Widening the boundary between decidable and undecidable hybrid systems ⋆

Eugene Asarin and Gerardo Schneider

VERIMAG, Centre Equation, 2 ave de Vignate, 38610 Gières, France,
Eugene.Asarin@imag.fr, Gerardo.Schneider@imag.fr

Abstract. We revisited decidability of the reachability problem for low dimensional hybrid systems. Even though many attempts have been done to draw the boundary between decidable and undecidable hybrid systems there are still many open problems in between. In this paper we show that the reachability question for some two dimensional hybrid systems are undecidable and that for other 2-dim systems this question remains unanswered, showing that it is as hard as the reachability problem for Piecewise Affine Maps, that is a well known open problem.

1 Introduction

Although many intense research activity in the last years have been done in the domain of hybrid systems (systems combining discrete and continuous behaviors), there is no clear boundary between what is decidable or not on such systems. In this paper we address only the reachability problem, we refer the reader interested in decidability of other problems, such as stability, to [10].

It is well known that for particular cases the reachability question is decidable. For continuous-time hybrid systems, the reachability is decidable for timed automata (TA) [3], their generalizations such as multirate automata [2, 30], some kinds of updatable timed automata [12, 13] and initialized rectangular automata [20, 32]. For all these models the decidability depends on existence of a finite bisimulation and holds for systems of any dimensions. Another class of decidability results concerns planar systems. The method was suggested in [27], where decidability was stated for 2-dim PCD (systems with the dynamics given by Piecewise Constant Derivatives). The results were extended to planar multi-polynomial systems in [16] and to non-deterministic planar polygonal systems (SPDI) in [8]. All these results are based on topological properties of the plane and the method does not work neither in higher dimension nor for systems with "jumping" discontinuous trajectories. On the negative side there are many undecidability results, and we cannot give an exhaustive list. For dimension 3 or more the reachability is undecidable for Linear Hybrid Automata [20], and even for PCDs [7]. Undecidability proofs are based on simulation of Turing or Minsky (counter) machines. A more economic simulation allows to prove undecidability

⋆ Partially supported by CNRS Project MathSTIC "Squash - Analyse Qualitative de Systèmes Hybrides".

for systems with stringent restrictions on the continuous dynamics, guards and resets: for example reachability is undecidable for rectangular automata with at least 5 clocks and one two-slope variable [20], or for TA with two skewed clocks [2].

Another group of results is related to the reachability problem for discrete-time dynamical systems, in particular iterations of piecewise affine or more complex functions. Roughly speaking, as it is well known since Poincaré's work, continuous-time systems of dimension $n+1$ are "as complex as" discrete iterations in dimension n. On the negative side, as stated in [25], TM can be simulated by iterations of 2-dim piecewise affine maps(PAM), and hence reachability is undecidable in dimension 2. In dimension one the known undecidable discrete-time systems involve rather complex dynamics, e.g. in [26] an elementary function (a combination of sines an cosines) that simulates Turing machines (TM) with an exponential slowdown is constructed. Another class of systems with undecidable reachability in dimension one are countable PAMs (PAMs with an infinite number of intervals). As for the most natural class of one dimensional systems: finite PAMs, the decidability of reachability is an old standing open question (see [24, 11] for a thorough discussion), related to other open questions in number theory and linear algebra. This problem (we call it REACH_{PAM}) plays the key rôle in this paper.

In this paper we analyze continuous-time hybrid systems which are close to the boundary between decidable and undecidable. As it was mentioned, planar systems with continuous trajectories are decidable, 3-dim are not. That is why we explore planar systems with jumps, and also systems with continuous trajectories on 2-dim manifolds. For such systems instead of proving decidability or undecidability, we establish an equivalence to the problem REACH_{PAM}. A finer analysis allows to show that the reachability for some constrained systems (e.g. with 2 clocks and affine resets) is also as hard as for REACH_{PAM}. For a little bit more complex 2-dim systems with a simple infinitary pattern we prove undecidability.

The paper is organized as follows. In section 2 we define several classes of hybrid automata, two dimensional manifolds, and our reference model: Piecewise affine maps (PAM). In section 3 we introduce Hierarchical PCDs (HPCD) and we show that the reachability problem for HPCD, PCD on manifolds, and some other classes of 2-dim systems is as hard as the reachability for PAM. In section 4 we show that enriching HPCD with one counter, or an infinite partition leads to the undecidability of the reachability question. We conclude in the last section with a summary. Due to space limitations we give only sketches of proofs for most results. The reader can find more details in the thesis [33].

2 Preliminaries

2.1 Hybrid automata

There are many (more or less) equivalent definitions of hybrid systems/automata (see for example [1, 21, 35]). We will adopt in this paper the following definition.

A hybrid system is a dynamical system that combines discrete and continuous components. A natural model for hybrid systems is *hybrid automata* [21] that are automata such that at each discrete location the dynamics is governed by a differential equation (over continuous variables) and whose transitions (between locations) are enabled by conditions on the values of the variables.

Formally, an *n-dimensional hybrid automaton* is a tuple $\mathcal{H} = (\mathcal{X}, Q, f, \mathsf{I}_0, \mathsf{Inv}, \delta)$ where

- $\mathcal{X} \subseteq \mathbb{R}^n$ is the *continuous state space*. Elements of $\mathcal{X}$ are written as $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, for $\{x_1, x_2, \ldots, x_n\} \in \mathsf{V}$, where V is a finite set of variables;
- Q is a finite set of *discrete locations*;
- $f : Q \to (\mathcal{X} \to \mathbb{R}^n)$ assigns a continuous vector field on $\mathcal{X}$ to each discrete location. While in discrete location $\ell \in Q$, the evolution of the continuous variables is governed by the differential equation $\dot{\mathbf{x}} = f_\ell(\mathbf{x})$. We say that the differential equation defines the *dynamics* of location ℓ;
- The *initial condition* $\mathsf{I}_0 : Q \to 2^{\mathcal{X}}$ is a function that for each state defines the initial values of the variables of $\mathcal{X}$;
- The *invariant* or *staying* conditions $\mathsf{Inv} : Q \to 2^{\mathcal{X}}$, $\mathsf{Inv}(\ell)$ is the condition that must be satisfied by the continuous variables in order to stay in location $\ell \in Q$;
- δ is a set of transitions of the form $tr = (\ell, g, \gamma, \ell')$ with $\ell, \ell' \in Q$. Such a quadruple means that a transition from ℓ to ℓ' can be taken whenever the guard $g \subset \mathcal{X}$ is satisfied and then the reset $\gamma : \mathcal{X} \to \mathcal{X}$ is applied.

In what follows we will consider deterministic systems unless the contrary be specified.

A *state* is a pair $(\ell, \mathbf{x})$ consisting of a location $\ell \in Q$ and $\mathbf{x} \in \mathcal{X}$. A state can change in two ways: (1) by *discrete* and *instantaneous* transition that changes both the location and the values of the variables according to the transition relation, and (2) by a *time delay* that changes only the values of the variables according to the dynamics of the current location. The system may stay at a location only if the invariant is true, and a transition must be taken before the invariant becomes false.

A *trajectory* of a hybrid automaton $\mathcal{H}$ is a function $\Theta : [0, T] \to Q \times \mathcal{X}$, $\Theta(t) = (\ell(t), \xi(t))$ such that there exists a sequence of times values $t_0 = 0 < t_1 < \ldots < t_n = T$ for which the following holds for each $1 \le i \le n$: (1) ℓ is constant on (t_i, t_{i+1}) (we describe its value there by ℓ_i) and ξ is derivable on (t_i, t_{i+1}), it is left continuous and with right limits everywhere; (2) There is a transition $(\ell(t_i), g, \gamma, \ell(t_{i+1})) \in \delta$ such that $\xi^-(t_{i+1}) \in g(\ell_i, \ell_{i+1})$ and $\xi(t_{i+1}) = \gamma(\xi^-(t_{i+1}))$[1]; (3) For any $0 \le i \le n$, for any $t \in (t_i, t_{i+1})$, $\dot{\xi}(t) = f_{l(t)}(\xi(t))$.

2.2 Rectangular and linear hybrid automata

A hybrid automaton $\mathcal{H}$ is *linear* [21, 1] if the following restrictions are met: (1) The initial and invariant conditions as well as the guard are boolean combinations of linear inequalities; (2) The dynamics are defined by differential equations

[1] $\xi^-(t)$ is the left limit of $\xi(t)$.

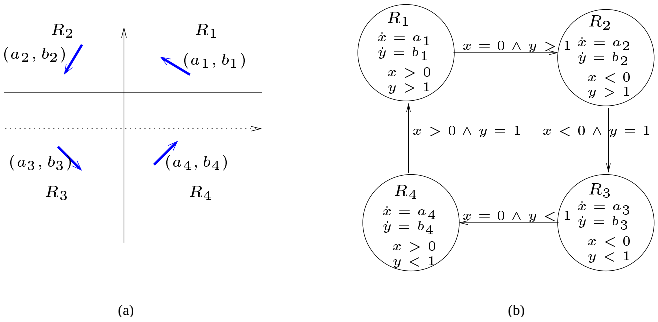


Fig. 1. (a) A simple PCD; (b) Its corresponding hybrid automaton.

of the form $\dot{x} = k_x$, one for each variable $x \in \mathsf{V}$, where $k_x \in \mathbb{Z}$ is an integer constant. We say that k_x is the *slope* (or *rate*) of the variable x at a given location.

We say that a variable x is a *memory cell* if it has slope 0 in every location of $\mathcal{H}$. A variable x is a *clock* if it has slope 1 in every location. A variable x is a *skewed clock* if there is a rational $k \in \mathbb{Q}\backslash\{0,1\}$ such that x has slope k in each location. The variable x is a *two-slope clock* if there is a rational k such that for each location $\dot{x} = k$ or $\dot{x} = 1$. A *stopwatch* is a two-slope clock with $k = 0$.

A *rectangle of dimension* n $R = \prod_{1 \leq i \leq n} I_i$ is the product of n intervals $I_i \subseteq \mathbb{R}$ of the real line with rational or infinite extremities.

A hybrid automaton is a *rectangular automaton* [21, 20, 32] if (1) all the initial conditions, invariants and graphs of resets are rectangles; (2) for each location ℓ, the dynamics has the form $\dot{\mathbf{x}} \in R_\ell$, where R_ℓ is a rectangle.

Another special case of linear hybrid automata are PCDs, that are described in next section.

2.3 PCD

A *piecewise constant derivative system* (PCD) [7, 27] is a pair $\mathcal{H} = (\mathbb{P}, \mathbb{F})$ with $\mathbb{P} = \{P_s\}_{s\in S}$ a finite family of non-overlapping convex polygonal sets in $\mathbb{R}^2$ with non-empty interiors, and $\mathbb{F} = \{\mathbf{c}_s\}_{s\in S}$ a family of vectors in $\mathbb{R}^2$. The dynamics of the PCD is determined by the equation $\dot{\mathbf{x}} = \mathbf{c}_s$ for $\mathbf{x} \in P_s$. Hence trajectories are broken lines.

A well known technique for planar differential equations and in particular for PCD is to replace the analysis of those systems by analysis of edge-to-edge discrete successors [7, 8, 27] (also known as Poincaré map [22]). Given an edge e, each point on e can be represented by a local one dimensional coordinate. A one-step edge-to-edge successor in such coordinates can be written as $\mathsf{Succ}_{ee'}(x) = ax+b$. In general, a n-step successor for a given sequence of edges $\sigma = e_1, e_2, \ldots, e_n$ is again a function of the above form (see for example [7] for a better understanding).

Notice that PCDs can be viewed as linear hybrid automata without reset. In Figure 1 a simple PCD and its corresponding hybrid automata are shown.

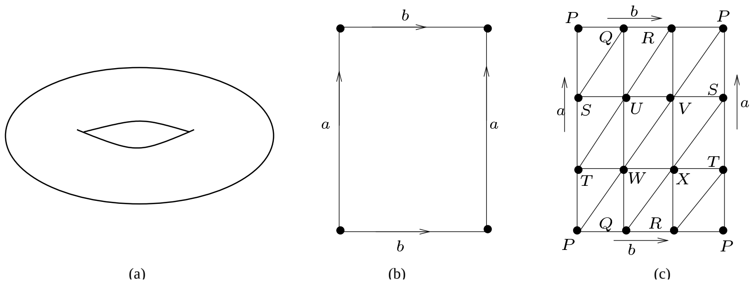


Fig. 2. Representations of a Torus: (a) a surface in $\mathbb{R}^3$; (b) a square with identified edges; (c) a triangulated surface.

2.4 Two dimensional manifolds

All the (topological) definitions, examples and results of this section are done following the *combinatorial method* and follow [19].

A topological space is *triangulable* if it can be obtained from a set of triangles by the identification of edges and vertices subject to the restriction that any two triangles are identified either along a single edge or at a single vertex, or are completely disjoint. The identification should be done via an affine bijection.

A *surface* (or *2-dim manifold*) is a triangulable space for which in addition: (1) each edge is identified with exactly one other edge; and (2) the triangles identified at each vertex can always be arranged in a cycle $T_1, \ldots, T_k, T_1$ so that adjacent triangles are identified along an edge. Typical examples are the sphere, the torus (see Figure 2) or the Klein's bottle.

A *surface with boundary* is a topological space obtained by identifying edges and vertices of a set of triangles as for surfaces except that certain edges may not be identified with another edge. These edges, which violate the definition of a surface, are called *boundary edges*, and their vertices, which also violate the definition of surface, are called *boundary vertices.* Typical examples of surfaces with boundary are the cylinder and the Möbius strip. Indeed, the cylinder is equivalent to a sphere with two disks cut out.

We state now an important theorem in the topological theory of surfaces:

Theorem 1 (Classification theorem(see [19], p.122)).

- *Every compact, connected surface is topologically equivalent to a sphere, or a connected sum of tori, or a connected sum of projective planes.*
- *Every compact, connected surface with boundary is equivalent to either a sphere, or a connected sum of tori, or a connected sum of projective planes, in any case with some finite number of disks removed.*

2.5 Piecewise Affine Maps (PAM)

We define in this section one dimensional Piecewise affine maps (PAM) [11, 24, 25]. We say that a function $f : \mathbb{R} \to \mathbb{R}$ is *piecewise affine* (PAM) if f is of the form $f(x) = a_i x + b_i$ for $x \in I_i$, where $I_i = [l_i, u_i]$ is an interval with $l_i, u_i \in \mathbb{Q}$. Coefficients a_i, b_i and the extremities of I_i are supposed to be rational.

Let REACH_{PAM}, the reachability problem for PAMs, be the following problem.

Problem 1. Given a PAM $\mathcal{A}$ and two points x and y, is y reachable from x?

Even for a function f with just two linear pieces, there is no known decision algorithm for the above problem. The same problem is known to be undecidable in dimension 2 and if piecewise affine maps are replaced by polynomials, the problem is open for any dimension [11, 24, 25].

3 Between Decidability and Undecidability

We show in this section that for several natural classes of 2-dim hybrid systems the reachability problem is s as hard as for 1-dim PAMs, for which such problem is known to be open. Recall that the reachability problem is decidable [27] for planar PCDs and undecidable for dimensions greater than two [6].

3.1 HPCD

Hierarchical piecewise constant derivative systems (HPCDs) can be seen as hybrid automata such that at each location the dynamics is given by a PCD. More formally, an HPCD is a hybrid automaton $H_{PCD} = (\mathcal{X}, Q, f, \mathsf{l}_0, \mathsf{Inv}, \delta)$ such that Q and l_0 are as before while the dynamics at each ℓ is a PCD and each transition $tr = (\ell, g, \gamma, \ell')$ is such that (1) its guard g is a predicate of the form $P(x, y) \equiv (ax + by + c = 0 \wedge x \in I \wedge y \in J)$ where I and J are intervals and a, b, c and the extremities of I and J are rational-valued and (2) the reset functions γ are affine functions: $\gamma(\mathbf{x}) = A\mathbf{x} + \mathbf{b}$. Last, Inv is defined as the negation of the union of the guards, i.e. we can stay in location ℓ as long as no guard is satisfied. If all the PCDs are bounded, then the HPCD is said to be *bounded*.

We need to introduce a 1-dim coordinate system on each edge e. We will denote a point with local coordinates x on edge e by (e, x) or whenever no confusion may arise, just as x.

It can be argued that the term *hierarchical* in the above definition is superfluous and that in fact HPCDs are just 2 dimensional linear hybrid automata. Even though this is true, the definition is intentional since we want to emphasize the fact that there are just "few" real discontinuities due to jumps and reset and that in general the trajectory behaves like a PCD.

Let REACH_{HPCD} be the following problem:

Problem 2. Given a HPCD $\mathcal{H}$ and two points $\mathbf{x}_0$ and $\mathbf{x}_f$, is $\mathbf{x}_f$ reachable from $\mathbf{x}_0$?

We will prove that HPCDs can simulate PAMs and vice versa. For that we show first that each HPCD $\mathcal{H}$ is simulated by a PAM $\mathcal{A}$ and that for each PAM

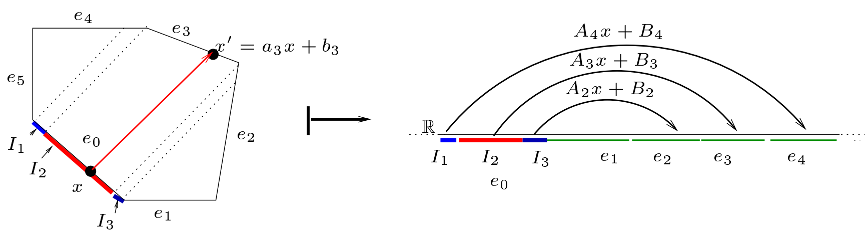


Fig. 3. Sketch of the simulation of a HPCD by a PAM.

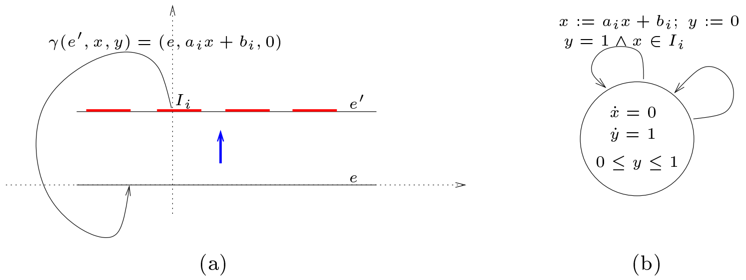


Fig. 4. (a) The HPCD that simulates a PAM. (b) An equivalent $\mathsf{RA}_{\mathsf{1cl1mc}}$.

$\mathcal{A}$ there is a HPCD $\mathcal{H}$ such that $\mathcal{H}$ simulates $\mathcal{A}$. For proving the first, we should: (1) Encode an initial and final point of $\mathcal{H}$ by points on some intervals of $\mathcal{A}$; (2) Represent a configuration of $\mathcal{H}$ by a configuration of $\mathcal{A}$; (3) Simulate an edge-to-edge transition of $\mathcal{H}$ by some function application on $\mathcal{A}$.

Lemma 1 (PAMs simulate HPCDs). *Every bounded 2-dimensional HPCD $\mathcal{H}$ can be simulated by a PAM.*

Sketch of the proof: We arrange all the edges of $\mathcal{H}$ in the Real line (in an arbitrary order) and we represent each edge-to-edge successor function and each reset function by an affine map (restricted to an interval). Assembling all those affine maps together yields the PAM $\mathcal{A}$ simulating $\mathcal{H}$ (see Figure 3). $\square$

Lemma 2 (HPCDs simulate PAMs). *Every PAM $\mathcal{A}$ can be simulated by a 2-dim HPCD.*

Proof: Let $\mathcal{A}$ be defined by $f(z) = a_iz + b_i$ if $z \in I_i$ for $i \in \{1, \cdots, k\}$ where $I_i = [l_i, u_i]$ are rational intervals. We define a one-location HPCD with a one-region PCD defined by $y \geq 0 \wedge y \leq 1$, i.e. there are two edges $e \equiv y = 0$ and $e' \equiv y = 1$, and dynamics defined by vector $(0, 1)$ as shown in figure 4-(a). There are as many transitions as intervals I_i of the PAM. The guards are of the form $y \in e \wedge x \in I_i$ and the reset functions associated with these guards are of the form $\gamma(e', x, y) = (e, a_ix + b_i, 0)$. The initial point z_0 of the PAM is encoded as

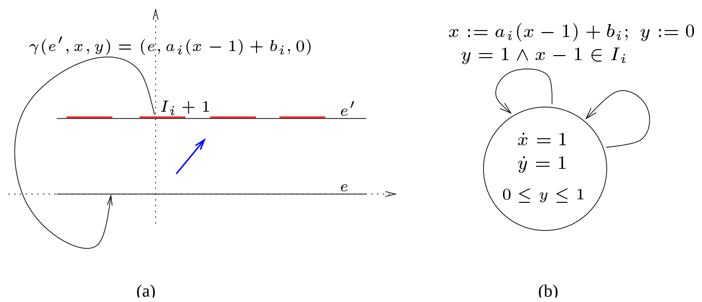


Fig. 5. (a) Another HPCD that simulates a PAM; (b) The corresponding $\mathsf{RA}_{\mathsf{2cl}}$.

a point $(x_0, y_0) \in e$ with local coordinate $\lambda_0 = x_0 = z_0$. Hence, it is easy to see that $z_f = f(z_0)$ iff $\lambda_f = \gamma(e', \lambda')$ where $\lambda' = \mathsf{Succ}_{ee'}(\lambda_0)$. $\square$
From the above two lemmas, we have then the following theorem.

Theorem 2 (HPCDs are equivalent to PAMs). *REACH_{HPCD} is decidable iff REACH_{PAM} is.* $\square$

Remark. It can be said that encoding everything in reset functions is not fair. Indeed, the simulation works for less general resets. In fact, it can be shown that any PAM can be simulated by an HPCD with isometric (length preserving) reset functions. Let us denote the corresponding HPCD by $\mathsf{HPCD}_{\mathsf{iso}}$ and its reachability problem by $\mathsf{REACH}_{HPCD_{iso}}$. Hence we have the following theorem (the exact construction can be found in [33]).

Theorem 3 ($\mathsf{HPCD}_{\mathsf{iso}}$ are equivalent to PAMs). *$\mathsf{REACH}_{HPCD_{iso}}$ is decidable iff REACH_{PAM} is.* $\square$

3.2 About rectangular and linear 2-dimensional hybrid automata

In this section we prove some corollaries of Theorem 2 and Theorem 3.

The class of rectangular hybrid automata with one clock y, one memory cell x, invariants of the form $C \leq y \leq D$, guards of the form $y = D$ and resets of the form $\gamma(x, y) = (ax + b, 0)$ will be denoted as $\mathsf{RA}_{\mathsf{1cl1mc}}$. It is easy to observe that the HPCD defined for simulating a PAM (see Figure 4-(a)) is in fact a $\mathsf{RA}_{\mathsf{1cl1mc}}$ (see Figure 4-(b)) and to deduce the following result.

Corollary 1 ($\mathsf{RA}_{\mathsf{1cl1mc}}$ are equivalent to PAMs). *Reachability for $\mathsf{RA}_{\mathsf{1cl1mc}}$ is decidable iff reachability for PAMs is. The same is true for one-state $\mathsf{RA}_{\mathsf{1cl1mc}}$.*

The class of rectangular hybrid automata with two clocks x and y, invariants of the form $C \leq y \leq D$, guards of the form $y = D$ and resets of the form $\gamma(x, y) = (ax + b, 0)$ will be denoted as $\mathsf{RA}_{\mathsf{2cl}}$.

Corollary 2 **($\mathsf{RA}_{\mathsf{2cl}}$ are equivalent to PAMs).** *Reachability for* $\mathsf{RA}_{\mathsf{2cl}}$ *is decidable iff reachability for PAMs is.*

Sketch of the proof: In Lemma 2 an HPCD $\mathcal{H}$ (see Figure 4) that simulates a PAM was built. We obtain another HPCD $\mathcal{H}'$ applying an affine transformation to $\mathcal{H}$, where the edge e remains unchanged whereas e' is translated by one unit to the right. $\mathcal{H}'$ is represented in Figure 5-(a), where given $I = [l, u]$ $I + 1$ is a short for $[l+1, u+1]$. It is not difficult to see that the automaton of Figure 5-(b) is a $\mathsf{RA}_{\mathsf{2cl}}$ equivalent to $\mathcal{H}'$. □

Notice that $\mathsf{RA}_{\mathsf{2cl}}$ automata can be considered as *updatable* timed automata [12, 13] with more general resets (of the form $y := ax + b$).

The next two corollaries are consequences of Theorem 3.

We denote by $\mathsf{RA}_{\mathsf{1sk1sl}}$, the class of rectangular hybrid automata with one two-slope clock x (taking values on $\{-1, 1\}$) and one positive n-skewed clock y with the following restrictions: (1) on each transition, x is reset to function of y of the form $x := y + d$ and y is reinitialized with a constant value c, where c is the inferior bound of y in ℓ'; (2) the values of the two variables are never compared, and (3) the guard of a transition from location ℓ to ℓ' is of the form $x = A$, where A is one of the bounds of x in the invariant of location ℓ.

It can be seen that the construction of Theorem 3 gives in fact a $\mathsf{RA}_{\mathsf{1sk1sl}}$.

Corollary 3 **($\mathsf{RA}_{\mathsf{1sk1sl}}$ are equivalent to PAMs).** *Reachability for* $\mathsf{RA}_{\mathsf{1sk1sl}}$ *is decidable iff reachability for PAMs is.*

Let $\mathcal{H}$ be a linear hybrid automaton with just two (mutually exclusive) stopwatches x and y with the following restriction: (1') whenever a transition is taken, x and y remain unchanged or the new value of x is a function of y of the form $x := y + d$ and y is reinitialized with a constant value c, where c is the inferior bound of y in ℓ'; (2') the guard of a transition from ℓ to ℓ' is of the form $x = A$ or $ax + by + c = 0$, where A is one of the bounds of x in the invariant of location ℓ and a, b and c are rational constants. We denote this class by $\mathrm{LA}_{\mathrm{St}}$.

It can be shown that $\mathrm{LA}_{\mathrm{St}}$ can simulate any $\mathsf{RA}_{\mathsf{1sk1sl}}$ which implies the last result of this subsection.

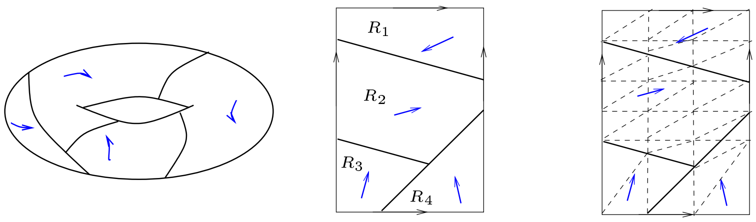
Corollary 4 **($\mathrm{LA}_{\mathrm{St}}$ are equivalent to PAMs).** *Reachability for* $\mathrm{LA}_{\mathrm{St}}$ *is decidable iff reachability for PAMs is.*

3.3 $\mathbf{PCD_{2m}}$: PCDs on 2-dimensional manifolds

Surfaces (or 2-dimensional manifolds) were introduced in section 2.4. To define a PCD on a triangulated surface S, a PCD should be defined on each of its triangles. We call this class of systems *PCD on 2-dimensional manifolds* ($\mathrm{PCD_{2m}}$).

In figure Figure 6 we define a PCD on a torus and show how to represent it as a family of PCDs on triangles.

A point $\mathbf{x}_f$ is *reachable* from another point $\mathbf{x}_0$ if there exists a trajectory from $\mathbf{x}_0$ to $\mathbf{x}_f$. We consider the following problem:



(a) (b) (c)

Fig. 6. A $\mathrm{PCD_{2m}}$ on the torus : three views.

Problem 3. Given a $\mathrm{PCD_{2m}}$ $\mathcal{H}$ and two points $\mathbf{x}_0$ and $\mathbf{x}_f$, is $\mathbf{x}_f$ reachable from $\mathbf{x}_0$?

Lemma 3 ($\mathsf{PAM_{inj}}$ **simulate** $\mathrm{PCD_{2m}}$)**.** *Every* $\mathrm{PCD_{2m}}$ *can be simulated by an injective PAM.*

Sketch of the proof: Let $\mathcal{H}$ be a $\mathrm{PCD_{2m}}$. The reduction is analog to the simulation of HPCDs by PAMs. Notice that $\mathcal{H}$ is in fact an HPCD where a jump is produced each time we reach an identified edge and the resets are the identifying bijections between the identified edges. We will not reproduce the proof here, see Lemma 1. The requirement that each edge is identified with exactly one other edge ensures injectivity. □

Lemma 4 ($\mathrm{PCD_{2m}}$ **simulate** $\mathsf{PAM_{inj}}$)**.** *Every injective (bounded) PAM can be simulated by a* $\mathrm{PCD_{2m}}$*.*

Sketch of the proof: Let $\mathcal{A}$ be an injective PAM defined as $f(z) = f_i(z) = a_i z + b_i$ if $z \in I_i$ for $1 \le i \le n$. We obtain a $\mathrm{PCD_{2m}}$ in the following way similar to the construction of lemma 2. In the rectangle $R = [-M; M] \times [0; 1]$ (with M large enough) the dynamics is defined by vector $(0, 1)$. In order to realize the function f by identification of edges, we introduce several new edges (see Figure 7: on the bottom side of the rectangle R we define $I_i^k = (f_i(I_i) \cap I_k) \times \{0\}$, on the top side we define $J_i^k = (I_i \cap f_i^{-1}(I_k)) \times \{1\}$). Injectivity of the PAM $\mathcal{A}$ guarantees that these intervals do not overlap.

Next we identify each non-empty J_i^k with I_i^k via the function f_i (which is an affine bijection between these two edges). It is easy to find a triangulation such that I_i^k and J_i^k are its edge, hence we have represented our system as a PCD on a compact surface with boundary.

By the Classification Theorem for Surfaces with Boundary (see Theorem 1) we have that this surface is equivalent to a sphere with some disks removed and we obtain then a $\mathrm{PCD_{2m}}$ just "sewing" the disks. We associate with these disks a zero slope vector. □

From the above two lemmas we have that $z_f = f^*(z_0)$ iff $\mathrm{Reach}(\mathcal{H}, \mathbf{x}_0, \mathbf{x}_f)$, where $\mathbf{x}_0$ has local coordinate $\lambda_0 = z_0$ on a given edge e and $\mathbf{x}_f$ has local coordinate $\lambda_f = z_f$ on an edge e'. Then the following theorem holds.

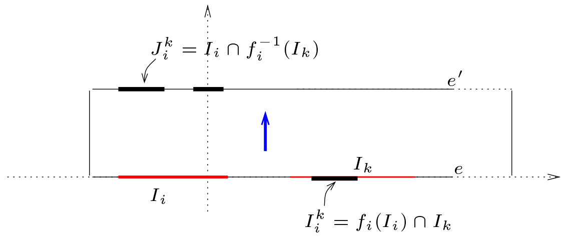


Fig. 7. Simulation of a $\mathrm{PCD_{2m}}$ by a $\mathsf{PAM_{inj}}$: edge J_i^k identified with I_i^k via f_i

Theorem 4 **($\mathrm{PCD_{2m}}$ are equivalent to $\mathsf{PAM_{inj}}$).** *Reachability for* $\mathrm{PCD_{2m}}$ *is decidable iff reachability for injective PAMs is.*

4 Undecidability results

We show in this section that modifying HPCD slightly by adding "something infinite" (a counter, an infinite partition, etc.) yields undecidable systems.

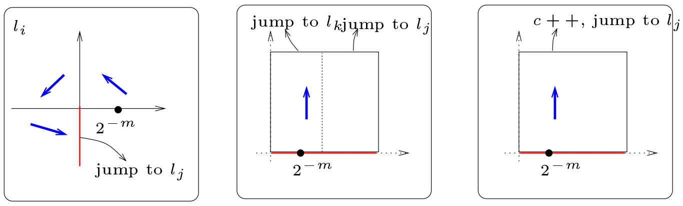
4.1 HPCD with one counter ($\mathbf{HPCD_{1c}}$)

Consider the class of $\mathrm{HPCD_{1c}}$ which are HPCD augmented with a counter c. In each location ℓ the state vector (x, y) evolves according to a PCD, while c remains constant. Guards have the form $P(x, y) \wedge Q(c)$ where $P(x, y)$ is as for HPCDs and $Q(c) \equiv c = 0 \mid c > 0 \mid \mathsf{true}$. Resets are as for HPCDs, but they can also increment or decrement c.

We prove that the reachability problem for $\mathrm{HPCD_{1c}}$ is undecidable showing that a $\mathrm{HPCD_{1c}}$ $\mathcal{H}$ can simulate Minsky (two counter) machines [28] for which reachability is known to be undecidable.

Proposition 1 **($\mathrm{HPCD_{1c}}$ simulates MM).** *Every Minsky Machine $\mathcal{M}$ can be simulated by a 2-dim HPCD with one counter. Hence reachability is undecidable for* $\mathrm{HPCD_{1c}}$.

Sketch of the proof: We associate with each q_i of $\mathcal{M}$ a location ℓ_i of $\mathrm{HPCD_{1c}}$. In order to encode a configuration of $\mathcal{M}$ which is a triple (q_i, m, n) (with m and n standing for contents of the two counters), we represent it in $\mathcal{H}$ by (ℓ_i, x, y, c) with the point $(x, y) = (2^{-m}, 0)$ representing the first counter of $\mathcal{M}$, and $c = n$ storing the second one. In any case, the PCD associated to the location ℓ_i simulates the instruction for the state q_i. Informally speaking, in order to increment or decrement m we just divide or multiply x by 2 (it can be easily done by a PCD). In order to test whether $m = 0$ we check whether $x > 0.75$. All the operations on n are done directly on the counter c. Figure 8 represents PCD simulating instructions $m{+}{+}$, $m = 0?$ and $n{+}{+}$. PCD for the three other instructions of Minsky machine ($m{-}{-}$, $n{-}{-}$ and $n = 0?$) are similar.
Putting all those PCDs together we obtain a $\mathrm{HPCD_{1c}}$ which simulates $\mathcal{M}$. $\square$



q_i : m++, goto q_j — (a)

q_i : if $m = 0$ then goto q_j else goto q_k — (b)

q_i : n++, goto q_j — (c)

Fig. 8. Sketch of the simulation of a MM by a HPCD$_{1c}$: location l_i.

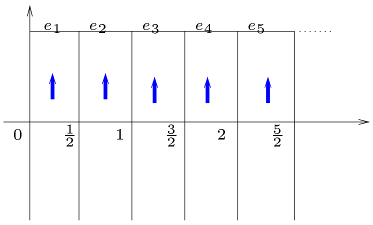


Fig. 9. Sketch of the simulation of a TM by a HPCD$_\infty$.

4.2 HPCDs with infinite partition (HPCD$_\infty$)

We will consider in this section HPCDs for which we relax the condition of having a finite number of regions. We call this class of systems, *HPCDs with infinite partition* (HPCD$_\infty$). We are not going to define this class formally, since we are just interested in showing that this additional feature (having an infinite partition) leads immediately to the undecidability of the reachability problem for HPCD$_\infty$.

Proposition 2 (HPCD$_\infty$ **simulate TMs).** *Every TM $\mathcal{M}$ with alphabet* $\{0;1\}$ *can be simulated by a 2-dimensional (unbounded) HPCD with infinite partition. Hence reachability is undecidable for* HPCD$_\infty$.

Sketch of the proof: The system $\mathcal{H}$ will have a location l_k for each state q_k of the TM. We represent the TM tape contents by a point on the x-axis with the abscissa $x = \sum_{i=-\infty}^{\infty} a_i 2^i$ (here a_0 is the symbol under the head of the TM) in a HPCD with infinite partition as in Figure 9. With such a partition it is easy to test whether the current symbol is 0 or 1: whenever an "even" edge is reached (e_i with $i = 2k$ for $k \in \mathbb{N}$), that corresponds to $\mathrm{frac_x} > \frac{1}{2}$, and hence the current symbol is 1, otherwise it is 0.

Hence, to simulate an instruction of the form $q_k 0 \to \ldots$ we make a jump from all the odd e_i edges of the location l_k. For an instruction $q_k 1 \to \ldots$ we make a jump from all the even edges. It is easy to see that this jump is always

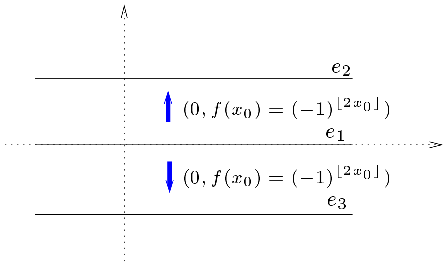


Fig. 10. Simulation of a TM by a $\mathrm{HPCD_x}$.

affine: shifting the head corresponds to division or multiplication of x by 2, and replacing the current symbol corresponds to addition or substraction of $\frac{1}{2}$. □

4.3 Origin-dependent rate HPCDs ($\mathbf{HPCD_x}$)

Another way of introducing "infinite patterns" is allowing continuous dynamics with some periodic behavior that depends on the initial points after a reset is done. An *origin-dependent rate PCD* is a PCD $\mathcal{H} = (\mathbb{P}, \mathbb{F})$ such that each region P_s has dynamics $\dot{\mathbf{x}} = \phi_s(\mathbf{x}_0)$ (as before, given a generic region P we will also use the notation $\phi(P, \mathbf{x}_0)$).

Notice, that after reaching an edge, the system evolves according to a fixed rate that depends on the initial value $\mathbf{x}_0$ of the variables when entering the region. The idea of having flows (dynamics) that depend on initial sates has been taken from [5].

In the construction of Proposition 3 we will use rather particular ϕ_s functions.

We extend the above definition to HPCDs: an *origin-dependent rate HPCD (*$\mathrm{HPCD_x}$*)* is a HPCD with an origin-dependent rate PCD at each location.

Proposition 3 ($\mathrm{HPCD_x}$ **simulate TMs).** *Every TM $\mathcal{M}$ can be simulated by a 2-dimensional unbounded* $\mathrm{HPCD_x}$ *$\mathcal{H}$. Hence the reachability is undecidable for such systems.*

Proof: We associate with each TM-state q_i a location ℓ_i, where the PCD_i is defined by four regions: R_1, $(y > 0) \wedge (y < 1)$; R_2, $(y < 0) \wedge (y > -1)$; R_3, $y < -1$; R_4, $y > 1$. The first two regions have dynamics given by the vector $(0, f(x_0))$ and the last two by (0,1).

Let e_1, e_2 and e_3 be as shown in Figure 10. Let $f(x_0) = (-1)^{\lfloor 2x_0 \rfloor}$, where x_0 is the first coordinate on edge e_0 of the initial point $\mathbf{x}_0$. Notice that $f(x) = 1$ if $\mathrm{frac_x} < \frac{1}{2}$ and $f(x) = -1$ otherwise.

There are two transitions from ℓ_i: (1) $tr_1 = (\ell_i, g_1, \gamma_1, \ell_j)$ where $g_1 \equiv e_2$ and $\gamma_1(e_2, x) = (e_1', f'(x))$; (2) $tr_2 = (\ell_i, g_2, \gamma_2, \ell_h)$ where $g_2 \equiv e_3$ and $\gamma_2(e_3, x) = (e_1, f''(x))$. Transitions tr_1 and tr_2 allow the trajectory to continue in locations ℓ_j and ℓ_h with a reset function that implement the instructions of the Turing machine as before. □

Notice that the above definition allows the dynamics to be defined by any function of the initial point, but in order to simulate a TM we need very particular kind of functions, those that have a periodic pattern. We could have chosen any periodic function like sine or cosine. In any case, the key idea is to obtain an "infinite pattern" as before.

5 Conclusion

The contribution of this paper is twofold. First, we have shown that between 2 dimensional PCDs (for which the reachability problem is decidable [27]) and 3 dimensional PCDs (reachability is undecidable [7]) there exist an interesting class, *2-dim HPCD*, for which the reachability question is still open. We have also shown that the same is true for other similar systems, namely 2-dim rectangular automata and 2-dim linear hybrid automata with some restrictions as well as for PCD on 2-dim manifolds. Second, we have proved that 2-dim HPCD are really in the boundary between decidability and undecidability, since adding a simple counter or allowing some kind of "infinite pattern" to these systems, makes the reachability problem undecidable.

References

1. R. Alur, C. Courcoubetis, N. Halbwachs, T.A. Henzinger, P.-H. Ho, X. Nicollin, A. Olivero, J. Sifakis, and S. Yovine. The algorithmic analysis of hybrid systems. *TCS*, 138:3–34, 1995.
2. R. Alur, C. Courcoubetis, T.A. Henzinger, and P.-H. Ho. Hybrid automata: An algorithmic approach to the specification and verification of hybrid systems. LNCS 736, p. 209–229. Springer-Verlag, 1993.
3. R. Alur and D.L. Dill. A theory of timed automata. *TCS*, 126:183–235, 1994.
4. R. Alur, T.A. Henzinger, and M.Y. Vardi. Parametric real-time reasoning. In *25th ACM STOC*, p. 592–601, 1993.
5. R. Alur, S. Kannan, and S. La Torre. Polyhedral flows in hybrid automata. In *HSCC'99*, LNCS 1569, p. 5–18. Springer-Verlag, 1999.
6. E. Asarin and O. Maler. On some relations between dynamical systems and transition systems. In *ICALP'94*, LNCS 820. Springer, 1994.
7. E. Asarin, O. Maler, and A. Pnueli. Reachability analysis of dynamical systems having piecewise-constant derivatives. *TCS*, 138:35–65, 1995.
8. E. Asarin, G. Schneider, and S. Yovine. On the decidability of the reachability problem for planar differential inclusions. In *HSCC'2001*, LNCS 2034, p. 89–104.
9. L.S. Bobrow and M.A. Arbib. *Discrete Mathematics*. W.B. Saunders, 1974.
10. V. Blondel and J. Tsitsilkis. Complexity of stability and controllability of elementary hybrid systems. *Automatica*, 35:479–489, 1999.
11. O. Bournez. *Complexité algorithmique des systèmes dynamiques continus et hybrides*. PhD thesis, ENS de Lyon, 1999.
12. P. Bouyer, C. Dufourd, E. Fleury, and A. Petit. Are timed automata updatable? In *CAV'2000*, LNCS 1855, p. 464–479.
13. P. Bouyer, C. Dufourd, E. Fleury, and A. Petit. Expressiveness of updatable timed automata. In *MFCS'2000*, LNCS 1893, p. 232–242.

14. M. S. Branicky. Universal computation and other capabilities of hybrid and continuous dynamical systems. *TCS*, 138(1), 1995.
15. K. Čerāns. *Algorithmic problems in analysis of real-time systems specifications.* PhD thesis, Univ. of Latvia, 1992.
16. K. Čerāns and J. Vīksna. Deciding reachability for planar multi-polynomial systems. In *Hybrid Systems III*, LNCS 1066. Springer, 1996.
17. R.L. Devaney. *An Introduction to Chaotic Dynamical Systems.* Addison-Wesley, Redwood City, 2nd edition, 1989.
18. J. Guckenheimer and P. Holmes. *Nonlinear Oscillations, Dynamical Systems and Linear Algebra.* Springer-Verlag, New York, 1990.
19. M. Henle. *A combinatorial introduction to topology.* Dover publications, Inc., 1979.
20. T.A. Henzinger, P.W. Kopke, A. Puri, and P. Varaiya. What's decidable about hybrid automata? In *STOC'1995*, p. 373–382. ACM Press.
21. T.A. Henzinger. The theory of hybrid automata. In *LICS'96*, p. 278–292, 1996.
22. M. W. Hirsch and S. Smale. *Differential Equations, Dynamical Systems and Linear Algebra.* Academic Press Inc., 1974.
23. Y. Kesten, A. Pnueli, J. Sifakis, and S. Yovine. Integration graphs: a class of decidable hybrid systems. In *Hybrid Systems*, LNCS 736, p. 179–208, 1993.
24. P. Koiran. My favourite problems. `www.ens-lyon.fr/~koiran/problems.html`.
25. P. Koiran, M. Cosnard, and M. Garzon. Computability with low-dimensional dynamical systems. *TCS*, 132(1):113–128, 1994.
26. P. Koiran and C. Moore. Closed-form analytic maps in one and two dimensions can simulate universal Turing machines. *TCS* , 210:217–223, 1999.
27. O. Maler and A. Pnueli. Reachability analysis of planar multi-linear systems. In *CAV'93*, LNCS 697, p. 194–209. Springer-Verlag, 1993.
28. M.L. Minsky. *Computation: Finite and Infinite Machines.* Prentice-Hall, 1967.
29. C. Moore. Unpredictability and undecidability in dynamical systems. *Physical Review Letters*, 64(20), 1990.
30. X. Nicollin, A. Olivero, J. Sifakis, and S. Yovine. An approach to the description and analysis of hybrid systems. In *Hybrid Systems*, LNCS 736, p. 149–178, 1993.
31. C.H. Papadimitriou. *Computational complexity.* Addison-Wesley, 1994.
32. A. Puri and P. Varaiya. Decidability of hybrid systems with rectangular differential inclusions. In *CAV'94*, LNCS 818, p. 95–104. Springer, 1994.
33. G. Schneider. *Algorithmic Analysis of Polygonal Hybrid Systems.* PhD thesis, VERIMAG – UJF, Grenoble, 2002.
34. K.S. Sibirsky. *Introduction to Topological Dynamics.* Noordhoff International Publishing, Leyden, 1975.
35. S. Simić, K. Johansson, S. Sastry, and J. Lygeros. Towards a geometric theory of hybrid systems. In *HSCC'00*, LNCS 1790. Springer, 2000.