} = (A, Q, \delta, \lambda)$ where $A = \{a_0, ..., a_{m-1}\}$ is a finite set of states, $Q = \{q_0, ..., q_{n-1}\}$ a finite set of symbols, $\delta : A \to \mathcal{P}(A)$ is the transition relation and $\lambda : A \to Q$ is the labelling maps.

The dynamics generated by an automaton $\mathscr{A}$ is the set of all sequences $\vec{s} = (\lambda(s_0), \lambda(s_1), \ldots)$ with $s_i \in A$ such that $s_{i+1} \in \delta(s_i)$, these sequences are called runs of the automaton. We can now define the following:

Definition 1.1.4 (Sofic shift)**.** *A shift Σ is* sofic *if it is generated by a finite automaton.*

Since the set of sofic shifts is dense in the space of all shifts on an alphabet A, sofic shifts represented by finite automata are a convenient way of approximating arbitrary shifts. An important result relates the entropy of a class of sofic shifts to the spectral radius of the adjacency matrix of the automaton:

Theorem 1.1.1. *Let Σ a sofic shift represented by an automaton such that each sequence corresponds to only one run (this representation exists for every sofic shift). Let M the adjacency matrix of the automaton of the sofic shift then:*

$$h(\Sigma) = \ln(\rho(M))$$

where $\rho(M)$ is the spectral radius of M.

A superclass of sofic shifts are the cocyclic shifts. Intuitively they can be associated to an automaton where some runs are eliminated when the product of some linear applications associated to their paths become zero:

Definition 1.1.5 (Cocyclic shift)**.** *A shift* Σ *is* cocyclic *if there is a group of linear applications defined on some space* $:M_1, ...M_k$ *such that* $s \in \Sigma$ *iff for any* $k, m \in \mathbb{N}$, $M_{s_k} M_{s_{k+1}} ... M_{s_{k+m}} \neq 0$.

1.1.2 Symbolic dynamics

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.

A *topological partition* of a space X is a finite collection $\mathcal{P} = \{P_1, P_1, ..., P_k\}$ of mutually disjoint open sets such that $X = \bigcup_{i=1}^{k} \overline{P}_i$. It is convenient to assume that the elements of $\mathcal{P}$ are *regular* sets, that is, $\overline{P}^{\circ} = P$ for all $P \in \mathcal{P}$ ($\overline{P}^{\circ}$ denotes the interior of the closure of P). 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 $\mathcal{Q}$ is a *refinement* of $\mathcal{P}$ if for all $Q \in \mathcal{Q}$, there exists $P \in \mathcal{P}$ such that $Q \subset P$. 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 1.1.6 (Piecewise-continuous map)**.** *A function* $f : X \to X$ *is* piecewise-continuous *if there is a topological partition* $\mathcal{P}$ *of* X *such that for all* $P \in \mathcal{P}$, $f|_P$ *is continuous, and that* $f_P := f|_P$ *extends to a continuous function* $\bar{f}_P$ *on* $\overline{P}$. *We let* $f^{\circ} = f|_{\bigcup \mathcal{P}}$

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

Definition 1.1.7 (Symbolic dynamics)**.** *Let* $\mathcal{Q}$ *be a topological partition. Suppose* $f : X \to X$ *is monotone continuous on each element of* $\mathcal{L}$.

- *The* lower symbolic dynamics $\underline{\Sigma}(f, \mathcal{Q})$ *of* f *is the the set of sequences* $\vec{s}$ *such that* $\forall m \in \mathbb{N}$ *there exists* $\epsilon > 0$ *such that for all* ϵ*-perturbations* f_ϵ *of* f *with the same monotone branches,* f_ϵ *has an orbit whose* $\mathcal{Q}$*-itinerary has the same* m*-prefix of* $\vec{s}$.

- *The* upper symbolic dynamics $\overline{\Sigma}(f, \mathcal{Q})$ *of* f *is the set of all sequences* $\vec{s}$ *such that* $\vec{s}$ *for all* $\epsilon > 0$, *there exists an* ϵ*-perturbation* f_ϵ *of* f *with the same monotone branches such that* f_ϵ *has an orbit with* $\mathcal{Q}$*-itinerary* $\vec{s}$.

The motivation for these definitions are that the lower and upper symbolic dynamics give under and over approximations to the symbolic dynamics which can be effectively computed.

The following results are trivial, but provide the main tools for determining whether an itinerary is present in the system.

Proposition 1.1.2. *If there is an orbit $\vec{x}$ such that $x_i \in \overline{R}_i$ with $\overline{R}_i \subset \overline{P}_i$ for all i, then $\bar{f}_{P_i}(\overline{R}_i) \cap \overline{R}_{i+1} \neq \emptyset$ for all i.*

Proposition 1.1.3. *Let f be a map defined on a one dimensional space and suppose $n \in \mathbb{N} \cup \{\infty\}$ and $(R_0, R_1, \ldots)$ is a sequence such that $f(R_i) \supset R_{i+1}$ for all $i < n$, and $f(R_i) \subset R_{i+1}$ for all $i \geq n$. Then there is an orbit $(x_0, x_1, \ldots)$ of f such that $x_i \in R_i$ for all i.*

The following result relates the inner and outer symbolic dynamics.

Proposition 1.1.4. *If the sets $\overline{Q}$ for $Q \in \mathcal{Q}$ are compact, then the inner symbolic dynamics of f is a subshift of the outer symbolic dynamics of f.*

Definition 1.1.8 (Topological entropy)**.** *The topological entropy of a map $f : X \longrightarrow X$ is defined as:*

$$h(f) = \lim_{\epsilon \to 0} \left(\limsup_{n \to \infty} \frac{1}{n} \log N(n, \epsilon) \right)$$

where $N(n, \epsilon) = \max_{S \in V_{n,\epsilon}} |S|$ and

$$V_{n,\epsilon} = \{S \subset X \mid \forall x, y \in S \max\{d(f^i(x), f^i(y)) : 0 \leq i < n\} \geq \epsilon\}$$

The interpretation of this quantity is the average exponential growth rate of the number of n-partial orbits which can be distinguished with accuracy ϵ, there is also a measure theoretic definition of entropy which has been proved to coincide with the above definition and can be interpreted as the rate of increase of disorder of a space under the action of the dynamical system. The entropy of the symbolic dynamics has the same value of the entopy of the map when the partition is generating

1.2 Symbolic dynamic in one dimension

Symbolic dynamics for one-dimensional maps is substantially easier than in higher dimensions. The partition elements $Q \subset \mathcal{Q}$ are intervals, so can easily be represented by their boundary points q_i. The topological entropy of a continuous interval map is always lower-semicontinuous [14], and can be shown to be upper-semicontinuous in C^{infty} if the critical points are non-degenerate [15, 16]. Of particular importance are the *monotone branches*,

sometimes known as *laps*. It is trivial that any point has at most one preimage in any lap, and this point is readily computed by bisection.

The wildest class of fucntion for which we study symbolic dynamics is the the class of piecewise-monotone-continuous functions. For this class we can consider perturbations by functions with the same critical and discontinuity points. This gives a refined notion of approximations to the symbolic dynamics within a class.

If $\mathcal{R}$ is a topological partition with boundary points $\{r_0, r_1, \ldots, r_m\}$ and elements $R_k = [r_k, r_{k+1}]$, then we introduce new symbols $\frac{0}{1}$, $\frac{1}{2}$ etc. as the code for the points $r_1, \ldots, r_m$. In other words, we use the symbol $\frac{1}{2}$ for a point in $\overline{R}_1 \cap \overline{R}_2$. This greatly facilitates the writing of itineraries. We also write $\mathrm{itin}(x^{\pm}) = \lim_{\epsilon \to 0} \mathrm{itin}(x \pm \epsilon)$.

The following example shows that the upper symbolic dynamics for maps with the same monotone branches can be different from the outer symbolic dynamics.

Example 1.2.1. Let $X = [-1, 1]$ and $f(x) = x/2$. Let $\mathcal{Q} = \{Q_0, Q_1\}$ with $Q_0 = (-1, 0)$ and $Q_1 = (0, 1)$.

Since any sequence of zeroes and ones is a valid $\overline{Q}$-itinerary of the point 0, the outer symbolic dynamics is the full shift on $\{0, 1\}$. Further, among all perturbations of f, we can generate orbits with all itineraries by fitting a small version of the tent map near 0.

For perturbations f_ϵ of f among monotone maps (which include all C^1-perturbations), the possible itineraries depend on whether $f_\epsilon(0) \gtrless 0$. For $f_\epsilon(0) > 0$, there exists N such that the $\mathcal{Q}$-itineraries are of the form 0^n1^* for $n \leq N$, and for $f_\epsilon(0) < 0$ we have itineraries 1^n0^* for $n \leq N$. For $f_\epsilon(0) = 0$, the $\mathcal{Q}$-itineraries are $\overline{0}$ and 1^*. Taking the size of the perturbation to zero, we obtain upper symbolic dynamics $\{0^n1^*, 1^n0^* \mid n = 0, 1, \ldots\}$. The inner and lower symbolic dynamics are both $\{0^*, 1^*\}$

In general, it is possible to show that the upper symbolic dynamics is a subset of the outer symbolic dynamics, and that the outer symbolic dynamics can be obtained by removing the restriction that the perturbations f_ϵ must have the same laps as f.

1.2.1 Kneading Theory

The original kneading theory was developed for multimodal maps by Milnor and Thurston [2]. The theory was extended to maps of the interval with "holes" by Rocha and Sousa Ramos [3, 4], which have a similar flavour to discontinuous maps. In this section we give a review of the kneading theory for multimodal maps without discontinuities. Rather than use Milnor and

Thurston's approach using formal power series and a kneading matrix, we will work with itineraries and covering relations, since this is simpler and ties in better with the computation of sofic shifts.

Let f be a multimodal map defined on the interval $I = [a, b]$ with critical points $a < c_1, \ldots, c_{l-1} < b$. Let $I_j = [c_{j-1}, c_j]$ for $j = 1, \ldots, l$, where we take $c_0 = a$ and $c_l = b$. For $j \in \{1, \ldots, l\}$, let ε_j be the sign of f over the interval I_j. For a sequence $\vec{s} \in \{1, \ldots, l\}^{\mathbb{N}}$, denote by $\varepsilon_{\vec{s},j}$ or $\varepsilon_{s_0,s_1,\ldots,s_{j-1}}$ the product $\varepsilon_{s_0}\varepsilon_{s_1}\cdots\varepsilon_{s_{j-1}}$.

We define an ordering on the itineraries of f. Suppose $\vec{s}$ and $\vec{t}$ are sequences, and $j = \min\{i | s_i \neq t_i\}$, and suppose $s_j < t_j$. Then $\vec{s} < \vec{t}$ iff $\epsilon\varepsilon\, s_j < \epsilon\varepsilon\, t_j$, where $\epsilon = \varepsilon_{\vec{s},j} = \varepsilon_{\vec{t},j}$.

We now consider whether there exists an orbit with a given itinerary. Suppose the itineraries of the images of the critical points $c_1, \ldots, c_{l-1}$ are $\vec{k}_j$, that is $\vec{k}_j = \iota(f(c_j))$. Suppose $\vec{t}$ is the itinerary of a point x. Then if $c_i < x < c_{i+1}j$, then either $f(c_i) < f(x) < f(c_{i+1})$ or $f(c_{i+i}) < f(x) < f(c_i)$ depending on the sign of ε_{i+1}. Since $\iota(f(x))$ is $\sigma(\iota(x))$, where $\iota(x)$ is the itinerary of x We deduce that $\vec{t}$ is an itinerary and only if, $\sigma^{n+1}(\vec{t}) \in [\vec{k}_{t_n}, \vec{k}_{t_n+1}]$, where the endpoints of the interval may be reversed. We may also consider the endpoints of the intervals.

We have the following result.

Theorem 1.2.1. *Let $\vec{s}$ be a sequence in $\{1, \ldots, l\}^{\mathbb{N}}$. Then $\vec{s}$ is the itinerary of an orbit $\vec{x}$ of f if $\sigma^{n+1}(\vec{s})$ lies between $\vec{k}_{s_n-1}$ and $\vec{k}_{s_n}$ for all n, where $\vec{k}_j$ is the itinerary of $f(c_j)$.*

In the unimodal case, we have $f(a) = f(b) = a$, and $\iota(a) = (0, 0, \ldots)$. Hence we obtain the classical result that $\vec{s}$ is an itinerary of a point x if, and only if $\sigma^{n+1}(\vec{s}) \leq \vec{k}$ for all n, where $\vec{k}$ is the itinerary of $f(c)$, the image of the critical point.

1.3 Symbolic dynamics in two higher and dimensions

The dynamics of systems in two and higher dimensions shows considerably more complexity than the dynamics of one dimensional systems. Kneading theory is not extendible to maps in two or higher dimensions. In this section we show two effective approaches, the first can be applied only to two dimension while the second also to higher dimensions.

1.3.1 Homoclinic Tangle

The tangle of a map f is the structure of intersections between the stable and unstable manifolds of one or more fixed points. There is already a well developed theory and software for computation of lower approximation of symbolic dynamics from tangles. Here we will provide the definition using the notation of [17], and we refer the reader the same paper for the theory.

Definition 1.3.1 (Tangle)**.** *If f is a diffeomorphism with a saddle fixed point p, then the stable and unstable manifolds of p are smooth curves which we denote W_S and W_U respectively. The pair $W = (W_U; W_S)$ is a homoclinic tangle for f.*

If W_U and W_S intersect at a point other than p, they must intersect infinitely often and have infinite length, hence are impossible to draw, compute or describe combinatorially. Therefore we give the following definition:

Definition 1.3.2 (Trellis)**.** *Let T_U and T_S be sub-intervals of W_U and W_S. If T_U and T_S both contain p and $f(T_U) \supset T_U$ and $f(T_S) \subset T_S$, we call the pair $T = (T_U; T_S)$ a trellis for f.*

A diagram of a simple trellis is shown in Figure 1.1.

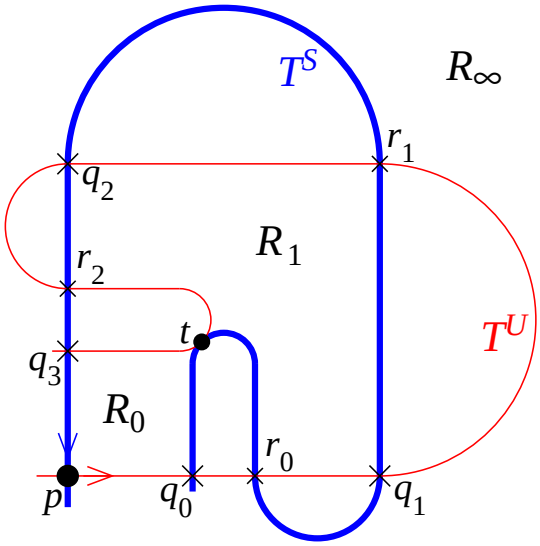


Figure 1.1: Trellis, T_U is drawn in red and T_S in blue, p is the fixed point, q_i and r_i are intersection points, R_i regions, t a tangency point

Here we present algorithms which deals with piecewise affine maps:

The algorithm which compute the symbolic dynamics from the combinatorial information of intersection of the trellis take as input a list of tuple organizes as in table **trellis_intersection**. For instance the corresponding entry for the intersection q_2 in figure 1.1 is $[q_2, r_2, r_2, -, q_3]$.

omputing intersections among the manifolds can be done by computing intersections of the segments of the manifolds. It is natural to order the intersections according the ordering of the segments of a manifolds, and this make straightforward computing for some intersection the next intersection on a manifold. The image of an intersection can be computed checking the image or preimage of the segment where the intersection lies. Finally the orientation of an intersection can be computed from the angle between two manifolds at the intersection as the orientation of a cross product between vectors.

The theory underling the tangle algorithm provides a lower approximation of the symbolic dynamic in term of regions of the tangle by proving that the mapping among some curves inside the regions prove the existence of periodic orbit (see Theorem 10 in [17]). Such mapping among curves can be computed by the topological relations among intersection of the trellis.

We refer the reader to [18] and [17] for a detailed explanation of the algorithm and the underlying theory.

1.3.2 Conley Index

The Conley index is an index defined via homology of a map over a region. It can give relevant information about the invariant set of a map.

We use the following notation throughout the thesis, $\mathrm{inv}_f^+(B) = \{x \in B \mid \forall i \in \mathbb{N}, f^i(x) \in B\}$ in other words denotes the maximal invariant subset of a set B with respect to a map f, $\mathrm{int}(B)$ and $\mathrm{cl}(B)$ denote respectively the interior and the closure of a set B.

Definition 1.3.3 (Isolating neighbourhood)**.** *A compact set N is an isolating neighbourhood for a map f if* $\mathrm{inv}_f(N) \subset \mathrm{int}(N)$.

Definition 1.3.4 (Index pair)**.** *A pair (W, Y) of compact sets with $Y \subset W$ is an index pair for an isolated invariant set S of a function f if:*

- $f(W \setminus Y) \subset W$
- $f(Y) \cap W \subset Y$
- $\mathrm{cl}(W \setminus Y)$ *is an isolating neighbourhood for f*

Definition 1.3.5 (Index map homology)**.** *Let (W, Y) be and index pair for a function f. Let us consider the maps:*

$$f : (W, Y) \longrightarrow (W \cup f(Y), Y \cup f(Y))$$

$$j : (W, Y) \longrightarrow (W \cup f(Y), Y \cup f(Y))$$

where j is the inclusion and by the excision property

$$H_*(j) : H_*(W, Y) \longrightarrow H_*(W \cup f(Y), Y \cup f(Y))$$

is an invertible map and if we pick the same homology generators in the domain and co-domain it is the identity. The index map homology is $\forall i \in \mathbb{N}$:

$$H_*(f_W) = H_*(j)^{-1} \circ H_*(f) : H_*(W, Y) \longrightarrow H_*(W, Y)$$

Proposition 1.3.1 (Decomposition)**.** *Let S be a compact set, a decomposition of S is a collection of pairwise disjoint compact sets $\{S_i\}$ such that $\bigcup_i S_i = S$.*

Proposition 1.3.2 (Wazewski property)**.** *If (W, Y) is an index pair for a map f and $\exists q : H_q(f_W)$ is not nilpotent, then* $\mathrm{inv}(\mathrm{cl}(W \setminus Y))$ *is non-empty.*

This property can be proved by showing that the index map of $(\emptyset, \emptyset)$ is the zero-matrix in any dimension for any map and that this is a property of related invariant set and not of the index pair. It is a very powerful method to prove the existence of invariant sets and - combined with the decomposition of the index pairs presented below - the way we can extract the symbolic dynamics of a map from the Conley index.

Here we give definitions and results from [19].

Definition 1.3.6 (Decomposition of an index pair)**.** *An index pair (W, Y) of an invariant set S is compatible with a decomposition $\{S_i\}$ of S if and only if there exists $\{D_i\}$ a decomposition of* $\mathrm{cl}(W \setminus Y)$ *such that $S_i = S \cap D_i$ for each i.*

Definition 1.3.7 (Decomposition of an index map)**.** *Let (W, Y) be an index pair for a map f compatible with a decomposition $\{S_i\}$ of S and let $\{D_i\}$ be the corresponding decomposition* $\mathrm{cl}(W \setminus Y)$*, then for each $x \in B$ we can define :*

$$r^i : W/Y \longrightarrow W/Y$$

by the following formula:

$$r^i([x]) = \begin{cases} [x] \text{ if } x \in D_i \\ [W] \text{ otherwise} \end{cases}$$

and we define $f^i = f \circ r^i$

Proposition 1.3.3. *Let (W, Y) be an index pair for a map f and an invariant set S compatible with the decomposition $\{S_i\}$ of S. Define:*

$$\Lambda = \{(s_i)_{i=0}^{\infty} : \forall n \in \mathbb{N} \quad H_*(f^{s_n}) \circ H_*(f^{s_{n-1}}) \circ .... \circ H_*(f^{s_0}) \neq 0\}$$

1.4 Computability theory and numerical computation

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 a compact interval $\lfloor x \rceil = [\underline{x}, \overline{x}]$ such that $\underline{x} \leq x \leq \overline{x}$. We let $\mathbb{I}$ denote the set of all rational intervals i.e. $\mathbb{I} = \{[a, b] \mid a, b \in \mathbb{Q}\}$.

In the theory of effective (Turing) computability over the reals, we represent a real number as a sequence of approximations, either as floating points x_n with n binary digits of accuracy, or as a convergent sequence of intervals. In this representation, all arithmetical operations are computable, but comparison tests are only *semidecidable* rather than being decidable. For if real numbers x and y differ, then we can determine $x < y$ or $x > y$ in finite time. If, however, $x = y$, then even if we know both x and y up to n bits, we can still never know that indeed $x = y$. This means that any algorithm which tests for (in)equality will not terminate on some inputs. In practice, we can replace the standard comparison operator on reals with an extended comparison operator, so that $\mathrm{cmp}_\epsilon(x, y)$ may (nondeterministically) return indeterminate value $\uparrow$ if $|x - y| < \epsilon$.

When using interval arithmetic, we can test for equality, but the result can only be true if both intervals are singletons. If $\lfloor x \rceil$, $\lfloor y \rceil$ and $\lfloor z \rceil$ overlap, then we cannot say anything about the relative ordering of x, y and z, it is even possible that all three values are equal. If, however, $\lfloor y \rceil$ overlaps both $\lfloor x \rceil$ and $\lfloor z \rceil$, but $\lfloor x \rceil < \lfloor z \rceil$ (i.e $\overline{x} < \underline{z}$) , then the only possible orderings are $y < x < z$, $x = y < z$, $x < y < z$, $x < y = z$ and $x < z < y$. Further, by increasing the precision, we can eventually compute $\lfloor y \rceil$ sufficiently accurately so that either $\lfloor y \rceil > \lfloor x \rceil$ or $\lfloor y \rceil < \lfloor z \rceil$, reducing the possible relative orderings.

More generally, we say that intervals $\lfloor y_0 \rceil, \cdots, \lfloor y_k \rceil$ form a *chain* if $\lfloor y_{i-1} \rceil$ overlaps $\lfloor y_i \rceil$ for $i = 1, \ldots, k$, but $\lfloor y_0 \rceil < \lfloor y_k \rceil$. In this case, increasing the precision of computation, and hence the accuracy of the interval approximations, will eventually partition the intervals so that (possibly after a reordering) there exists m such that $\lfloor y_i \rceil < \lfloor y_j \rceil$ whenever $i \leq m < j$. We say intervals $\lfloor y_1 \rceil, \ldots, \lfloor y_k \rceil$ form a *cluster* if $\lfloor y_i \rceil$ intersects $\lfloor y_j \rceil$ for all $i, j = 1, \ldots, k$. Given a finite set of points with interval bounds, by increasing

the precision, we can always partition the points into clusters by increasing the precision.

In order to work with a continuous function, we assume that given an interval approximation $\lfloor x \rceil$ to x, we can effectively compute an approximation $\lfloor y \rceil$ to $f(x)$.

Definition 1.4.1 (Interval extension)**.** *Let $f : I^n \to I^n$ be a continuous function. Then an* interval extension *of f is a function $\lfloor f \rceil : \mathbb{I}^n \to \mathbb{I}^n$ such that*

1. *$f(\lfloor x \rceil) \subset \lfloor f \rceil(\lfloor x \rceil)$ for all $\lfloor x \rceil \in \mathbb{I}^n$,*
2. *if $\lfloor x \rceil \subset \lfloor y \rceil$, then $\lfloor f \rceil(\lfloor x \rceil) \subset \lfloor f \rceil(\lfloor y \rceil)$, and*
3. *if $\bigcap_{i=1}^{\infty} \lfloor x_n \rceil = \{x\}$, then $\bigcap_{i=1}^{\infty} \lfloor f \rceil(\lfloor x_n \rceil) = \{f(x)\}$.*

If f is n-times continuously-differentiable, we also assume that an interval extension is available for derivatives $f^{(i)}(x)$ for $i = 1, \ldots, n$.

Definition 1.4.2 (Enumeration of open sets)**.** *An enumeration of open set $\mathcal{U}$ is a list of boxes $I_i \in \mathbb{I}^n$ such that $\mathcal{U} = \bigcup_{i=0}^{\infty} I_i$*

Definition 1.4.3 (Representation of piecewise-continuous functions)**.** *Let f be a piecewise-continuous function with pieces $\mathcal{P} = \{P_1, \ldots, P_k\}$. An interval extension of a piecewise-continuous function consists of*

1. *An enumeration of each partition element $P_i \in \mathcal{P}$, by boxes and*
2. *An interval extension of each piece f_i over the closure of P_i.*

If f is piecewise-continuous, then we can compute interval approximations $\lfloor d \rceil$ to the discontinuity points d from the given data. Note that from the data given, we can compute approximations to the discontinuity points, since a point d lies on the boundary of two partition elements if, and only if, for all intervals $J \ni d$, there exist $\bar{I}_{i_1} \subset P_{i_1}$ and $\bar{I}_{i_2} \subset P_{i_2}$ such that $J \cap I_{i_1} \neq \emptyset$ and $J \cap I_{i_2} \neq \emptyset$.

In practice, we combine numerical and symbolic approaches. For example, in the unimodal map $x \mapsto \mu x(1-x)$, we know that $f(0) = f(1) = 0$ and that $f'(1/2) = 0$. Letting $a = 0$, $b = 1$ and $c = 1/2$, we know exactly that $b \mapsto a \mapsto a$ and that c is a critical point.

We may sometimes want to *define* boundary points of the partition $\mathcal{Q}$ as fixed, periodic or critical points. For example, if we want to use the period-two orbit (q_1, q_2) of the unimodal map at $\mu = 3.92$ with negative derivative as partition boundary elements, we can define the position of q numerically by $\lfloor q_1 \rceil = [0.356, 0.357]$ and $\lfloor q_2 \rceil = [0.898, 0.899]$ and symbolic images $q_1 \mapsto q_2 \mapsto q_1$.

Chapter 2

Computation of symbolic dynamics in one dimension

In this chapter we show algorithms for the computation of symbolic dynamics of one dimensional dynamical systems. One-dimensional discrete-time dynamical systems are widely used to describe physical and engineering systems in a simplified way, and often arise as return maps of continuous or hybrid systems in dimension two. The dynamics of such systems can be complicated because even in one dimension the system can exhibit chaotic behaviour.

We present some algorithms for computing the symbolic dynamics of one-dimensional piecewise-continuous maps. Each of the algorithms constructs sofic shifts on the symbol space which over- and under-approximate the exact shift space of the maps. Further, the algorithms are designed to use interval methods so can be rigorously implemented on a digital computer. Under certain nondegeneracy conditions the shift maps obtained converge to the exact symbolic dynamics as the precision is increased. The approximation of the symbolic dynamics allows us also computing arbitrary accurate approximation of the topological entropy of a map.

There has been considerable work on symbolic dynamics of interval maps. A large class of results rely on the elementary fact that if there is a sequence of intervals $(I_n)_{n\in\mathbb{N}}$ such that $f(I_n) \supset I_{n+1}$ for all $n \in \mathbb{N}$, then there is a point x such that $f^n(x) \in I_n$ for all $n \in \mathbb{N}$. The celebrated theorem of Sharkovskii [20] on coexistence of periodic orbits with different periods can be proved in this way, and stronger results can be obtained by considering relative orderings of points forming finite invariant sets.

One of the most important contribution to one-dimensional dynamics is the kneading theory of Milnor and Thurston [2] for continuous multimodal maps. The theory gives necessary and sufficient conditions for an itinerary to

be realised by an orbit in terms of the itineraries of the orbits of the critical points. The theory is cleanest and received most attention for unimodal maps, since the itinerary of the critical point is sufficient to determine the symbolic dynamics. A more complete understanding of kneading theory for bimodal and trimodal maps was developed by Block and Keesling [21] and Lampreia and Sousa Ramos and co-workers [22]. Maps with discontinuities or holes were considered by Rocha and Sousa Ramos [3]. The theory has been extended from interval maps to tree maps; see Alves and Sousa Ramos [23] for details.

Another motivation is related to a problem posed by Milnor on computability of topological entropy [24]. The topological entropy of a continous interval map is always lower-semicontinuous [14], and can be shown to be upper-semicontinuous in C^2 if the critical points are non-degenerate [15, 16]. The constructions used in the proofs demonstrate that convergent approximations to the entropy can be effectively computed. In this chapter we consider computability of symbolic dynamics, and convergence of the entropy of the generated shift space.

The main contributions of this chapter are related to the practical implementation of the rigorous computation of symbolic dynamics based on the existing theory. Rather than start from a given periodic or invariant skeleton described combinatorially, we consider the computation of partition refinements, and compare the performance of different methods. We use rigorous interval arithmetics for the numeric computations, and show how to handle the difficulties which arise when extracting combinatorial data, including the use of references to store image and preimages, and how to resolve overlapping intervals. We also extend the kneading theory to be able to extract information about the dynamics from finite orbits, where the standard theory only considers infinite itineraries. We also consider the computation of symbolic dynamics relative to an arbitrary partition as opposed to the natural partition given by the critical points. A formal definition of the "effectively computable" symbolic dynamics is introduced, and an analysis of the convergence of the algorithms with respect to the theoretical restrictions on what is possible using numerical (as opposed to algebraic or symbolic) methods is also given.

A further motivation for this work was to obtain insight into computation of symbolic dynamics which could be applied to higher-dimensional systems. For this reason, we do not only focus on methods based on kneading theory, which appear to be most powerful in dimension one, but also on methods based on backward iteration and covering relations, which are also applicable in higher dimensions.

The main mathematical techniques used in this article include interval

analysis, symbolic dynamics, one-dimensional dynamical systems theory and kneading theory. Good references to these topics are include the books [25, 26, 27, 28, 29, 30]. Important standard references on kneading theory include [2, 31, 32, 33, 34, 35, 23]; other references include [22, 3, 36, 4, 37, 38].

The chapter is organised as follows. In Section 2.1 we present algorithms for computing combinatorial mapping data about the map, and extracting the symbolic dynamics using covering relations. In Section 2.2 we present methods for improving the symbolic dynamics obtained from the mapping data by considering kneading theory. In Section 2.3 we demonstrate the effectiveness of the method by applying it on several examples of continuous and discontinuous maps. In Section 2.4 we analyze two case studies of return maps of hybrid systems: an affine switching hybrid automaton and the singular limit of the Van Der Pol oscillator. Finally, in Section 2.5 we give some conclusions and suggestions for further research.

2.1 Computing Symbolic Dynamics using Covering Relations

We will consider the computation of symbolic dynamics for a $\mathcal{P}$-continuous map f with respect to the partition $\mathcal{Q}$. We let B be the boundary points of $\mathcal{Q}$, C the critical points of f and D the discontinuity points of f.

Our basic strategy for computing symbolic dynamics is as follows

Algorithm Scheme 1.

1. *Compute the topological partition $\mathcal{L}$ such that f is monotone and continuous on each piece of $\mathcal{L}$*

2. *Refine the partition $\mathcal{L} \vee \mathcal{Q}$ to obtain a partition $\mathcal{R}$.*

3. *On the refined partition $\mathcal{R}$, compute the symbolic dynamics by considering covering relations $f(R) \supset R'$, $f(R) \subset R'$ and $f(R) \cap R' \neq \emptyset$.*

There are a number of difficulties which arise when actually trying to implement this strategy. The first is that the discontinuity and critical points may be difficult to locate exactly, and we need to use approximations. This can be the case even if we are given the map in explicit form. A second difficulty is that there are many ways of refining the partition, and we need to find ways which give good approximations to the dynamics with as little computational effort possible. A third difficulty is that the boundary points of the refined partition may be difficult to compute exactly, and even the

relative ordering of the boundary points on the interval may be difficult to compute. Finally, the basic method of extracting symbolic dynamics using covering relations can be quite crude, and it is often useful to improve the results by using kneading theory.

2.1.1 Mapping data

In order to refine the partition, we need to add extra points to the partition and determine their images and/or preimages. In general, given a point p, we will not be able to compute its image $q = f(p)$ arbitrarily accurately, so we generally need to represent points by intervals $q \in \lfloor q \rceil$. However, since $\lfloor q \rceil$ is an interval containing $f(p)$, we do know that $f(p) \in \lfloor q \rceil$. Similarly, computing the image $r = f(q) = f^2(p)$, we obtain the interval $\lfloor r \rceil \supset f(\lfloor q \rceil)$, though we know that $\lfloor r \rceil$ is an interval containing the exact image r of the point $q \in \lfloor q \rceil$.

The data we can compute about our function will therefore consist of a finite set of points, for which we know the image map exactly, but know the value of the points only approximately. We formalise this notion as a *mapping dataset* for the function.

Definition 2.1.1. *A* mapping dataset $\mathscr{F}$ *consists of a finite set* Y *of point labels, together with*

1. *an* image *function* $\rightsquigarrow : Y' \to Y$*, and*
2. *a* value *function* $\lfloor \cdot \rceil : Y \to \mathbb{I}$*.*

We alternatively write $f(x) = y$ *if* $x \rightsquigarrow y$*. From the interval values we can deduce*

3. *a partial ordering* $\leq$ *on* Y*, defined by* $x \leq y \iff \lfloor x \rceil \leq \lfloor y \rceil$ *(by which we mean* $\overline{x} \leq \underline{y}$*).*

Further, there is a finite totally-ordered subset Z *of* Y *such that if* $z_i \leq x \leq y \leq z_{i+1}$*, then either* $f(z_i) \leq f(x) \leq f(y) \leq f(z_{i+1})$ *or* $f(z_{i+1}) \leq f(y) \leq f(x) \leq f(z_i)$*. The points of* Z *correspond to the critical points* c_i *and discontinuity points* $d_i^{\pm}$*.*

The data type representing a point y therefore has two fields, a `value` field which is (an approximation to) the numerical value of y, and a `image` field which is a reference or pointer to the object representing $f(y)$. If $f(y)$ has not been computed, then the image field is `null`. If d is a discontinuity point, then we store two image points, namely the image of d under both branches of $\bar{f}$ at d, which we denote $f(d^-)$ and $f(d^+)$.

In the case that we can compute images and/or preimages exactly, then the value field of each point is an exact real number, the resulting mapping dataset is totally ordered, and the complement of the set Y forms a topological partition. Otherwise, we cannot extract a topological partition from the mapping data set, but instead obtain a topological partition for each linear order compatible with the mapping dataset. We can assume that the ordering is *clustered*, which means that if $w < y$, then for all x, either $w < x$ or $x < y$. If these interval values are disjoint, so $\lfloor x \rceil \gtrless \lfloor y \rceil$ if $x \neq y$, then we can recover a topological partition.

To compute the critical points numerically, 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 chapter we assume for simplicity that these cases do not arise.

2.1.2 Partition refinement strategies

In this section, we give the basic refinement strategies used.

Definition 2.1.2. *Let $\mathcal{Q}$ be the initial partition and $\mathcal{L}$ the partition into monotone branches. The forward-refinement partition at step n is the partition whose end points are*

$$Y = \partial\mathcal{Q} \cup \bigcup_{z \in \partial\mathcal{L}} \{f^j(z_i) \mid j = 0, \ldots, n\}$$

.

Definition 2.1.3. *Let $\mathcal{Q}$ be the initial partition and let $\mathcal{P}$ be the partition into continuous branches. The backward-refinement partition at step n is the partition whose end points are*

$$Y = \bigcup_{z \in \partial\mathcal{P} \cup \partial\mathcal{Q}} \{f^{-j}(z) \mid j = 0, \ldots, n\}$$

The forward refinement of a partition $\mathcal{R}$ can be easily computed, since we need simply to compute the images of all boundary points of $\mathcal{R}$. To compute the backward refinement of $\mathcal{R}$ we need to compute preimages of $r \in \partial\mathcal{R}$. Since f is strictly monotone on each element of $\mathcal{L}$, so we can use a bisection strategy to locate the preimage of y in each $L \in \mathcal{L}$, which must be unique (if it exists).

The advantage of the forward refinement strategy are that better results can usually be obtained with fewer partition points, since the growth of the number of endpoints is linear in the number of steps. The advantage of backward refinement is that each partition element of an n-step backward refinement of $\mathcal{Q}$ determines the first n elements of a $\mathcal{Q}$-itinerary. Define sets $Q_{i_0,i_1,\ldots,i_{n-1}}$ recursively by $Q_{i_0} \in \mathcal{Q}$ and $Q_{i_0,i_1,\ldots,i_{n-1}} = Q_{i_0} \cap f^{-1}(Q_{i_1,\ldots,i_{n-1}})$. Then it is easy to see that $Q_{i_0,i_1,\ldots,i_{n-1}} = \{\vec{y} \mid f^k(\vec{y}) \in Q_{i_k} \text{ for } k = 0,\ldots,n-1\}$. Further, the endpoints of $Q_{i_0,i_1,\ldots,i_{n-1},i_n}$ are points in $f^{-n}(B \cup D)$.

The following partition refinement strategy attempts to combine the efficiency of forward refinement with the better theoretical properties of backward refinement.

Definition 2.1.4. *Let $\mathcal{Q}$ be the initial partition, let $\mathcal{L}$ be the partition into monotone branches, and $\mathcal{R}_0 = \mathcal{Q} \vee \mathcal{L}$. The forward-looking backward-refinement partition at step n is the partition $\mathcal{R}_n$ whose end points are*

$$\{p \in f^{-1}(\partial\mathcal{R}_{n-1}) \mid \exists q \in \partial\mathcal{R}_{n-1} : f^{-1}(\partial\mathcal{R}_{n-1}) \cap \operatorname{conv}\{p, f(q)\} = \{p\}\}.$$

In other words among all preimages up to order m of endpoints of the initial partition this refinement considers only the ones which are the closest to at least one endpoint of the initial partition, in this way we minimize the size of elements of the partition which do not cover exactly other elements of the partition, and consequently the gap between lower and upper symbolic dynamics.

The above strategy gives a growth of end points which is linear with the number of steps and both lower and upper approximation converge monotonically.

Despite this great improvement also an algorithm using the latest strategy has an exponential running time in the number of steps because it still requires to compute the preimages which grow exponentially in the number of steps. The only algorithm which has linear running time and converges to the symbolic dynamics is the one using the kneading theory as will be explained in Section 2.2.

2.1.3 Eventually periodic discontinuity and critical points

Suppose that the backward images of the partition boundary points are all disjoint from each other, and disjoint from the critical and discontinuity points. Then the partitions computed using backward refinement are all disjoint, and so can be ordered given enough information about the function. However, it may be the case that certain preimages intersect, or are equal to critical or discontinuity points, and in this case they cannot be ordered.

Similarly, if the forward images of critical or discontinuity points are not all disjoint, or touch partition boundary points, then we lose the ability to order them.

In some cases, we may be able to use algebraic information about the function to prove equality. For example, if we know that a partition boundary is also a fixed point, then we can set the image explicitly, rather than use approximations. The symbolic dynamics is discontinuous at a point for which $f^n(c) = q$, since we introduce the possibility of a new transition after n steps.

During our interval computations, we will find cases for which (say) $f^n(c) \approx_\epsilon q$ for a partition boundary point q. If this is the case, we do not know whether $f^n(c) < q$, $f^n(c) > q$ or $f^n(c) = q$, though we do know that in either of the former two possibilities, increasing the precision will eventually give the correct answer. In order to study this situation, we continue iterating the interval. Note that if $\lfloor x \rceil$ and $\lfloor y \rceil$ overlap, then so do $\lfloor f(x) \rceil$ and $\lfloor f(y) \rceil$, and therefore $\lfloor f \rceil(\lfloor x \rceil)$ and $\lfloor f \rceil(\lfloor y \rceil)$.

Upon further computation, we may find a case in which an interval $\lfloor y \rceil$ intersects two disjoint intervals $\lfloor x \rceil$ and $\lfloor z \rceil$. In this case, we increase the precision until $\lfloor y \rceil$ is disjoint from either $\lfloor x \rceil$ or $\lfloor z \rceil$.

To resolve the overlap, we consider the dynamics for the three cases $x < y$, $x = y$ and $x > y$. In the case of equality, we can combine the exact orbits. In the case of inequality, suppose that y is not a critical or discontinuity point, then we know the monotone branch that y lies in, and hence the relative ordering of $f^i(x)$ and $f^i(y)$ for as long as these have been computed.

If there are multiple overlaps, then we need to consider all possible orderings. In principle, this can be very expensive, but in practice the condition that the algorithm increases the precision if we find a chain of overlaps means that the number of combinations remains small (at most l overlaps, where l is the number of laps).

2.1.4 Covering relations

The simplest way to extract symbolic dynamics is directly via covering relations. Given a partition $\mathcal{R}$ and covering relations between its elements we want to construct sofic shifts under- and over-approximating the symbolic dynamics.

Definition 2.1.5. *Let $f : X \to X$ be a $\mathcal{P}$-continuous map and $\mathcal{Q}$ a partition of X. Let $\mathcal{R}$ be a partition of X refining $\mathcal{Q}$ and $\mathcal{P}$. Let $\mathcal{R}$ be the partition $\mathcal{L}$ into monotone continuous pieces of f.*

Define covering relations *relations $R \longrightarrow R'$ and $R- \!\!\rightarrow R'$ on $\mathcal{R} \times \mathcal{R}$ by $R \longrightarrow R'$ if $f(R) \supset R'$ and $R- \!\!\rightarrow R'$ if $f(R) \cap R' \neq \emptyset$.*

We define the attracting subset $\mathcal{A} \subset \mathcal{R}$ *to be the maximal subset of* $\mathcal{R}$ *such that for all* $A \in \mathcal{A}$*, there exists* $A' \in \mathcal{A}$ *such that* $f(A) \subset A'$*. Define relation* $\hookrightarrow$ *by* $A \hookrightarrow A'$ *if* $f(A) \subset A'$*.*

We define a lower symbolic dynamics $\underline{\Sigma}(f, \mathcal{Q}, \mathcal{R})$ *to be determined by the transitions* $\longrightarrow$ *and* $\hookrightarrow$*, and* upper symbolic dynamics $\overline{\Sigma}(f, \mathcal{Q}, \mathcal{R})$ *by the transitions* $- \!\!\rightarrow$*.*

We can represent the symbolic dynamics visually by a graph with vertices $\mathcal{R}$, each vertex R being labelled with the element $Q \in \mathcal{Q}$ with $R \subset Q$, and edges $- \!\!\rightarrow$, $\longrightarrow$ or $\hookrightarrow$ determined by the transition relations.

It is clear that the lower symbolic dynamics is an under-approximation to the inner symbolic dynamics, $\underline{\Sigma}(f, \mathcal{Q}, \mathcal{R}) \subset \underline{\Sigma}(f, \mathcal{Q})$, and the upper symbolic dynamics is an over-approximation to the inner symbolic dynamics, $\overline{\Sigma}(f, \mathcal{Q}, \mathcal{R}) \supset \underline{\Sigma}(f, \mathcal{Q})$

However, we may not have $\overline{\Sigma}(f, \mathcal{Q}, \mathcal{R}) \supset \overline{\Sigma}(f, \mathcal{Q})$ due to problems with the endpoints of partition elements. For example, if c is a critical point and $f(c) \in \{r\} = R_i \cap R_{i+1}$, then we should allow broken arrows from each interval containing c to R_i and R_{i+1} to obtain an over-approximation to the upper symbolic dynamics. If exact mapping data is available, we can augment the graph with vertices $z \in B \cup C \cup D^{\pm}$ with the appropriate labelling and arrows.

In order to determine the transitions $R \longrightarrow R'$ and $R - \!\!\rightarrow R'$, 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 of $\mathcal{R}$ do not overlap, we can determine the relative ordering by taking a sufficiently high precision. However, if the boundary points do overlap, then there is more than one topological partition consistent with the mapping data.

In order to obtain under- and over-approximations to the symbolic dynamics, we consider *all* possible topological partitions consistent with the mapping data. Note that knowledge of the monotone branches of f may help reduce the number of possibilities. For example, if the numerical values of r_i and r_j overlap, then we need to consider partitions with $r_i < r_j$, $r_i = r_j$ and $r_i > r_j$. However, if r_i, r_j lie in an increasing branch, then $f(r_i) \gtrless f(r_j)$ according to $r_i \gtrless r_j$. We can also dispense with the case $r_i = r_j$, since this will always provide better approximations than the cases $r_i \neq r_j$. We therefore only need to consider two cases if $\lfloor r_i \rceil$ and $\lfloor r_j \rceil$ overlap.

We therefore have the following algorithm for computing over- and under-approximations to the symbolic dynamics from a mapping dataset $\mathscr{F}$.

Algorithm Scheme 2.

1. *Compute the set of all partitions* $\mathcal{R}$ *consistent with the mapping dataset* $\mathscr{F}$ *and for which there is no equality in uncertain comparisons.*

2. *On each refined partition $\mathcal{R}$, compute lower and upper symbolic dynamics $\underline{\Sigma}(f, \mathcal{Q}, \mathcal{R})$ and $\overline{\Sigma}(f, \mathcal{Q}, \mathcal{R})$.*

3. *Let the lower symbolic dynamics $\Lambda = \underline{\Sigma}(f, \mathcal{Q}, \mathscr{F})$ be the intersection of all $\underline{\Sigma}(f, \mathcal{Q}, \mathcal{R})$ with $\mathcal{R}$ consistent with $\mathscr{F}$, and the upper symbolic dynamics $\Upsilon = \overline{\Sigma}(f, \mathcal{Q}, \mathscr{F})$ be the union of all $\overline{\Sigma}(f, \mathcal{Q}, \mathcal{R})$ with $\mathcal{R}$ consistent with $\mathscr{F}$*

It is clear that Λ is a subshift of $\underline{\Sigma}(f, \mathcal{Q})$, and Υ is a supershift of $\underline{\Sigma}(f, \mathcal{Q})$. In Section 2.1.6 we shall see that the symbolic dynamics converges to the upper and lower symbolic dynamics of f, and that this convergence is monotone using backward refinement.

2.1.5 Combining intervals

The dynamics obtained using covering relations can be improved by *combining intervals*. The motivation for this trick is as follows. Suppose I_1 and I_2 are adjacent intervals, and $I_1 \cap I_2$ is a point x with no computed image under f. Then we can make a new formal symbol I_{12} which covers any interval covered by $I_1 \cup I_2$. Whenever an interval J covers I_1 and I_2, we replace the arrows $J \longrightarrow I_1$ and $J \longrightarrow I_2$ with $J \longrightarrow I_{12}$. Whenever an interval K is covered by $I_1 \cup I_2$, we add an arrow $I_{12} \to K$. The resulting symbolic dynamics is a supershift of the previous dynamics, since any sequence passing $J \longrightarrow I_i \to K$ for $i = 1, 2$ either remains as is, or is changed to $J \longrightarrow I_{12} \longrightarrow K$.

We formalise the conditions on the combined intervals as follows.

Definition 2.1.6. *Let $\mathcal{I} \supset \mathcal{R}$ be a collection of intervals such that every $I \in \mathcal{I}$ is contained in some $L \in \mathcal{L}$ and has endpoints in $\partial\mathcal{R}$. Define relations $\longrightarrow$, and $-\!\!\to$ satisfying*

1. *If $f(I) \supset J$, then there exists $K \supset J$ such that $I \longrightarrow K$.*
2. *If $f(I) \cap J \neq \emptyset$, then there exists $K \subset J$ such that $I -\!\!\to K$.*
3. *If $I \longrightarrow J$, then $f(I) \supset J$.*
4. *If $I -\!\!\to J$ and $K \subset J$, then $f(I) \cap K \neq \emptyset$.*
5. *If $I -\!\!\to J$ and $I -\!\!\to K$ or $I \longrightarrow J$ and $I \longrightarrow K$, then $J \cap K = \emptyset$.*

Condition 5 ensures that every orbit is assigned a unique itinerary.

The next result shows that the symbolic dynamics computed using combined intervals improves that computed directly.

Theorem 2.1.1. *Let Λ and Υ be the lower and upper shifts defined by the partition $\mathcal{R}$, and Λ' and Υ' defined by the relations $\longrightarrow$ and $-\!\!\rightarrow$ on $\mathcal{I}$. Then $\Lambda \subset \Lambda' \subset \underline{\Sigma}(f, \mathcal{Q}) \subset \Upsilon' \subset \Upsilon$.*

Proof. Any $\mathcal{Q}$-itinerary $\vec{s}$ in Λ is also in Λ': Take a sequence $\vec{R}$ with $R_i \subset Q_i$ and $f(R_i) \supset R_{i+1}$. Define $I_0 = R_0$, and I_{i+1} recursively such that $R_{i+1} \subset I_{i+1}$ and $I_i \longrightarrow I_{i+1}$. The resulting sequence $\vec{I}$ shows that $\vec{s}$ is in Λ'.

Any $\mathcal{Q}$-itinerary in Λ' also is an itinerary of f: If $I_i \longrightarrow I_{i+1}$ in the shift Λ', then $f(I_i) \supset I_{i+1}$.

Any $\mathcal{Q}$-itinerary of f is also in Υ': If $\vec{x}$ is an orbit with itinerary $\vec{s}$ with $x_i \in R_i$ for $R_i \in \mathcal{R}$, we can choose $I_0 \supset R_0$ and then recursively find I_{i+1} such that $I_{i+1} \supset R_{i+1}$ and $I_i -\!\!\rightarrow I_{i+1}$ since $f(I_i) \cap R_{i+1} \neq \emptyset$.

Any $\mathcal{Q}$-itinerary in Υ' is also in Υ: Suppose $I_i -\!\!\rightarrow I_{i+1}$ for all $i \in \mathbb{N}$. Suppose $R_j \in \mathcal{R}$ with $R_j \subset I_j$. Then $f(I_{j-1}) \cap R_j \neq \emptyset$, so there exists $R_{j-1} \subset I_{j-1}$ with $f(R_{j-1}) \cap R_j \neq \emptyset$. □

2.1.6 Convergence to the symbolic dynamics

We now describe how the sofic shifts computed by Algorithm 1 approximate the symbolic dynamics of f, and give sufficient conditions under which the approximations converge.

We first consider monotonicity of the computed shift dynamics.

Theorem 2.1.2. *Suppose f is a $\mathcal{P}$-monotone-continuous map, $\mathcal{Q}$ is a topological partition, that $\mathcal{R}_1$ is a refinement of $\mathcal{P} \vee \mathcal{Q}$ and $\mathcal{R}_2$ a refinement of $\mathcal{R}_1$. Then the over-approximation to the $\mathcal{Q}$-itineraries of f induced by $\mathcal{R}_2$ is a subset of the set of $\mathcal{Q}$-itineraries induced by $\mathcal{R}_1$.*

Proof. Suppose $I, J \in \mathcal{R}_1$, $K, L \in \mathcal{R}_2$, $I \subset K$ and $J \subset L$. Then if $f(K) \cap L \neq \emptyset$ we have also $f(I) \cap J \neq \emptyset$. Hence if there is a broken arrow from K to L, there is also a broken arrow from I to J. The result follows. □

When using forward refinement, it is not necessarily true that the topological entropy of the finite-type shift generated by $\mathcal{R}_2$ is lower than that of the finite-type shift generated by $\mathcal{R}_1$.

Theorem 2.1.3. *Suppose f is a $\mathcal{P}$-monotone-continuous map, $\mathcal{Q}$ is a topological partition, that $\mathcal{R}_1$ is a refinement of $\mathcal{P} \vee \mathcal{Q}$, and $\mathcal{R}_2$ a partial backward refinement of $\mathcal{R}_1$. Then the under-approximation to the $\mathcal{Q}$-itineraries of f induced by $\mathcal{R}_2$ is a superset of the set of $\mathcal{Q}$-itineraries induced by $\mathcal{R}_1$.*

Proof. Suppose $I_k, J_k \in \mathcal{R}_1$ and $f(I_k) \supset J_k$. If $J_k = [c, d]$, then $a, b \in I_k$ such that $f(a, c) \subset (c, d)$ and either $f(a) = c$ and $f(b) = d$, or $f(a) = d$

and $f(b) = c$. For take a', b' such that $f(a') = c$ and $f(b') = d$ and assume $a' < b'$ (the case $a' > b'$ is similar). Let $a = \sup\{x \in [a', b'] \mid f(a) = c\}$ and $b = \inf\{x \in [a, b'] \mid f(b) = d\}$. Then clearly $f(x) \notin \{c, d\}$ for all $x \in (a, b)$. Since $\mathcal{R}_2$ is a backward refinement of $\mathcal{R}_1$, there exists an interval $I_{k+1} \in f^{-(k+1)}(\mathcal{P} \vee \mathcal{Q})$ such that $[a, b] \subset I_{k+1}$ and so $f(I_{k+1}) \supset J_k$. Hence $\exists I_{k+1} \in I_k,\ \forall J_{k+1} \subset J_k,\ f(I_{k+1}) \supset J_{k+1}$.

Now suppose $\vec{s}$ is a $\mathcal{R}_1$-itinerary, so $f(I_{k,n_i}) \supset I_{k,n_{i+1}}$ for all i for intervals $I_{k,n_i} \subset I_{s_i}$. By the above, there exist intervals I_{k+1,m_i} of $\mathcal{R}_2$ such that $I_{k+1,m_i} \subset I_{k,n_i}$ and $f(I_{k+1,m_i}) \supset I_{k,n_i}$, and immediately $f(I_{k+1,m_i}) \supset I_{k+1,m_{i+1}}$. Hence $\vec{s}$ is a $\mathcal{R}_2$-itinerary. □

We now consider convergence of the approximation using backwards refinement. Unfortunately, it is not the case that the lower-approximation to the symbolic dynamics always converges in the Hausdorff metric.

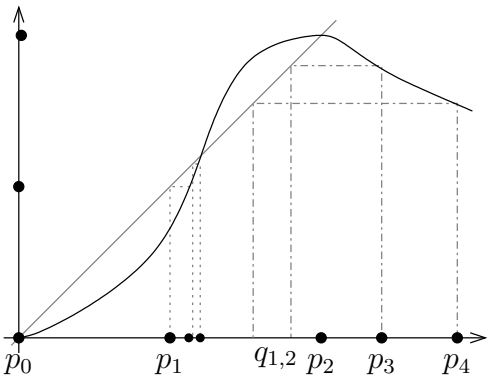


Figure 2.1: A map for which the approximations to the lower symbolic dynamics computed using backwards refinement do not converge. For any stage of backward refinement, the interval $I_3 = (p_3, p_4)$ has no outgoing solid edges.

Example 2.1.1. *Let f be a function which is monotone-continuous on the partition $\mathcal{P}$ with boundary elements $p_0, \ldots, p_4$ such that $f(p_0) = p_0$, $p_1 < f(p_1) < f(p_2) = p_2$, and $f(p_3) = q_2$, $f(p_4) = q_1$ with $f(p_1) < f(q_1) < q_1 < q_2 < f(q_2) < p_2$, as shown in Figure 2.1. Then $f^n([p_3, p_4]) = [f^{n-1}(q_1), f^{n-1}(q_2)] \subset (f^m(p_1), f^m(p_2))$ for all m, n, so there are no preimages of $\mathcal{P}$ in (p_3, p_4). Let J_n be the interval $(f^{-n}(p_1), p_2)$, $I_n = (f^{1-n}(p_1), f^{-n}(p_1))$ and $L = (p_4, p_5)$ Then $f(L) \subset J_n$ but $f(L) \not\supset J_n$ for all n, and $f(J_n) \supset I_n \cup J_n$. Hence there is no solid outgoing arrow from L, and L is not part of an attractor bounded by points of $f^{-n}(\partial\mathcal{P})$, so the itineraries starting in L are not captured by any lower approximation $\underline{\Sigma}(f, \mathcal{P}, \mathcal{R}_n)$*

Theorem 2.1.4. *Suppose $f : X \longrightarrow X$ is a $\mathcal{P}$-monotone-continuous map. Then the over-approximations to the $\mathcal{Q}$-symbolic dynamics computed by Algorithm 1 using backwards refinement converge to the upper symbolic dynamics of f as the precision and maximum number of steps increase.*

Proof. Let $\mathcal{R}_0$ be the partition into monotone-continuous branches refining $\mathcal{P} \vee \mathcal{Q}$. Let $\mathcal{R}_n$ be the refinement of $\mathcal{R}_0$ computed with n steps of the backward refinement strategy. Notice that if R is a subinterval of $\mathcal{R}_n$ whose boundary points are not in $\partial\mathcal{R}_0$, then the image of R exactly covers subintervals of $\mathcal{R}_n$.

To prove that the upper approximations converge to $\overline{\Sigma}(f, \mathcal{Q})$, suppose that $(s_0, s_1, \ldots, s_m)$ is a word which does not occur in $\overline{\Sigma}(f, \mathcal{Q})$. Then $\{x \mid f^k(x) \in Q_{s_k} \text{ for } k = 0, \ldots, m\}$ is empty. However, this set consists of unions of subintervals with endpoints in $f^{-n}(\partial\mathcal{R}_0)$. Hence by checking the itineraries of these subintervals, we can tell which points are absent.

If we do not have the ability to compute exact preimages, but $f^n(p_i^{\pm}) \neq p_j$ for all j, then for a fixed number of backward steps, by taking a high enough precision, the mapping dataset will yield a total order on the points, and we will be in the same situation as if we have exact information. There may be difficulties if $f^n(p_i^{\pm}) = p_j$ for some $p_i, p_j \in \partial\mathcal{P}$. For then the interval approximation $\lfloor x \rceil$ for some $x \in f^{-n}(p_j)$ will overlap $\lfloor p_i \rceil$. Since f is potentially discontinuous at p_i, we can not determine whether $f^{-n}(p_j)$ really has an element x near $d_i = p_i^{\pm}$. However, assuming the point x exists, we know that a neighbourhood of x intersects both intervals adjoining p_j, whereas if there were no preimage, the interval containing d_i would only intersect one interval adjoining p_j. Hence to obtain an over-approximation to the symbolic dynamics, we should assume that such a point x exists and is not equal to p_i.

Convergence of the over-approximations follows since if $x = p_i$, then all backward preimages of p_j intersect p_i. In this case, in the limit, the extra orbit is one of the pair with itinerary $\vec{k}|_{n-1}(d) \cdot \text{itin}(p_j^{\pm})$, where $\vec{k}|_{n-1}(d)$ is the word consisting of the first $n-1$ elements of the kneading sequence of d, and we itin$(p_j^{\pm})$ are the two possible itineraries of p_j. □

The proof of convergence is similar in nature to proofs of continuity of the topological entropy [14, 39]. See [30], Theorem 16.3.1 for details. However, in our case, the lower and upper limits may be slightly different.

2.2 Computing Symbolic Dynamics using Kneading Theory

We now consider how to use the kneading theory to improve the symbolic dynamics. The basic kneading theory framework is first extended in a straightforward way to consider arbitrary regions for the symbols and maps with discontinuity points. We give some intuition on modifying kneading sequences to reduce the symbolic dynamics forced, and then present an algorithm to extend the partial kneading information obtained from the mapping dataset to full kneading information which is either forced by, or forces, the dynamics of the original map. Finally, we prove that this algorithm really does yield over- and under-approximations to the symbolic dynamics.

2.2.1 Kneading theory for discontinuous maps

Suppose f is $\mathcal{P}$-continuous, $\mathcal{L}$-monotone-continuous, and we wish to compute the symbolic dynamics relative to a partition $\mathcal{Q}$. For simplicity, we assume that every point $q \in \partial\mathcal{Q}$ is either known to be disjoint from $\partial\mathcal{L}$, or is known to coincide exactly with a point $l \in \partial\mathcal{L}$. We let $\mathcal{B}$ be the partition $\mathcal{L} \vee \mathcal{Q}$. We therefore consider the case that f is $\mathcal{B}$-monotone-continuous, and we are looking for symbolic dynamics relative to $\mathcal{B}$. When considering perturbations of f, we will allow perturbations which are $\mathcal{B}$-continuous, and not restrict to those which are merely $\mathcal{P}$-continuous; this means that we allow perturbations which destroy continuity at the critical points and points of $\partial\mathcal{Q}$.

Let b_i be the i^{th} point of $\partial\mathcal{B}$ ordered along X, so that $B_i = (b_i^+, b_{i+1}^-)$. We let D be the set of boundaries of monotone-continuous branches, so $D = \{b_0^+, b_1^-, b_1^+, \ldots, b_{l-1}^+, b_l^-\}$.

We define kneading invariants $\vec{k}(d) = \text{itin}(f(d))$ for $d \in D$. If $f^n(d) \in \partial\mathcal{B}$ for some n, then $f(d)$ has two (or more) possible itineraries, and we take $\vec{k}(d) := \lim_{x \to d} \text{itin}(f(x))$ where the limit is taken through points in the same monotone-continuous branch as $d = b^{\pm}$. By the standard kneading theory, $\vec{s}$ is an itinerary of some point x under f if $\sigma^{n+1}(\vec{s})$ lies between $\vec{k}(b_{s_n}^+)$ and $\vec{k}(b_{s_n+1}^-)$ for all $n \in \mathbb{N} = \{0, 1, 2, \ldots\}$.

Since we can only compute partial kneading information about points, we will be especially interested in modifications to the function f or its kneading invariants $\vec{k}_f(d)$ which either decrease or increase the symbolic dynamics. The critical observation is that if d is a maximum (the left-hand end of a decreasing branch or right-hand end of an increasing branch), then decreasing $\vec{k}(d)$ decreases the symbolic dynamics.

We let $\delta_k(d)$ for $d \in D$ be equal to $+1$ if f^k has a local minimum at d, and -1 if f^k has a local maximum at d.

2.2.2 Relaxing and stretching

We now give an informal analysis of homotopies and their effect on the kneading invariants.

If $d \in D$ is a discontinuity and $\vec{s}$ is an itinerary, we write $\sigma^{n-1}(\vec{k}(d)) \succ \vec{s}$ if f^n has a maximum at d and $\sigma^{n-1}(\vec{k}(d)) > \vec{s}$, or $\sigma^{n-1}(\vec{k}(d)) \succ \vec{s}$ if f^n has a minimum at d and $\sigma^{n-1}(\vec{k}(d)) < \vec{s}$. This means that relaxing f moves $f^n(d)$ towards the point x with itinerary s. If d is a discontinuity, and $\vec{k}_1$ and $\vec{k}_2$ are two possible kneading sequences for d, we write $\vec{k}_1 \prec \vec{k}_2$ if d is a maximum and $\vec{k}_1 < \vec{k}_2$, or d is a minimum and $\vec{k}_1 < \vec{k}_2$. Clearly, if f and g are two maps with the same discontinuity and critical points, and if $\vec{k}_f(c_j) \prec \vec{k}_g(c_j)$, then every itinerary of f is an itinerary of g.

Suppose f has a local maximum at $d = p^{\pm}$. Let f_s be a homotopy with $f_0 = f$ such that $f_s(d)$ decreases as s increases. Then images $f_s^k(d)$ move to the left if $\delta_k(d) = -1$ and to the right if $\delta_k(d) = +1$ for as long as $f_s^i(d)$ remains away from $\partial\mathcal{B}$ for $i \leq k$. Intuitively, this will reduce the kneading invariant $\vec{k}_{f_s}(d)$, as long as $f_s^i(d)$ crosses $\partial\mathcal{B} \setminus b$ for some $i < j$ before $f_s^k(d)$ crosses b. For in this case, $f_s^k(d) = f^{k-1}(f_s(d)) \gtrless f^k(d)$ depending on $\delta_k(d)$.

We call a homotopy which decreases the kneading invariant $\vec{k}_{f_s}(d)$ of a maximum, or increases the kneading invariant of a minimum, a *relaxation*, and say that $\vec{k}_f(d)$ *relaxes* to $\vec{k}_{f_s}(d)$ and that d is relaxed. Similarly, a homotopy which increases the kneading invariant of a maximum, or decreases the kneading invariant of a minimum, is a *stretching*.

Now suppose f^k has a maximum at $d \in D$ and that $x \in \partial\mathcal{B}$. Then we can homotope f in a neighbourhood of d to set $f_s(d) = f_s(x) = f(x)$. This will have essentially the same effect as a relaxation in a neighbourhood of d to set $f^k(f_s(d)) = f(x)$.

We define the relations $\prec$ and $\succ$ on forward images of discontinuity points by saying $f^k(d) \prec x$ if $\delta_k(d)$ has the same sign as $f^k(d) - x$, and $f^k(d) \succ x$ if $\delta_k(d)$ has the opposite sign as $f^k(d) - x$.

There is a critical difference between the locality of relaxations and stretching. For if f has a maximum at $d \in \partial B$, then to set $f_s(d) = y < f(d)$ but preserve monotonicity on B, we need to change f_s on the entire neighbourhood $[d, x]$ of d with $f(x) = y$. However, to stretch to $f_s(d) = y > f(d)$, we only need to change f_s on an arbitrarily small neighbourhood of d by introducing a very narrow "spike". Note that as shown in 2.2, it is not necessary to preserve continuity at b, though this could be done by relaxing or

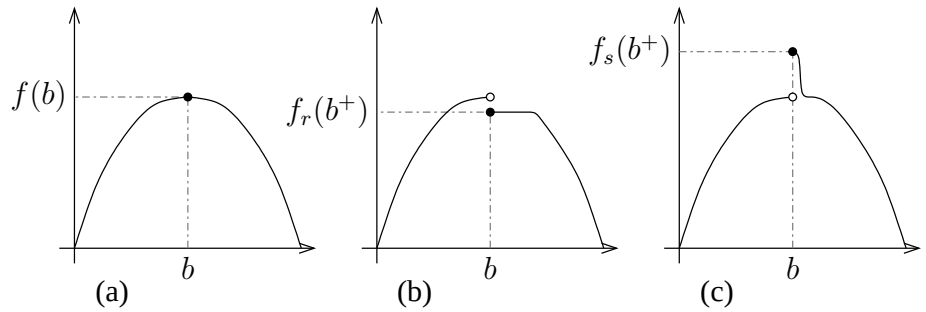


Figure 2.2: Stretching and relaxing at a critical point. (a) A critical point b of f. (b) Relaxing b^+ to reduce the symbolic dynamics. (c) Stretching at b^+ to increase the symbolic dynamics.

stretching at b^- as well as at b^+.

We will need the following result on forcing relations between maps due to their kneading invariants.

Theorem 2.2.1. *Let f and g be $\mathcal{B}$-monotone-continuous maps with discontinuities $D = \partial\mathcal{B}^{\pm}$. Suppose that for all discontinuities $d \in D$, either*

1. *$\vec{k}_g(d) \prec \vec{k}_f(d)$, where $\prec$ is $<$ when f has a maximum at d, and $>$ when f has a minimum at d.*

2. *$g^n(d) = d'$ for some d' with $\sigma^n(\vec{k}_f(d)) \succ \mathrm{itin}(d')$.*

Then every $\mathcal{B}$-itinerary of g is an itinerary of f.

Proof. Consider an itinerary $\vec{s}$ of g, and let x be a point with itinerary $\vec{s}$. Suppose $s_n = i$, so $g^n(x) \in [b_i^+, b_{i+1}^-]$. Then by the kneading condition, we have $\sigma^{n+1}(\vec{s})$ between $\vec{k}_g(b_i^+)$ and $\vec{k}_g(b_{i+1}^-)$.

If $\vec{k}_g(d)$ is determined by Condition 1 above, then $\sigma^{n+1}(\vec{s})$ has the same relation to $\vec{k}_f(d)$ as $\vec{k}_g(d)$.

If $\vec{k}_g(d)$ is determined by Condition 2, then $g^m(d) = b_j$ for some $b_j \in \partial\mathcal{B}$, and $g^l(d)$ and $f^l(d)$ lie in the same regions for $l < m$. Then if the itineraries of d and $f^n(x)$ differ in the first m symbols, then $\sigma^{n+1}(\vec{s})$ and $\vec{k}_f(d)$ differ in the first $m-1$ symbols, so the bound is satisfied. Otherwise we have $s_{m+n} \in \{j, j+1\}$, and since σ is a valid itinerary for g, so we have $\sigma^{m+n+1}(\vec{s}) \gtrless \vec{k}_g(b_j^{\pm})$. Then $\sigma^k(\vec{s})$ satisfies the kneading conditions for $k < n+m$ since the conditions are the same for f and g. If $\vec{k}_g(b_j^{\pm})$ is determined by Condition 1, then σ^{m+n} satisfies the kneading conditions as above. If $\vec{k}_g(b_j^{\pm})$ is also determined by Condition 2, then we need to continue

until either we find a critical point determined by Condition 1, or by some iterate differing from a kneading invariant, or we have an infinite loop. □

2.2.3 Kneading invariants from mapping data sets

Ideally, given a mapping dataset $\mathscr{F} = (Y, f, \lfloor\cdot\rceil, \leq)$ as in Definition 2.1.1, we would like to extend the data by extending the ordering to a total ordering, and extending the mapping information to a total function, such that the resulting kneading information forces only orbits which are present in all maps consistent with the mapping data. Unfortunately, this is not always possible. We shall see examples of mapping datasets (which can be taken to be totally ordered) for which there is no extension to a total function with minimal symbolic dynamics. Instead, we need to relax the ordering on the mapping dataset in certain instances.

Given a mapping dataset, if we know the ordering of the partition boundary points Q, the discontinuity points D and the critical points C, and all their kneading sequences, then we can determine the ordering of all images of points of P. If all points of P are eventually periodic, then so are the kneading sequences, and can be determined purely from the image data.

Definition 2.2.1. *Kneading invariants $\vec{k}(d)$, $d \in D$ are* consistent *with a mapping dataset $(Y, f, \lfloor\cdot\rceil, \leq)$ if*

1. $\sigma^j(\vec{k}(d)) \in [\vec{k}(p^+_{k_j(d)}), \vec{k}(p^-_{k_j(d)+1})]$, *and*

2. *if $f^j(d) \geq f^{j'}(d')$, then $\sigma^j(\vec{k}(d)) \geq \sigma^{j'}(\vec{k}(d'))$.*

If every kneading invariant is eventually periodic, then consistency can be checked by a finite algorithm.

In order to compute lower- and upper-approximations to the symbolic dynamics, we therefore need to set the image of each $f^n(d)$ for which the image is not already given by the mapping data. By the kneading theory, we should aim to relax the kneading invariants to reduce the symbolic dynamics, and stretch kneading invariants to increase the symbolic dynamics.

If $x < y$, we say that x and y are *adjacent* if there is no point z such that $x < z$ and $z < y$.

Algorithm Scheme 3. *Given a mapping dataset $\mathscr{F}$, we aim to find a map g such that any map f compatible with $\mathscr{F}$ has all the itineraries of g.*

1. *Choose $d \in D$ with exactly n forward images in the mapping data. Let $x = f^m(c)$ be the point adjacent to d which $f^n(d)$ relaxes to. Consider the following cases:*

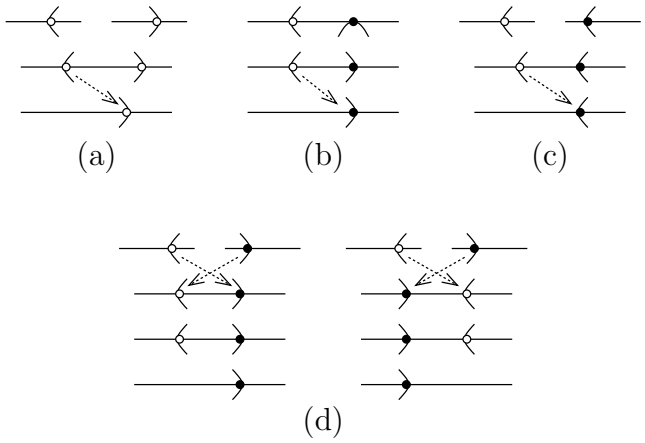


Figure 2.3: Relaxations to the map g under-approximating the symbolic dynamics made in Algorithm 3.

(a) *$d = c$. Then $f^n(d)$ and x lie in the same orbit. Extend the mapping data by setting $g(g^n(d)) = g^m(c)$.*

(b) *$d \neq c$ and relaxing g at c moves $g^m(c)$ away from $g^n(d)$. Then we set $g(g^n(d)) = g^{m+1}(c)$; note that $g^{m+1}(c)$ need not be defined yet.*

(c) *$d \neq c$ and $g^{n-m+i}(d)$ is adjacent to $f^i(c)$ for $i = 0, \ldots, m$. Then set $g(g^n(d)) = g^{m+1}(c)$.*

(d) *$d \neq c$, $m > 0$ and relaxing g at c moves $g^m(c)$ towards $g^n(d)$. Let l be maximal such that $g^{n-i}(d)$ and $g^{m-i}(c)$ lie in the same lap for $0 \leq i \leq l$. Then remove the ordering requirement between $g^{n-i}(d)$ and $g^{m-i}(c)$.*

2. *Whenever $\sigma^i(\mathrm{itin}(d)) \neq \sigma^j(\mathrm{itin}(c))$ (i.e. the known part of the itineraries definitely differ), then add the ordering $g^i(d) \lesseqgtr g^j(c) \iff \sigma^i(\mathrm{itin}(d)) \lesseqgtr \sigma^j(\mathrm{itin}(c))$.*

In order to find h which forces all itineraries of f, we use the same construction, but set $h(h^n(d))$ in the direction of stretching at d, rather than relaxing.

Note that if we set $g(g^n(d))$ equal to $g^{m+1}(c)$, and $g^n(d)$ and $g^m(c)$ lie in the same monotone branch, then g is constant on $[g^n(d), g^m(c)]$.

We can apply Algorithm 3 without any ordering data except for comparisons of $f^i(d)$ with c for $d, c \in D$. In this case, we obtain the dynamics forced by the symbolic itineraries. Notice that is c is a critical point of f which is not a discontinuity point, then relaxing and strectching at c can be performed keeping the maps continuous.

Theorem 2.2.2. *The symbolic dynamics of the map g constructed by Algorithm 3 is a subshift of that of f, and the symbolic dynamics of h is a supershift of f.*

Proof. Suppose $g(f^n(d))$ is determined by (a). Then setting $g^{n+1}(d)$ to $f^{m+1}(d)$ makes d eventually periodic. Further, if f^{n-m} is orientation-reversing, then $f^{2n-m}(d)$ lies on the same side of $f^n(d)$ as does $f^m(d)$. This means that $k_g(d)$ is an itinerary of f.

Suppose $g(f^n(d))$ is determined by (b). Then $\vec{k}_g(d) \prec \vec{k}_f(d)|_{[0,n-1]}\sigma^{m-1}\vec{k}_g(c)$, so satisfies Condition 2 of Theorem 2.2.1.

Suppoe $g(f^n(d))$ is determined by (c). Then $\vec{k}_g(d) = \vec{k}_f(d)|_{[0,l-1]}\vec{k}_g(c)$. Although this may not relax $\vec{k}_f(d)$, the kneading invariant $\vec{k}_g(d)$ satisfies Condition 2 of Theorem 2.2.1.

Suppose we are in the situation of (d). Then the kneading invariant $\vec{k}_g(d)$ is not determined, but since we remove some ordering relations, the symbolic dynamics must decrease for g. □

We now give an example of a map f with a mapping dataset computed using forward refinement which does not admit a representative with minimal symbolic dynamics.

Example 2.2.1. *Let $X = [0,3]$, and $\mathcal{Q}$ be the partition with endpoints $\{0, \frac{1}{3}, \frac{2}{3}, 1, 2, 2\frac{1}{3}, 2\frac{2}{3}, 3\}$. Let f be a system such that $f(0) = 1$, $f(1) = 2$, $1 = f(0) < f(\frac{2}{3}) < f(\frac{1}{3}) < f(1) = 2$, and $f(2) = f(2\frac{2}{3}) = 3$, $f(2\frac{1}{3}) = f(3) = 2$. Define symbols $q_i = [\frac{i}{3}, \frac{i+1}{3}]$ and $q_{i+4} = [2\frac{i}{3}, 2\frac{i+1}{3}]$ for $i = 0, 1, 2$, and $q_3 = [1, 2]$.*

Then since $f(I_0) \cup f(I_2) = I_3$ and $f(I_3) = f(I_4) = f(I_5) = f(I_6) = I_4 \cup I_5 \cup I_6$, we know that for any sequence $\vec{s} \in \{4, 5, 6\}^\omega$, there exists a point with itinerary $03\vec{s}$ or $23\vec{s}$. However, by taking $f(\frac{1}{3})$ very close to $f(0)$, we can rule out the itinerary $03\overline{5}$, whereas taking $f(\frac{2}{3})$ very close to $f(1)$, we can rule out the itinerary $23\vec{s}$ (as long as $\vec{s}$ is not either $\overline{4}$ or $\overline{6}$.

2.2.4 The kneading algorithm for unimodal maps

Let f be a unimodal map on $[a, b]$ with $f(b) = f(a) = a$ and single critical point at c. Let $P_0 = (a, c)$ and $P_1 = (c, b)$. The computation of the topological entropy using kneading data for unimodal maps has been considered in [40, 41]. When applying Algorithm 3 we are always in case (a) of choosing the image of $f^n(c)$. This means that if $f^i(c) < f^n(c) < f^j(c)$ and f^n has a maximum at c, then we set $g(f^n(c)) = f^{i+1}(c)$, and $h(f^n(c)) = f^{j+1}(c)$. In the former case, we collapse the interval $[f^i(c), f^n(c)]$ and map $[f^n(c), f^j(c)]$ to $[f^{i+1}(c), f^{j+1}(c)]$. The resulting kneading sequence is eventually periodic,

and may be periodic if $f^n(c)$ relaxes onto c itself. If the mapping data does not give a total order on the images of c, then we may be able to deduce some of the ordering from the kneading invariant.

The following example shows that just relying on the kneading sequences can lose information.

Example 2.2.2. *Consider a unimodal map f with $c_2 < c_3 < c_5 < c_0 < c_4 < c_1$, where $c_i = f^i(c)$. Then $c_4 < c_6 < c_1$, $c_2 < c_7 < c_5$ and $c_3 < c_8 < c_1$, so the kneading sequence is $1001010?? \cdots$.*

Continuing the kneading sequence in a consistent way with as little dynamics as possible gives $\vec{k} = \overline{1001010}$, which induces topological entropy 0.522. However, for this kneading invariant, the itinerary of c_3 is $010101\cdots$ and the itinerary of c_5 is $010100\cdots$ which is lower. Hence the kneading information is not consistent with the interval information.

If we are to respect the condition $c_3 < c_5$, then we can only change the kneading sequence to $10\overline{01}$, in which c_3 and c_5 have the same periodic itinerary, and which induces topological entropy 0.528.

2.3 Examples

We now give some simple examples illustrating the main features of the chapter and the results of our algorithms. We consider the quadratic unimodal map

$$f_1(x) = \mu x(1-x),$$

the cubic bimodal map

$$f_2(x) = ax + (1-a-b)x^2 + bx^3$$

and the normal form of a discontinuous border-collision bifurcation of a stable fixed-point with a square-root singularity [42].

$$f_3(x) = \begin{cases} ax + \epsilon \text{ if } x \leq 0; \\ \sqrt{bx - c} \text{ if } x \geq 0. \end{cases} \tag{2.1}$$

2.3.1 The unimodal map

Consider the unimodal map:

$$f_1(x) = \mu x(1-x) \text{ with } \mu = 3.92$$

Let $a = 0$ and $b = 1$ be the endpoints of the interval I and $c = 0.5$ be the critical point. Let $I_0 = [a, c]$ and $I_1 = [c, b]$.

The orbit of the critical point is given by $f^i(c) = c_i$ with

$$c_0 = 0.5,\ c_1 = 0.98,\ c_2 \approx 0.077,\ c_3 \approx 0.278,\ c_4 \approx 0.787,$$
$$c_5 \approx 0.657,\ c_6 \approx 0.883,\ c_7 \approx 0.405,\ c_8 \approx 0.945,\ c_9 \approx 0.204.$$

The itinerary of $f_1(c)$ is therefore

$$\text{itin}(f_1(c)) = 100111010\cdots$$

Consider the partition into six intervals with boundary points

$$p_0 = a, p_1 = c_2, p_3 = c_3, p_4 = c_0, p_5 = c_4, p_6 = c_1, p_7 = b.$$

Note that $f(p_5) \in (p_4, p_5)$. Taking $J_i = [p_i, p_{i+1}]$ we have

$$f_1(J_1) \supset J_2 \cup J_3;\ f_1(J_2) \supset J_4; f_1(J_3) \supset J_4;\ f_1(J_4) \supset J_1 \cup J_2$$

and also

$$f_1(J_3) \cap J_3 \neq \emptyset \text{ and } f(J_4) \cap J_3 \neq \emptyset.$$

Now consider the computation of entropy. Using the lower and upper symbolic dynamics on the partition, we obtain entropies of 0.41962 and 0.73286. The upper shift on the refined partition has multiple orbits with the same itinerary on I_0 and I_1. If we consider the entropy of the shift itself (by using additional states J_{12} and J_{34}) we obtain 0.583.

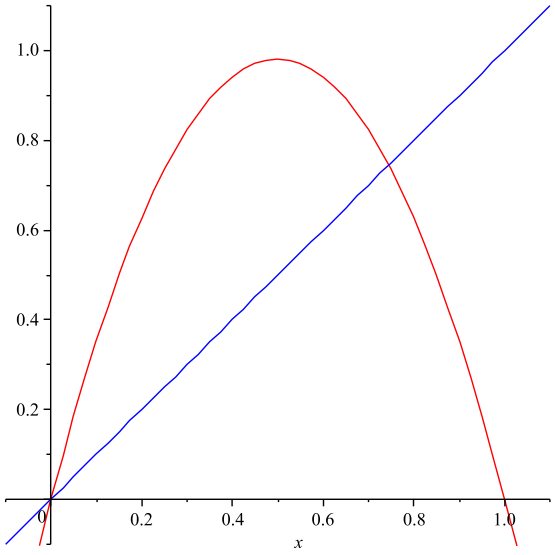


Figure 2.4: Unimodal map

steps	entropy	running time backward	running time hybrid
1	[0:0.69314]	0.05	0.11
2	[0.48121:0.69315]	0.11	0.12
3	[0.48121:0.60938]	0.24	0.26
4	[0.54353:0.60938]	0.55	0.58
5	[0.54353:0.58356]	1.26	1.21
6	[0.54353:0.56240]	2.68	2.63
7	[0.54761:0.56240]	6.73	5.74
8	[0.55642:0.56240]	17.74	11.04
9	[0.55642:0.56099]	49.32	18.86
10	[0.55842:0.56099]	149.26	30.44
11	[0.55990:0.56099]	450.37	47.78
12	[0.55990:0.56073]	1246.19	76.07
13	[0.55990:0.56026]		116.80
14	[0.55998:0.56026]		186.57
15	[0.56014:0.56026]		309.09

Table 2.1: Entropy of the unimodal map using backward and hybrid refinement

Improvement of entropy using kneading theory

If we use the kneading theory, we see that by setting $f_1(c_4) = c_4$ we have $\text{itin}(f_1(c)) = 1001111\cdots$ which is higher in the unimodel order, and setting $f_1(c_4) = c_0$ we have $\text{itin}(f(c)) = 1001^0_1100\cdots$ which is lower than $10011101\cdots$. The corresponding lower and upper entropies are 0.54354 and 0.571.

Consider a unimodal map with kneading invariant $\overline{1001^0_1}$, so c is periodic and $f_1^5(c) = c$. The images of c are ordered $a < f_1^2(c) < f_1^3(c) < c < f_1^4(c) < f_1(c) < b$. We can compute the topological entropy of the shift, and obtain a value of 0.54354.

Consider a unimodal map with kneading invariant $100\bar{1}$, so c is eventually periodic and $f_1^5(c) = f_1^4(c)$. The images of c are ordered $a < f_1^2(c) < f_1^3(c) < c < f_1^4(c) < f_1(c) < b$. We can compute the topological entropy of the shift, and obtain a value of 0.57058.

Now consider a unimodal map with kneading invariant $\vec{k} = 10011\cdots$. From the kneading theory, we know $10011 \succ 1001^0_1$ so we have $h_{\text{top}}(f_1) \geq 0.64$. Define $R_1 = [f_1^2(c), f_1^3(c)]$, $R_1 = [f_1^3(c), c]$, $R_3 = [c, f_1^4(c)$ and $R_4 = [f_1^4(c), f_1(c)]$ When using the forward refinement strategy, since $f_1^5(c) \in R_3$, the interval R_3 maps to $[f_1^5(c), f_1(c)]$ and the interval $[f^4(c), f(c)]$ maps to $[f_1^2(c), f_1^5(c)]$ which together cover R_3, but neither does individually. Hence

neither of the transitions $R_3 \longrightarrow R_3$ nor $R_4 \longrightarrow R_3$ in the automaton for the lower symbolic dynamics. The entropy bound obtained drops to 0.41962.

The overall effect of the kneading theory is to "choose" a transition, either $R_3 \longrightarrow R_3$ or $R_4 \longrightarrow R_3$, to put in the lower symbolic dynamics, while still ensuring that the dynamics is a lower bound. The chosen transition is the one giving least entropy.

steps	entropy	running time
1	[0:0.693147180559]	0.01
2	[0:0.693147180559]	0.04
3	[0.481211825059:0.6931471805599]	0.03
4	[0.481211825059:0.6093778634360]	0.03
5	[0.543535072497:0.5705796667792]	0.06
6	[0.543535072497:0.5705796667792]	0.10
7	[0.555194599694:0.5623991486459]	0.10
8	[0.557934930430:0.5623991486459]	0.10
9	[0.558939519816:0.5623991486459]	0.13
10	[0.560046256097:0.560988810813]	0.10
15	[0.560216078753:0.560259207813]	0.09
20	[0.560235149472:0.560235821005]	0.16
25	[0.560235528970:0.560235669923]	0.20
30	[0.560235632260:0.560235638765]	0.27
35	[0.560235635949:0.560235636493]	0.36
40	[0.560235636370:0.560235636375]	0.55

Table 2.2: Entropy of the unimodal map with kneading algorithm

2.3.2 A bimodal map

$$f_2(x) = 0.35x^3 - 2.75x^2 + 4.5x + 3.3$$

The graph of the map is shown in Fig. 2.5. The initial partition has the following end-points: $\text{fix}_0 \approx -0.619$, $c_0 \approx 1.015$, $c_1 \approx 4.223$, $\text{fix}_2 \approx 5.886$

In Table 2.3 we show the computed values of the entropy of the map with backward and hybrid refinement up to 10 steps of refinement. In Table 2.4 we show the computed values of the entropy of the map with kneading algorithm up to 35 steps of refinement.

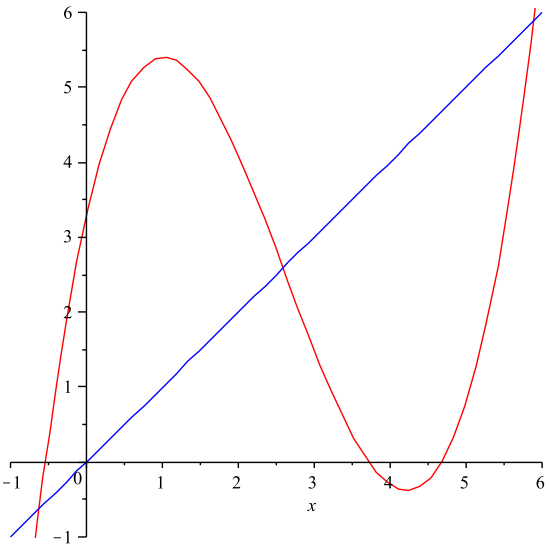


Figure 2.5: Bimodal map

steps	entropy	running time backward	running time hybrid
1	[0:0.88137]	0.20	0.26
2	[0.69314:0.83412]	0.77	0.82
3	[0.75832:0.81443]	1.15	1.23
4	[0.77727:0.80327]	3.66	3.52
5	[0.77727:0.78803]	12.65	8.80
6	[0.78109:0.78636]	51.15	19.97
7	[0.78303:0.78533]	233.61	42.91
8	[0.78351:0.78463]	1057.99	83.19
9	[0.78383:0.78441]		141.64
10	[0.78408:0.78425]		279.64

Table 2.3: Entropy of the bimodal map with backward and hybrid refinement

2.3.3 A discontinuous border-collision

A discontinuous border-collision bifurcation of a stable fixed-point with a square-root singularity gives the following normal form [42].

$$f_3(x) = \begin{cases} ax + e \text{ if } x \leq 0; \\ \sqrt{bx} - c \text{ if } x \geq 0. \end{cases} \tag{2.2}$$

where the parameters values are: $a = -3.5$, $e = 1.5$, $b = 2$, $c = 2$. In Table 2.5 we show the computed values of the entropy of the map with

steps	entropy	running time
1	[0 :1.098612288668]	0.01
2	[0 :0.881373587019]	0.01
3	[0.6931471806139 :0.7949452427288]	0.05
4	[0.7641997080283 :0.7949452427288]	0.06
5	[0.7772705789959 :0.7949452427288]	0.06
6	[0.7772705789959 :0.7880304603014]	0.07
7	[0.7825090975765 :0.7847829456947]	0.10
8	[0.7839408308854 :0.7844986535653]	0.12
9	[0.7840431481027 :0.7844184271750]	0.14
10	[0.7840949704856 :0.7842660362345]	0.16
15	[0.7841523953806 :0.7841538593989]	0.31
20	[0.7841531278765 :0.7841531492005]	0.51
25	[0.7841531371710 :0.7841531375475]	0.71
30	[0.7841531373109 :0.7841531373218]	1.02
35	[0.7841531373175 :0.7841531373177]	1.50

Table 2.4: Entropy of the bimodal map with kneading algorithm

kneading algorithm up to 40 steps of refinement.

2.3.4 Comparison of different strategies

In this section we compare the effectiveness of different refinement strategies and methods for extracting the symbolic dynamics.

In Table 2.1 after the seventh steps the running time increases by a factor which ranges over the interval $[2.5, 3]$ for backward refinement while with a factor which ranges over the interval $[1.5, 1.7]$ for hybrid refinement. In Table 2.3 after the fifth steps the running time increases by a factor which ranges over the interval $[4.5, 4.6]$ for backward refinement while with a factor which ranges over the interval $[1.7, 2]$ for hybrid refinement.

This observations suggest that the running time follows an exponential law in terms of the steps of refinement, in the hybrid refinement the exponential rate is lower than with the backward refinement. This outcome matches the structure of the algorithms where in both cases we need to deal with exponentially growing number of preimages of endpoints as we said in section 3.2. The hybrid approach is faster because it filters out most of the endpoints obtaining a partition whose cardinality grows linearly.

Instead in the forward refinement kneading approach the data approximatively match a liner function for the running time in term of the steps. This

steps	entropy	running time
1	[0:0.481211]	0.02
2	[0:0.481211]	0.02
3	[0,0.240606]	0.09
4	[0,0.240606]	0.10
5	[0,0.240606]	0.10
6	[0,0.240606]	0.12
7	[0,0.240606]	0.17
8	[0,0.240606]	0.26
9	[0,0.227270]	0.28
10	[0,0.227270]	0.17
15	[0.214105,0.221197]	0.23
20	[0.217382,0.218459]	0.41
25	[0.217429,0.217601]	0.75
30	[0.217535,0.217601]	0.87
35	[0.217540,0.217561]	1.21
40	[0.217547,0.217559]	1.89

Table 2.5: Entropy of the discontinuous border-collision map with kneading algorithm

algorithm is much faster because by one hand the computation of forward images is easier then backward images and by the other hand the number of endpoints increases linearly.

In all these three approaches the accuracy of the entropy approximation increases monotonically, despite not always strictly monotonically. Finding an exact law which expresses the rate of convergence of the computation of topological entropy in terms of the steps of refinement and the running time is not straightforward. This problem requires further investigation and the guess of the authors is that it deeply depends on the structure of the map.

2.4 Case Studies

In this section, we present three case studies: a discontinuous border-collision singularity, a simple hysteresis system and the singular limit of the Van der Pol equation.

2.4.1 A hysteresis switching system

We now consider a piecewise-affine model of a system governed by hysteresis switching [43]. Let $H(x)$ be the hysteresis map, informally given by $H(x) = 0$ for $x \leq 1$ and $H(x) = 1$ for $x \geq 0$. Consider the system:

$$\begin{aligned} \dot{x} &= y + a_1 H(x/b) \\ \dot{y} &= -x - 2\sigma y + a_2 H(x/b). \end{aligned} \tag{2.3}$$

The return map is defined on the set $P = \{(x, y) \in \mathbb{R}^2 \mid x = 0,\ y > 0\}$ as: $f(p) = \phi(p, \bar{t})$ where $\bar{t} = \min_t\{t \mid q = \phi(p, t) \in P \text{ and } H(x_q) = 0\}$ and $\phi(p, t)$ is the integral map of the system 2.4. 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. 2.6. We take an

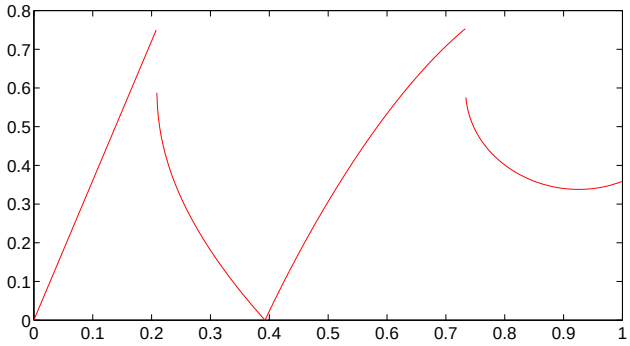


Figure 2.6: The return map for the hysteresis system (2.4).

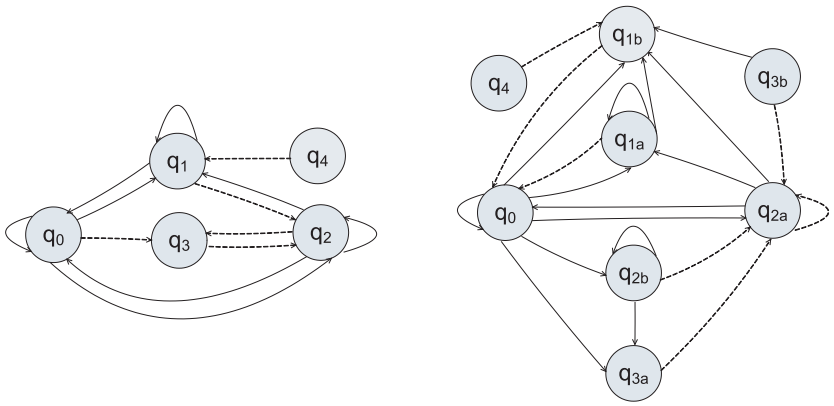
initial partition $\mathcal{Q}$ which are the domains of the monotone branches. The partition elements are $Q_0 = [p_0, p_1]$, $Q_1 = [p_1, p_3]$, $Q_2 = [p_3, p_5]$, $Q_3 = [p_5, p_7]$ and $Q_4 = [p_7, p_8]$ where the boundary points are

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

The associated symbolic dynamics is in Fig. 2.7(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 i.e. $f(p_1^-) = f(p_5^-)$ and $f(p_1^+) = fz(p_5^+)$. The partition after one iteration of forward refinements has the following additional 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. 2.7(b).



(a)

(b)

Figure 2.7: Lower and upper approximation of symbolic dynamics for the hysteresis system (2.4) for (a) the initial partition and (b) the forward refinement of the initial partition

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 Table 2.6. In Table 2.7 we show instead the much more accurate results are obtained using the kneading approach.

2.4.2 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

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

Table 2.6: Entropy of the return map of the hysteresis system with forward refinement

steps	entropy with kneading algorithm
1	[1.05757681,1.14673525]
2	[1.09861228,1.11532470]
3	[1.10409066,1.10921416]
4	[1.10732895,1.10921416]
5	[1.10763700,1.10800210]
6	[1.10770695,1.10783470]
7	[1.10777981,1.10780229]
8	[1.10779612,1.10780078]
9	[1.10779674,1.10780027]
10	[1.10779826,1.10779894]

Table 2.7: Entropy of the return map of the hysteresis system with kneading algorithm

singular limit of the forced Van der Pol oscillator have been studied in [44]. In this chapter we analyse the following version of the equation:

$$\ddot{x} + \mu(x^2 - 1)\dot{x} + x = a(x^2 - 1)\sin(2\pi\nu\tau) \tag{2.4}$$

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:

$$\begin{aligned} \varepsilon\dot{x} &= y + x - x^3/3; & (2.5)\\ \dot{y} &= -x + a(x^2 - 1)\sin(2\pi\theta); & (2.6)\\ \dot{\theta} &= \omega. \end{aligned}$$

The *fast subsystem* is defined by (2.5), 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 $|x| \leq 1$, and that when this occurs, the value of x jumps instantaneously to one of the stable fixed points of (2.5).

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:

$$\begin{aligned} \dot{x} &= -x + a(x^2 - 1)\sin(2\pi\theta) \\ \dot{\theta} &= \omega(x^2 - 1) \end{aligned} \tag{2.7}$$

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

$$G = \{(x, \theta) \mid |x| = 1\}; \qquad r(x, \theta) = (-2\,\mathrm{sgn}(x), \theta). \tag{2.8}$$

In other words, when the guard condition $x = \pm 1$ 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. 2.8. We have computed the

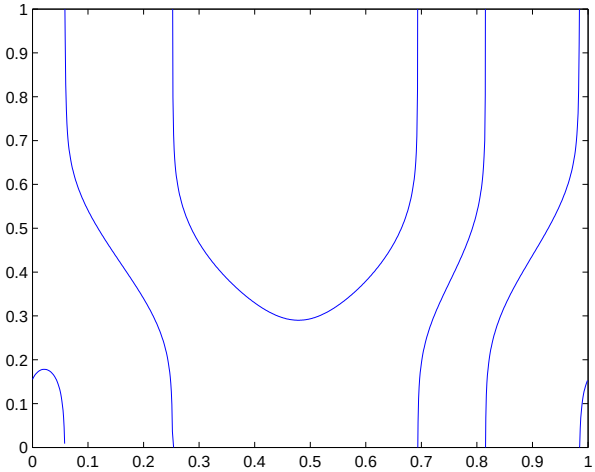


Figure 2.8: The half return map for the singular limit of the forced Van der Pol oscillator (2.7).
lower and upper symbolic dynamics with respect to the partition given by the continuous branches using forward refinement.

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. 2.8 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.

2.5 Final Remarks

In this chapter we have considered the computation of symbolic dynamics relative to an arbitrary partition for piecewise-monotone-continuous maps of the interval. We have considered the case in which images and preimages cannot be computed exactly, but only approximations using interval arithmetic. We have considered both forwards and backwards refinements of the initial partition into monotone branches, and the computation of symbolic dynamics using both covering relations and kneading theory. We have given a number of illustrative examples from the unimodal and trimodal families, and two case studies.

As mentioned in the introduction, an important motivation for this work was to gain intuition in methods for computing symbolic dynamics in a simple case, in the hope that some of this can be carried over to higher dimensions. The backward refinement strategy for computing the upper symbolic dynamics can be carried directly over to higher dimensions. As we have seen, the

entropy bounds for the sofic shift are typically very good, but care must be taken to reduce the dynamics of the generating directed graph.

Computing the lower symbolic dynamics with convergence in entropy is complicated in higher dimensions since we can prove the existence of orbits without the covering relation. This problem is analyzed in the following chapter.

Chapter 3

Computation of symbolic dynamic in two dimensions

3.1 Introduction

Despite the many relevant and general results for the one dimensional case, tackling the study of maps in higher dimension is still particularly challenging. Even in the two dimensional case the analysis of such systems is in many ways still mostly an open field of research. The task is significantly more difficult than the case of one dimension, first of all for the obvious reason of impossibility to visualize the graph of the map which lies in a four dimensional space, second for the complexity which an additional dimension can add to the dynamics and the more complicated structure of partitions of the space. Kneading theory is not extendible to higher dimension, the covering relations approach is extendible but ineffective in general to provide a lower approximation of the symbolic dynamics of the map. Besides, differently from one dimension, there are no results on finding a partition which make sure the symbolic dynamics grasps all the complexity of the map.

There are three important existing methods for computing under approximations of symbolic dynamics of two dimensional maps. The theory of *trellises* has already been successfully used to develop algorithms for computation of symbolic dynamics for two-dimensional nonlinear maps with homo/heteroclinic tangles in [17]. The Conley index which has been used to prove the existence of localised symbolic dynamics of the Hénon map by Mischaikow et al. in [12] and Day et al. in [13]. Another approach based on parallelotopic covering relations was used by Galias and Zgliczynski in [45].

In this chapter we develop an algorithm to compute global symbolic dynamics of two-dimensional piecewise-affine maps (not necessarily continu-

ous), a relatively simple class of map which has the important property of being dense in the set of piecewise continuous maps. The technique we use for the computation of the symbolic dynamics is Conley index and its decomposition developed by Szymczak in [19]. The main improvement of our results over existing work is that we obtain a good approximation to the *global* symbolic dynamics, which we validate by computing over-approximations to the symbolic dynamics. We also explicitly consider the discontinuous case.

In our approach we want to investigate a way to generalize and make automatic the computation of the symbolic dynamics and related needed information such isolating neighbourhood and index pair for the above mentioned group of maps. We use compare the results of the algorithm we develop with the result of the algorithm based on homoclinic tangle.

3.2 Piecewise affine maps and polyhedra

In this chapter we focus the analysis on two dimensional piecewise affine maps $f : \mathbb{R}^2 \longrightarrow \mathbb{R}^2$

$$f(x) = A_i x + b_i \quad x \in P_i \tag{3.1}$$

where $P_1, P_2, ...P_n$ is a topological partition of $\mathbb{R}^2$ whose members are convex polyhedra.

3.2.1 Computability theory and numerical computation for piecewise affine maps

We note that in principle by working with a piecewise affine map with rational coefficients it is possible to compute exact images of rational points, because they are invariant in the set of rationals.

Nevertheless we deal with approximation of vertices and linear inequalities which define polyhedra of the partitions. Vertices of polyhedra and linear inequalities of polyhedra can have respectively irrational coordinates and coefficients such as the stable and unstable manifold of the map and their intersections with the axes. Because of that we need to adopt the interval approximation.

3.2.2 Calculus of polyhedra

In our work we develop algorithms on piecewise-affine maps. Because the set of polyhedra is closed with respect to piecewise affine maps and set operations such as intersection and union, it turned to be convenient to model this kind of sets in a object oriented programming language as shown in Data

Type 3.2 and to develop a calculus of polyhedra. Data Type 3.1 shows the model of convex polyhedra which is necessary for the definition of Data Type 3.2. We describe convex polyhedra by a set of linear inequalities, all operations defined over them can be performed by only using linear inequalities without needing explicit information about the vertices. The polyhedra are represented a as collection of convex polyhedra whose interiors are pairwise disjoint.

Data Type 3.1 Convex Polyhedron

OBJECT FIELDS:

- $[A], [b]$ # interval matrices which defines a polyhedron via the inequality $[A]x \leq [b]$.

METHODS:

- constructor($[A], [b]$)
- image(f) # implemented by computing the image of each linear inequality which defines the polyhedron.
- preimage(f)
- union(self,$object_2$) # return output only when the union of and self $object_2$ is a convex polyhedron
- intersection($object_2$)

We remark the fact that the condition of polytopes having mutually disjoint interiors and sharing one side is numerically undecidable. Nevertheless because we start with polytopes defined by linear inequalities with exact values, by keeping track of the generation of all linear inequalities at each step of the algorithms in the following sections, we can decide when two polytopes intersect each other only by one side by checking if they share a linear inequality with opposite sign, even if the inequalities are approximated by interval inequalities (inequalities whose coefficients are interval instead of numbers).

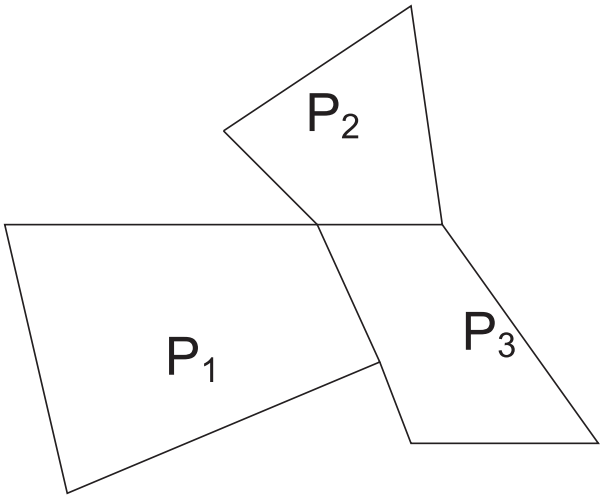


Figure 3.1: Polyhedron

Data Type 3.2 Polyhedron

OBJECT FIELDS:

- $[P_1, , , , , , P_n]$ # list of pairwise disjoint open convex polyhedra such that: $\bigcup_{i=1}^{n} \overline{P}_i$ is the whole polyhedron

METHODS:

- constructor($[Q_1, , , , , , Q_n]$) # the constructor takes care of checking whether the convex polyhedra are pairwise disjoint and if not to compute a list of pairwise disjoint convex polyhedra which cover the same region
- image(f) # return constructor($[f(P_1), , , , , , f(P_n)]$)
- union($object_2$) #return constructor(object.$[P_1, , , , , , P_{n_1}]$ + object2.$[P_1, , , , , , P_{n_2}]$)
- intersection($object_2$)

3.3 Algorithms for computation of symbolic dynamics for piecewise affine maps based on covering relations

Given a topological partition $\mathcal{Q}$, we wish to compute an approximation to the symbolic dynamics $\Sigma(f, \mathcal{Q})$. The simplest way to extract symbolic dynamics

is directly via covering relations, using Propositions 1.1.2 and 1.1.3. Since using $\mathcal{Q}$ directly only yields a coarse approximations to the symbolic dynamics, we use a refinement $\mathcal{R}$ of $\mathcal{Q}$, obtaining a finite-type shift approximating $\Sigma(f, \mathcal{R})$ and a sofic shift approximating $\Sigma(f, \mathcal{Q})$.

While in one-dimension, the under-approximations obtained using the covering relations $f(R_i) \supset R_j$ are good, in higher dimensions they usually yield no useful information. However, the over-approximations to the symbolic dynamics $\Sigma(f, \mathcal{Q})$ based on the covering relations $f(R_i) \cap R_j \neq \emptyset$ are useful.

Definition 3.3.1 (Upper symbolic dynamics)**.** *The* upper symbolic dynamics $\Upsilon(f, \mathcal{Q})$ *consists of all* $\overline{\mathcal{Q}}$*-itinaries of* f*.*

The approximation $\Upsilon(f, \mathcal{Q}, \mathcal{R})$ *to* $\Sigma(f, \mathcal{Q})$ *is define by* $\vec{s} \in \Upsilon(f, \mathcal{Q}, \mathcal{R})$ *if there exists a sequence* (t_i) *such that* $R_{t_i} \subset Q_{s_i}$ *and* $f(\overline{R}_{t_i}) \cap \overline{R}_{t_{i+1}} \neq \emptyset$ *for all* i*.*

It can be easily shown that

$$\Upsilon(f, \mathcal{Q}) \subset \Upsilon(f, \mathcal{Q}, \mathcal{R}),$$

and that $\Upsilon(f, \mathcal{Q}, \mathcal{R})$ converges to $\Upsilon(f, \mathcal{Q})$ as the partition $\mathcal{R}$ becomes arbitrarily fine.

Algorithm 1 computes Υ by computing the covering relations. The output is a matrix which represents the sofic shift.

Algorithm 1 compute_symbolic_dynamics_via_covering_relations

INPUT:
•f # function
•$\mathcal{Q} = \{Q_1, ..., Q_k\}$ # topological partition of the state space
•$\mathcal{R} = \{R_1, ..., R_n\}$ # refinement of $\mathcal{Q}$
OUTPUT:
•Upper symbolic dynamics $\Upsilon(f, \mathcal{Q}, \mathcal{R})$
for i, j **in** $[1, ..., n]$ **do**
  $\{\Upsilon\}_{ij} \leftarrow 0$
  if $f(\overline{R}_i) \cap \overline{R}_j \neq \emptyset$ **then**
    $\{\Upsilon\}_{ij} \leftarrow 1$
  end if
end for
return Λ,Υ

As we saw in the previous chapter covering relations combined with refinement of partitions make an effective method for computing arbitrarily

accurate lower and upper approximation of the symbolic dynamics of maps in one dimension. In two and higher dimension there are some obstacles in using covering relations which do not allowed to obtain such results for lower approximations.

First of all - unlike the one dimensional case where any partition whose members the map is monotone on is generating - there are no general results about finding a generating partition or deciding whether some partition is generating or not.

A second issue is that unlike the case in one dimensions [46] in general due to lack of any cycle of full coverings of elements of some partition, the lower approximation of the symbolic dynamics is the empty set.

Despite these limitations in two and higher dimensions the covering relations are still useful general approach for computing the over approximations of the symbolic dynamics of some partition. The finer is the partition $\mathcal{R}$ the better is the approximation. We will see in Section 3.5 how the Conley index decomposition will provide a natural partition of some invariant set which can arbitrarily improved.

3.4 Algorithms for computation of symbolic dynamics for piecewise affine maps based on tangles of fixed points

In Chapter 1 we introduced the definition of Tangle as a structure of intersections between the stable and unstable manifolds of one or more fixed points. There is already a well developed theory and software for computation of lower approximation of symbolic dynamics from tangles.

In this section we present algorithms which deals with piecewise affine maps. The algorithm which compute the symbolic dynamics from the combinatorial information of intersection of the trellis take as input a list of tuple organizes as in Data Type 2. For instance the corresponding entry for the intersection q_2 in Figure 1.1 is $[q_2, r_2, r_2, -, q_3]$.

Algorithm 3 performs the computation of the the two manifolds which form the trellis.

Computing intersections among the manifolds can be done by computing intersections of the segments of the manifolds. It is natural to order the intersections according the ordering of the segments of a manifolds, and this make straightforward computing for some intersection the next intersection on a manifold. The image of an intersection can be computed checking the image or preimage of the segment where the intersection lies. Finally the

Data Type 5.1 Trellis Intersection

OBJECT FIELDS:
•id # integer label
•next_unstable # label of the next intersection on the unstable manifold
•next_stable # pointer to the next intersection on the stable manifold
•orientation # orientation of the intersection, + or −
•image # pointer to the image of the intersection

orientation of an intersection can be computed from the angle between two manifolds at the intersection as the orientation of a cross product between vectors.

The theory underling the tangle algorithm provides a lower approximation of the symbolic dynamic in term of regions of the tangle by proving that the mapping among some curves inside the regions prove the existence of periodic orbit (confront Theorem 10 in [17]). Such mapping among curves can be computed by the topological relations among intersection of the trellis.

We refer the reader to [18] and [17] for a detailed explanation of the algorithm and the underlying theory.

3.5 Algorithms for computation of symbolic dynamics for piecewise affine maps based on decomposition of Conley index pair

In this section we present an approach to compute an index pair for a piecewise-affine map relative to the invariant set containing the bounded dynamic. We then present an algorithm to compute a set of linear independent homology generators and computing the symbolic dynamics.

3.5.1 Computing forward and backward diverging sets

In our study we restrict our attention to piecewise-affine maps whose chaotic dynamics is bounded, we are then interested in computing an over approximation of the set of chaotic dynamics.

In order to perform this task we first compute forward and backward diverging sets.

Definition 3.5.1 (Diverging sets)**.** *A set F is a* forward diverging set *of f if*

Algorithm 3 computation_of_stable_and_unstable_curves

INPUT:
•f # Piecewise affine function $f(x) = f_i(x) = A_i x + b_i \quad x \in P_i, i : 1, \ldots, n$
•m # $m \in \{1, .., n\}$ region of f whose fixed point we compute the curve
•k # number of steps we perform the approximation
OUTPUT:
•S,U # stable curve S, unstable curve U of the fixed point of f in P_i. S and U are expressed
as list of segments, each segment expressed as a list of two points in $\mathbb{R}^2$

$U \leftarrow$ [**Unstable_manifold**$(A_m x + b_m) \cap P_m$]
$S \leftarrow$ [**Stable_manifold**$(A_m x + b_m) \cap P_m$]
new_segments_unstable $\leftarrow U$
new_segments_stable $\leftarrow S$
buffer $\leftarrow$ []
for i $\in [1, ..., k]$ **do**
  for s $\in$ new_segments_stable **do**
    for j $\in [1, ..., n]$ **do**
      new_segment$\leftarrow f(s \cap f^{-1}(P_j))$
      $U \leftarrow U$+[new_segment]
      buffer $\leftarrow$ buffer+[new_segment]
    end for
  end for
  new_segments_unstable $\leftarrow$ buffer
  buffer $\leftarrow$ []
  for s $\in$ new_segments_unstable **do**
    for j $\in [1, ..., n]$ **do**
      new_segment $\leftarrow f^{-1}(s \cap f(P_j))$
      $S \leftarrow S$+[new_segment]
      buffer $\leftarrow$ buffer+[new_segment]
    end for
  end for
  new_segments_stable $\leftarrow$ buffer
  buffer $\leftarrow$ []
end for
return S, U

- $\forall x \in F : \forall n \in \mathbb{N},\ f^n(x) \in F,\ \textit{and}\ \lim_{n \to +\infty} ||f^n(x)|| = \infty$.

A set B is a backward diverging set *of f if*

- $\forall x \in B : \forall n \in \mathbb{N},\ f^{-n}(x) \subset B,\ \textit{and}\ \lim_{n\to+\infty} \min_{z\in f^{-n}(\{x\})} ||z|| = \infty.$

We now present an approach to computing forward basins of divergence for piecewise-affine maps. The method relies on computing basins with given periodic itineraries. Although we do not prove convergence in general of the algorithm which implements the method, the method works in all cases we consider.

Definition 3.5.2 (Basin operators)**.** *We define two operators* FB *(forward basin of divergence) and* BB *(backward basin of divergence) which have as input a piecewise continuous function $f : \mathbb{R}^2 \longrightarrow \mathbb{R}^2$ and a subset S of $\mathbb{R}^2$.*

- $\mathrm{FB}(f, S) = \{x \in S \mid \forall n,\ f^n(x) \in S,\ \textit{and}\ \lim_{n\to+\infty} ||f^n(x)|| = \infty\}$
- $\mathrm{BB}(f, S) = \{x \in S \mid \forall n,\ f^{-n}(x) \subset S,\ \textit{and}\ \lim_{n\to+\infty} \min_{z\in f^{-n}(\{x\})} ||z|| = \infty\}$

The set $\mathrm{FB}(f, S)$ consists of all points such that the forward orbit remains in S and diverges to infinity. The set BB consists of all points such that all backwards orbits remain in S and diverge to infinity at a uniform rate.

The following result asserts that the basin operators $\mathrm{FB}(f, P)$ and $\mathrm{BB}(f, P)$ can be effectively computed for an affine (*not* piecewise-affine) map f and a polyhedral set P.

Proposition 3.5.1. *For an affine function $f : \mathbb{R}^2 \longrightarrow \mathbb{R}^2$ and a polyhedral set $P \subset \mathbb{R}^2$ the operators* $\mathrm{RFB}(f, P)$ *and* $\mathrm{BB}(f, P)$ *are computable. More precisely, for any such f and P it is possible to locally approximate the basin at desired accuracy by a polyhedron (proof in appendix).*

We use the basin operators for affine maps to construct an algorithm to compute forward and backward basins of divergence for piecewise affine maps in two dimensions. The strategy of the algorithm is to consider basins with periodic itineraries. It can turns out that there is no diverging set according one branch in any P_i, still the analysis can be extended to composition of two or more branches. We show this approach in Algorithm 4.

Let $\mathcal{F} = \{f_1, f_2, ..., f_n\}$ be a collection of functions $\mathbb{R}^2 \to \mathbb{R}^2$, let $\mathcal{P} = \{P_1, P_2, ..., P_n\}$ be a collection of sets, and let $l = (l_0, l_1, ..., l_{k-1}) \in \{1, 2, ..., n\}^k$. Define $f_{l_0,...,l_{k-1}} = f_{l_{k-1}} \circ \cdots \circ f_{l_1} \circ f_{l_0}$, taking $f_\epsilon = \mathrm{id}$, where ϵ is the empty string. Let

$$\begin{aligned} P_{l,\mathcal{F}} &= \textstyle\bigcap_{j=0}^{k} f^{-1}_{l_0,...,l_{j-1}}(P_{l_{j \bmod k}}) \\ &= \{x \in \mathbb{R}^2 : f_{l_{j-1}} \circ \cdots \circ f_{l_0}(x) \in P_{l_{j \bmod k}}\ \forall\, j = 0, \ldots, k\}. \end{aligned} \tag{3.2}$$

In other words, $P_l(\mathcal{F})$ is the set of points starting in P_{l_1} with initial itinerary $l_0, l_1, \ldots, l_{k-1}, l_0$.

We use Algorithm 4 to approximate basins of divergence. In the algorithm we want to make sure that computed basins of divergence are robust, more precisely we want to make sure they have the property $f(\mathrm{cl}(F)) \subset \mathrm{int}(F)$ and $f^{-1}(\mathrm{cl}(B)) \subset \mathrm{int}(B)$. For this purpose we introduce the operator *shrink* defined in this way: $F_2 =$ *shrink* (F_1) implies $F_2 \subset F_1$ and each side of F_2 is parallel to a corresponding side of F_1 shifted of a suitably small quantity δ as illustrated in Fig. 3.2. We can assign to the δ the smallest available value in our computational setting. According to the analysis of affine maps explained in Appendix C is trivial to see that the outputs of the operator *shrink* are still diverging sets and satisfies the two properties es $f(\mathrm{cl}(F)) \subset \mathrm{int}(F)$ and $f^{-1}(\mathrm{cl}(B)) \subset \mathrm{int}(B)$.

Algorithm 4 approximation_of_basins_of_divergence

INPUT:
•f # piecewise affine map as expressed in Formula 3.1
OUTPUT:
•F, B # basins of forward and backward divergence respectively
$k = 1$, $F = \emptyset$, $B = \emptyset$
while $F = \emptyset$ OR $B = \emptyset$ **do**
  for $l \in \{1...n\}^k$ **do**
    $F \leftarrow F \cup \mathrm{RFB}(f_l, P_l(f))$
    $B \leftarrow B \cup \mathrm{RFB}(f_l, P_l(f))$
  end for
  $k \leftarrow k + 1$
end while
$F \leftarrow \mathrm{shrink}(F)$
$B \leftarrow \mathrm{shrink}(B)$
return F, B

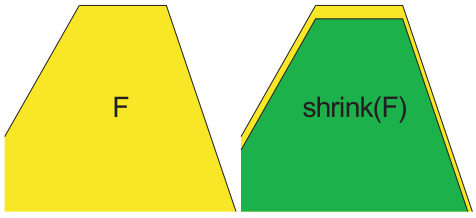


Figure 3.2: Shrink of a polyhedron

What Algorithm 4 does is finding regions of diverging orbits whose sym-

bolic itineraries in terms of the alphabet given by the partition $\{P_1, P_2, ...P_n\}$ of $\mathbb{R}^2$ are periodic. Regions corresponding to diverging orbits whose itineraries are not periodic can not be found by it and if such regions make the whole basin of diverging orbits or the diverging set is empty the algorithm does not halt. Even when it halts it might have computed only a subset of the whole basin of divergence. The results of the algorithm which relies on the computability of diverging sets for affine functions is then partial due to the many uncomputable or semicomputable problems related to piece-wise affine functions. The fact of being dense in the set of piecewise continuous function gives an insight of the complexity.

Despite these restrictions, even when the above algorithm fails to compute an approximation of the basin of divergence the following theory can still be applied using non general analytic techniques customized to the map which needs to be studied to find the diverging set of the map, such as finding simple properties which can be proved but not computed for some particular map.

The diverging sets we found in the previous section are (unbounded) polyhedra and we can work with them with the calculus of polyhedra we defined in Section 3.3.

3.5.2 Algorithms for computation of isolating neighbourhood and index pair

To compute an isolating neighbourhood we need to expand the basin of divergence by computing preimages of the forward diverging set and images of the backward diverging set at the stage that the intersections of the complementaries is bounded and to following stages in order to get a better approximation of the set where the chaotic dynamics takes place.

We note that if F and B are open forward and backward basins of divergence and $C := X \setminus (F \cup B)$ is bounded, then $\text{inv}(C)$ is an isolated invariant set. If F and B are closed, then $N := \text{cl}(X \setminus (F \cup B))$ is an isolating neighbourhood.

Proposition 3.5.2. *The output of Algorithm 5 is an approximation of an index pair under the assumption that* $X \setminus W = N = f^{-1}(f(N))$

Proof. Let us consider each of the three conditions of the definition of index pair:

1. $f(X \setminus W) \subset X$: because B is backward invariant, $N = f^{-1}(f(N))$ and $N \cap B = \emptyset$ we have that $f(N) \cap B = \emptyset$.

Algorithm 5 computation_of_index_pair

INPUT:
•f # piecewise affine #
•n # steps of refinement#
OUTPUT:
•(X, W) # Index pair set

$F, B \leftarrow$ **computation_of_basins_of_divergence_sets**(f)
for $i = 1$ to n **do**
  $F \leftarrow f^{-1}(F)$
  $B \leftarrow f(B)$
end for
$N \leftarrow F^C \cap B^C$
$W \leftarrow f(N) \cap F$
$X \leftarrow N \cup W$.
return (X, W)

2. $f(W) \cap X \subset W$: this is true because W is a subset of forward basin F which is forward invariant and does not intersect $N = X \setminus W$.

3. $\mathrm{cl}(X \setminus W)$ is an isolating neighbourhood for f : the bounded dynamics of f is in the interior of $N = X \setminus W$ by construction.

□

The index pair is defined by a pair of of polyhedra. We remark that in our work we deal with polyhedra by linear inequalities whose coefficients are in general irrational numbers, so there is no way to represent them exactly with floating points numbers in a computer. As we have explained in Section 3.2 we adopt interval approximation of real numbers to deal with numerical computation. The output is an approximation of an index pair and any operation on it is decidable up to some minimal accuracy.

3.5.3 Algorithms for computation of Conley index and symbolic dynamics

In order to compute the Conley index and so to obtain the symbolic dynamics we need to find for each connected component of the isolating neighbourhood a group of independent homology generators. First of all we need to compute the connected components of the polyhedron which approximate the invariant set; Algorithm 6 performs this task.

Algorithm 6 compute_connected_components

INPUT:
•N # isolating neighbourhood for f
OUTPUT:
•$[N_1, \ldots, N_k]$ # components of N

Set_of_components ← [ ]
for p ∈ N.convex_polyhedra **do**
  Set_of_components ← Set_of_components +[p]
end for
while Δ Set_of_components ≠ 0 **do**
  for C_i, C_j ∈ Set_of_components **do**
    for $p_k \in C_i$,$p_l \in C_j$ **do**
      if p_k share_boundary_with p_l **then**
        merge C_i, C_j
        jump next while
      end if
    end for
  end for
end while
return Set_of_components

On each connected component we set the homology generators with Algorithm 8 but before we need to define a new object to model homology generator which is a key for the computation of the Conley index and symbolic dynamics. This is given in Data Type 7. We start with a topological pair (P, Q) where P and Q are polyhedral sets, and let $N = \mathrm{cl}(P \setminus Q)$. Figure 3.5.3 shows the information related to an homology generator: the green set is N and the two red strips are the set Q. The *region* of α is the polyhedral set N.

The *poles* of the homology generator α are the one-dimensional faces containing the endpoints of α; these are always common faces of Q and N. The *segments* are the faces of the convex polyhedra which are crossed by α. This data type represents the class of equivalence of homology generators associated with this pair of poles. The operator selector return a member of this class, and it is represented in Fig. 3.5.3 as a dotted arrow.

After the setting of the homology generators we can compute the Conley index by computing the way the homology generators map each others, this is performed in Algorithm 9.

Data Type 7 Homology Generator

OBJECT FIELDS:

- region # a polyhedron where the generator can lies
- poles # segments which make the extremes of the homology generator
- segments # boundaries among convex polyhedra of the polyhedron

METHODS:

- constructor(region,poles,segments)
- image(f) # return constructor(f(region),f(poles), f(segments))
- selector # return an oriented polygonal curve connecting the poles lying in the interior of the polyhedron, which represents a possible choice of the generator
- covers($object_1$,$object_2$) # let v_1 and v_2 be the vectors which express $object_1$ and $object_2$ respectively in terms of a basis of the homology space (where each member is a generator whose selectors intersect the exit set only at the extremes), if for each component i $v_2[i] \leq v_1[i]$ return true, false otherwise. In other words every selector of $object_2$ is a sub curve of e a selector of $object_1$.
- compare_orientation($object_1$,$object_2$) # if $object_1$ covers $object_2$ or vice versa return 1 if they have the same orientation, -1 if they have opposite orientation.

The lower approximation of the symbolic dynamics corresponds to a cocyclic shift defined by a set of matrices, one for each connected component the index pair, each matrix has as non zero rows only the rows of the Conley index corresponding to the homology generators which are defined on the corresponding connected component.

3.5.4 Dealing with the discontinuous case

In this section we consider the case of a discontinuous map with discontinuity on the boundaries of the partition. Also in this case Algorithm 9 still

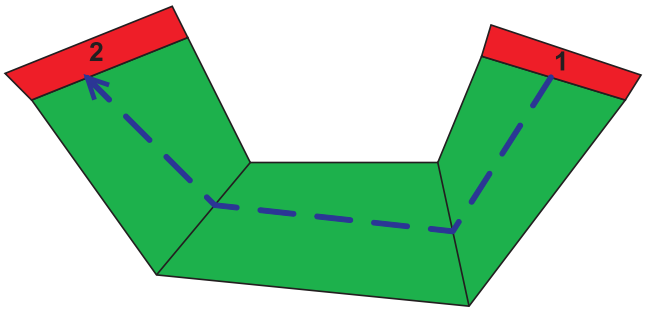


Figure 3.3: Homology generator and selector

Algorithm 8 compute_homology_generators

INPUT:
•Set_of_components # components of the topological pair
OUTPUT:
•homology_generators # list of homology generators
for C ∈ Set_of_components **do**
  Exit_sides ← $\partial C \cap \partial W$
  for i ∈ length(Exit_sides)-1 **do**
    s ← Exit_sides[i], t ← Exit_sides[$i+1$]
    Homology_generators ← Homology_generators + [Homology_generator (s,t, Shortest_Path(s,t))]
  end for
end for
return Homology_generators

Algorithm 9 computation_of_the_Conley_index

INPUT:
•Homology_generators
OUTPUT:
•CI # a matrix representing the Conley index

for h_i ∈ Homology_generators **do**
  for h_j ∈ Homology_generators **do**
    $\{CI\}_{ij} \leftarrow f(h_i).\text{covers}(h_j)$
  end for
end for
return CI

computes a correct over approximation of the set of the bounded dynamics.

The algorithm which computes the index pair produces an output which in this case can not be an index pair of the function because the index pair as the Conley index is defined only for continuous functions. Fortunately under some hypotheses we can approximate the function with a continuous function, then the output of Algorithm 5 is an index pair for such approximation and we can compute a lower approximation of the symbolic dynamics:

Theorem 3.5.3. *Let $N = X\backslash(F \cup B)$ be the isolating neighbourhood of an invariant set S of f computed by Algorithm 9 and D the set of discontinuity. Suppose N is simply connected. If*

$$N \cap D = \emptyset \text{ or } (\text{int}(N) \cap D = \emptyset \text{ and } f(\partial N \cap D \cap \partial F) \cap \text{int}(N) = \emptyset) \quad (3.3)$$

*then we can construct a continuous map f_c, which has the same value of f over N, and so the computation of the lower symbolic dynamics of f can be done computing the Conley index of f_c over N. The output of **computation_of_index_pair**(f) is an index pair for f_c.*

Define $R = \{x \in S \cap D : \exists i, j, \forall \epsilon I_\epsilon(x) \cap P_i \cap S \neq \emptyset$ and $I_\epsilon(x) \cap P_j \cap S \neq \emptyset\}$. Then condition (3.3) implies that $R = \emptyset$.

Proof. 1. Let us consider the set $\partial N \cap D$ and one of his connected component $C \subset \partial P_i \cap \partial P_j$. Without loss of generality let assume in any enough small neighbourhood of C $\partial P_j \cap \text{int}(N) = \emptyset$. Let K be a local half neighbourhood of D in P_j (which intersects the interior of P_j but not the interior of P_i) . We create a new function f_c which outside K coincides with f, on $\partial K \cap P_i$ coincides with f_i while on $\partial K \cap \text{int}(P_j)$ coincides with f_j and map each curve connecting $\partial K \cap P_i$ to $\partial K \cap \text{int}(P_j)$ to a line which does not intersect $\text{int}(N)$. This is possible because each components of N is simply connected. This means it is possible to continuously extend f in a neighbourhood of D in order to map such neighbourhood outside N.

So if $f_c(N) \cap D \neq \emptyset$ then $f_c(N) \cap D \subset Q$, but for what we said $f_c(Q \cap D) \cap N = \emptyset$, so Q respects the property of exit set for an index pair and the pair computed by **computation_of_index_pair**(f) is an index pair for f_c.

2. If a point of S has sequences of points of S converging to it on both sides of the discontinuity, any local over approximation of S intersect D in its interior.

□

The second condition is necessary but not sufficient for being able to extend our approach with a continuous function, because also when the condition holds in general it can be impossible to approximate S with the algorithm in such away to have $\text{int}(N) \cap D = \emptyset$ in finite steps.

3.6 Case studies

In this section we analyse a number of case studies, for each of them we compute the symbolic dynamics with the three different approaches: tangle, Conley index and covering relations and we compare the results.

3.6.1 Lozi Map

In this subsection we are going to show results for the Lozi map:

$$\begin{array}{rcl} x(k+1) & = & 1 - \alpha|x(k)| + y(k) \\ y(k+1) & = & \beta x(k) \end{array} \tag{3.4}$$

This piecewise affine functions was shown to have chaotic behaviour [47]. We start with the following value of the parameter: $\alpha = 2$, $\beta = 0.5$. Fig. 3.4 shows a trellis of the map.

The related symbolic dynamics is a sofic shift represented by the following matrices where the symbols corresponds to the regions in the same ordered as in Fig. 3.5 :

$$\left[\begin{array}{c|ccc} & R_1 & R_2 & R_3 \\ \hline R_1 & 0 & 0 & 1 \\ R_2 & 0 & 0 & 1 \\ R_3 & 1 & 1 & 0 \end{array}\right]$$

the entropy is $\frac{1}{2}\log 2 \backsimeq 0.346$.

While the over-approximation to the symbolic dynamic of the covering relations over the same regions is the following sofic shift over the same symbols:

$$\left[\begin{array}{c|ccc} & R_1 & R_2 & R_3 \\ \hline R_1 & 1 & 1 & 1 \\ R_2 & 1 & 1 & 1 \\ R_3 & 1 & 1 & 1 \end{array}\right]$$

the entropy is $\log 3 \backsimeq 1.098$

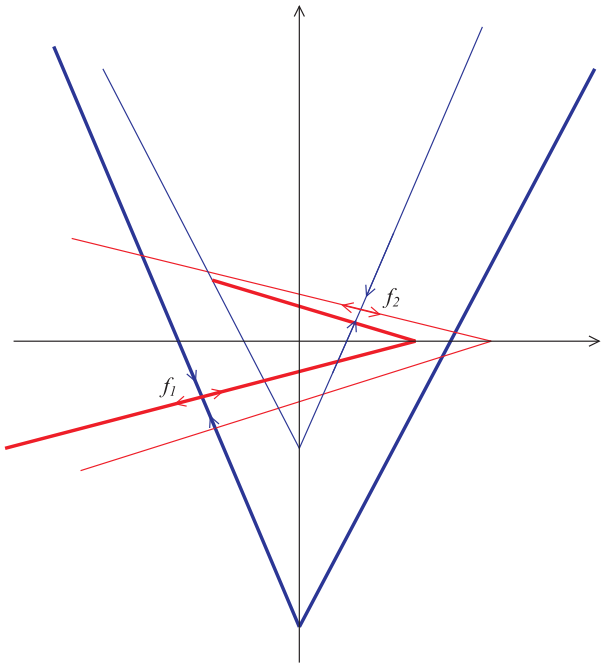


Figure 3.4: Unstable (red) and stable (blue) manifolds of the fixed points of the Lozi map with parameters $\alpha = 2$, $\beta = 0.5$

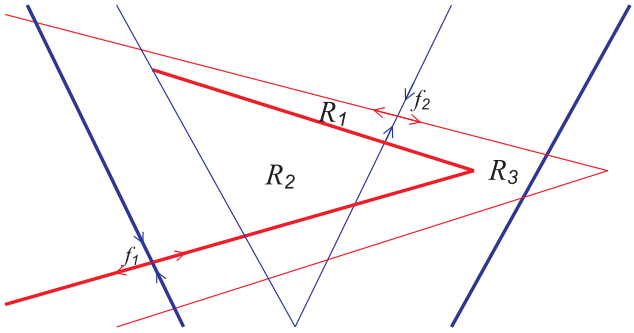


Figure 3.5: Expansion of manifolds of Lozi map with parameters $\alpha = 2$, $\beta = 0.5$

Fig. 3.6 shows an index pair of the map with the homology generators. Note that the regions $R_1, \ldots, R_4$ are different from regions of the trellis shown in Fig. 3.4.

We obtain the following Conley index, each group of rows corresponds to generators of one region:

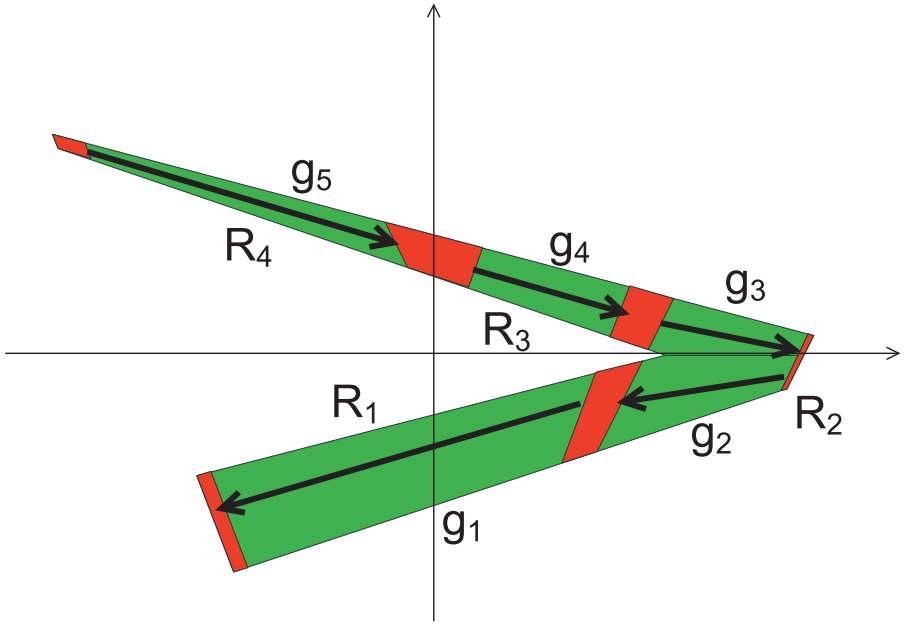


Figure 3.6: An index pair for the Lozi map with parameters $\alpha = 2$, $\beta = 0.5$, together with homology generators

	g_1	g_2	g_3	g_4	g_5
g_1	1	1	1	-1	0
g_2	0	0	0	0	-1
g_3	0	0	0	0	1
g_4	0	0	-1	1	0
g_5	1	1	0	0	0

The corresponding cocyclic has cancellations. Nevertheless we can compute an equivalent sofic shift in this way: we add an extra symbol corresponding to the linear dependent homology generator $g_2 + g_3$ in region R_2 and this gives the following matrix:

	R_1	R_2			R_3	R_4
R_1	1	0	0	1	1	0
R_2	0	0	0	0	0	1
	0	0	0	0	0	1
	0	0	0	0	0	0
R_3	0	0	1	0	1	0
R_4	1	1	0	0	0	0

The covering relation algorithm provides over approximation of the sym-

bolic dynamics in the following finite type shift:

$$\left[\begin{array}{c|cccc} & R_1 & R_2 & R_3 & R_4 \\ \hline R_1 & 1 & 1 & 1 & 0 \\ R_2 & 0 & 0 & 0 & 1 \\ R_3 & 0 & 1 & 1 & 0 \\ R_4 & 1 & 1 & 0 & 0 \end{array}\right]$$

The entropies of the two shifts are respectively:
$\ln\left(1/2 - 1/6\,d + 1/6\,\sqrt{3}\sqrt{\frac{6d\,\sqrt[3]{54+6\,\sqrt{129}}+d\left(54+6\,\sqrt{129}\right)^{2/3}-12\,d+18\,\sqrt[3]{54+6\,\sqrt{129}}}{d\,\sqrt[3]{54+6\,\sqrt{129}}}}\right) \backsimeq$
0.540 where $d = \sqrt{\frac{9\,\sqrt[3]{54+6\,\sqrt{129}}-3\left(54+6\,\sqrt{129}\right)^{2/3}+36}{\sqrt[3]{54+6\,\sqrt{129}}}}$ and $\ln\left(1/2+1/2\,\sqrt{3+2\,\sqrt{5}}\right) \backsimeq$ 0.624.

Now we show results for the values of the parameter: $\alpha = 2$, $\beta = -0.5$. Fig. 3.7 shows the trellis. The corresponding symbolic dynamics is represented by the following matrix where the symbols corresponds to the regions of Fig. 3.8:

$$\left[\begin{array}{c|cccc} & R_1 & R_2 & R_3 & R_4 \\ \hline R_1 & 0 & 1 & 0 & 0 \\ R_2 & 0 & 0 & 1 & 1 \\ R_3 & 0 & 1 & 0 & 0 \\ R_4 & 1 & 1 & 1 & 0 \end{array}\right]$$

the entropy is

$$\ln\left((1/3)\,\sqrt[3]{27+3\,\sqrt{78}}+\frac{1}{\sqrt[3]{27+3\,\sqrt{78}}}\right) \backsimeq 0.419.$$

The symbolic dynamics of the covering relations over the same regions is the following sofic shift:

$$\left[\begin{array}{cccc} 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{array}\right].$$

The entropy is

$$\ln\left(1/3\,\sqrt[3]{37+3\,\sqrt{114}}+7/3\,\frac{1}{\sqrt[3]{37+3\,\sqrt{114}}}+1/3\right) \backsimeq 0.819.$$

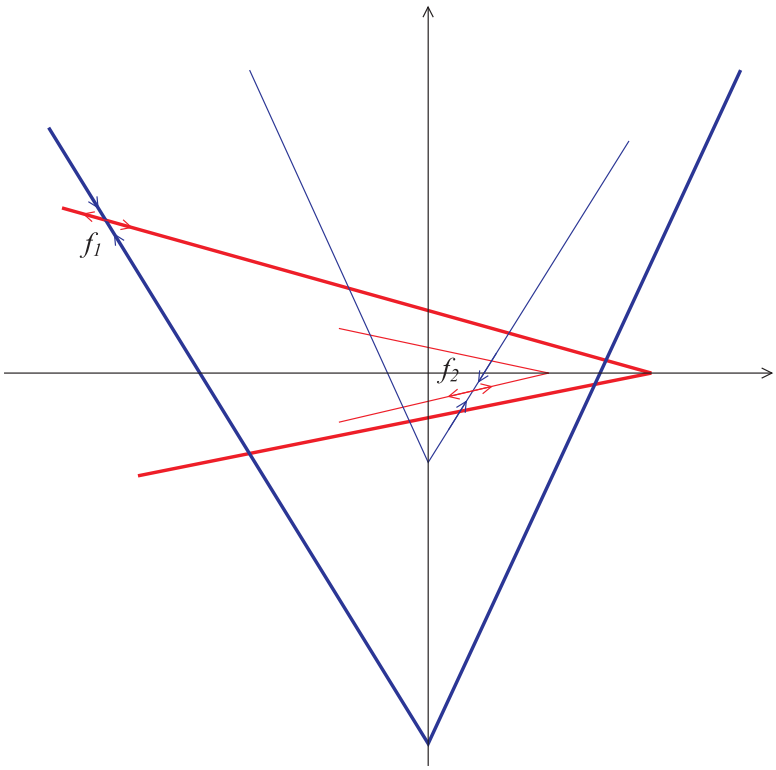


Figure 3.7: Stable and unstable manifolds of the fixed points of the Lozi map with parameters $\alpha = 2$, $\beta = -0.5$

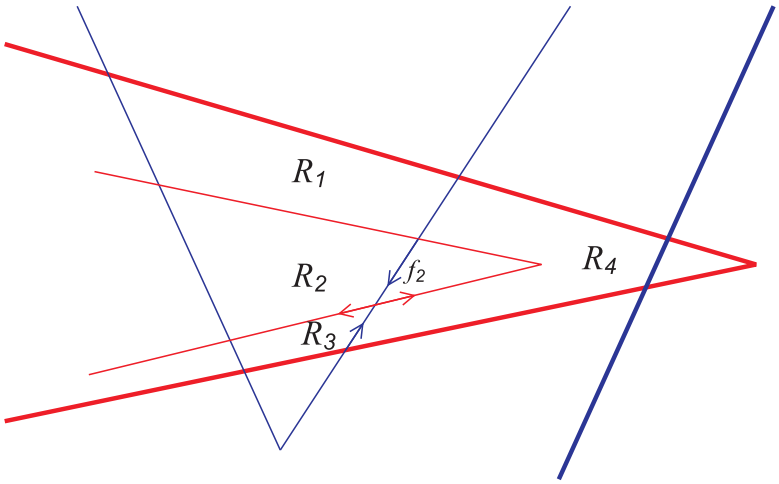


Figure 3.8: Expansion of manifolds of Lozi map with parameters $\alpha = 2$, $\beta = -0.5$

Fig. 3.9 shows the index pair. The Conley index is:

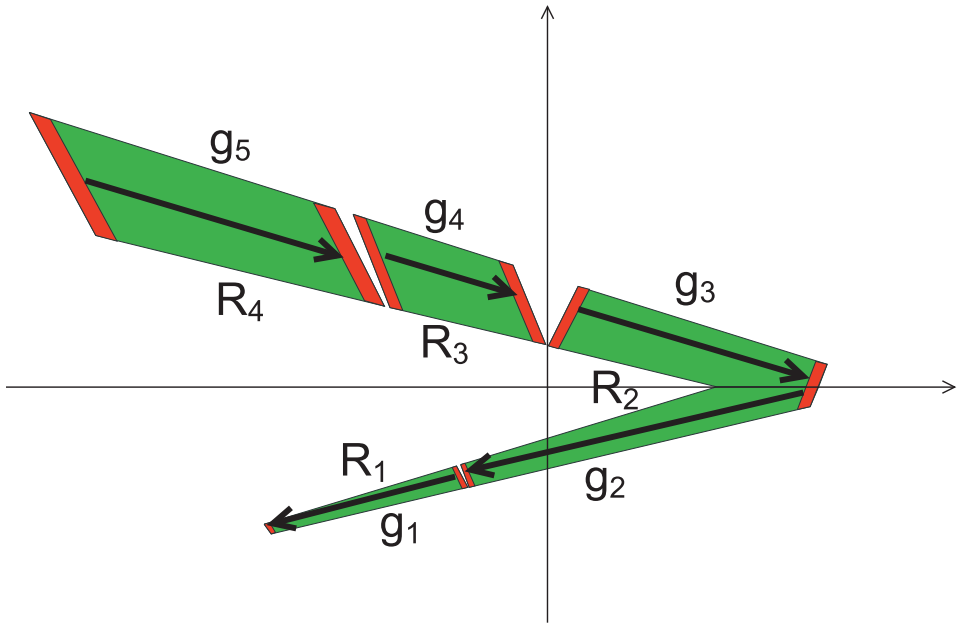


Figure 3.9: Index Pair of Lozi map with parameters $\alpha = 2$, $\beta = -0.5$ with homology generators

	g_1	g_2	g_3	g_4	g_5
g_1	0	0	0	1	-1
g_2	-1	1	-1	0	0
g_3	1	-1	0	0	0
g_4	0	0	1	0	0
g_5	0	0	0	-1	1

The cocyclic shift associated to the index does not have cancellation, so the symbolic dynamic is the sofic shift:

	R_1	R_2		R_3	R_4
R_1	0	0	0	1	1
R_2	1	1	1	0	0
	1	1	0	0	0
R_3	0	0	1	0	0
R_4	0	0	0	1	1

This is also the shift obtained by covering relation algorithm over the same regions, so the symbolic dynamics of the map over these regions is

exactly this shift. The entropy of the shift and the map is :

$$\ln\left(1/6\sqrt[3]{100+12\sqrt{69}}+2/3\,\frac{1}{\sqrt[3]{100+12\sqrt{69}}}+2/3\right)\backsimeq 0.562$$

From these two examples we can conclude that for a comparable number of steps the Conley index approach provide a much better lower approximation of the topological entropy and symbolic dynamics than the tangle approach, in the second case provide even the exact value. The covering relations can be used with either methods to provide an upper bound to both symbolic dynamics and entropy with the relative partition. The partitions provided by the two different approaches trellis, and Conley Index are different in general and we notice that in the case studies the partition of Conley index approach provides a better approximation of entropy and symbolic dynamics.

3.6.2 Discontinuous Lozi Map

In this subsection we study a discontinuous map:

$$\begin{array}{rcl} x(k+1) & = & -\operatorname{sgn}(x)-|x(k)|+y(k) \\ y(k+1) & = & \beta x(k)+1 \end{array} \tag{3.5}$$

The Conley Index is:

$$\left[\begin{array}{c|ccccc} & g_1 & g_2 & g_3 & g_4 & g_5 \\ \hline g_1 & 1 & 1 & 0 & 0 & 0 \\ g_2 & 0 & 0 & 1 & 0 & 0 \\ g_3 & 0 & 0 & 0 & 1 & 1 \\ g_4 & 0 & 0 & -1 & 0 & 0 \\ g_5 & -1 & -1 & 0 & 0 & 0 \end{array}\right]$$

Because each region has one and only open generator there can not be cancellation and so the lower approximation of the symbolic dynamics is expressed by the following shift:

$$\left[\begin{array}{c|ccccc} & R_1 & R_2 & R_3 & R_4 & R_5 \\ \hline R_1 & 1 & 1 & 0 & 0 & 0 \\ R_2 & 0 & 0 & 1 & 0 & 0 \\ R_3 & 0 & 0 & 0 & 1 & 1 \\ R_4 & 0 & 0 & 1 & 0 & 0 \\ R_5 & 1 & 1 & 0 & 0 & 0 \end{array}\right]$$

The lower approximation of the entropy is : $\ln\left(1/2\sqrt{5}+1/2\right)\backsimeq 0.481$

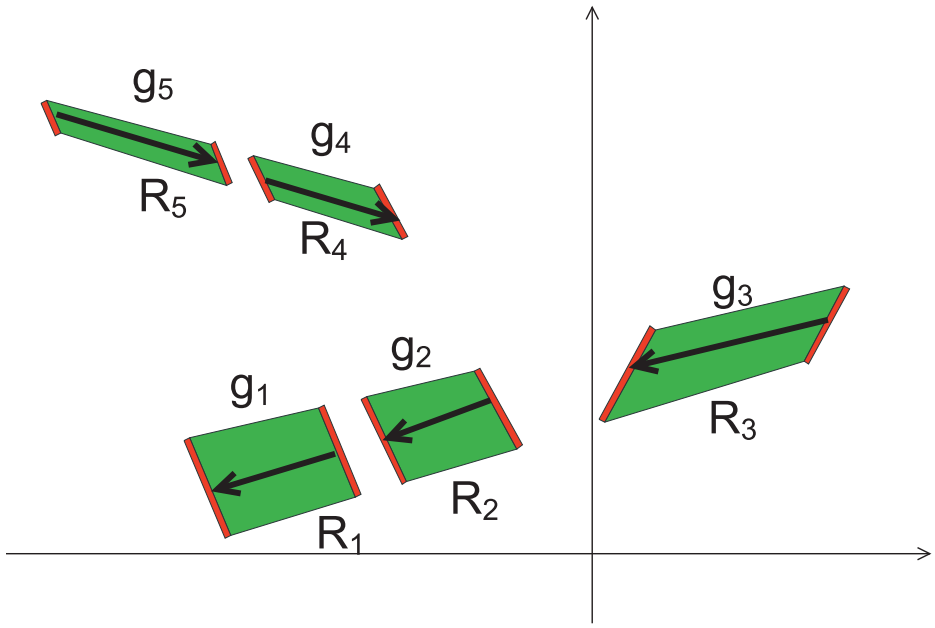


Figure 3.10: Index pair of discontinuous Lozi-like map with homology generators

Because the covering relation algorithm produces the same shift we conclude that these are respectively the exact symbolic dynamics and entropy of the map in terms of the regions in Fig. 3.10. It would have not be feasible to study this map with the tangle approach.

3.6.3 The study of an hybrid system via return map

In this section we study the hybrid system in Fig. 1, the dynamic in each mode is affine and the switch conditions are defined by affine equation and inequalities.

The values of the parameters are:

$\mathbf{u^T} = [1/3, 2/3, 1/3]$,$\mathbf{v_1^T} = [0, -1, 0]$, $\mathbf{v_2^T} = [0, 2, 0]$, $\mathbf{w^T} = [0, 0, 1]$

$$\mathbf{A} = \begin{bmatrix} -1/3 & 0 & 2/3 \\ 0 & -1 & 0 \\ 4/3 & 0 & 1/3 \end{bmatrix} + \begin{bmatrix} 1 & 1/2 & -1/2 \\ -1 & -1/2 & 1/2 \\ -1 & -1/2 & 1/2 \end{bmatrix} \log(3)$$

$$\mathbf{a} = \begin{bmatrix} -5/3 \\ 1 \\ -7/3 \end{bmatrix} + \begin{bmatrix} -3 \\ 3 \\ 3 \end{bmatrix} (\log 3/2)^{-1}$$

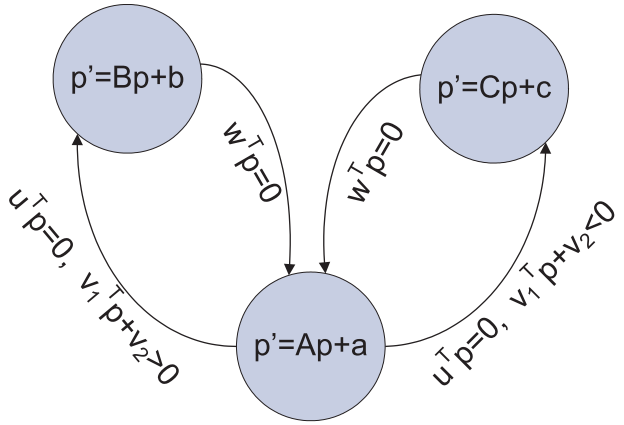


Figure 3.11: Hybrid system

$$\mathbf{B} = \begin{bmatrix} 1 & 8 & 0 \\ 0 & -1 & 0 \\ \frac{1+\log(3)}{5} & \frac{24-16\log(3)}{5} & -\log(3) \end{bmatrix}$$

$$\mathbf{b} = \begin{bmatrix} -23/2 \\ 1 \\ \frac{5-11\ \log(3)}{2\log(3)} \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} 1 & 8/3 & 0 \\ 0 & -1 & 0 \\ \frac{1+\log(3)}{5} & \frac{-16+8\log(3)}{5} & -\log(3) \end{bmatrix}$$

$$\mathbf{c} = \begin{bmatrix} -1/3 \\ -1 \\ \frac{3-5\ \log(3)}{3\log(3)} \end{bmatrix}$$

where all the logarithm are in base 2. The coefficients of the matrices of the modes above and the guard sets below are intentionally chosen in order to consider a piecewise affine return map. We consider the return map defined on the guard set $\mathbf{w^T p} = 0$. Fig. 2 shows that the return map maps a point p_1 to a point p_3, the corresponding orbit cross the other guard set at the point p_2

The analytic computation of the return for this system is simple because of the property that the mode $\dot{\mathbf{p}} = \mathbf{Ap} + \mathbf{a}$ maps every point of the guard set $\mathbf{w^T p} = 0$ to a point of guard set $\mathbf{u^T p} = 0$ after time $t = \ln(2)$ and the

modes $\dot{\mathbf{p}} = \mathbf{Bp} + \mathbf{b}$ and $\dot{\mathbf{p}} = \mathbf{Cp} + \mathbf{c}$ is map every point of the corresponding half of the guard set $\mathbf{u^T p} = 0$ to a point of the guard set $\mathbf{w^T p} = 0$ adter time $t = \ln(2)$.

By the simple fact that the solution of the equation $\dot{\mathbf{p}} = \mathbf{Fp} + \mathbf{f}$ with $\mathbf{p}(0) = \mathbf{p}_0$ is $\mathbf{p}(t) = e^{\mathbf{F}t}\mathbf{p}_0 - \mathbf{F}^{-1}\mathbf{f}$ we can express the integral map between guard set from $\mathbf{w^T p} = 0$ to $\mathbf{u^T p} = 0$ as:

$$\mathbf{M_1 p} + \mathbf{m_1} = e^{\mathbf{A}\ln(2)}\mathbf{p} - \mathbf{A}^{-1}\mathbf{a}$$

from $\mathbf{u^T p} = 0, \mathbf{v_1^T p} + \mathbf{v_2^T} \geq 0$ to $\mathbf{w^T p} = 0$ as :

$$\mathbf{M_2 p} + \mathbf{m_2} = e^{\mathbf{B}\ln(2)}\mathbf{p} - \mathbf{B}^{-1}\mathbf{b}$$

from $\mathbf{u^T p} = 0, \mathbf{v_1^T p} + \mathbf{v_2^T} \leq 0$ to $\mathbf{w^T p} = 0$ as :

$$\mathbf{M_3 p} + \mathbf{m_3} = e^{\mathbf{C}\ln(2)}\mathbf{p} - \mathbf{C}^{-1}\mathbf{c}$$

we obtain the following expressions for these matrices:

$$\mathbf{M_1} = \begin{bmatrix} 2 & 1/2 & 0 \\ -1 & 0 & 1/2 \\ 0 & -1/2 & 2 \end{bmatrix} \quad \mathbf{m_1} = \begin{bmatrix} 4 \\ -2 \\ 0 \end{bmatrix}$$

$$\mathbf{M_2} = \begin{bmatrix} 2 & 6 & 0 \\ 0 & 1/2 & 0 \\ 1/3 & 2/3 & 1/3 \end{bmatrix} \quad \mathbf{m_2} = \begin{bmatrix} 7/2 \\ 1 \\ 0 \end{bmatrix}$$

$$\mathbf{M_3} = \begin{bmatrix} 2 & 2 & 0 \\ 0 & 1/2 & 0 \\ 1/3 & 2/3 & 1/3 \end{bmatrix} \quad \mathbf{m_3} = \begin{bmatrix} -3 \\ 1 \\ 0 \end{bmatrix}$$

The return map is a piecewise affine maps whose pieces are $\mathbf{M_2 M_1 p} + \mathbf{M_2 m_1} + \mathbf{m_2}$ and $\mathbf{M_3 M_1 p} + \mathbf{M_3 m_1} + \mathbf{m_3}$.

The expression of such map on the set $\mathbf{w^T p} = 0$ is the following :

$$\begin{bmatrix} 2 & 1 \\ -0.5 & 0 \end{bmatrix} \begin{bmatrix} |\,\mathrm{x}\,| \\ \mathrm{y} \end{bmatrix} + \begin{bmatrix} -0.5 \\ 1 \end{bmatrix}$$

this map is very similar to the Lozi map being the difference only in the constant term.

Symbolic dynamics and discrete abstraction

We now apply our approach by computing the index pair and the Conley index. Fig 3.13 show the basin of forward and backward divergence respectively. We than compute and index pair by computing iteratively backward

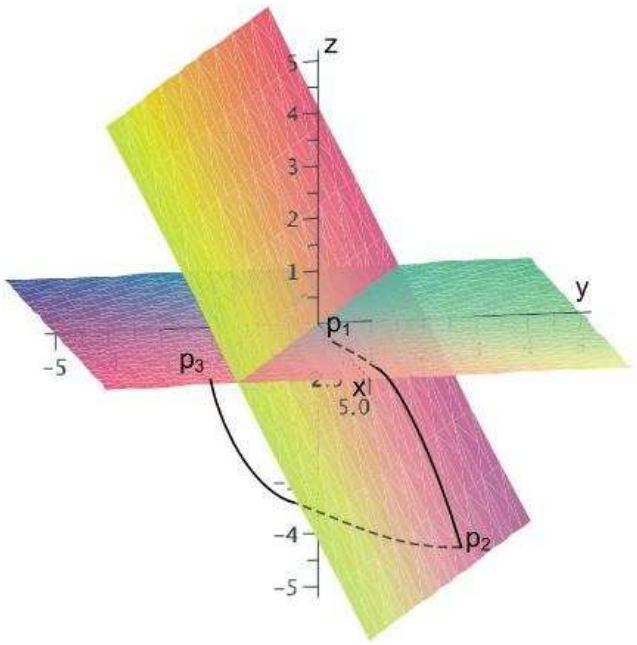


Figure 3.12: Return map

and forward images respectively of the two basins. The index pair we obtained is shown in Fig. 3.14.

The discontinuity, the line $x = 0$ in this case, does not intersect the interior of $P \backslash Q$. Setting the homology generator is straightforward because each connected component has only two exit sides, so there is only one homology generator for each component.

We then obtain the following Conley index:

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 \end{bmatrix}$$

Because each connected component of the index pair has only one homology generator the decomposition of the Conley index on each region corresponds exactly to one row of the matrix, then there are are no cancellation, the shift is sofic and can be expressed by the following matrix where we have inverted the sign of negative entries of the Conley index:

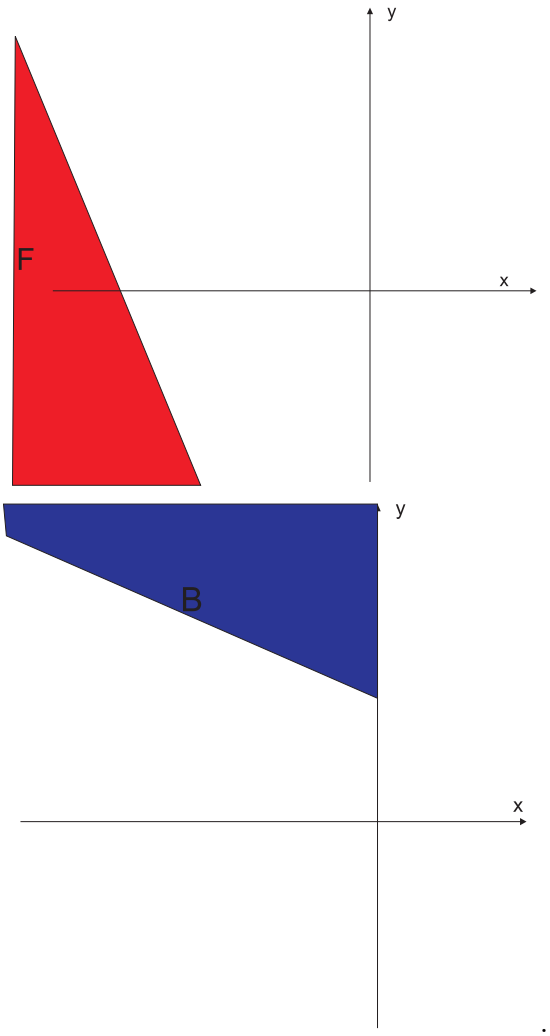


Figure 3.13: Basins of forward and backward divergence

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{bmatrix}$$

The sofic shift is represented in Fig. 3.15 where yellow node correspond to components whose trajectories switch from mode a to c while the blue

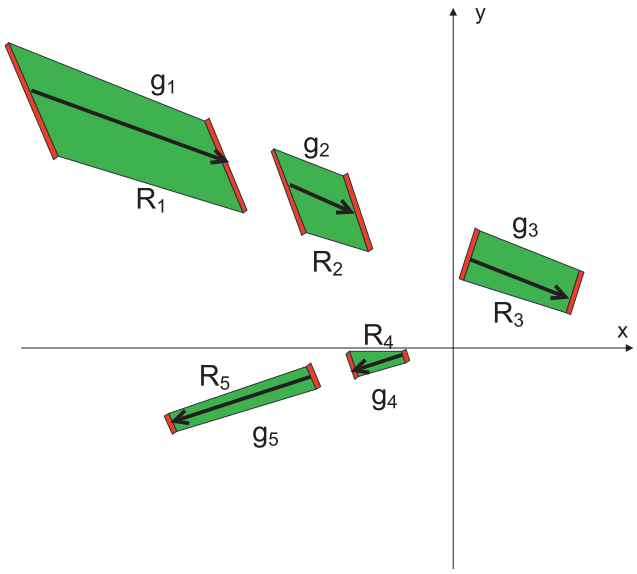


Figure 3.14: Index pair

node correspond to trajectory which switch from mode a to b.

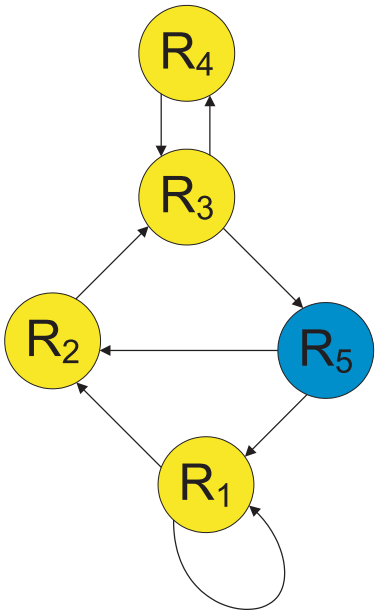


Figure 3.15: Shift space

The entropy of the shift can be computed as the logarithm of the Frobenius eigenvalue of the matrix which is about 0.481. This means that the return map is chaotic and so also the hybrid system.

We finally compute lower approximation of the discrete abstraction which can be deduced from the symbolic dynamics, this information is represented in Fig. 3.16. We finally compute lower approximation of the discrete abstraction which can be deduced from the symbolic dynamics, this information is represented in Fig. 3.16 which can be simplified in the representation in Fig. 3.17.

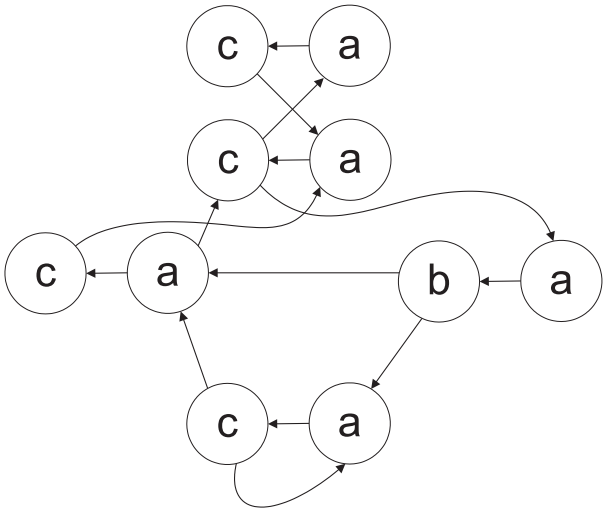


Figure 3.16: Discrete abstraction

3.7 Final remarks

In this chapter we have tackled the problem of formulating a general approach for the computation of symbolic dynamics of piecewise affine two dimensional maps. This class of maps are relatively simple but interesting because dense in the set of piecewise continuous maps. We developed algorithms to compute the tangle of piecewise affine map and exploiting the theory and already available software for computation of symbolic dynamics.

We have implemented a way to over approximate with arbitrary accuracy the bounded dynamic set where the chaotic dynamics takes place. We developed and implemented a group of algorithms for a series of intermediate tasks: computation over approximation of the set of the chaotic dynamic, computa-

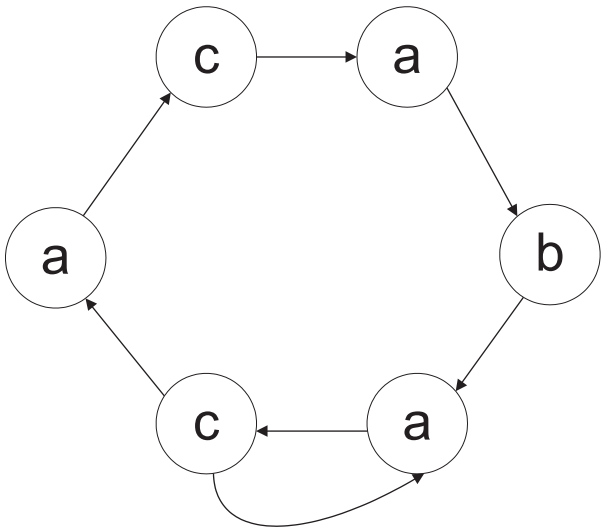


Figure 3.17: Sipmlified discrete abstraction

tion of the index pair, computation of set homology generators, computation of the Conley index and finally computation the symbolic dynamics.

The approach we developed provides results which for comparable number of steps have performances at least as good or better than the approach of the tangle. This approach also deals with discontinuous maps, important for the studies of hybrid systems which has in general discontinuous return maps. The whole work provides a contribution for several issues related to discrete time systems such as: chaos detection, expression of itineraries in terms of regions, approximation of chaotic attractors or repellers, computation of topological entropy, analysis of flows and hybrid systems via return maps.

We show how the algorithms work on the Lozi map, and a discontinuous variant of the Lozi map.

Finally a simple analysis of the whole approach shows that nothing is constraint to two dimensions as in the case of the tangle approach. We make the following considerations: finding invariant and diverging set of linear maps is not significantly more difficult for higher dimension once Jordan form of the matrix of the system has been computed. The calculus of polyhedral complexes is rigorously possible also for higher dimension. Computation of homology generators can be modelled by pairs of polyhedra also in higher dimension.

Chapter 4

Uncomputability of discrete dynamics of hybrid systems in 3D

4.1 Introduction

In the paper [48] Misiurewicz constructed a sequence of non-chaotic diffeomorphism which converges to a chaotic diffeomorphism in C^∞-topology. The minimal dimension for which this result applies is three; it was already proved that there is no such sequence in two dimensions.

In this chapter we describe a sequence of hybrid systems in three dimensions and related return map, which are smooth non invertible-maps in two dimensions and we show that maps are non-chaotic while the limit is chaotic and from this result we are able to deduce that the discrete abstraction of the hybrid system of the sequence is non-chaotic while the discrete abstraction of the limit is chaotic.

4.2 Non-convergence in Entropy

A sequence of hybrid systems

Let us consider in $\mathbb{R}^3$ a system of cylindrical coordinates ρ, θ, z where for simplicity θ ranges over $[0, 1]$ instead of usual $[0, 2\pi]$ (i.e. the value of a full turn angle is 1 instead of 2π).

We consider the system in Fig 4.1, the dynamics of the modes are:
$F_1(\rho, \theta, z) = (0, 0, z)$
$F_2(\rho, \theta, z) = (0, 0, -z)$

$F_3(\rho, \theta, z) = (0, 0, -z)$

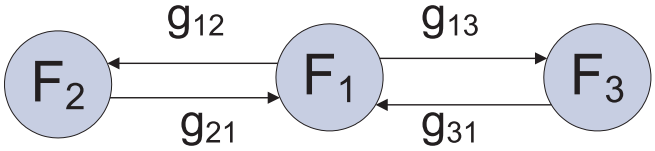


Figure 4.1: hyrbid automaton

Let us considere the following sets:

$S_1 = \{z = 1\}$
$S_2 = \{z = e, \rho < 1/2\}$
$S_3 = \{z = e, \rho > 1/2\}$

and let us consider the following switching conditions:
$g_{12} : z = e, \rho < 1/2$
$g_{21} : z = 1$
$g_{13} : z = e, \rho > 1/2$
$g_{31} : z = 1$

The system switches from mode 1 to mode 2 at S_2 and from mode 1 to mode 3 at S_3, in both cases resetting the state according the following reset map: $r_n(\rho, \theta) = [r^\rho(\rho, \theta), r_n^\theta(\rho, \theta)] : S_2 \cup S_3 \longrightarrow S_2 \cup S_3$:

$$r^\rho(\rho, \theta) = \begin{cases} 2\rho + 2\theta - 1 & \text{if } \rho < 1/2 \wedge \theta < 1/2 \wedge \rho + \theta > 1/2 : \\ -2\rho + 2\theta + 1 & \text{if } \rho < 1/2 \wedge \theta > 1/2 \wedge \theta - \rho < 1/2 \\ 2\rho - 2\theta + 1 & \text{if } \rho > 1/2 \wedge \theta < 1/2 \wedge \rho - \theta < 1/2 \\ -2\rho - 2\theta + 3 & \text{if } \rho > 1/2 \wedge \theta > 1/2 \wedge \theta + \rho < 3/2 \\ 0 & \text{otherwise} \end{cases}$$

$$r_n^\theta(\rho, \theta) = \begin{cases} 0 & \text{if } \theta \in [0, 0.25] \\ (2 - 2^{2-n})\theta - (2^{-1} - 2^{-n}) & \text{if } \theta \in [0.25, 0.5] \\ -(1 - 2^{1-n})\theta + 3(2^{-1} - 2^{-n}) & \text{if } \theta \in [0.5, 0.75] \\ 0 & \text{if } \theta \in [0.75, 1] \end{cases}$$

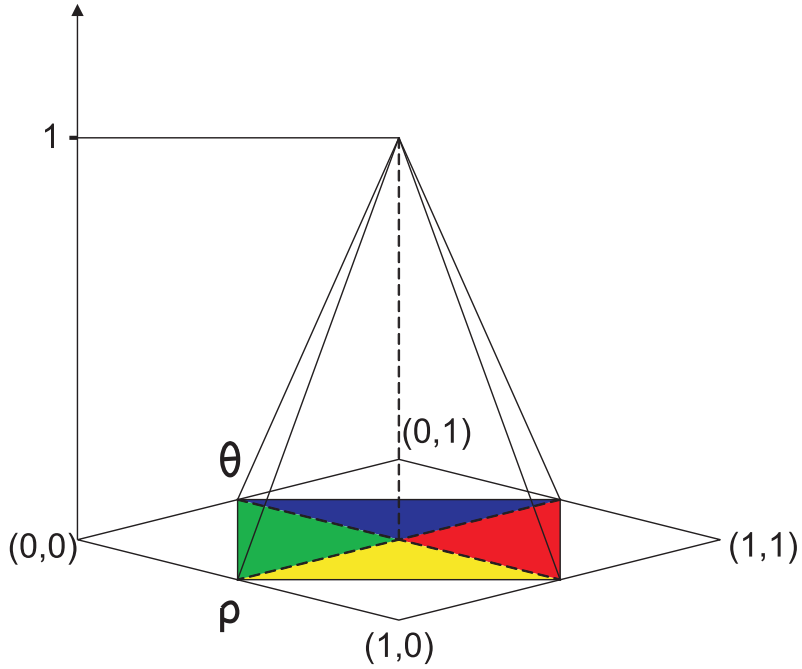


Figure 4.2: The function r^{ρ}

The system switches from mode 3 and 2 to mode 1 at S_1 without resetting. It is easy to see that the reset map is also the return map at the section $S_2 \cup S_3$.

The limit $\lim_{n\to\infty} r_n^{\theta} = r^{\theta}$ is:

$$r^{\theta}(\rho,\theta) : \begin{cases} 0 & \text{if } \theta \in [0, 0.25] \\ 2\theta - 2^{-1} & \text{if } \theta \in [0.25, 0.5] \\ -\theta + 3(2^{-1}) & \text{if } \theta \in [0.5, 0.75] \\ 0 & \text{if } \theta \in [0.75, 1] \end{cases}$$

For all n the return map is not chaotic beacause the set of non-wandering points in S_1 is $[0,1] \times \{1\}$ and all points on such set are fixed points. Instead the limit is chaotic because r^{θ} has a fixed point at $1/2$ and $r_{\infty}(\rho, 1/2) : [0,1] \times \{1/2\} \longrightarrow [0,1] \times \{1/2\}$ is the tent map.

Because of this fact we can deduce that the discrete dynamics of the hybrid system has only sequences eventually ending with the sequence $\{12\}^{\omega}$, while for the limit the discrete dynamics is $\{12 + 13\}^{\omega}$.

This example shows that the discrete dynamic of hybrid system in three dimension is discontinuous in entropy. This implies the entropy can not be correctly approximated at the discontinuity and so it is uncomputable. Because of this fact we can not then use the entropy as parameter to check

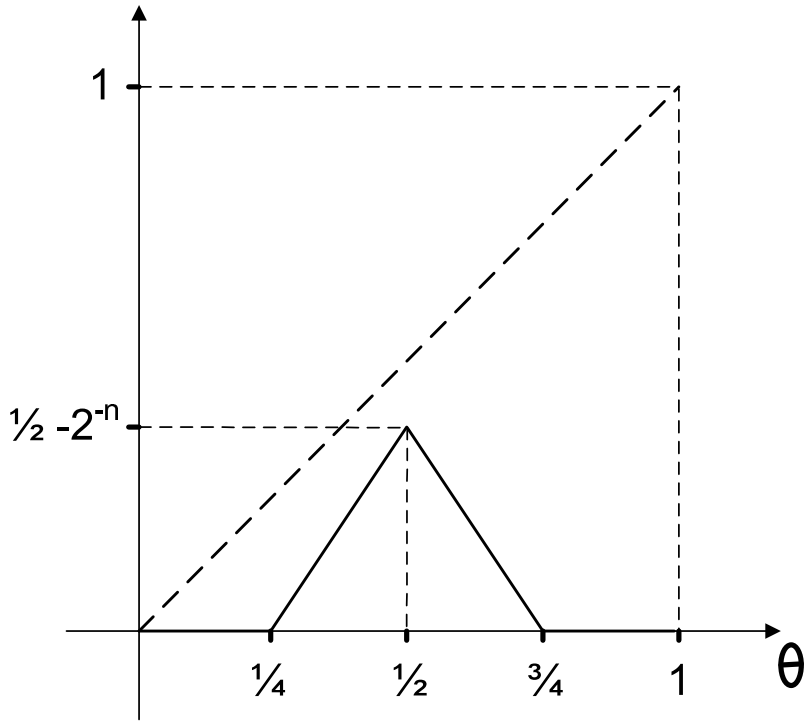


Figure 4.3: The function r_n^θ

how close are lower and upper approximations of the symbolic dynamics of the return map and discrete abstratcion of the hybrdi systems, this is what we mean by uncomputability of discrete abstraction, despite it may still be possible to use a parameter different from entropy like the Hausdorff distance which can be continuous while the entropy is discontinuous as it happens in the excample of this chapter.

A sequence of switched hybrid systems

In this section we prove that the same results we proved in the previous section holds also for hybrid systems without jumps. As before let us consider in $\mathbb{R}^3$ a system of cylindrical coordinates ρ, θ, z where by simplicity θ ranges over $[0, 1]$ instead of usual $[0, 2\pi]$ (i.e. the value of a full turn angle is 1 instead of 2π).

Let us define a sequence of hybrid systems whose hybrid automaton is in Fig. 4.4 and whose modes are given by the following expressions:

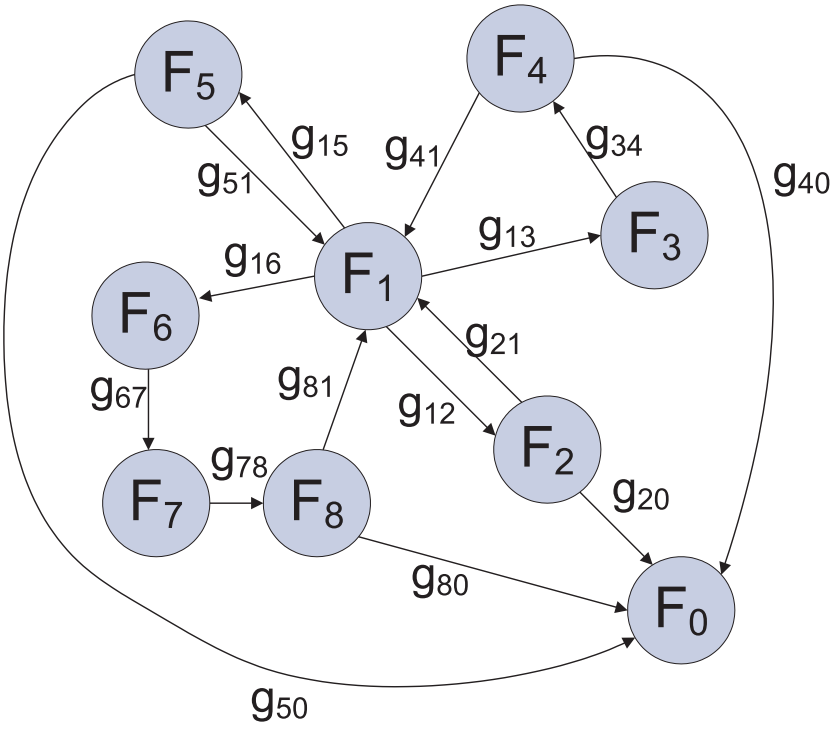


Figure 4.4: hybrid automaton

$$F_0(\rho,\theta,z) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \rho \\ \theta \\ z \end{bmatrix}$$

$$F_1(\rho,\theta,z) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \rho \\ \theta \\ z \end{bmatrix}$$

$$F_2(\rho,\theta,z) = \begin{bmatrix} 1 & -\frac{2^{1+n}\left(-1+\ln\left(1-2^{1-n}\right)\right)}{2^n+2} & 0 \\ 0 & \ln\left(1-2^{1-n}\right) & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} \rho \\ \theta \\ z \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$F_3(\rho,\theta,z) = \begin{bmatrix} 1/2\,\ln(2) & \sqrt{2}/2\,\pi & 0 \\ -\sqrt{2}/4\,\pi & 1/2\,\ln(2) & 0 \\ 0 & 0 & -1/2 \end{bmatrix} \begin{bmatrix} \rho \\ \theta \\ z \end{bmatrix}$$

$$F_4(\rho,\theta,z) = \begin{bmatrix} \frac{(w-1)\ln(1-w)+(1+w)\ln(1+w)}{2w} & \frac{(\ln(1-w)-\ln(1+w))\left(w^2-1\right)}{2w\left(-1+2^{1-n}\right)} & 0 \\ \frac{\left(-1+2^{1-n}\right)(\ln(1-w)-\ln(1+w))}{2w} & \frac{\ln(1-w)(1+w)+\ln(1+w)(w-1)}{2w} & 0 \\ 0 & 0 & -1/2 \end{bmatrix} \begin{bmatrix} \rho \\ \theta \\ z \end{bmatrix} +$$

$$1/\Delta \begin{bmatrix} -1/2\,\frac{\ln(1-w)(1+w)+\ln(1+w)(w-1)}{w} \\ 1/2\,\frac{\left(-1+2^{1-n}\right)(\ln(1-w)-\ln(1+w))}{(ad-bc)w} \\ 0 \end{bmatrix}$$

$$F_5(\rho,\theta,z) = \begin{bmatrix} \ln(2) & \frac{2^{1+n}\left(-\ln(2)+\ln\left(1+2^{-n}\right)\right)}{2^n-1} & 0 \\ 0 & \ln\left(1+2^{-n}\right) & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} \rho \\ \theta \\ z \end{bmatrix} + \begin{bmatrix} \frac{-\ln\left(1+2^{-n}\right)2^n-3\,\ln\left(1+2^{-n}\right)+4\,\ln(2)}{\left(2^n-1\right)\ln(2)\ln\left(1+2^{-n}\right)} \\ \frac{2^{1-n}}{\ln\left(1+2^{-n}\right)} \\ 0 \end{bmatrix}$$

$$F_6(\rho,\theta,z) = \begin{bmatrix} \sqrt{3}\ln(v)-\frac{\pi}{3v} & -\frac{\sqrt{3}\pi}{9v} & \frac{\sqrt{3}\pi}{9v^2} \\ \frac{\sqrt{3}\pi\left(1+v^2\right)}{9v} & \ln(v)+\frac{\sqrt{3}\pi}{9v} & \frac{\pi\,(-1+v)\sqrt{3}}{9v^2} \\ \frac{\pi\left(v+2^{1-n}+v2^{1-n}\right)\sqrt{3}}{9v^2} & -1/9\,\pi\,v\sqrt{3} & \ln(v) \end{bmatrix} \begin{bmatrix} \rho \\ \theta \\ z \end{bmatrix}$$

$$F_7(\rho,\theta,z) = \begin{bmatrix} \ln(2) & 0 & 0 \\ -4/5\,\ln(2)\,2^{-n}-3/5\,\ln(2)-1/5\,\pi\,2^{-n}+1/10\,\pi & 0 & -1/2\,\pi \\ 1/5\,\ln(2)-1/5\,\ln(2)\,2^{1-n}+3/10\,\pi+1/5\,\pi\,2^{1-n} & 1/2\,\pi & 0 \end{bmatrix} \begin{bmatrix} \rho \\ \theta \\ z \end{bmatrix} + \begin{bmatrix} -3\,(\ln(2))^{-1} \\ -3/5\,\frac{-2\,\ln(2)+4\,\ln(2)2^{-n}-3\,\pi-4\,\pi\,2^{-n}}{\pi\,\ln(2)} \\ e/5\,\frac{4\,\ln(2)2^{-n}+18\,\ln(2)+6\,\pi\,2^{-n}-3\,\pi}{\pi\,\ln(2)} \end{bmatrix}$$

$$F_8(\rho,\theta,z) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} \rho \\ \theta \\ z \end{bmatrix}$$

The values of costant w and v are: $w = \sqrt{5-4\,2^{1-n}}$ and $v = \sqrt[3]{1-2^{1-n}}$, the switching conditions are given by the following expressions:

$g_{12} : z = e, \rho < 1/2, \theta < 1/2, \rho + \theta > 1/2$
$g_{20} : z = 1$ and $x \notin D_1$
$g_{21} : z = 1$ and $x \in D_1$
$g_{13} : z = e, \rho > 1/2, \theta < 1/2, \rho - \theta < 1/2$
$g_{34} : z = e^{1/2}$
$g_{40} : z = 1$ and $x \notin D_1$
$g_{41} : z = 1$ and $x \in D_1$
$g_{15} : z = e, \rho < 1/2, \theta > 1/2, \theta - \rho < 1/2$
$g_{50} : z = 1$ and $x \notin D_1$
$g_{51} : z = 1$ and $x \in D_1$
$g_{16} : z = e, \rho > 1/2, \theta > 1/2, \rho + \theta > 1, \theta + \rho < 3/2$
$g_{67} : \rho + \theta = e$
$g_{78} : z = e$
$g_{80} : z = 1$ and $x \notin D_1$
$g_{81} : z = 1$ and $x \in D_1$

Despite the apparent complexity the behaviour of these hybrid systems is relatively simple. The hybrid automaton has a radial structure with centre mode 1 and four branches made by modes: (2), (3, 4) , (5), (6, 7, 8).

Let us consider Fig. 4.5 where over the plane of coordinates ρ, θ is represented the unit circle in polar coordinates. The switching of the hybrid system takes place in the unit circle over the planes $z = 1$ and $z = e$.

Mode 1 project the points from the plane $z = 1$ to the plane $z = e$, the four branches project a point from the plane $z = e$ to $z = 1$ in order to produce four different affine values of the return map defined in the next section in its $\rho\times\theta \to \rho$ component over the four differently coloured regions as shown in Fig.4.5 (and matching also the value of $\rho\times\theta \to \theta$ component). Any trajectory which starts outside the coloured regions is eventually mapped to mode 0.

Let us define the sets:

$$D_1 = \{z = 1, \rho + \theta > 1/2, \rho - \theta < 1/2, \theta - \rho < 1/2, \theta + \rho < 3/2\}$$
$$D_2 = \{z = e, \rho + \theta > 1/2, \rho - \theta < 1/2, \theta - \rho < 1/2, \theta + \rho < 3/2\}$$
$$D_3 = \{z = \sqrt{e}, \rho + \theta > 1/2, \rho - \theta < 1/2, \theta - \rho < 1/2, \theta + \rho < 3/2\}$$

We consider return map $r_n(\rho, \theta) : D_1 \longrightarrow D_1$ and to compute the expression of this return map we first integrate the flow of each mode as follows:

Mode 1 maps from D_1 to D_2, the integral map is:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & e \end{bmatrix} \begin{bmatrix} \rho \\ \theta \\ z \end{bmatrix}$$

Mode 2 maps from D_2 to D_1, the integral map is:

$$\begin{bmatrix} -2 & 2 & 0 \\ 0 & 1-2\,n^{-1} & 0 \\ 0 & 0 & e^{-1} \end{bmatrix} \begin{bmatrix} \rho \\ \theta \\ z \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

Mode 3 maps from D_2 to D_3 in time $1/2$, the integral map is:

$$\begin{bmatrix} 0 & 2 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & e^{-1/2} \end{bmatrix} \begin{bmatrix} \rho \\ \theta \\ z \end{bmatrix}$$

Mode 4 maps from D_3 to D_1 in time 1/2, the integral map is:

$$\begin{bmatrix} 1 & 2 & 0 \\ 1/2 - 1/2\,2^{1-n} & 0 & 0 \\ 0 & 0 & e^{-1/2} \end{bmatrix} \begin{bmatrix} \rho \\ \theta \\ z \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

Mode 5 maps from D_2 to D_1 in time 1/2, the integral map is:

$$\begin{bmatrix} -2 & -2 & 0 \\ 0 & 1 + 2\,n^{-1} & 0 \\ 0 & 0 & e^{-1} \end{bmatrix} \begin{bmatrix} \rho \\ \theta \\ z \end{bmatrix} + \begin{bmatrix} 1 \\ -2^{1-n} \\ 0 \end{bmatrix}$$

Mode 6 maps from D_3 to D_2 in time 1/2, the integral map is:

$$\begin{bmatrix} -1 & -1 & 0 \\ 1 & 1 & 1 \\ 1 + 2^{1-n} & 0 & 0 \end{bmatrix} \begin{bmatrix} \rho \\ \theta \\ z \end{bmatrix}$$

Mode 7 maps from D_3 to D_2 in time 1/2, the integral map is:

$$\begin{bmatrix} 2 & 0 & 0 \\ -1 - 2^{1-n} & 0 & -1 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} \rho \\ \theta \\ z \end{bmatrix} + \begin{bmatrix} 3 \\ -2^{1-n} \\ 0 \end{bmatrix}$$

Mode 8 maps from D_3 to D_2 in time 1/2, the integral map is:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & e^{-1} \end{bmatrix} \begin{bmatrix} \rho \\ \theta \\ z \end{bmatrix}$$

Combining all this integral map we obtain the following return map:

$$r_n(\rho,\theta) = \begin{cases} \begin{bmatrix} 2 & 2 \\ 0 & 1-2^{1-n} \end{bmatrix} \begin{bmatrix} \rho \\ \theta \end{bmatrix} + \begin{bmatrix} -1 \\ 0 \end{bmatrix} & \text{if } \rho < 1/2 \wedge \theta < 1/2 \wedge \rho+\theta > 1/2 \\ \begin{bmatrix} 2 & -2 \\ 0 & 1+2^{1-n} \end{bmatrix} \begin{bmatrix} \rho \\ \theta \end{bmatrix} + \begin{bmatrix} 1 \\ -2^{1-n} \end{bmatrix} & \text{if } \rho < 1/2 \wedge \theta > 1/2 \wedge \theta-\rho < 1/2 \\ \begin{bmatrix} -2 & 2 \\ 0 & 1-2^{1-n} \end{bmatrix} \begin{bmatrix} \rho \\ \theta \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} & \text{if } \rho > 1/2 \wedge \theta < 1/2 \wedge \rho-\theta < 1/2 \\ \begin{bmatrix} -2 & -2 \\ 0 & 1+2^{1-n} \end{bmatrix} \begin{bmatrix} \rho \\ \theta \end{bmatrix} + \begin{bmatrix} 3 \\ -2^{1-n} \end{bmatrix} & \text{if } \rho > 1/2 \wedge \theta > 1/2 \wedge \theta+\rho < 3/2 \end{cases}$$

The limit $\lim_{n\to\infty} r_n^\theta = r^\theta$ is:

$$r(\rho,\theta) := \begin{cases} \begin{bmatrix} 2 & 2 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \rho \\ \theta \end{bmatrix} + \begin{bmatrix} -1 \\ 0 \end{bmatrix} & \text{if } \rho < 1/2 \wedge \theta < 1/2 \wedge \rho+\theta > 1/2 : \\ \begin{bmatrix} 2 & -2 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \rho \\ \theta \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} & \text{if } \rho < 1/2 \wedge \theta > 1/2 \wedge \theta-\rho < 1/2 \\ \begin{bmatrix} -2 & 2 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \rho \\ \theta \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} & \text{if } \rho > 1/2 \wedge \theta < 1/2 \wedge \rho-\theta < 1/2 \\ \begin{bmatrix} -2 & -2 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \rho \\ \theta \end{bmatrix} + \begin{bmatrix} 3 \\ 0 \end{bmatrix} & \text{if } \rho > 1/2 \wedge \theta > 1/2 \wedge \theta+\rho < 3/2 \end{cases}$$

For all n the return map is not chaotic beacause the set of non-wandering points in D_1 is $[0,1]\times\{1\}$ and all points on such set are fixed points. Instead the limit for $n \to \infty$ is chaotic because $r_\infty(\rho, 1/2) : [0,1]\times\{1/2\} \longrightarrow [0,1]\times\{1/2\}$ is the tent map.

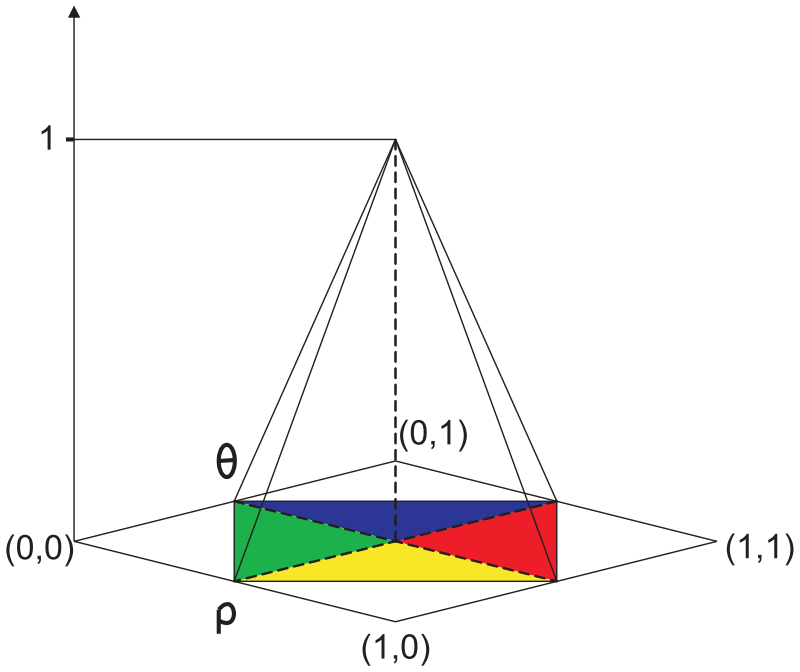


Figure 4.5: The $\rho \times \theta \to \rho$ component of the return map

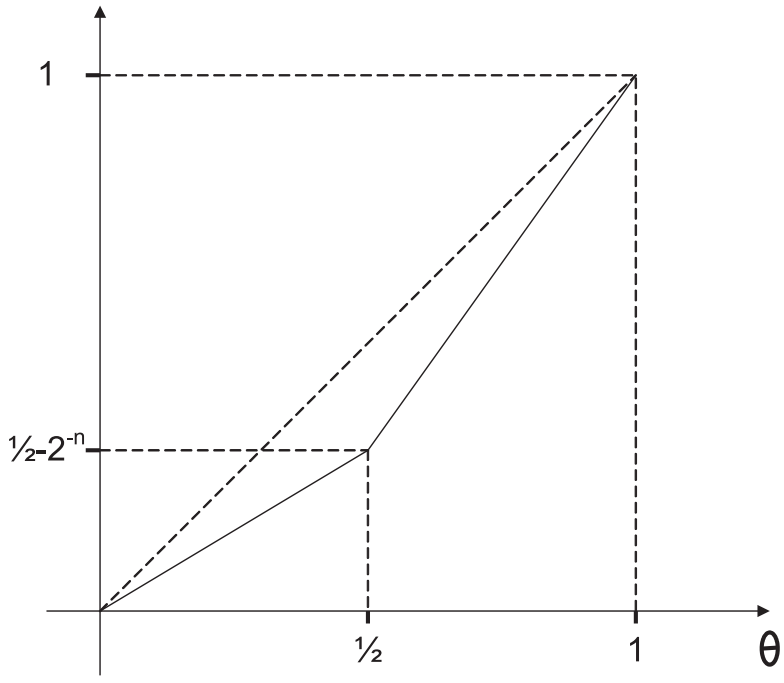


Figure 4.6: The $\rho \times \theta \to \theta$ component of the return map which turns to be only a function of θ

Conclusion

Symbolic dynamics is a way of translating orbits of a dynamical systems defined on a space X to a symbolic system where the dynamic is given by the shift operator over a set of symbolic sequences defined in A^∞ where A is a finite alphabet where each of the symbol is associated to an element of a topological partition of X. There is a semiconjugacy between the original system and the symbolic system. The symbolic system is much easier to study to understand important qualitative and topological properties of the original systems. Frequently for a system from applied science the partition of X can be chosen in such a way that the sequences have a very relevant physical meaning.

In this thesis we have presented techniques to compute symbolic dynamics for piecewise continuous maps in one dimension and piecewise affine maps in two dimensions. This second technique is extendible to higher dimensions. The advantage of these techniques compared to other techniques for computation of symbolic dynamics in the literature is that they deal with dynamic with discontinuity both in one and in two dimension and can either be proved to converge or yield good results in practice.

The work for computation of symbolic dynamics of one dimensional piecewise continuous maps combines both covering relations and kneading theory providing algorithms which produce rigorous lower and upper approximation of the symbolic dynamics and under some conditions of absence of periodic orbit along the boundaries of the topological partition converge to the symbolic dynamics under suitable refinement strategy.

The work for computation of symbolic dynamics of two dimensional piecewise affine maps is based of the decomposition of the Conley index and requires the implementation of a calculus of non convex polyhedra.

The algorithms can be used to analyse flows via return maps and to compute discrete dynamics of hybrid systems, a type of system which combine continuous and discrete dynamic via suitable return map. For hybrid systems algorithms which compute symbolic dynamics for discontinuous maps are essential because the general return map of an hybrid system is discontin-

uous. All the algorithms we developed perform accurate computation, they deal with numerical error by using interval arithmetic.

Related future work can be focused on implementation of algorithms for computation of symbolic dynamics based on the decomposition of Conley index extended to higher dimension and non affine maps.

Appendix A

Kneading theory

Milnor & Thurston's original approach to kneading theory

For multimodal maps with l branches (laps) $I_1, \ldots, I_l$, we define a *invariant coordinate function* $\vec{\theta}(x)$ as a power series $\vec{\theta}(x) = \sum_{k=0}^{\infty} \vec{\theta}_k(x) t^k$. Using the formal coordinates of the critical points, we define an $(l-1) \times l$ *kneading matrix* N with entries in the formal power series ring $\mathbb{Z}[[t]]$. From the kneading matrix, we can define the kneading determinant $D(t)$ as a formal power series over t, and thence estimate the topological entropy.

We now define the invariant coordinate $\vec{\vec{\theta}}(x)$ of a point x. Define a base of $\mathbb{Q}^l$ by $\vec{e}_1, \ldots, \vec{e}_l$ corresponding to the laps $I_1, \ldots, I_l$. For each $j = 1, \ldots, l$, define $\varepsilon_j = \operatorname{sgn}(f')$ over I_j. For a point x with itinerary $\vec{s}$ such that $f^n(x)$ is not a critical point for all n, define

$$\vec{\theta}_i = \operatorname{sgn}\big((f^i)'(x)\big)\, \vec{e}_{s_i} = \varepsilon_{s_0} \varepsilon_{s_1} \cdots \varepsilon_{s_{i-1}} \vec{e}_{s_i} \in \mathbb{Q}^l,$$

and the *invariant coordinate* as the power series $\vec{\theta} = \sum_{i=0}^{\infty} \vec{\theta}_n t^n$.

We order the sequences $\vec{\vec{\theta}}(x)$ lexicographically. Hence $\vec{\vec{\theta}} < \vec{\vec{\theta}}'$ if $\vec{\theta}_i = \vec{\theta}'_i$ for all $i < j$, and $\vec{\theta}_j < \vec{\theta}'_j$. Here, we have the ordering

$$-\vec{e}_1 < -\vec{e}_2 < \cdots < -\vec{e}_l < \vec{e}_l < \cdots < \vec{e}_1.$$

Then if $x < y$, we have $\vec{\vec{\theta}}(x) \leq \vec{\vec{\theta}}(y)$. We define the formal power series $\vec{\theta}$ by

$$\vec{\theta}(t) = \sum_{i=0}^{\infty} \vec{\theta}_i t^i$$

We now define the kneading matrix. For each of the turning points $c_1, \ldots, c_{l-1}$, we construct the limits from above and below of the invariant coordinates, $\vec{\theta}(c_i^{\pm})$. We define the *kneading invariants* by $\nu_i = \vec{\theta}(c_i^+) - \vec{\theta}(c_i^-)$. The kneading matrix is then given by

$$\vec{\nu_i} = \sum_{j=1}^{l} N_{ij}\vec{e}_j.$$

We write

$$N_{ij} = \sum_{k=0}^{\infty} N_{ij}^k t^k.$$

where for $k \geq 1$, the N_{ij}^k are given by

$$N_{ij}^k = \begin{cases} +2 \text{ if } f^k(c_i) \in I_j \text{ and } f^k \text{ has a local minimum at } c_i; \\ -2 \text{ if } f^k(c_i) \in I_j \text{ and } f^k \text{ has a local maximum at } c_i; \\ 0 \text{ otherwise.} \end{cases}$$

and for $k = 0$ by

$$N_{i,i+1}^0 = +1;\ N_{i,i}^0 = -1;\ N_{i,j}^0 = 0 \text{ otherwise.}$$

We now define the kneading determinant. Let $D_j(t)$ be the determinant of the matrix obtained from N by removing the j^{th} column. Then we can show that the formal power series $D(t)$ defined by

$$D(t) = (-1)^{j+1} D_j(t)/(1 - \varepsilon_j t)$$

is independent of j. The power series $D(t)$ is the *kneading determinant*.

We now obtain a formula for the topological entropy in terms of the kneading determinant. The topological entropy is given by

$$h_{\text{top}}(f) = \log 1/s, \text{ where } s \text{ is the smallest positive root of } D(t) = 0.$$

The main disadvantage of Milnor and Thurston's original approach is that it only deals with the infinite kneading invariants, and does not directly apply to finite pieces. It is easy to see that the coefficients of the kneading matrix, and hence the kneading determinant, only depend on this initial pieces of orbits of the critical points. A secondary disadvantage is that it requires a substantial machinery to set up the power series.

Old material on kneading

Lemma A.0.1. *Let f and g a piecewise-monotone maps with critical points c_i. Suppose $k_f(c_i) \leq k_g(c_i)$ whenever c_i is a maximum, and $k_f(c_i) \geq k_g(c_i)$ whenever c_i is a minimum. Then any itinerary of f is also an itinerary of g.*

The proof is trivial.

The following result shows how we can construct a map with increased symbolic dynamics.

Lemma A.0.2. *Let f be a piecewise-monotone map with critical points C. That there is a finite set P disjoint from C such that $f(P) \subset P \cup C$. Choose p_i in P such that $f(c_i) \leq p_i$ if c_i is a maximum, and $f(c_i) \geq p_i$ if c_i is a maximum. Then there is a map g with critical set C such that $g(c_i) = p_i$ and $g(p) = f(p)$ for all $p \in P$. Further, any itinerary of g is an itinerary of f.*

The following result shows how we can construct a map with decreased symbolic dynamics.

Lemma A.0.3. *Let f be a piecewise-monotone map with critical points C. Let P be a finite invariant set for f containing at least two points from each monotone branch of f. Let g be the piecewise-affine extension of $f|_P$. Then any itinerary of g is an itinerary of f.*

Remark A.0.1. *In the following theorem, we assume that the forward orbits of all critical points have been computed for the same number of steps. This is a necessary condition (though can be weakend); there are examples for which the symbolic dynamics is not appropriately decreased or increased if this condition is not satisfied.*

Lemma A.0.4. *Let f be a piecewise-monotone map with critical set C, and let $R = \bigcup_{i=0}^{n} f^i(C)$. Assume that the orbits of C are distinct up to length n. Further assume that for any two critical points c, c', there exists $i \in 1, \ldots, n$ and $c'' \in C$ such that $c'' \in [f^i(c), f^i(c')]$.*

Consider the map g given by $f(f^n(c)) = f(x)$, where $x = \max\{y \in R \mid y < f^n(c)\}$ if f^n has a maximum at c. Assume futher that $f(x) \in R$. Then any itinerary of an orbit of g is an itinerary of an orbit of f.

Proof. Consider the kneading invariant of an orbit of f. The first n symbols are the same for both f and g. Since f and g agree on $f^i(c)$ for $i < n$, the first n symbols of the itinerary of c are the same for both f and g. Thereafter, the orbit c under g follows the orbit of x under f for some steps. If the itineraries vary here, then the itinerary of c under g is less forcing than that of f. □

Theorem A.0.5. *Let $\mathcal{Q}$ be a partition of I and f be a $\mathcal{P}$-continuous map.*

- *If c is a critical point with $f^j(c) \in (p, q)$, then changing the map so that $f(f^{j-1}(c)) = p$ increases the symbolic dynamics if $\delta_{\vec{k},j} > 0$, and decreases symbolic dynamics if $\delta_{\vec{k},j} < 0$. Similarly, changing the map so that $f(f^{j-1}(c)) = p$ decreases symbolic dynamics if $\delta_{\vec{k},j} > 0$, and increases symbolic dynamics if $\delta_{\vec{k},j} < 0$.*

- *If d is a discontinuity point with $f^j(c) \in (p, q)$, then changing the map so that $f(f^{j-1}(d)) = p$ increases entropy if $\varepsilon_{\vec{k},j} > 0$, and decreases entropy if $\varepsilon_{\vec{k},j} < 0$.*

Proof. **Decreasing entropy** Let R be a collection of forward images of critical points. Let c be a critical point; without loss of generality suppose R contains exactly n iterates of c, and that f and f^n both have a maximum at c. Let r be the maximum point of R less that $f^n(c)$. Assume that the image of r lies in R. Let P consist of all n^{th} preimages of r. Let p be the maximum point of P less than $f(c)$.

Consider the effect of changing f such that $f(f^n(c)) = f(r)$. We aim to show that this has the same effect as changing f so that $f(c) = f(p)$. Firstly, if we take $f(c) = f(p)$, then $f(f^n(c)) = f^n(f(p)) = f(f^n(p)) = f(r)$.

Increasing entropy

□

We can reduce the discontinuous case to the continuous case by making an infinitesimal perturbation splitting the discontinuity point d into two points $_{\pm}$, and interpolating monotonically between them. In the resulting theory, the entropy bounds will be too high, but we can obtain the correct values by removing orbits passing through the artificially introduced regions.

Appendix B

Homology

In this section we give the main few definitions of homology theory in an intuitive and simplified way to help the reader unfamiliar with the subject to understand the algorithms in the previous sections.

This very essential presentation of homology is very similar to explanation of homology provided by the abelianization of the fundamental group, which is the group of close curves with the concatenation operation. The fundamental group can only explain homology related to curves, the first homology which is exactly the homology we used in the chapter, but a generalization to higher dimension is straightforward.

We refer the readers to the many good text books explaining homology with the above mentioned abelianization of the fundamental group [49], simplicial complexes or categorical axioms [50] and cubical complexes [12] for a rigorous and detailed explanation of this theory.

Definition B.0.1 (Homology of a set)**.** *The homology of a compact orientable topological space P is a sequence of groups*

$$H^i(P) = \mathbb{Z}^{n_i}$$

where n_i is called the $i-th$ Betti number and depends on the number of holes in the $i-th$ dimension in the topological space. The number n_0 corresponds to the number of connected components in the topological space. The boundary of a two dimensional hole is associated with an homology generator of the first homology, in other words an element of a basis of this homology. Then each element of the first homology space can be associated to a curve which turns around the i-th holes as many times as the i-th component of the element.

A generator corresponding to an hole of dimension i is associated to the boundary of this hole, a closed hypersurface of dimension i.

When the space where the map is defined is non orientable such as Moebius band or Klein bottle the homology has a torsional component:

$$H^i(P) = \mathbb{Z}^{n_i} \oplus \mathbb{Z}/m_i\mathbb{Z}$$

Definition B.0.2 (Homology of a function)**.** *The homology of a function f is a sequence of linear application*

$$H^i(f) : H^i(P) \longrightarrow H^i(P)$$

each of them is determined by the way the function maps each homology generator of the corresponding dimension to other homology generators of the same dimension. The $j-th$ row of $H^i(f)$ is the element of $\mathbb{Z}^{n_i}$ corresponding to the image of the $j-th$ homology generator.

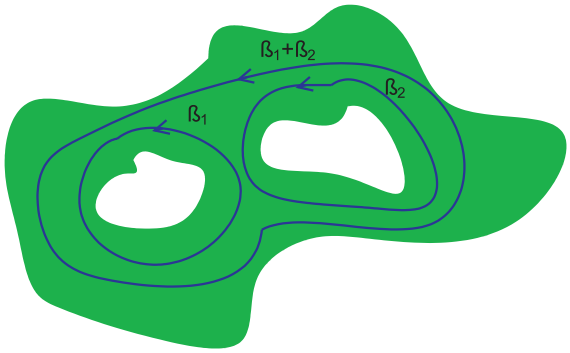


Figure B.1: First homology generators

Now we are going to give the definition of relative homology also known as homology of a pair of spaces (P, Q), where $Q \subseteq P$. Also this homology is a sequence of groups as the homology of a space, but to give a suitable definition in this context before we mention the excision property of the homology of a pair: $H^i(P, Q) = H^i(P \backslash A, Q \backslash A)$ where A is an open set. Then we can assume Q contractible, otherwise we can consider an open set A with $P_1 = P \cup A$, $Q_1 = Q \cup A$ such that Q_1 is contractible and by the excision property $H^i(P, Q) = H^i(P_1, Q_1)$.

Definition B.0.3 (Relative homology)**.** *We define the homology of a pair (P, Q) the subgroup of the homology of P made by points associated to curves which make at least one turn around a hole whose boundary intersect $P \backslash Q$.*

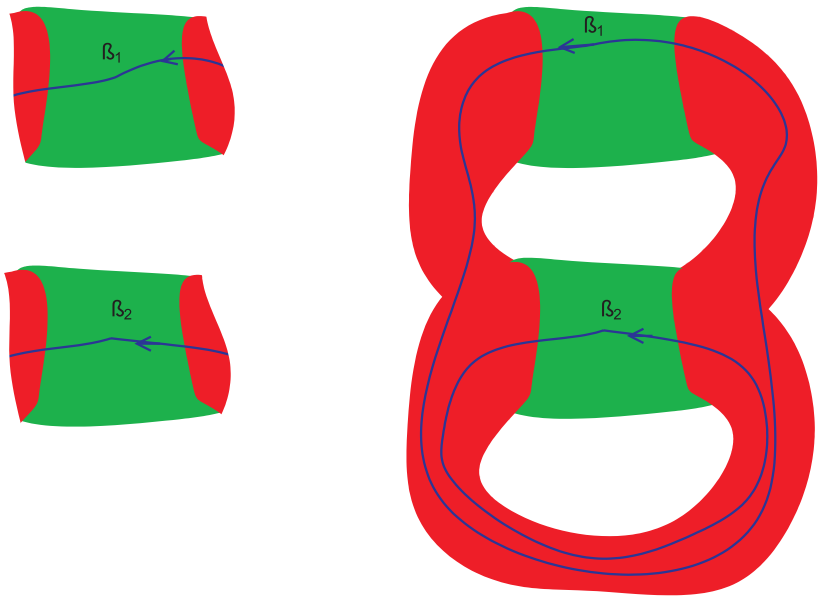


Figure B.2: Relative homology

Appendix C

Computability of subbasin of divergence for affine maps

Proof of Proposition 3.5.1. Let $f(x) = Ax + b$, we are going to prove the result on linear system with matrix in Jordan form because there is always an affine transformation $z = T^{-1}(x+(I-A)^{-1}b)$ which transforms the system $Ax + b$ to the system Jz where J is the Jordan form of A and can be one of the following cases:

$$\begin{bmatrix} \lambda & 0 \\ 0 & \mu \end{bmatrix} \begin{bmatrix} \lambda & 1 \\ 0 & \lambda \end{bmatrix}$$

where λ and μ are eigenvalues. In the case the eigenvalues are complex conjugate is more convenient to write it as:

$$\begin{bmatrix} \sigma & \omega \\ -\omega & \sigma \end{bmatrix} = \begin{bmatrix} \rho\cos(\theta) & \rho\sin(\theta) \\ -\rho\sin(\theta) & \rho\cos(\theta) \end{bmatrix}$$

where $\lambda = \sigma + j\omega = \rho e^{j\theta}$

In this system the equation of an orbit starting at an initial point z_0 is given by $z(n) = J^n z_0$, where J^n can be written as:

$$\begin{bmatrix} \lambda^n & 0 \\ 0 & \mu^n \end{bmatrix} \begin{bmatrix} \lambda^n & \lambda^{n-1}n \\ 0 & \lambda^n \end{bmatrix} \begin{bmatrix} \rho^n\cos(n\theta) & \rho^n\sin(n\theta) \\ -\rho^n\sin(n\theta) & \rho^n\cos(n\theta) \end{bmatrix}$$

to determine the invariant curve where orbit lays we consider the following limit which is well defined only for matrices with positive real eigenvalues or pairs of conjugate complex eigenvalues:

$$Y_i = \lim_{h\to 0} \frac{J_i^h - I}{h}$$

which gives respectively :

$$\begin{bmatrix} \ln\lambda & 0 \\ 0 & \ln\mu \end{bmatrix} \begin{bmatrix} \ln\lambda & 1/\lambda \\ 0 & \ln\lambda \end{bmatrix} \begin{bmatrix} \ln\rho & \theta \\ -\theta & \ln\rho \end{bmatrix}$$

This means that each invariant curve of the discrete time system $z(n+1) = Jz(n)$ corresponds to a trajectory of the continuous time dynamical system $\dot{z} = Yz$. In the case of different real eigenvalues where at least one is negative we can extend the result in this way:

$$J^{<\lambda,\mu>} \begin{bmatrix} \lambda & 0 \\ 0 & \mu \end{bmatrix}$$

$w = [w_x, w_y] = J^{<|\lambda|,|\mu|>} z(n)$

1. $\lambda < 0, \mu > 0,\ z(n+1) = [-w_x, w_y]$
2. $\lambda > 0, \mu < 0,\ z(n+1) = [w_x, -w_y]$
3. $\lambda < 0, \mu < 0,\ z(n+1) = [-w_x, -w_y]$

So in cases 1, 2, 3 each invariant curves of the discrete time system $z(n+1) = J^{<\lambda,\mu>} z(n)$ corresponds to two trajectories of the dynamical system $\dot{z} = Y^{<|\lambda|,|\mu|>} z$, respectively symmetric for the axis x, y and the origin.

We are going to prove only the computability of the subbasin of forward divergence because it is possible to reduce the subbasin of backward divergence to the subbasin of the forward divergence of the inverted map.

Without loss of generality we restrict the case to P convex, if P is not convex we can partition it in convex components and compute the subbasin in each of them. Detect weather P is convex or not is computable and partitioning it in a finite collection of convex polyhedra is computable. Boundedness of P is computable and if P is bounded the operator returns the empty set. If P is unbounded then it has exactly two unbounded sides by convexity. In the following part of the proof we show how to check if the subbasin is empty or not by checking the sign of the eigenvalues or function of the eigenvalues. If the subbasin is not empty we show a procedure to compute it.

- *Case real positive eigenvalues, fixed point node:*

  If the node is an attractor the subbasin of divergence is empty.

  Otherwise let us consider Y on one of the unbounded side of P. If the side is a subset of the manifold of the fixed point we know that not flow line cross it. Otherwise let us consider the following equations:

$$\begin{bmatrix} \ln(\lambda) & 0 \\ 0 & \ln(\mu) \end{bmatrix} \begin{bmatrix} c_1 s + c_2 \\ d_1 s + d_2 \end{bmatrix} = \begin{bmatrix} \ln(\lambda)(c_1 s + c_2) \\ \ln(\mu)(d_1 s + d_2) \end{bmatrix}$$

$$\begin{bmatrix} \ln(\lambda) & \lambda^{-1} \\ 0 & \ln(\lambda) \end{bmatrix} \begin{bmatrix} c_1 s + c_2 \\ d_1 s + d_2 \end{bmatrix} = \begin{bmatrix} \ln(\lambda)(c_1 s + c_2) + \frac{d_1 s + d_2}{\lambda} \\ \ln(\lambda)(d_1 s + d_2) \end{bmatrix}$$

We define a function $\eta(s)$ over an unbounded sides of P:

$$\eta(s) = \sin(\alpha) - \sin(\arg \dot{x}(s))$$

where α is the slope of the side. The sign of $\eta(s)$ tells if the flow is directed inside or outside of P on that side.

$$\eta_1(s) = \frac{1}{\sqrt{\frac{{c_1}^2}{{d_1}^2} + 1}} - \frac{1}{\sqrt{\frac{(\ln(\lambda))^2}{(\ln(\mu))^2} \frac{(c_1 s + c_2)^2}{(d_1 s + d_2)^2} + 1}}$$

limit

$$\lim_{s \to \infty} \eta_1(s) = \frac{1}{\sqrt{\frac{{c_1}^2}{{d_1}^2} + 1}} - \frac{1}{\sqrt{\frac{(\ln(\lambda))^2 {c_1}^2}{(\ln(\mu))^2 {d_1}^2} + 1}}$$

derivative

$$\frac{d\eta_1(s)}{ds} = \frac{(c_1 s + c_2)(-d_1 c_2 + d_2 c_1)(\ln(\mu))^3}{\Delta_1(s)}$$

$$\eta_2(s) = \frac{1}{\sqrt{\frac{{c_1}^2}{{d_1}^2} + 1}} - \frac{1}{\sqrt{(\ln(\lambda))^2 (d_1 s + d_2)^2 \left(\ln(\lambda)(c_1 s + c_2) + \frac{d_1 s + d_2}{\lambda}\right)^{-2} + 1}}$$

limit

$$\lim_{s \to \infty} \eta_2(s) = \frac{1}{\sqrt{\frac{{c_1}^2}{{d_1}^2} + 1}} - \frac{1}{\sqrt{1 + (\ln(\lambda))^2 {d_1}^2 \left(\ln(\lambda) c_1 + \frac{d_1}{\lambda}\right)^{-2}}}$$

derivative

$$\frac{d\eta_2(s)}{ds} = -\frac{(\ln(\lambda))^3 (d_1 s + d_2) \lambda^3 (-d_1 c_2 + d_2 c_1)}{\Delta_2(s)}$$

$\forall s\ \Delta_1(s) > 0$ and $\Delta_2(s) > 0$. In both cases the function has only one critical point and so at most two zeroes. The critical point can be computed from the expression of the derivative and according the value of the function on the critical point and the limit we can compute the number of the zeroes and the values. After the last zero the condition of the flow of being directed inside or outside P is constant and depends on the value of the limit.

In conclusion for each side of P we are able to compute if there is point on the side after which the flow Y is directed inside P or outside P. In the second case we can compute the point.

We can join the two point by a line. In this way we define a new polyhedron which is candidate to be a subbasin. We consider the direction of Y on the bounded side, as we saw before we are able to compute if is directed inward or outward the polyhedron. In the case is outward on some part of the bounded side we can apply this procedure: on both unbounded sides.

Let us consider a series of points $\{p_i\}_i$ such that p_0 is the initial point of the side and the distance along the side between p_0 and p_k is $k \times d$ where d is some arbitrarily chosen step . For each k we consider two points, one for side, p_k^1 p_k^2 which define a new side and then a new polyhedron. We proceed on this construction until we find a side $[p_k^1, p_k^2]$ such that the direction of Y over the side is fully inward. This procedure terminates after finite number of steps because otherwise there is a trajectory whose alpha set is infinity. A contradiction, being the alpha set of every trajectory the fixed point.

- *Case real positive eigenvalues, fixed point a saddle:*

  We assume the interior of P does not intersect the stable manifold otherwise we can partition P in two polyhedron with such property. Clearly P contains a subbasin of divergence if and only if P contain half unstable manifold (half starting at any point, not necessary the fixed point). This condition can be easily checked comparing the slope of the two unbounded side of P. To determine the point on the side after which the flow is inward the polyhedron we can proceed as before. We can then connect the two sides with a side lying on a line parallel to the stable manifold on which the flow is inward.

- *Case real eigenvalues:*

  In this case we consider non convex polyhedra made of convex polyhedra which are the symmetric image of each other according to the

symmetry of the systems induced by the eigenvalues.

The apply the procedure of the two cases above (according the type of fixed point) to each of them, with a system with positive eigenvalues.

- *Case complex eigenvalues:*

  If $\frac{\theta}{\pi}$ is irrational, the polyhedron contains a subasin if and only if the complementary of P is bounded. Then any orbit of the dynamical system $\theta(n+1) = \measuredangle(x(n+1)) = \measuredangle(A_i x(n) + b_i)$ is dense in the interval $[0, 2\pi]$, we refer to [51] for this result. If $\frac{\theta}{\pi} = \frac{q_1}{q_2}$ is rational it is clear that P contains a subbasin if and only if it contains a set of cones with vertex in the fixed point whose slope cover the interval $\bigcup_{k=0}^{2q_2}[\phi + \frac{k}{q_2}\pi, \phi + \alpha + \frac{k}{q_2}\pi]$ except for at most a bounded part of it, for some $\phi \in [0, 2\pi]$ and some $\alpha \in [0, \frac{1}{q_2}\pi)$.

□

Bibliography

[1] Zgliczyński, P., Gidea, M.: Covering relations for multidimensional dynamical systems. J. Differential Equations **202**(1) (2004) 32–58

[2] 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

[3] Rocha, J.L., Sousa Ramos, J.: On iterated maps of the interval with holes. J. Difference Equ. Appl. **9**(3-4) (2003) 319–335 Dedicated to Professor Alexander N. Sharkovsky on the occasion of his 65th birthday.

[4] Rocha, J.L., Ramos, J.S.: Weighted kneading theory of one-dimensional maps with a hole. Int. J. Math. Math. Sci. (37-40) (2004) 2019–2038

[5] Rom-Kedar, V.: Homoclinic tangles—classification and applications. Nonlinearity **7**(2) (1994) 441–473

[6] McRobie, F.A., Thompson, J.M.T.: Knot-types and bifurcations sequences of homoclinic and transient orbits of a single-degree-of-freedom driven oscillator. Dynam. Stab. Sys. **9** (1994) 223–251

[7] Mitchell, K.A., Delos, J.B.: A new topological technique for characterizing homoclinic tangles. Phys. D **221**(2) (2006) 170–187

[8] Birman, J.S., Williams, R.F.: Knotted periodic orbits in dynamical systems. i. lorenz's equations. Topology **22**(1) (1983) 47–82

[9] Birman, J.S., Williams, R.F.: Knotted periodic orbits in dynamical systems-ii: Knot holders for fibered knots. In: Low-dimensional topology. Number 20 in Contemporary Mathematics, Providence, Rhode Island, American Mathematical Society (1983) 1–60

[10] Fried, D.: Periodic points and twisted coefficients. In: Geometric dynamics (Rio de Janeiro, 1981). Volume 1007 of Lecture Notes in Math. Springer, Berlin (1983) 261–293

[11] Bestvina, M., Handel, M.: Train-tracks for surface homeomorphisms. Topology **34**(1) (1995) 109–140

[12] Kaczynski, T., Mischaikow, K., Mrozek, M.: Computational homology. Volume 157 of Applied Mathematical Sciences. Springer-Verlag, New York (2004)

[13] Day, S., Junge, O., Mischaikow, K.: Towards automated chaos verification. In: EQUADIFF 2003. World Sci. Publ., Hackensack, NJ (2005) 157–162

[14] Misiurewicz, M.: Horseshoes for mappings of the interval. Bull. Acad. Polon. Sci. Sér. Sci. Math. **27**(2) (1979) 167–169

[15] Block, L.: An example where topological entropy is continuous. Trans. Amer. Math. Soc. **231**(1) (1977) 201–213

[16] Misiurewicz, M., Szlenk, W.: Entropy of piecewise monotone mappings. Studia Math. **67**(1) (1980) 45–63

[17] Collins, P.: Symbolic dynamics from homoclinic tangles. Internat. J. Bifur. Chaos Appl. Sci. Engrg. **12**(3) (2002) 605–617

[18] Collins, P.: Dynamics of surface diffeomorphisms relative to homoclinic and heteroclinic orbits. Dyn. Syst. **19**(1) (2004) 1–39

[19] Szymczak, A.: The Conley index for decompositions of isolated invariant sets. Fund. Math. **148**(1) (1995) 71–90

[20] Sharkovskii, O.M.: Co-existence of cycles of a continuous mapping of the line into itself. Ukrain. Mat. Ž. **16** (1964) 61–71

[21] Block, L., Keesling, J.: Computing the topological entropy of maps of the interval with three monotone pieces. J. Statist. Phys. **66**(3-4) (1992) 755–774

[22] Lampreia, J.P., Sousa Ramos, J.: Trimodal maps. Internat. J. Bifur. Chaos Appl. Sci. Engrg. **3**(6) (1993) 1607–1617

[23] Alves, J.F., Sousa Ramos, J.: Kneading theory for tree maps. Ergodic Theory Dynam. Systems **24**(4) (2004) 957–985

[24] Milnor, J.: Is entropy effectively computable. `www.math.sunysb.edu/ jack/comp-ent.ps`

[25] Moore, R.E.: Interval analysis. Prentice-Hall Inc., Englewood Cliffs, N.J. (1966)

[26] Jaulin, L., Kieffer, M., Didrit, O., Walter, É.: Applied interval analysis. Springer-Verlag, London (2001)

[27] Aberth, O.: Introduction to Precise Numerical Methods. Academic Press (2007)

[28] Lind, D., Marcus, B.: An introduction to symbolic dynamics and coding. Cambridge University Press, Cambridge (1995)

[29] Kitchens, B.P.: Symbolic dynamics. Universitext. Springer-Verlag, Berlin (1998) One-sided, two-sided and countable state Markov shifts.

[30] 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)

[31] Baladi, V., Ruelle, D.: An extension of the theorem of Milnor and Thurston on the zeta functions of interval maps. Ergodic Theory Dynam. Systems **14**(4) (1994) 621–632

[32] Preston, C.: What you need to know to knead. Adv. Math. **78**(2) (1989) 192–252

[33] Misiurewicz, M., Ziemian, K.: Horseshoes and entropy for piecewise continuous piecewise monotone maps. In: From phase transitions to chaos. World Sci. Publ., River Edge, NJ (1992) 489–500

[34] Mori, M.: Fredholm determinant for piecewise linear transformations. Osaka J. Math. **27**(1) (1990) 81–116

[35] Alves, J.F., Sousa Ramos, J.: Kneading theory: a functorial approach. Comm. Math. Phys. **204**(1) (1999) 89–114

[36] Lampreia, J.P., Sousa Ramos, J.: Symbolic dynamics of bimodal maps. Portugal. Math. **54**(1) (1997) 1–18

[37] Chen, Z.X., Zhou, Z.: Entropy invariants. I. The universal order relation of order-preserving star products. Chaos Solitons Fractals **15**(4) (2003) 713–727

[38] Chen, Z.X., Zhou, Z.: Entropy invariants. II. The block structure of Stefan matrices. Chaos Solitons Fractals **15**(4) (2003) 729–742

[39] Yomdin, Y.: Volume growth and entropy. Israel J. Math. **57**(3) (1987) 285–300

[40] Collet, P., Crutchfield, J.P., Eckmann, J.P.: Computing the topological entropy of maps. Comm. Math. Phys. **88**(2) (1983) 257–262

[41] Block, L., Keesling, J., Li, S.H., Peterson, K.: An improved algorithm for computing topological entropy. J. Statist. Phys. **55**(5-6) (1989) 929–939

[42] Nusse, H.E., Yorke, J.A.: Border-collision bifurcations for piecewise smooth one-dimensional maps. Internat. J. Bifur. Chaos Appl. Sci. Engrg. **5**(1) (1995) 189–207

[43] Newcomb, R.W., El-Leithy, N.: Chaos generation using binary hysteresis. Circuits Systems Signal Process. **5**(3) (1986) 321–341

[44] 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)

[45] Galias, Z., Zgliczyński, P.: Abundance of homoclinic and heteroclinic orbits and rigorous bounds for the topological entropy for the Hénon map. Nonlinearity **14**(5) (2001) 909–932

[46] Sella, L., Collins, P.: Discrete dynamics of two-dimensional nonlinear hybrid automata. In: Hybrid Systems: Computation and Control, Lecture Notes in Computer Science, Berlin, Springer-Verlag (2008) 486–499

[47] Misiurewicz, M.: Strange attractors for the Lozi mappings. In: Nonlinear dynamics (Internat. Conf., New York, 1979). Volume 357 of Ann. New York Acad. Sci. New York Acad. Sci., New York (1980) 348–358

[48] Misiurewicz, M.: On non-continuity of topological entropy. Bull. Acad. Polon. Sci. Sér. Sci. Math. Astronom. Phys. **19** (1971) 319–320

[49] Hatcher, A.: Algebraic topology. Cambridge University Press, Cambridge (2002)

[50] Munkres, J.R.: Elements of algebraic topology. Addison-Wesley Publishing Company, Menlo Park, CA (1984)

[51] Gilmore, R., Lefranc, M.: The topology of chaos. Wiley-Interscience [John Wiley & Sons], New York (2002) Alice in Stretch and Squeezeland.

Samenvatting

Dynamische systemen zijn wiskundige objecten die goede modellen zijn voor verschillende verschijnselen in het natuur. Vaak zijn ze heel complex, en het is daarom onmisbaar om theoretische en praktische methodes te ontwikkelen om ze te kunnen begrijpen.

Dit proefschrift vormt een bijdrage aan deze taak door methodes voor het analyseren van discrete-tijd dynamische systemen over de reële getallen voor te stellen. Het algemene kader is het vertalen van reële dynamische systemen naar symbolische dynamische systemen. Symbolische dynamische systemen bevatten oneindige reeksen symbolen uit een eindig alfabet; de dynamica wordt bepaald door het schuif-operator, die een reeks transformeert naar een reeks met alle symbolen één plaats opgeschoven. Zulke systemen zijn eenvoudiger om te analyseren en om te begrijpen dan systemen over reële getallen; het is vooral makkelijker voor eigenschappen als de topologische entropie, die een schatting van het complexiteit van de systeem geeft.

In Hoofdstuk 1 wordt het basistheorie van dynamische systemen uitgelegd, en de hoofdstellingen uit het al bestaande literatuur samengevat over het vertalen van systemen over rele toestandsruimtes naar symbolische systemen. In dimensie één is een van de belangrijkste resultaten de knedingstheorie ontwikkeld door Milnor en Thurston, terwijl in dimensie twee een van de belangrijkste resultaten de theorie van de homoclinische verwarring van een vaste punt is.

In Hoofdstuk 2 worden algoritmes voor het berekenen van de symbolische dynamica van stuksgewijs-continue systemen in dimensie één aangegeven; deze methodes zijn gebaseerd op dekkingsrelaties. De toestandsruimte van het systeem wordt verdeeld in paarsgewijs disjuncte open verzamelingen die bijectief corresponderen met het alfabet van de symbolische systeem. De dekkingsrelaties beschrijven hoe elk element van het partitie op een andere afbeeldt, of totaal, of partieel of geen dekking. Strategieën voor de verfijning van de partities laten verbeteringen van de benadering van de symbolische dynamica toe, en verzekeren convergentie van het algoritme tot de exacte dynamica. Door dekkingsrelaties met knedingstheorie te combineren, krijgen

we een opmerkelijke verbetering in de tijdsduur van het computerprogramma.

In Hoofdstuk 3 worden algoritmes voor het berekenen van de symbolische dynamica van stuksgewijs-affine systemen in dimensie twee aangegeven. Voor dit doel zijn geavanceerdere technieken nodig dan in het één-dimensionale geval, omdat in het algemeen de dekkingsrelaties niet geschikt zijn om een goede onder benadering van de dynamica te berekenen. De methode maakt gebruik van algebraïsche topologie, met name van de decompositie van de Conley index, om de symbolische dynamica van de afbeelding van onder te benaderen. Om deze methode te implementeren, wordt een calculus van niet-noodzakelijk convex polyhedra ontwikkeld om de index-paren te beschrijven. Deze methodes zijn ook nuttig voor discontinue afbeeldingen, en voor het bestuderen van de discrete dynamica van hybride systemen met behulp van de symbolische dynamica van de Poincaré afbeelding.

Tenslotte, in Hoofdstuk 4 wordt een bewijs gegeven dat de discrete abstractie van hybride systemen in het algemeen onberekenbaar is. Dit resultaat wordt bewezen door twee voorbeelden van convergente reeksen hybride systemen te tonen waarvan de entropie van de discrete dynamica niet naar de entropie van de discrete dynamica van de limiet convergeert. Dit werk maakt gebruik van het al bestaande werk van Misiurewicz.

Summary

Dynamical systems are mathematical objects which model quite effectively many phenomena in nature. Frequently they are very complex and it is indispensable to develop theoretical and practical tools to understand them.

This thesis gives a contribute to this task by providing techniques to analyse discrete time dynamical systems over real numbers. The general framework where this thesis sets is the translation of dynamical systems over the reals to symbolic dynamical systems. Symbolic systems consist of infinite sequences of symbols over a finite alphabet, the dynamics is given by the shift operator which maps a sequence to a sequence shifted of one position. These kind of systems are easier to analyse and understand than systems over real numbers, in particular it is simpler to compute properties such as topological entropy which give a measure of the level of complexity of the system.

In Chapter 1 we explain the basic theory of symbolic systems and the main results in the literature on translating real value systems to symbolic systems. In one dimension one of the main results is Kneading theory developed by Milnor and Thurston, while in two dimensions one of the main results is the theory of fixed point tangles.

In Chapter 2 we present some algorithms for the computation of symbolic dynamics of piecewise continuous systems in one dimension, these techniques are based on covering relations. The state space where the system is defined is partitioned in pairwise disjoint open sets which represent the alphabet of the symbolic system. The covering relations describe how each element of the partition map an other, covering it totally, partially or not at all. Refinement strategies of the partitions allow improvement of the approximation of the symbolic dynamics and ensure convergence of the algorithm to the exact dynamics. A combination of covering relations with Kneading theory provides an remarkable improvement in the running time of the algorithm.

In Chapter 3 we present algorithms to compute the symbolic dynamics of piecewise affine dynamical systems in two dimensions. For these purpose we need more sophisticated technique than in one dimension, covering relations do not provide in general a good lower approximation of the symbolic

dynamics. The explained method uses algebraic homology, more precisely the decomposition of the Conley index to compute a lower approximation of the symbolic dynamics of the map. To implement this method we develop a calculus of generally non convex polyhedra, by which we model the index pair. The methods reveal to be useful also for discontinuous maps, and so also to study discrete dynamics of hybrid systems via symbolic dynamic of a suitable return map.

Finally in Chapter 4 we prove that the discrete abstraction of hybrid systems is in general uncomputable. We prove this results by showing two examples of converging sequences of hybrid systems whose entropy of the discrete dynamics does not converge to the entropy of the discrete dynamics of the limit. This work exploit the result of a previous work by Misiurewicz.