Words and bisimulations of dynamical systems

Thomas Brihaye

Discrete Mathematics and Theoretical Computer Science DMTCS vol. **9:2**, 2007, 11–32

Words and bisimulations of dynamical systems

Thomas Brihaye†

Université de Mons-Hainaut
Institut de Mathématique
6, Avenue du Champ de Mars
7000 Mons, Belgique

received 15 Jan 2005, revised 31 May 2005, accepted 16 Jun 2005.

In this paper we study bisimulations on *dynamical systems* through a given partition. Our aim is to give a new vision of the notion of bisimulation by using *words*. To achieve this goal, we encode the *trajectories* of the transition system as words. This method was introduced in our paper "On o-minimal hybrid systems" in order to give a new proof of the existence of a finite bisimulation for *o-minimal hybrid systems* (as previously proved in a paper by Lafferriere G., Pappas G.J. and Sastry S.). Here we want to provide a systematic study of this method in order to obtain a procedure for building finite bisimulations based on words.

1 Introduction

More and more real-life systems are automatically controlled. It is of a capital importance to know whether the programs governing these systems are correct. In order to be able to manipulate these real-life systems, various mathematical models have been introduced (*timed automata* [AD94], *hybrid systems* [Hen96],...) making the study of the abstract systems a wide and interesting domain of research. Unfortunately even the abstract systems are not always that easy to handle, the main problem being their infinite size. One way to solve this problem is to reduce these infinite systems to finite systems in such a way that enough information is preserved. It is known that *bisimulations* (see [Acz88, Cau95, Hen95]) are a "reduction" of particular interest since they preserve a lot of interesting properties (*reachability problem*, *model-checking branching logic...* [HNSY94, ACH+95, AHLP00]). That is why we focus our attention on systems admitting a finite bisimulation.

In [BMRT04] in order to prove the existence of a finite bisimulation for an extended class of *o-minimal hybrid systems*(i), we encode the continuous dynamics through *words* (see also [BM05]). In the previous two papers we limit ourselves to the encoding of *o-minimal dynamical systems* (i.e. dynamical systems definable in an o-minimal structure; see [vdD98] for a nice overview on o-minimality.). In particular we only had to manipulate *finite words*. Let us mention that some

†This author is supported by the following research programs: FRFC 2.4.530.02., FRFC 2.4.564.02, Modnet MRTN-CT-2004-512234 and by a grant from the National Bank of Belgium

(i) introduced in [LPS00].

analogue already appears in the literature (the notion of signature for example in [ASY01]). Let us also notice that bisimulations of dynamical systems has been studied independently in [JdS04] but in a different framework. They studied dynamical system as defined in [Wil91].

Our technique was used by Korovina and Vorobjov in order to compute a doubly exponential bound on the size of the coarsest finite bisimulation of *pfaffian hybrid systems* (see [KV04]). They recently improved their results by reducing the bound to a single exponential and prove that this bound is tight (see [KV06]).

In this paper, we want to give a systematic study of this encoding technique. In particular we give a Procedure (*Bisiω*) that aims to build a bisimulation on a dynamical system through a partition. Our hope is that this systematic study will lead to the discovery of some new general classes of dynamical systems (through partition) which admit finite bisimulations. Beyond the fact that dynamical systems are of interest in their own, they are an essential component of hybrid systems. In particular, when *strong reset conditions* are assumed on the hybrid system, finding finite bisimulations of the hybrid system reduces in finding a finite bisimulation on each location (which is endowed with a dynamical system) w.r.t. the partition induced by the guards, resets and invariant. It is the case for o-minimal hybrid systems, see [LPS00]. Moreover a recent point of view on the theory of hybrid systems allows to see an hybrid system as a dynamical system (see the notion of *hybrifold* in [SJSL00]).

The rest of the paper is organized as follows. In section 2, we recall classical definitions and properties of bisimulations on a transition system, we also describe the well-known *bisimulation algorithm* ([BFH91, KS90, Hen96]), which is in fact a semi-algorithm. We end this section by defining what we call a *dynamical system* in this paper. Section 3 is the main section of the paper. We start by explaining how to associate a *word* with a trajectory; we introduce the notion of *dynamical type* which allows in some sense to recover the continuous dynamics through the partition. These tools being formalized we introduce a conceptual semi-algorithm called Procedure *Bisiω* and we prove that this procedure computes a bisimulation. We also describe several variants of our procedure. Finally we discuss in which case *Bisiω* provides the coarsest bisimulation of a dynamical system through a given partition. In Section 4 we provide an extensive list of examples.

2 Preliminaries

In this section, we recall some basic definitions and results concerning bisimulations on a transition system (see [Acz88, Cau95, Hen95] for general references). We also recall the well-known *bisimulation algorithm* ([BFH91, KS90, Hen96]). Then we give definition of *dynamical systems* and associate with them a natural transition system.

2.1 Transition systems and bisimulation

Definition 2.1 *A* transition system $T = (Q, \Sigma, \rightarrow)$ *consists of a set of states Q (which may be uncountable), Σ a finite alphabet of events, and $\rightarrow \subseteq Q \times \Sigma \times Q$ a transition relation.*

A transition $(q_1, a, q_2) \in \rightarrow$ is denoted by $q_1 \xrightarrow{a} q_2$. A transition system is said finite if Q is finite. If the alphabet of events is reduced to a singleton, $\Sigma = \{a\}$, we will denote the transition system $(Q, \rightarrow)$ and omit the event a.

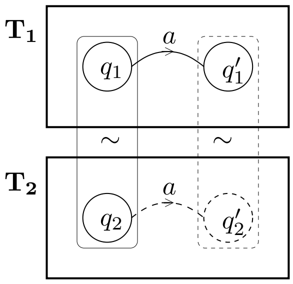

Fig. 1: Forward stable relation

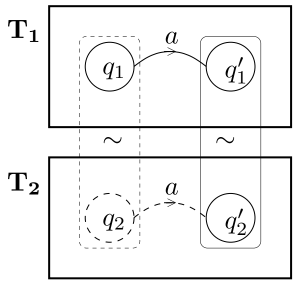

Fig. 2: Backward stable relation

Definition 2.2 *Given a transition system $T = (Q, \Sigma, \rightarrow)$, a* finite path in T *is a finite sequence of transitions $q_0\ q_1\ q_2\ \cdots\ q_n$ such that for all $i = 1, ..., n$ there exists $a_i \in \Sigma$ such that $q_{i-1} \xrightarrow{a_i} q_i$. We denote it as follows:*

$$\rho = q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} q_2 \cdots \xrightarrow{a_n} q_n.$$

Definition 2.3 *Given two transition systems on the same alphabet of events, $T_1 = (Q_1, \Sigma, \rightarrow_1)$ and $T_2 = (Q_2, \Sigma, \rightarrow_2)$, a* partial simulation of T_1 by T_2 *is a binary relation $\sim\ \subseteq Q_1 \times Q_2$ which satisfies the following condition:*

$$\begin{aligned} &\forall q_1, q_1' \in Q_1,\ \forall q_2 \in Q_2,\ \forall a \in \Sigma, \\ &\quad \big(q_1 \sim q_2 \text{ and } q_1 \xrightarrow{a}_1 q_1'\big) \Rightarrow \big(\exists q_2',\ q_1' \sim q_2' \text{ and } q_2 \xrightarrow{a}_2 q_2'\big) \end{aligned} \tag{1}$$

The condition (1) is read T_2 partially simulates *T_1.*

Definition 2.4 *Given $\sim$ a partial simulation of T_1 by T_2, we say that $\sim$ is a* simulation of T_1 by T_2 *if, for each $q_1 \in Q_1$, there exists $q_2 \in Q_2$ such that $q_1 \sim q_2$.*

Definition 2.5 *Given two transition systems on the same alphabet of events, $T_1 = (Q_1, \Sigma, \rightarrow_1)$ and $T_2 = (Q_2, \Sigma, \rightarrow_2)$, a* bisimulation between T_1 and T_2 *is a relation $\sim\ \subseteq Q_1 \times Q_2$ such that $\sim$ is a simulation of T_1 by T_2 and the inverse relation*(ii) *$\sim^{-1}$ is a simulation of T_2 by T_1. In this case we say that* T_1 and T_2 are bisimilar.

Remark 2.6 *One could consider a different notion of bisimulation, let us call it* back-bisimulation *or* backward bisimulation *(see [HKPV98]). This would come from the notion of* partial backward-simulation *defined as partial simulation (Definition 2.3) where the condition (1) is replaced by:*

$$\begin{aligned} &\forall q_1, q_1' \in Q_1,\ \forall q_2' \in Q_2,\ \forall a \in \Sigma, \\ &\quad \big(q_1' \sim q_2' \text{ and } q_1 \xrightarrow{a}_1 q_1'\big) \Rightarrow \big(\exists q_2,\ q_1 \sim q_2 \text{ and } q_2 \xrightarrow{a}_2 q_2'\big) \end{aligned}$$

We say that a bisimulation is a forward stable *relation and that the back-bisimulation is a* backward stable *relation. The difference between these two notions is illustrated on Figures 1 and 2.*

Definition 2.7 *Given a transition system $T = (Q, \Sigma, \rightarrow)$, we can look at bisimulations on $Q \times Q$; they are called* bisimulations on T.

(ii) If $\sim = \{(q_1, q_2) \in Q_1 \times Q_2 \mid q_1 \sim q_2\}$, then $\sim^{-1} = \{(q_2, q_1) \in Q_2 \times Q_1 \mid q_1 \sim q_2\}$.

As already mentioned in the introduction, a motivation for the study of bisimulation is the *reachability problem*. Let us make this problem more precise:

Reachability Problem 2.8 *Given $T = (Q, \Sigma, \to)$ a transition system, $Init \subseteq Q$ and $Fin \subseteq Q$ two subsets of states, is there a* finite path[(iii)] *ρ from $Init$ to Fin?*

If $T = (Q, \to)$ is a *reflexive*[(iv)] transition system then there exists several *trivial "partial" bisimulations on T* given by $\sim_q = \{(q, q') \mid q' \in Q\}$ for $q \in Q$. This implies that there exists a bisimulation between T and a *one-state system* T_0, where $T_0 = (Q_0, \to_0)$ with $Q_0 = \{q_0\}$ and $\to_0 = \{(q_0, q_0)\}$. The bisimulation between T and T_0 is given by $\sim_0 = \{(q, q_0) \mid q \in Q\}$.

Regarding the *reachability problem 2.8*, the bisimulations $\sim_q$ and $\sim_0$ are completely irrelevant. One can have a bisimulation between a completely disconnected reflexive transition system and a single state system (T_0). This gives a motivation for the definition of *bisimulation w.r.t. a partition*. This notion leads to a preservation result on the Reachability Problem (see Lemma 2.15).

Moreover the study of finite bisimulation w.r.t. a partition on dynamical systems leads to the existence of finite bisimulations on subclasses of *hybrid systems*, for examples see [LPS00, Dav99, BMRT04, KV04, BM05, KV06].

Let us give the definition of *bisimulation w.r.t. a partition*.

Definition 2.9 *Given T a transition system, $\mathcal{P}$ a partition of Q and $\sim \subseteq Q \times Q$ a bisimulation, we say that the bisimulation $\sim$* respects the partition *$\mathcal{P}$ if given any $p, q \in Q$ such that $p \sim q$ then p and q belong to the same piece of the partition $\mathcal{P}$. We will speak of* bisimulations w.r.t. $\mathcal{P}$.

Definition 2.10 *Given T a transition system, $\mathcal{P}$ a partition of Q we can define the* coarsest bisimulation on T w.r.t. $\mathcal{P}$, *it is denoted $\sim_{\mathcal{P}}$:*

$$\sim_{\mathcal{P}} \;=\; \bigcup \{\sim \mid \sim \textit{ is a bisimulation on } T \textit{ w.r.t. } \mathcal{P}\}$$

Remarks 2.11 *Definition 2.10 makes sense since the union of bisimulations on T w.r.t. $\mathcal{P}$ is still a bisimulation on T w.r.t. $\mathcal{P}$.*
One can show that the coarsest bisimulation on T w.r.t. $\mathcal{P}$ is an equivalence relation, moreover each piece of the partition $\mathcal{P}$ is an union of equivalence classes of $\sim_{\mathcal{P}}$.

In the case of bisimulations which are equivalence relations, we can define the notion of *quotient of a transition system by such a bisimulation.*

Definition 2.12 *Given a transition system $T = (Q, \Sigma, \to)$ and $\sim$ a bisimulation on T which is an equivalence relation. We can consider the* quotient of T by $\sim$, *denoted by $T/_\sim = (Q/_\sim, \Sigma, \to_\sim)$ and defined as follows:*

- *$Q/_\sim = \{[q]_\sim \mid q \in Q\}$ where $[q]_\sim = \{q' \mid q \sim q'\}$*
- *$[q_1]_\sim \xrightarrow{a}_\sim [q_2]_\sim$ if and only if there exists $q_1' \in [q_1]_\sim$ and $q_2' \in [q_2]_\sim$ such that $q_1' \xrightarrow{a} q_2'$.*

Remark 2.13 *Definition 2.12 makes sense even when we consider an equivalence relation $\sim$ which is not a bisimulation.*

(iii) i.e. $\rho = q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} q_2 \cdots \xrightarrow{a_n} q_n$ with $q_0 \in Init$ and $q_n \in Fin$.
(iv) i.e. for all $q \in Q$ we have that $q \to q$.

Lemma 2.14 *Given T a transition system, $\sim$ a bisimulation on T which is an equivalence relation, then the graph of the natural map $[\cdot]_\sim : Q \to Q/_\sim$ is a functional bisimulation from T to its quotient transition system $T/_\sim$ (see [Cau95, Lemma A.1 p. i]).*

We end this subsection by making precise the folk result that states that bisimulations preserve the *Reachability Problem.*

Lemma 2.15 *Given T, $Init$, Fin as in the* Reachability Problem 2.8, *$\mathcal{P}$ a partition of Q given by $\big\{Init \cap Fin, Fin \setminus Init, Init \setminus Fin, Q \setminus (Init \cup Fin)\big\}$ and $\sim_\mathcal{P}$ a bisimulation on T which is an equivalence relation w.r.t. $\mathcal{P}$. There exists a finite path in T from $Init$ to Fin if and only if there exists a finite path in $T/_{\sim_\mathcal{P}}$ from $Init/_{\sim_\mathcal{P}}$ to $Fin/_{\sim_\mathcal{P}}$.*

Let us notice that the same result holds for back-bisimulation.

2.2 Bisimulation Algorithm

As already mentioned previously, it is an important question to know whether a given infinite system admits a finite bisimulation. Since, for example, the reachability problem is decidable for a finite system effectively described. Moreover it would be nice to have an automatic procedure to build this finite bisimulation. These facts lead to the introduction of the *bisimulation algorithm* which appeared in [BFH91, KS90, Hen96]. Given a transition system $T = (Q, \Sigma, \to)$ and $\mathcal{P}_0$ a finite partition of Q, the bisimulation algorithm iterates the computation of predecessors(v) of the pieces of the partition, let us recall it:

Algorithm 2.16
Initialization: $\mathcal{P} := \mathcal{P}_0$
While $\exists P, P' \in \mathcal{P}\ \exists a \in \Sigma$ *such that* $\varnothing \neq P \cap \mathrm{Pre}_a(P') \neq P$
    Set $P_1 = P \cap \mathrm{Pre}_a(P')$ *and* $P_2 = P \setminus \mathrm{Pre}_a(P')$
    Refine $\mathcal{P} := (\mathcal{P} \setminus \{P\}) \cup \{P_1, P_2\}$
End while
Return $\mathcal{P}$

The following are well-known results on the bisimulation algorithm.

Lemma 2.17 *Given T a transition system and $\mathcal{P}_0$ a finite partition of Q, the bisimulation algorithm terminates if and only if there exists a finite bisimulation on T w.r.t. $\mathcal{P}_0$.*

Lemma 2.18 *If the bisimulation algorithm terminates it provides the coarsest bisimulation on T w.r.t. $\mathcal{P}_0$.*

2.3 Dynamics

Definition 2.19 *A* dynamical system(vi) *is a pair $(\mathcal{M}, \gamma)$ where:*

(v) Given T a transition system and $q \in Q$, the set of a-predecessors of q, denoted $\mathrm{Pre}_a(q)$, is defined by $\mathrm{Pre}_a(q) = \{q' \in Q \mid q' \xrightarrow{a} q\}$, and if $P \subseteq Q$, $\mathrm{Pre}_a(P) = \bigcup_{q \in P} \mathrm{Pre}_a(q)$.

(vi) Our definition of dynamical system is an attempt to generalize the continuous dynamics of hybrid systems ([Hen96]) with no explicit reference to differential equations. This definition, even if rather close, is different from the one given in [Wil91]. Deeper investigation on the links between the two definitions would be relevant work.

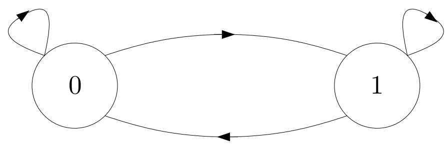

Fig. 3: A finite automaton

- *$\mathcal{M} = \langle M, < \rangle$ is a totally ordered structure,*
- *$\gamma : M^{k_1} \times M \rightarrow M^{k_2}$ is a function.*

The function γ is called the dynamics of the dynamical system. More generally, we can consider the case where γ is defined on subsets of $\mathcal{M}$ that is $\gamma : V_1 \times V \rightarrow V_2$ with $V_1 \subseteq M^{k_1}$, $V \subseteq M$ and $V_2 \subseteq M^{k_2}$.

In the sequel we assume the range of γ is equal to M^{k_2}. Classically, when M is the field of the reals, we see M as the time, $M^{k_1} \times M$ as the space-time, M^{k_2} as the (output) space and M^{k_1} as the input space. We keep this terminology in the more general context of a structure $\mathcal{M}$.

In this presentation time and space have the same underlying structure (i.e. $\mathcal{M}$) this comes from our presentation in [BMRT04] where we needed the whole dynamical system to be definable in the o-minimal structure $\mathcal{M}$. However we can imagine dynamical system with dynamics $\gamma : V_1 \times V \rightarrow V_2$ where V is a totally ordered set and V_1, V_3 are defined in completely different structure. This should not affect the results presented in the sequel.

The definition of *dynamical system* encompasses a lot of different behaviors. Let us give some examples.

Example 2.20 *Let $\mathcal{M} = \langle \mathbb{N}, < \rangle$ and the dynamics $\gamma : \{0,1\} \times \{0,1\} \rightarrow \{0,1\}$ is given by $\gamma(x,t) = (x+t) \mod 2$. The transition system associated with this dynamical system (see Definition 2.24) is in fact a finite automaton (see Figure 3).*

Example 2.21 *We can recover the continuous dynamics of the* timed automaton *(see [AD94]). In this case, we have that $\mathcal{M} = \langle \mathbb{R}, < \rangle$ and the dynamics $\gamma : \mathbb{R}^n \times [0, +\infty[\rightarrow \mathbb{R}^n$ is defined as follows.*

$$\gamma(x_1, ..., x_n, t) = (x_1 + t, ..., x_n + t)$$

Example 2.22 *Definition 2.19 also allows dynamical systems with non deterministic*(vii) *behavior. Let us consider $(\mathcal{M}, \gamma)$ where each point of the plane has two possible behaviors: "to go to the right" or "to go up" (see Figure 4 on page 22). More precisely we have that $\mathcal{M} = \langle \mathbb{R}, < \rangle$ and $\gamma : \mathbb{R}^3 \times \mathbb{R} \rightarrow \mathbb{R}^2$ is defined as follows.*

$$\gamma(x_1, x_2, p, t) = \begin{cases} (x_1 + t, x_2) & \text{if } p \geqslant 0 \\ (x_1, x_2 + t) & \text{if } p < 0 \end{cases}$$

Definition 2.23 *If we fix a point $x \in M^{k_1}$, the set $\Gamma_x = \{\gamma(x,t) \mid t \in M\} \subseteq M^{k_2}$ is called the trajectory determined by x.*

(vii) The non determinism comes in fact from the associated transition system, see Definition 2.24.

We define a transition system associated with the dynamical system, this definition is an adaptation to our context of the classical *continuous transition* in the case of hybrid system (see [LPS00] for example).

Definition 2.24 *Given $(\mathcal{M}, \gamma)$ a dynamical system, we define a* transition system $T_\gamma = (Q, \rightarrow_\gamma)$ associated with the dynamical system *by:*

- *the set Q of states is M^{k_2};*
- *the transition relation $y_1 \rightarrow_\gamma y_2$ is given by:*

$$\exists x \in M^{k_1},\ \exists t_1, t_2 \in M,\ \big(t_1 \leqslant t_2 \text{ and } \gamma(x, t_1) = y_1 \text{ and } \gamma(x, t_2) = y_2\big)$$

Remark 2.25 *Let us notice that T_γ is a reflexive transition system.*

Remark 2.26 *The transition system T_γ is in general not transitive. To illustrate this fact, let us consider Example 2.22. Given the three points of the output space $y_1 = (0,0)$, $y_2 = (0,1)$ and $y_3 = (1,1)$, we clearly have that $y_1 \not\rightarrow_\gamma y_3$ since $y_1 \rightarrow_\gamma y_2$ and $y_2 \rightarrow_\gamma y_3$. Indeed $y_1 = \gamma(0,0,-1,0)$, $y_2 = \gamma(0,0,-1,1) = \gamma(0,1,1,0)$ and $y_3 = \gamma(0,1,1,1)$.*

3 Words and dynamics

Given a dynamical system $(\mathcal{M}, \gamma)$ and $\mathcal{P}$ a finite partition of the space M^{k_2}, an interesting question is to know if there exists a finite bisimulation of $(\mathcal{M}, \gamma)$ w.r.t. $\mathcal{P}$. If such a bisimulation exists the *bisimulation algorithm* 2.16 provides the coarsest one by iterating the computation of the predecessors of the pieces of the partition $\mathcal{P}$. The goal of this section is to give another procedure that computes the coarsest bisimulation on a dynamical system $(\mathcal{M}, \gamma)$ (i.e. a bisimulation on T_γ) w.r.t. a partition $\mathcal{P}$. Our approach is in some sense more global that the *bisimulation algorithm*. We use the idea introduced in [BMRT04] which consists in encoding the dynamics of $(\mathcal{M}, \gamma)$ through the partition $\mathcal{P}$ by words on this partition. Let us first explain how we associate a word with a trajectory.

3.1 Encoding trajectories by words

First let us define the notion of *word* in this general (possibly uncountable) context. This definition is inspired from [BC01], see also [Tru89, Rab03].

Definition 3.1 *Given $\mathcal{P}$ a finite set (called the* alphabet*), M a totally ordered set, a* word ω on $\mathcal{P}$ *is a function from M to $\mathcal{P}$; the word ω is also denoted in a sequence-like notation by $(\omega_i)_{i \in M}$ where $\omega_i \in \mathcal{P}$ is the image of the element i under the function ω.*

We recover the classical finite words or ω-words when the set M is respectively finite or equal to $\mathbb{N}$.

Example 3.2 *Let us consider the finite set $\mathcal{P} = \{A, B\}$. We give three examples of words on $\mathcal{P}$.*

1. *Given the finite set $M_1 = \{1,2,3,4\}$ equipped with the natural ordering and the function $\omega_1 : M_1 \rightarrow \mathcal{P}$ such that $\omega_1(1) = A$, $\omega_1(2) = B$, $\omega_1(3) = A$ and $\omega_1(4) = B$, we recover an example of* finite word. *In this case ω_1 is classically denoted $ABAB$.*

2. *Given the set of natural number $M_2 = \mathbb{N}$ equipped with the natural ordering and the function $\omega_2 : M_2 \to \mathcal{P}$ such that $\omega_2(n) = A$ if n is even and $\omega_2(n) = B$ if n is odd, we recover an example of ω-*word. *In this case ω_2 is classically denoted $(AB)^\omega$.*

3. *Given the set of real number $M_3 = \mathbb{R}$ equipped with the natural ordering and the function $\omega_3 : M_3 \to \mathcal{P}$ such that $\omega_3(n) = A$ if $n \in \mathbb{Q}$ and $\omega_2(n) = B$ if $n \in \mathbb{R} \setminus \mathbb{Q}$, we have a "degenerated" example of word.*

We need to introduce basic notions related to words in this general context. For finite words, we adopt the classical notations.

Definition 3.3 *Given $\omega : M \to \mathcal{P}$ a word on $\mathcal{P}$, a* subword of ω *is given by $\omega_s : M' \to \mathcal{P}$ where $M' \subseteq M$ is an arbitrary subset of M considered with the order induced from M.*

Definition 3.4 *Given $\omega : M \to \mathcal{P}$ a word on $\mathcal{P}$, a* suffix of ω *is a subword of a particular form. A subword $\omega_s : M' \to \mathcal{P}$ is a* suffix *if and only if $M' = \{t \mid t \geqslant t_0\}$ or $M' = \{t \mid t > t_0\}$ for some $t_0 \in M$. In the same way we can define the notion of* prefix.

Definition 3.5 *Given $\omega_1 : M_1 \to \mathcal{P}$ and $\omega_2 : M_2 \to \mathcal{P}$ two words on $\mathcal{P}$, the* concatenation of the words ω_1 and ω_2 *is defined by the word $\omega_1\omega_2 : M_1 \dot\cup M_2 \to \mathcal{P}$ where $\omega_1\omega_2\restriction_{M_1} = \omega_1$ and $\omega_1\omega_2\restriction_{M_2} = \omega_2$ and where the order on $M_1 \dot\cup M_2$ is the order induced from M_1 on M_1, the order induced from M_2 on M_2 and $\forall m_1 \in M_1$, $\forall m_2 \in M_2$ we have that $m_1 < m_2$ in $M_1 \dot\cup M_2$.*

We are now ready to build words associated with trajectories. Given $(\mathcal{M}, \gamma)$ a dynamical system and $\mathcal{P}$ a finite partition of M^{k_2}, given $x \in M^{k_1}$ we associate a word with the trajectory Γ_x in the following way. We consider the sets $\{t \in M \mid \gamma(x,t) \in P\}$ for each $P \in \mathcal{P}$. This gives a partition of the time M. In order to define a word on $\mathcal{P}$ associated with the trajectory determined by x, we need to define the set of intervals $\mathcal{F}_x$.

$$\begin{aligned}\mathcal{F}_x = \big\{ I \;\big|\; &(I \text{ is a time interval or a point}) \text{ and is maximal for the property} \\ &\exists P \in \mathcal{P},\ \forall t \in I,\ \gamma(x,t) \in P \big\}.\end{aligned}$$

For each x, the set $\mathcal{F}_x$ is totally ordered by the order induced from M. Let us note that the set $\mathcal{F}_x$ can be equal to M itself. This allows us to define *the word on $\mathcal{P}$ associated with* Γ_x denoted ω_x.

Definition 3.6 *Given $x \in M^{k_1}$,* the word associated with Γ_x *is given by the function $\omega_x : \mathcal{F}_x \to \mathcal{P}$ defined by:*

$$\omega_x(I) = P \qquad \text{where } I \in \mathcal{F}_x \text{ is such that } \forall t \in I \quad \gamma(x,t) \in P.$$

Let us note that given $x \in M^{k_1}$, there exists a unique word ω_x on $\mathcal{P}$ associated with the trajectory Γ_x. The intuition behind the introduction of $\mathcal{F}_x$ is the following. We want successive[(viii)] letters of the words ω_x to be different.

Definition 3.7 *We denote by $\Omega_{\mathcal{P}}$ the* set of words associated with $(\mathcal{M}, \gamma)$ w.r.t. $\mathcal{P}$. *We have that $\Omega_{\mathcal{P}}$ is a set of words on $\mathcal{P}$.*

[(viii)] The notion of successive letters is only defined for *"well behaving"* dynamical systems.

The set $\Omega_{\mathcal{P}}$ gives in some sense a complete *static* description of the dynamical system $(\mathcal{M}, \gamma)$ through the partition $\mathcal{P}$. In order to recover the *dynamics*, we need further information. This is the object of the following subsection.

3.2 Dynamical type

Given a point x of the input space M^{k_1}, we have associated with x a trajectory Γ_x and a word ω_x. If we consider (x,t) a point of the space-time $M^{k_1} \times M$, it corresponds to a point $\gamma(x,t)$ lying on Γ_x. To recover in some sense the position of $\gamma(x,t)$ on Γ_x from ω_x, we associate with (x,t) a suffix of the word ω_x denoted $\omega_{(x,t)}$. The construction of $\omega_{(x,t)}$ is similar to the construction of ω_x. We need to introduce the set of intervals

$$\mathcal{F}_{(x,t)} = \big\{ I \cap \{t' \mid t' \geqslant t\} \;\big|\; I \in \mathcal{F}_x \big\}.$$

For each (x,t), the set $\mathcal{F}_{(x,t)}$ is totally ordered by the order induced from M. This allows us to define *the suffix of the word ω_x associated with time t* denoted $\omega_{(x,t)}$.

Definition 3.8 *Given $(x,t) \in M^{k_1} \times M$,* the suffix of the word ω_x associated with time t *is given by the function $\omega_{(x,t)} : \mathcal{F}_{(x,t)} \to \mathcal{P}$ defined by:*

$$\omega_{(x,t)}(I) = P \qquad \textit{where } I \in \mathcal{F}_{(x,t)} \textit{ is such that } \forall t' \in I \quad \gamma(x,t') \in P.$$

Due to the particular form of the suffixes $\omega_{(x,t)}$, it makes sense to define *the first letter of $\omega_{(x,t)}$*.

Definition 3.9 *Given $(x,t) \in M^{k_1} \times M$,* the first letter of the suffix $\omega_{(x,t)}$ *is given by $\omega_x(I)$ where I is the interval of $\mathcal{F}_x$ such that $t \in I$. We denote the first letter of $\omega_{(x,t)}$ by $\mathbf{F}(\omega_{(\mathbf{x},\mathbf{t})})$.*

Let us notice that given (x,t) a point of the space-time $M^{k_1} \times M$ there is a unique suffix $\omega_{(x,t)}$ of ω_x associated with (x,t).

Given a point $y \in M^{k_2}$ it may have several (x,t) such that $\gamma(x,t) = y$ and so several suffixes are associated with y. In other words, given $y \in M^{k_2}$, the *future* of y is non deterministic, and so a single suffix $\omega_{(x,t)}$ is not enough to recover the dynamics of the transition system through the partition $\mathcal{P}$. To encode the dynamical behavior of a point y of the output space M^{k_2} through the partition $\mathcal{P}$, we introduce several notions of *dynamical type* of a point y w.r.t. $\mathcal{P}$.

Definition 3.10 *Given a dynamical system $(\mathcal{M}, \gamma)$, a finite partition $\mathcal{P}$ of M^{k_2}, a point $y \in M^{k_2}$ the* suffix dynamical type of y w.r.t. $\mathcal{P}$ *is denoted $Suf_{\mathcal{P}}(y)$ and defined by:*

$$Suf_{\mathcal{P}}(y) = \{\omega_{(x,t)} \mid \gamma(x,t) = y\}.$$

We have that $Suf_{\mathcal{P}}(y)$ is a subset of suffixes of words of $\Omega_{\mathcal{P}}$.

Definition 3.11 *Given a dynamical system $(\mathcal{M}, \gamma)$, a finite partition $\mathcal{P}$ of M^{k_2}, an integer $n \in \mathbb{N}$, a point $y \in M^{k_2}$ the* n-subword dynamical type of y w.r.t. $\mathcal{P}$ *is denoted $nSub_{\mathcal{P}}(y)$ and defined by:*

$$\begin{aligned} nSub_{\mathcal{P}}(y) =\{\omega \mid \gamma(x,t) = y \textit{ and } \omega \textit{ is a subword of } \omega_{(x,t)} \\ \textit{and the length of } \omega, |\omega| \leqslant n \textit{ and } \mathbf{F}(\omega) = \mathbf{F}(\omega_{(\mathbf{x},\mathbf{t})})\}. \end{aligned}$$

Definition 3.12 *Given a dynamical system* $(\mathcal{M}, \gamma)$*, a finite partition* $\mathcal{P}$ *of* M^{k_2}*, a point* $y \in M^{k_2}$ *the* subword dynamical type of y w.r.t. $\mathcal{P}$ *is denoted* $Sub_{\mathcal{P}}(y)$ *and defined by:*

$$Sub_{\mathcal{P}}(y) = \bigcup_{n \in \mathbb{N}} nSub_{\mathcal{P}}(y).$$

We have that $nSub_{\mathcal{P}}(y)$ *is a subset of* $Sub_{\mathcal{P}}(y)$ *for all* $n \in \mathbb{N}$.

Notations 3.13 *If we want to talk about a dynamical type of the point* y *without specifying if it is a subword,* n*-subword or suffix dynamical type, we use the notation* $\mathcal{T}_{\mathcal{P}}(y)$.

Our goal is to refine the partition $\mathcal{P}$ in order to build a bisimulation w.r.t. $\mathcal{P}$. For this purpose we consider the equivalence relation between points of the output space M^{k_2} "*to have same dynamical type w.r.t.* $\mathcal{P}$". This equivalence relation induces a new partition of the output space M^{k_2} which refines $\mathcal{P}$.

Definition 3.14 *We denote by* $\mathcal{T}(\mathcal{P})$ *the refinement of the partition* $\mathcal{P}$ *obtained by considering the equivalence relation* $\equiv_{\mathcal{T}(\mathcal{P})}$ *on* M^{k_2} *given by:*

$$y_1 \equiv_{\mathcal{T}(\mathcal{P})} y_2 \quad \text{if and only if} \quad \mathcal{T}_{\mathcal{P}}(y_1) = \mathcal{T}_{\mathcal{P}}(y_2).$$

Notations 3.15 *The partition* $\mathcal{T}(\mathcal{P})$ *is respectively denoted* $Suf(\mathcal{P})$, $nSub(\mathcal{P})$ *and* $Sub(\mathcal{P})$ *in the case of the suffix,* n*-subword and subword dynamical type.*

Remark 3.16 *The* $nSub(\mathcal{P})$ *partitions are only relevant for* $n \geqslant 2$. *Indeed,* $0Sub(\mathcal{P}) = \{M^{k_2}\}$ *and* $1Sub(\mathcal{P}) = \mathcal{P}$. *This is why in the sequel of the paper when we talk about* n*-subword dynamical typen we always assume* $n \geqslant 2$.

Remark 3.17 *The different dynamical types induced different partitions. Those partitions are related as follows in term of refinement.*

$$\mathcal{P} \supseteq 2Sub(\mathcal{P}) \supseteq ... \supseteq nSub(\mathcal{P}) \supseteq ... \supseteq \bigcap_{i \in \mathbb{N}} (iSub(\mathcal{P})) = Sub(\mathcal{P}) \supseteq Suf(\mathcal{P})$$

Once $\mathcal{T}(\mathcal{P})$ is computed[(ix)] two possibilities can occur. On one hand we can have that $\mathcal{P} = \mathcal{T}(\mathcal{P})$, in this situation, we have that $\mathcal{P}$ is in fact a bisimulation on $(\mathcal{M}, \gamma)$ w.r.t. $\mathcal{P}$ (see Theorem 3.27). On the other hand we can have that $\mathcal{P} \neq \mathcal{T}(\mathcal{P})$. In this case we can refine $\mathcal{T}(\mathcal{P})$ by considering the dynamical types on $\mathcal{T}(\mathcal{P})$. We start by building words on $\mathcal{T}(\mathcal{P})$ associated with the trajectories Γ_x to obtain $\Omega_{\mathcal{T}(\mathcal{P})}$ and finally we obtain the different kinds of dynamical types w.r.t. $\mathcal{T}(\mathcal{P})$. This leads to a third partition $\mathcal{T}(\mathcal{T}(\mathcal{P}))$ denoted $\mathcal{T}^2(\mathcal{P})$. Again two situations can occur: $\mathcal{T}(\mathcal{P}) = \mathcal{T}^2(\mathcal{P})$ or $\mathcal{T}(\mathcal{P}) \neq \mathcal{T}^2(\mathcal{P})$. This allows us to consider a general procedure that we describe in the following subsection.

Remark 3.18 *The readers familiar with the classical* bisimulation algorithm *(2.16) realised that the partition induced by* $2Sub(\mathcal{P})$ *is sufficent in order to compute bisimulation. We investigate the other dynamical types in order to "accelerate" in some sense the construction of the bisimulation, in particular when the* bisimulation algorithm *does not terminate (see Corollary 3.37 and Example 4.1).*

(ix) the meaning of the word "*computed*" is discussed in Remark 3.20.

3.3 Procedure $Bisi\omega$

By starting with some initial partition $\mathcal{P}_0$ we have seen how to build a new partition $\mathcal{T}(\mathcal{P}_0)$. We iterate the construction to obtain a sequence of partitions $\left(\mathcal{T}^i(\mathcal{P}_0)\right)_{i\in\mathbb{N}}$ such that for each $i \in \mathbb{N}$ we have that the partition $\left(\mathcal{T}^{i+1}(\mathcal{P}_0)\right) = \mathcal{T}\left(\mathcal{T}^i(\mathcal{P}_0)\right)$ corresponds to the partition induced by the dynamical types w.r.t. $\mathcal{T}^i(\mathcal{P}_0)$. This construction is summarized by the following procedure, we call this procedure $Bisi\omega$.

Procedure 3.19
Initialization: $\mathcal{P} := \mathcal{P}_0$
Do
  Compute *the set of words* $\Omega_{\mathcal{P}}$
  Associate $T_{\mathcal{P}}(y)$ *with each* $y \in M^{k_2}$ *use it to* ***Build*** $\mathcal{T}(\mathcal{P})$
  If $\mathcal{P} = \mathcal{T}(\mathcal{P})$
    Then Return $\mathcal{P}$
    Else $\mathcal{P} := \mathcal{T}(\mathcal{P})$
End Do

Remark 3.20 *Procedure* $Bisi\omega$ *is merely conceptual. Indeed in general it is far to be computable. One main problem to be settled is to determine when two general words, as defined in our context, are equal. Let us be more explicit, assume that* ω_x *and* $\omega_{x'}$ *are words respectively associated with the trajectories* Γ_x *and* $\Gamma_{x'}$*. The problem is that* ω_x *and* $\omega_{x'}$ *are not equal as functions since their domains are different: they are respectively* $\mathcal{F}_x$ *and* $\mathcal{F}_{x'}$*. Since the order on* $\mathcal{F}_x$ *and* $\mathcal{F}_{x'}$ *is possibly not discrete, and even not well-founded, we need to introduce a general notion of* synchronization *for ordered sets which is nothing else than an isomorphism of ordered structures. So we will say that* ω_x *and* $\omega_{x'}$ *are equal if and only there exist an isomorphism* σ *between the ordered structures* $\mathcal{F}_x$ *and* $\mathcal{F}_{x'}$ *such that for all* $I \in \mathcal{F}_x$ *we have that* $\omega_x(I) = \omega_{x'}(\sigma(I))$*.*
Let us remark that the partition $\mathcal{T}(\mathcal{P})$ *is in general even not* definable *by a first-order* $\mathcal{L}$*-formula where* $\mathcal{L}$ *is the language given by the order and the initial partition:* $\mathcal{L} = \{<, \mathcal{P}, \gamma\}$*.*
However we have shown that in the case of o-minimal structures *the first step of Procedure* $Bisi\omega$ *already provides interesting results. A discussion about computation of the words and the dynamical types in this particular case can be found in [BM05].*

Remark 3.21 *Given* $P \in \mathcal{T}(\mathcal{P})$ *it can be seen as a subset of* M^{k_2}*, or it can be seen as a dynamical type w.r.t.* $\mathcal{P}$ *i.e. a set of words on* $\mathcal{P}$*.*

Lemma 3.22 *Given* y *a point of the output space* M^{k_2}*, the first letter of each* $\omega \in \mathcal{T}_{\mathcal{P}}(y)$ *is* P *where* $P \in \mathcal{P}$ *and* $y \in P$*. This is true for the each kind of dynamical type defined previously.*

Proof: This is an immediate consequence of the different definitions of the dynamical types. □

Lemma 3.23 *Given a dynamical system* $(\mathcal{M}, \gamma)$ *and* $\mathcal{P}$ *a partition of* M^{k_2} *we have that* $\mathcal{T}(\mathcal{P})$ *refines* $\mathcal{P}$*.*

Proof: This a direct consequence of Remark 3.17 and the fact that $\mathcal{P} = 1\mathrm{Sub}(\mathcal{P})$. □

Remark 3.24 *By Lemma 3.23 we have that Procedure Bisiω generates a decreasing sequence of partitions:*

$$\mathcal{P} \supseteq \mathcal{T}(\mathcal{P}) \supseteq \mathcal{T}^2(\mathcal{P}) \supseteq \cdots \supseteq \mathcal{T}^i(\mathcal{P}) \supseteq \cdots$$

To illustrate how Procedure *Bisiω* works, let us give an example.

Example 3.25 *We consider the dynamical system $(\mathcal{M}, \gamma)$ of Example 2.22. We associate to $(\mathcal{M}, \gamma)$ the initial partition $\mathcal{P} = \{A_1, A_2\}$ where $A_1 = \{(0,0)\}$ and $A_2 = \mathbb{R}^2 \setminus \{(0,0)\}$ (see Figure 4). We apply Procedure Bisiω on $(\mathcal{M}, \gamma)$ with $\mathcal{P}$ as initial partition and using the* suffix dynamical type.

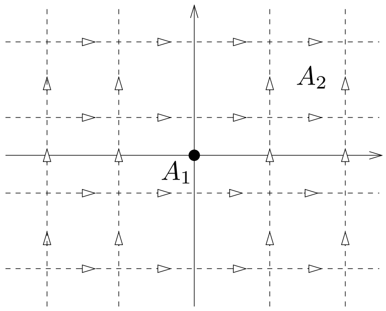

Fig. 4: $\mathcal{P} = \{A_1, A_2\}$

First, we compute the set of words w.r.t. $\mathcal{P}$,

$$\Omega_{\mathcal{P}} = \{A_2, A_1A_2, A_2A_1A_2\}.$$

From $\Omega_{\mathcal{P}}$ we see that there exist three dynamical types w.r.t. $\mathcal{P}$:

$$B_1 = \{A_2, A_2A_1A_2\}\ ;\ B_2 = \{A_1A_2\}\ ;\ B_3 = \{A_2\}.$$

These dynamical types lead to the new partition $Suf(\mathcal{P}) = \{B_1, B_2, B_3\}$ (see Figure 5) where $B_1 = \{(y_1, 0) \mid y_1 < 0\} \cup \{(0, y_2) \mid y_2 < 0\}$, $B_2 = \{(0,0)\}$ and $B_3 = \mathbb{R}^2 \setminus (B_1 \cup B_2)$. Notice that $Suf(\mathcal{P})$ is a strict refinement of $\mathcal{P}$, so we iterate the construction. We compute the set of words w.r.t. $Suf(\mathcal{P})$,

$$\Omega_{Suf(\mathcal{P})} = \{B_1B_2B_3, B_3B_1B_3, B_2B_3, B_1B_3, B_3\}.$$

From $\Omega_{Suf(\mathcal{P})}$ we see that there exist four dynamical types w.r.t. $Suf(\mathcal{P})$:

$$C_1 = \{B_1B_2B_3, B_1B_3\}\ ;\ C_2 = \{B_3B_1B_3\}\ ;\ C_3 = \{B_2B_3\}\ ;\ C_4 = \{B_3\}$$

Again these dynamical types lead to a new partition $Suf^2(\mathcal{P}) = \{C_1, C_2, C_3, C_4\}$ (see Figure 6). Let us compute the set of words w.r.t. $Suf^2(\mathcal{P})$,

$$\Omega_{Suf^2(\mathcal{P})} = \{C_2C_1C_4, C_1C_3C_4, C_1C_4, C_3C_4, C_4\}.$$

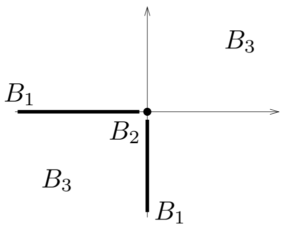

Fig. 5: Suf $(\mathcal{P})$

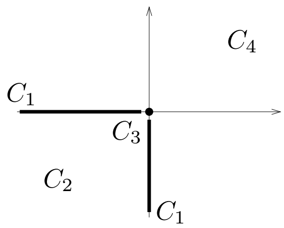

Fig. 6: $\mathrm{Suf}^2(\mathcal{P})$

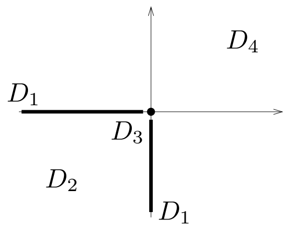

Fig. 7: $\mathrm{Suf}^3(\mathcal{P})$

From $\Omega_{Suf^2(\mathcal{P})}$ *we see that there exist four dynamical types w.r.t.* $Suf^2(\mathcal{P})$*:*

$$D_1 = \{C_1C_3C_4, C_1C_4\}\ ;\ D_2 = \{C_2C_1C_4\}\ ;\ D_3 = \{C_3C_4\}\ ;\ D_4 = \{C_4\}$$

Those four dynamical types do not refine the partition $Suf^2(\mathcal{P})$ *(see Figure 7). In other words, we have the following equality* $Suf^2(\mathcal{P}) = Suf^3(\mathcal{P})$*. One can check that* $Suf^2(\mathcal{P})$ *is a bisimulation on* $(\mathcal{M}, \gamma)$ *w.r.t.* $\mathcal{P}$*.*

Remark 3.26 *The dynamical system of Example 3.25 is non-deterministic. Indeed two trajectories are associated with each point* y *of the output space* M^{k_2}*. In the papers [BMRT04, BM05], we were unable to deal with such situations.*

The situation of Example 3.25 is not a particular case. Indeed if Procedure *Bisi*ω terminates, it provides a bisimulation. We can now state the main result of the paper:

Theorem 3.27 *Let* $(\mathcal{M}, \gamma)$ *be a dynamical system, let* T_γ *be the associated transition system on* M^{k_2}*, and let* $\mathcal{P}_0$ *be a finite partition of* M^{k_2}*. If Procedure Bisi*ω *terminates, then it provides a bisimulation on* T_γ *w.r.t.* $\mathcal{P}_0$*.*

Proof: By hypothesis, Procedure *Bisi*ω returns a partition $\mathcal{P}$ such that $\mathcal{P} = \mathcal{T}(\mathcal{P})$. To prove that the equivalence relation induced from $\mathcal{P}$ is a bisimulation on T_γ w.r.t. $\mathcal{P}_0$. We will show that given any $y_1, y_2 \in A$ and $y_1' \in B$ (for some A, $B \in \mathcal{P}$) if $y_1 \to_\gamma y_1'$ then there exists $y_2' \in B$ such that $y_2 \to_\gamma y_2'$.

Since $\mathcal{P} = \mathcal{T}(\mathcal{P})$, A corresponds to a dynamical type on $\mathcal{P}$ (i.e. an element of $\mathcal{T}(\mathcal{P})$). Hence we have that $T_\mathcal{P}(y_1) = T_\mathcal{P}(y_2)$. Depending of the kind of dynamical type, the argument to find y_2' is slightly different. We do the rest of the proof with the suffix dynamical type, the other[(x)] cases are similar.

Since $y_1 \to_\gamma y_1'$ there exists $x_1 \in M^{k_1}$ and t_1, $t_1' \in M$ with $t_1 \leqslant t_1'$ such that $\gamma(x_1, t_1) = y_1$ and $\gamma(x_1, t_1') = y_1'$. By definition of the suffix dynamical type, $\omega_{(x_1,t_1)} \in \mathrm{Suf}_\mathcal{P}(y_1)$. Since $y_1 \in A$ and $y_1' \in B$, we have that AB is a subword[(xi)] of $\omega_{(x_1,t_1)}$. Using the fact that $\mathrm{Suf}_\mathcal{P}(y_1) = \mathrm{Suf}_\mathcal{P}(y_2)$, we can find $x_2 \in M^{k_1}$ and $t_2 \in M$ such that $\gamma(x_2, t_2) = y_2$ and $\omega_{(x_2,t_2)} = \omega_{(x_1,t_1)}$. Hence it is possible to find an interval (or a point) $I \in \mathcal{F}_{(x_2,t_2)}$ such that $\omega_{(x_2,t_2)}(I) = B$. We pick any point $t_2' \in I$ and clearly we have that $y_2' = \gamma(x_2, t_2')$ is the desired point.

(x) This of course does not hold for the 0-subword and the 1-subword dynamical types.

(xi) Formally, we have to take $\{t_1, t_1'\} = M' \subseteq M$.

We have that $\mathcal{P}$ respects $\mathcal{P}_0$ by iterating Lemma 3.23. □

Corollary 3.28 *Under the assumptions of Theorem 3.27 we have that if there exists $\mathcal{P}_0'$ a refinement of $\mathcal{P}_0$ such that $\mathcal{P}_0' = \mathcal{T}(\mathcal{P}_0')$ then $\mathcal{P}_0'$ is a bisimulation on T_γ w.r.t. $\mathcal{P}_0$.*

Unfortunately, Procedure $Bisi\omega$ does not provide in general the coarsest bisimulation on T_γ w.r.t. $\mathcal{P}$. Here are two examples that illustrate this fact.

Example 3.29 *We consider a dynamical system where the output space consists of two parallel straight lines and the dynamics is completely deterministic, given a point on one of the lines, it goes to infinity without leaving the line. In other words, we have that $\mathcal{M} = \langle \mathbb{R}, < \rangle$ and the dynamics $\gamma : \mathbb{R} \times \{0,1\} \times \mathbb{R} \to \mathbb{R} \times \{0,1\}$ is defined by $\gamma(x_1, x_2, t) = (x_1 + t, x_2)$. We associate with $(\mathcal{M}, \gamma)$ the partition $\mathcal{P} = \{A, B\}$ where $B = \mathbb{R} \times \{0,1\} \setminus A$ and A is defined as follows:*

$$A = \big\{ (n - (1/m), 0) \ \mid n \in \mathbb{N}, m \in \mathbb{N} \setminus \{0\} \big\} \cup \big\{ (n, 1) \mid n \in \mathbb{N} \big\}.$$

Let us consider the suffix dynamical type of the two points $y_1 = (1/2, 0)$ and $y_2 = (1,1)$:

$$Suf_{\mathcal{P}}(y_1) = \big((AB)^\omega\big)^\omega \qquad \text{and} \qquad Suf_{\mathcal{P}}(y_2) = (AB)^\omega.$$

Clearly, y_1 and y_2 do not have the same suffix dynamical type w.r.t. $\mathcal{P}$ however one can show that $\mathcal{P}$ is the coarsest bisimulation w.r.t. $\mathcal{P}$.

Remark 3.30 *In the previous example, the fact that the partition is too fine is due to the fact that the bisimulation does not distinguish $(AB)^\omega$ and $((AB)^\omega)^\omega$. Indeed, in this case the transition system T_γ is completely deterministic, so the bisimulation only need to know that the dynamics goes infinitely often from A to B and from B to A. The bisimulation does not care about the* "kind of infinity"*. It is well-known that in the case of deterministic finite transition systems, the bisimulation and language correspond. Example 3.29 shows that considering more complex system make not clear how the notions of language equivalence and bisimulation are related.*

Remark 3.31 *Let us notice that if we consider n-subword or subword dynamical type on Example 3.29 we obtain the coarsest bisimulation.*

However when considering non deterministic system, the next example shows that using 3-subword dynamical type leads already to a too fine analysis.

Example 3.32 *Let us consider the dynamical system of Figure 8 with the partition $\mathcal{P} = \{A, B, C\}$. Let us consider the* 3*-subword dynamical type of the two points y_1 and y_2.*

$$3Sub_{\mathcal{P}}(y_1) = \{ABC\} \qquad \text{and} \qquad 3Sub_{\mathcal{P}}(y_2) = \{AB, AC\}.$$

Clearly, y_1 and y_2 do not have the same 3*-subword dynamical type w.r.t. $\mathcal{P}$, however one can show that $\mathcal{P}$ is the coarsest bisimulation w.r.t. $\mathcal{P}$.*

Nevertheless if we look at the 2-subword dynamical type, we always obtain the coarsest bisimulation as stated in the following theorem.

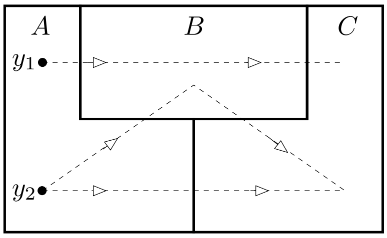

Fig. 8: 3-subword dynamical types do not provide the coarsest bisimulation

Theorem 3.33 *Let $(\mathcal{M}, \gamma)$ be a dynamical system, let T_γ be the associated transition system on M^{k_2}, and let $\mathcal{P}_0$ be a finite partition of M^{k_2}. If Procedure Bisiω terminates with the 2-subword dynamical type, then it provides the coarsest bisimulation on T_γ w.r.t. $\mathcal{P}_0$.*

Proof: By Theorem 3.27 we already know that Procedure *Bisiω* provides a bisimulation on T_γ w.r.t. $\mathcal{P}_0$, it remains to show that it is the coarsest. We proceed *ab absurdo*. Hence we suppose there exists some step of Procedure *Bisiω* and y_1, $y_2 \in M^{k_2}$ such that $y_1 \sim_{\mathcal{P}_0} y_2$ and $2\mathrm{Sub}_{\mathcal{P}}(y_1) \neq 2\mathrm{Sub}_{\mathcal{P}}(y_2)$. We can choose this step such that each piece of $\mathcal{P}$ is a union of equivalence classes for $\sim_{\mathcal{P}_0}$. We have that $y_1 \in A$ for some $A \in \mathcal{P}$. Since $2\mathrm{Sub}_{\mathcal{P}}(y_1) \neq 2\mathrm{Sub}_{\mathcal{P}}(y_2)$, we can suppose $AB \in 2\mathrm{Sub}_{\mathcal{P}}(y_1)$ and $AB \notin 2\mathrm{Sub}_{\mathcal{P}}(y_2)$ for some $B \in \mathcal{P}$. This means that there exists $y_1' \in B$ with $y_1 \to_\gamma y_1'$ and that it is impossible to find $y_2' \in B$ with $y_2 \to_\gamma y_2'$. Since $y_1 \sim_{\mathcal{P}_0} y_2$ and B is a union of equivalence classes for $\sim_{\mathcal{P}_0}$, this contradicts that $\sim_{\mathcal{P}_0}$ is a bisimulation. □

Corollary 3.34 *Let $(\mathcal{M}, \gamma)$ be a dynamical system, let T_γ be the associated transition system on M^{k_2}, and let $\mathcal{P}_0$ be a finite partition of M^{k_2}. Procedure Bisiω terminates with the 2-subword dynamical type if and only if there exists a finite bisimulation on T_γ w.r.t. $\mathcal{P}_0$.*

Proof: If there exists a finite bisimulation on T_γ w.r.t. $\mathcal{P}_0$, the proof of Theorem 3.33 shows that Procedure *Bisiω* terminates.

Let us now suppose that Procedure *Bisiω* terminates. Since $\mathcal{P}_0$ is finite, the number of 2-subword dynamical types is finite, (i.e. $2\mathrm{Sub}(\mathcal{P}_0)$ is finite). By an easy induction using the same argument, one can see that $2\mathrm{Sub}^i(\mathcal{P}_0)$ is finite for all $i \in \mathbb{N}$. Hence if Procedure *Bisiω* terminates, we clearly have that there exists a finite bisimulation on T_γ w.r.t. $\mathcal{P}_0$. □

In the sequel, we investigate extra assumptions which provide that Procedure *Bisiω* terminates with the coarsest bisimulation.

Theorem 3.35 *Let $(\mathcal{M}, \gamma)$ be a dynamical system and let $\mathcal{P}_0$ be a finite partition of M^{k_2} such that for all $n \in \mathbb{N}$ and for all $y \in M^{k_2}$ we have that $Suf_{Sub^n(\mathcal{P}_0)}(y)$ reduces to a singleton, and let T_γ be the associated transition system on M^{k_2}. If Procedure Bisiω terminates with the subword dynamical type, then it provides the coarsest bisimulation on T_γ w.r.t. $\mathcal{P}_0$.*

Proof: The proof is similar to the proof of Theorem 3.33. We also proceed ab absurdo. Hence we can find some step of Procedure *Bisiω* and y_1, $y_1' \in M^{k_2}$ such that $y_1 \sim_{\mathcal{P}_0} y_1'$ and $\mathrm{Sub}_{\mathcal{P}}(y_1) \neq$

$\mathrm{Sub}_{\mathcal{P}}(y'_1)$. We can choose this step such that each piece of $\mathcal{P}$ is a union of equivalence classes for $\sim_{\mathcal{P}_0}$.

Given any $\omega = A_1...A_n \in \mathrm{Sub}_{\mathcal{P}}(y_1)$, we can build the following sequence of transitions.

$$y_1 \to_\gamma y_2 \to_\gamma ... \to_\gamma y_n,$$

with $y_i \in A_i$ for $i = 1, ..., n$. Since $y_1 \sim_{\mathcal{P}} y'_1$ we can build a similar sequence of transitions.

$$y'_1 \to_\gamma y'_2 \to_\gamma ... \to_\gamma y'_n,$$

with $y_i \sim_{\mathcal{P}_0} y'_i$ for $i = 1, ..., n$. Since each A_i is a union of equivalence classes for $\sim_{\mathcal{P}_0}$, we have that $y'_i \in A_i$ for $i = 1, ..., n$. Let us now prove that the suffix uniqueness hypothesis implies that there exists $x \in M^{k_1}$ and $t_1,...,t_n \in M$ with $t_1 \leqslant ... \leqslant t_n$ such that $\gamma(x, t_i) \in A_i$ for $i = 1, ..., n$; meaning that $\omega \in \mathrm{Sub}_{\mathcal{P}}(y'_1)$. Clearly we can find x, t_1, t_2 with $t_1 \leqslant t_2$, $\gamma(x, t_1) \in A_1$ and $\gamma(x, t_2) \in A_2$ (since $y'_1 \to_\gamma y'_2$). Let us suppose, for a contradiction, that given x, t_1, t_2 such that $t_1 \leqslant t_2$, $\gamma(x, t_1) \in A_1$ and $\gamma(x, t_2) \in A_2$ we have that $\gamma(x, t_3) \notin A_3$ for all $t_3 \geqslant t_2$. In particular, using the suffix uniqueness hypothesis, this means that the unique word of $\mathrm{Suf}_{\mathcal{P}}(y'_2)$ does not contain the letter A_3. This contradicts the existence of the transition $y'_2 \to_\gamma y'_3$ where $y'_3 \in A_3$. Thus we can find t_3 with the desired conditions. Iterating the same argument we find the other t_i's.

Similarly, given any $\omega \in \mathrm{Sub}_{\mathcal{P}}(y'_1)$, we can prove that $\omega \in \mathrm{Sub}_{\mathcal{P}}(y_1)$. This contradicts that $\mathrm{Sub}_{\mathcal{P}}(y_1) \neq \mathrm{Sub}_{\mathcal{P}}(y'_1)$. □

The assumptions of Theorem 3.35 are very strong. To weaken these assumptions, one could investigate cases where T_γ is transitive or deterministic.

Corollary 3.36 *Under the hypothesis of Theorem 3.35, if there exists a finite bisimulation on T_γ w.r.t. $\mathcal{P}_0$, Procedure Bisiω terminates with the subword dynamical type.*

Corollary 3.37 *Under the hypothesis of Theorem 3.35, if some step of Procedure Bisiω, with the subword dynamical type, provides an infinite partition $\mathcal{P}$ there is no finite bisimulation on T_γ w.r.t. $\mathcal{P}_0$.*

Corollary 3.38 *Under the hypothesis of Theorem 3.35, when Procedure Bisiω has terminated, we have that:*

$$2Sub(\mathcal{P}) = ... = nSub(\mathcal{P}) = ... = Sub(\mathcal{P}) \supseteq Suf(\mathcal{P})$$

Remark 3.39 *The assumptions of Theorem 3.35 are satisfied when $\gamma(.,.)$ is a flow of a vector field $F : \mathbb{R}^n \to \mathbb{R}^n$ which does not depend on the time (this is the assumption in [LPS00]). In this case, T_γ is both transitive and deterministic.*

Remark 3.40 *An interesting question is of course to know when Procedure Bisiω terminates. In [BM05] Theorem 4.21 gives a condition of termination for Procedure Bisiω.*

Remark 3.41 *Given $(\mathcal{M}, \gamma)$ a dynamical system and $\mathcal{P}$ a finite partition of M^{k_2} such that there is no finite bisimulation on T_γ w.r.t. $\mathcal{P}$, there are examples where Procedure Bisiω terminates with the subwords (or suffix) dynamical types (see Example 4.1).*

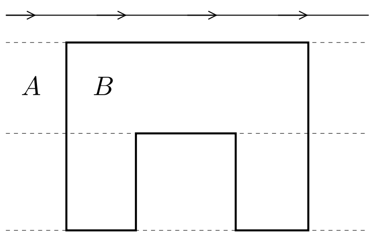

Fig. 9: Dotted words partition

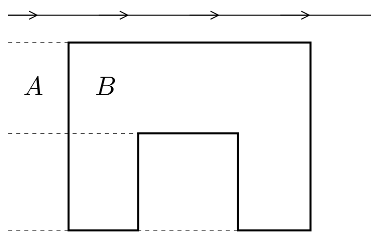

Fig. 10: Suffixes partition

Remark 3.42 *In order to obtain similar results on back-bisimulation, one could apply an analog to Procedure Bisiω where the suffixes are replaced by the prefixes.*

Remark 3.43 *In [BMRT04] in order to define the dynamical type of a point w.r.t. some partition, we introduced the notion of (multi)dotted words (instead of the suffixes). One can show that the finite bisimulation obtained with (multi)dotted words is both forward and backward stable. However the use of suffixes instead of (multi)dotted words leads to a coarser bisimulation than the one obtained with the suffixes. This is illustrated in Figures 9 and 10. In Figure 9, the partition induced by the dotted words have nine pieces corresponding to the dotted words $\dot{A}$, $\dot{A}BA$, ..., $AB\dot{A}$, $\dot{A}BABA$, ..., $ABAB\dot{A}$. In Figure 10, the partition induced by the suffixes have five pieces corresponding to the non empty suffixes of the word $ABABA$.*

4 Examples

This section illustrates Procedure *Bisiω* on some examples. In each case, we give a dynamical system $(\mathcal{M}, \gamma)$ and an initial partition $\mathcal{P}$ and we observe how Procedure *Bisiω* behaves.

Example 4.1 *We consider a dynamical system $(\mathcal{M}, \gamma)$ related to the spiral example of [LPS00]. We have $\mathcal{M} = \langle \mathbb{R}, < \rangle$ and $\gamma : \mathbb{R}^2 \times \mathbb{R} \to \mathbb{R}^2$ are defined as follows.*

$$\gamma(x,t) = e^t\,(x\cos t, x \sin t)$$

The dynamics γ is a solution of the system of differential equations (2) which is not time depending. Hence we can apply Corollary 3.37 (see Remark 3.39) to this example.

$$\begin{pmatrix} \dot{X} \\ \dot{Y} \end{pmatrix} = \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} X \\ Y \end{pmatrix} \tag{2}$$

The dynamics γ describes spirals moving away from the origin when time elapses. We associate with this dynamical system the partiton $\mathcal{P} = \{A, B\}$ where $A = \{(y_1, 0) \mid 0 \leqslant y_1 \leqslant 1\}$ and $B = \mathbb{R}^2 \setminus \{A\}$. Let us focus on the the trajectory $\Gamma_1 = \{(e^t \cos t, e^t \sin t) \mid t \in \mathbb{R}\}$. We divide the trajectory Γ_1 into two distinct parts:

$$\Gamma_1^- = \{(e^t\cos t, e^t \sin t) \mid t \leqslant 0\} \text{ and } \Gamma_1^+ = \{(e^t\cos t, e^t \sin t) \mid t > 0\}$$

We have that Γ_1^- is included in the ball of radius 1 centered at the origin $(0,0)$ and Γ_1^+ has no interesection with this ball. In particular we have that the subword dynamical type of any

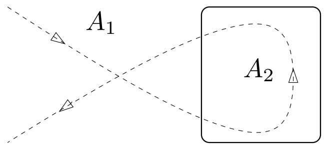

Fig. 11: A simple loop

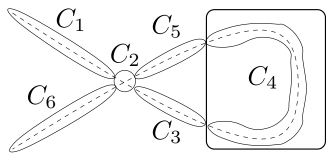

Fig. 12: $\mathcal{T}^2(\mathcal{P}) = \mathcal{T}^3(\mathcal{P})$

point $y \in \Gamma_1^+$ is given by $\{B\}$. If we now consider points on Γ_1^-, one can see that their subword dynamical consists of words in $(AB)^+$ or $(BA)^+$. Let us now show that there are infinitely many subword dynamical types by describing explicitely the dynamical types of the points on Γ_1^-. Given $y \in \Gamma_1^-$, we have that $y = \gamma(1,t)$ for some $t \leqslant 0$, two cases can occur.

$$\text{If } t = -2k\pi \quad \text{then } (AB)^{k+1} \in Sub_{\mathcal{P}}(y) \text{ and } (AB)^{k+2} \notin Sub_{\mathcal{P}}(y),$$
$$\text{if } t \in]-2(k+1)\pi, -2k\pi[\quad \text{then } (BA)^{k+1}B \in Sub_{\mathcal{P}}(y) \text{ and } (BA)^{k+2}B \notin Sub_{\mathcal{P}}(y).$$

Hence the first step of Procedure Bisiω with subword dynamical types, already provides an infinite partition $Sub(\mathcal{P})$. This shows that there is no finite bisimulation on T_γ w.r.t. $\mathcal{P}$ by Corollary 3.37. However one can see that $Sub(\mathcal{P}) = Sub^2(\mathcal{P})$. This means that $Sub(\mathcal{P})$ is the coarsest bisimulation on T_γ w.r.t. $\mathcal{P}$.

Remark 4.2 *In Example 4.1, we have just seen that T_γ does not admit a finite bisimulation w.r.t. $\mathcal{P}$. However T_γ admits a finite back-bisimulation w.r.t. $\mathcal{P}$. In particular when considering points on the trajectory Γ_1, we only have two prefixes to consider, $(AB)^\omega$ and $(BA)^\omega$. That justifies the interest of considering both back-bisimulations and bisimulations given a dynamical system.*

We now consider an example with self intersecting curve(xii).

Example 4.3 *We consider the dynamical system of Figure 11 with initial partition $\mathcal{P} = \{A_1, A_2\}$. There are four suffix dynamical types w.r.t. $\mathcal{P}$:*

$$B_1 = \{A_1A_2A_1\}\ ;\ B_2 = \{A_2A_1\}\ ;\ B_3 = \{A_1\}\ ;\ B_4 = \{A_1, A_1A_2A_1\}.$$

This leads to the four pieces partition $\mathcal{T}(\mathcal{P})$. The set $\Omega_{\mathcal{T}(\mathcal{P})}$ consists of the unique word $B_1B_4B_1B_2B_3B_4B_3$. There are six dynamical types w.r.t. $\mathcal{T}(\mathcal{P})$:

$$C_1 = \{B_1B_4B_1B_2B_3B_4B_3\}\ ;\ C_2 = \{B_4B_1B_2B_3B_4B_3, B_4B_3\}\ ;$$

$$C_3 = \{B_1B_2B_3B_4B_3\}\ ;\ C_4 = \{B_2B_3B_4B_3\}\ ;\ C_5 = \{B_3B_4B_3\}\ ;\ C_6 = \{B_3\}.$$

We obtain the partition $\mathcal{T}^2(\mathcal{P})$. One can easily check that $\mathcal{T}^2(\mathcal{P}) = \mathcal{T}^3(\mathcal{P})$.

(xii) This kind of behavior motivated the notion of multidotted words in [BMRT04].

5 Conclusion

In this paper we introduced a merely conceptual algorithm called Procedure *Bisi*ω. This procedure aims to build a finite bisimulation of a given dynamical system w.r.t. a given partition using words. Procedure *Bisi*ω gives a more "global" vision of the bisimulation than the well-known *bisimulation algorithm*. The papers [KV04, KV06] illustrates that Procedure *Bisi*ω can help to compute complexity bound on the size of the coarsest bisimulation.

Two of the main challenges for futur work are the following questions, *"When is Procedure Bisi*ω *effective?"*, *"When does Procedure Bisi*ω *terminate?"*.

Another question to address is the following. In Section 3, we introduced several "intermediate"equivalence relations (see Definition 3.14). These equivalence relations deserve to be investigate for their own. At present we did not manage to find any relevant property of these equivalence relations.

Acknowledgments. The author would like to thank Véronique Bruyère for helpfull discussions and Christian Michaux for careful reading of the drafts of this paper and numerous relevant comments. He also want to thank the referees for their serious work and useful remarks.

References

[ACH+95] R. Alur, C. Courcoubetis, N. Halbwachs, T.A. Henzinger, P;-H. Ho, A. Olivero, J. Sifakis, and S. Yovine. The algorithmic analysis of hybrid systems. *Theoret. Comput. Sci.*, 138(1):3–34, 1995. Special issue on hybrid systems.

[Acz88] P. Aczel. *Non-well-founded sets*, volume 14 of *CSLI Lecture Notes*. Stanford University Center for the Study of Language and Information, Stanford, CA, 1988. With a foreword by Jon Barwise [K. Jon Barwise].

[AD94] R. Alur and D.L. Dill. A theory of timed automata. *Theoretical Computer Science*, 126(2):183–235, 1994.

[AHLP00] R. Alur, T.A. Henzinger, G. Lafferriere, and G.J. Pappas. Discrete abstractions of hybrid systems. *Proceedings of the IEEE*, 88:971–984, 2000.

[ASY01] Eugene Asarin, Gerardo Schneider, and Sergio Yovine. On the decidability of the reachability problem for planar differential inclusions. In Di Benedetto and Maria Domenica, editors, *HSCC'2001*, volume 2034 of *Lecture Notes in Computer Science*, pages 89–104. Springer, 2001.

[BC01] V. Bruyère and O. Carton. Automata on linear orderings. In *Mathematical foundations of computer science, 2001 (Mariánské Lázně)*, volume 2136 of *Lecture Notes in Comput. Sci.*, pages 236–247. Springer, Berlin, 2001.

[BFH91] A. Bouajjani, J.-C. Fernandez, and N. Halbwachs. Minimal model generation. In E. M. Clarke and R. P. Kurshan, editors, *Computer-Aided Verification: Proc. of the 2nd International Conference CAV'90*, pages 197–203. Springer, Berlin, Heidelberg, 1991.

[BM05] Thomas Brihaye and Christian Michaux. On the expressiveness and decidability of o-minimal hybrid systems. *Journal of Complexity*, 21(4):447–478, 2005.

[BMRT04] T. Brihaye, C. Michaux, C. Rivière, and C. Troestler. On o-minimal hybrid systems. In Rajeev Alur and George J. Pappas, editors, *HSCC'2004*, volume 2993 of *Lecture Notes in Computer Science*, pages 219–233. Springer, 2004.

[Cau95] D. Caucal. Bisimulation of context-free grammars and of pushdown automata. In *Modal logic and process algebra (Amsterdam, 1994)*, volume 53 of *CSLI Lecture Notes*, pages 85–106. CSLI Publ., Stanford, CA, 1995.

[Dav99] J.M. Davoren. Topologies, continuity and bisimulations. *Theor. Inform. Appl.*, 33(4-5):357–381, 1999. Fixed points in computer science (Brno, 1998).

[Hen95] T.A. Henzinger. Hybrid automata with finite bisimulations. In *ICALP 95: Automata, Languages, and Programming*, Lecture Notes in Computer Science 944, pages 324–335. Springer-Verlag, 1995.

[Hen96] T.A. Henzinger. The theory of hybrid automata. In *Proceedings of the 11th Annual Symposium on Logic in Computer Science*, pages 278–292. IEEE Computer Society Press, 1996.

[HKPV98] Thomas A. Henzinger, Peter W. Kopke, Anuj Puri, and Pravin Varaiya. What's decidable about hybrid automata? *J. Comput. System Sci.*, 57(1):94–124, 1998. 27th Annual ACM Symposium on the Theory of Computing (STOC'95) (Las Vegas, NV).

[HNSY94] T.A. Henzinger., X. Nicollin, J. Sifakis, and S. Yovine. Symbolic model checking for real-time systems. *Inform. and Comput.*, 111(2):193–244, 1994. 1992 IEEE Symposium on Logic in Computer Science.

[JdS04] A.A. Julius and A.J.van der Schaft. State maps of general behaviors, their lattice structure and bisimulations. In *Proceedings of the Sixteenth International Symposium on Mathematical Theory of Networks and Systems (MTNS2004)*, 2004.

[KS90] Paris C. Kanellakis and Scott A. Smolka. CCS expressions, finite state processes, and three problems of equivalence. *Inform. and Comput.*, 86(1):43–68, 1990.

[KV04] M. Korovina and N. Vorobjov. Pfaffian hybrid systems. In Jerzy Marcinkowski and Andrzej Tarlecki, editors, *CSL 2004*, volume 3210 of *Lecture Notes in Computer Science*, pages 430–441. Springer-Verlag, 2004.

[KV06] Margarita V. Korovina and Nicolai Vorobjov. Upper and lower bounds on sizes of finite bisimulations of Pfaffian hybrid systems. In *CiE*, volume 3988 of *Lecture Notes in Computer Science*, pages 267–276. Springer, 2006.

[LPS00] G. Lafferriere, G. J. Pappas, and S. Sastry. O-minimal hybrid systems. *Math. Control Signals Systems*, 13(1):1–21, 2000.

[Rab03] A. Rabinovich. Automata over continuous time. *Theoret. Comput. Sci.*, 300(1-3):331–363, 2003.

[SJSL00] S. N. Simic, K. H. Johansson, S. Sastry, and J. Lygeros. Towards a geometric theory of hybrid systems. In N. Lynch et al, editor, *HSCC'2000*, volume 1790 of *Lecture Notes in Computer Science*, pages 421–436. Springer, 2000.

[Tru89] J. K. Truss. Infinite permutation groups. II. Subgroups of small index. *J. Algebra*, 120(2):494–515, 1989.

[vdD98] L. van den Dries. *Tame topology and o-minimal structures*, volume 248 of *London Mathematical Society Lecture Note Series*. Cambridge University Press, Cambridge, 1998.

[Wil91] Jan C. Willems. Paradigms and puzzles in the theory of dynamical systems. *IEEE Trans. Automat. Control*, 36(3):259–294, 1991.