Reachability and Mortality Problems for Restricted Hierarchical Piecewise Constant Derivatives

Paul C. Bell[1], Shang Chen[1], and Lisa Jackson[2]

[1] Department of Computer Science, Loughborough University
[2] Department of Aeronautical and Automotive Engineering, Loughborough University
{P.Bell,S.Chen3,L.M.Jackson}@lboro.ac.uk

Abstract. We show the NP-hardness of the reachability and mortality problems for a three dimensional variant of Piecewise Constant Derivative (PCD) system called a bounded 3-dimensional Restricted Hierarchical PCD (3-RHPCD). Both problems are shown to be in PSPACE, even for n-dimensional RHPCD. This is a restricted model with similarities to other models in the literature such as stopwatch automata, rectangular automata and PCDs. We also show that for an unbounded 3-RHPCD, both problems become undecidable via a simulation of a Minsky machine.

1 Introduction

The model of *Piecewise Constant Derivative* (PCD) system is a natural and intuitive hybrid system model. An n-dimensional PCD is a finite set of non-overlapping bounded or unbounded convex n-dimensional regions, for which each region is assigned a constant derivative. This derivative defines the direction of flow of points within that region, with the derivative changing when the trajectory passes from one region to the next. See Section 2 for formal definitions.

Among the possible problems one may consider for PCDs is the *reachability* problem. The reachability problem asks, given a PCD and two points x and y, does the trajectory starting at point x ever reach point y after some finite amount of time? It was shown in [11] that the reachability problem for 2-PCDs is *decidable*. In contrast, it was shown in [2] that reachability for 3-PCDs is actually *undecidable*.

In [4], a related model, called a *Hierarchical Piecewise Constant Derivative* (HPCD) system was introduced. An HPCD is a two-dimensional hybrid automaton where the dynamics in each discrete location is given by a 2-PCD (formal details are given in Section 2). Certain edges in the HPCD are called (transition) guards and cause the HPCD to change location if ever the trajectory reaches such an edge. When transitioning between locations, an affine reset rule may be applied. If all regions of the underlying PCDs are bounded, then the HPCD is called bounded. This model can thus be seen as an extension of a 2-PCD. Indeed, the reachability problem for a one-dimensional Piecewise Affine Map

J. Ouaknine, I. Potapov, and J. Worrell (Eds.): RP 2014, LNCS 8762, pp. 32–44, 2014.

(1-PAM), which is a longstanding open problem, was shown to be equivalent to that of reachability for a bounded HPCD with either: i) comparative guards, identity resets and elementary flows in Proposition 3.20 of [3] or else ii) affine resets, non-comparative guards and elementary flows in Lemma 3.4 of [3] (See Section 2 for definitions).

Further results for HPCDs were shown in [5]. The model of *Restricted HPCD* (RHPCD) was defined, which is an HPCD with restricted components. We aimed to study which restrictions of an HPCD lead to decidable reachability results. Essentially, the HPCD must have identity resets, elementary flows (derivatives of all continuous variables come from $\{0, \pm 1\}$) and non-comparative guards (all guards aligned with the x and y axes). These restrictions on the resets, derivatives and guards seem natural ones to consider. For example, restricting to identity resets means the trajectory will not have discontinuities in the continuous component, which is similar to a PCD trajectory. Restricting the derivatives to elementary flows ($\{0, \pm 1\}$) has similarities to a *stopwatch automaton*, for which all derivatives are from $\{0, 1\}$. Restricting the guards to be non-comparative gives strong similarities to the guards of a *rectangular automaton* [9], as well as the diagonal-free clock constraints of an *updatable timed automaton* [7].

Reachability for 2-RHPCDs was shown to be decidable. Together with the results in [3] mentioned above, the reachability problem for HPCDs was shown to be equivalent to that of 1-PAMs when the HPCD only has one of the following: comparative guards, linear resets or arbitrary constant flows. Furthermore, if the model is endowed with a non-deterministic transition function between locations, then the reachability problem becomes NP-hard.

Related to the reachability problem is the *mortality problem*. The mortality problem is the problem of determining if *all* valid initial points eventually reach some particular fixed point configuration (the mortal configuration). There is potentially more than one way to define the mortality problem for HPCDs. In this paper, we define the mortality problem to mean that from any valid initial configuration, the trajectory will reach some fixed point $(0, 0, 0)$ in a finite amount of time, after which the point never changes. Thus the trajectory can be said to halt at this stage.

In this paper, we consider an n-dimensional analogue of RHPCDs, which we denote n-RHPCD. In an analogous way to [3], our aim is to study the following question: "What is the simplest class of hybrid systems for which reachability is intractable or undecidable?" We show a lower bound that the reachability and mortality problems for bounded 3-RHPCDs are NP-hard by an encoding of the simultaneous incongruences problem. We then show that the reachability problem for unbounded 3-RHPCDs is actually undecidable by an encoding of a Minsky machine. Note that the reachability problem for a 3-dimensional HPCD is undecidable, even with only one location, since HPCDs are a superclass of 3-dimensional PCDs for which reachability is undecidable [2]. Finally, we give an upper bound by showing that the reachability and mortality problems for bounded n-RHPCD are in PSPACE.

Note that the systems we construct in this paper deal with trajectories of 'tubes' instead of single lines, which means tiny perturbations will not affect our results. This seems to coincide with the definition of *tube languages* introduced in [10] and implies that our models are *robust* in the properties considered in this paper, but we do not give full details in this conference version of the paper.

2 Preliminaries

Intervals of the form $(s,t), [s,t), (s,t], [s,t]$ are called open, half-open or closed bounded rational intervals (respectively), where $s,t \in \mathbb{Q}$. Let $S \in \mathbb{R}^n$ be a set in the n-dimensional Euclidean space. We define the *closure* of S to be the smallest closed set containing S, denoted $\overline{S}$. We use similar definitions as [3] for the following.

Definition 1. (HA) *An n-dimensional Hybrid Automaton (HA) [1] is a tuple* $\mathcal{H} = (\mathcal{X}, Q, f, l_0, Inv, \delta)$ *consisting of the following components:*

(1) A continuous state space $\mathcal{X} \subseteq \mathbb{R}^n$. Each $\boldsymbol{x} \in \mathcal{X}$ can be written $\boldsymbol{x} = (x_1, \dots, x_n)$, and we use variables $x_1, \dots, x_n$ to denote components of the state vector.
(2) A finite set of discrete locations Q.
(3) A function $f : Q \to (\mathcal{X} \to \mathbb{R}^n)$, which assigns a continuous vector field on $\mathcal{X}$ to each location. In location $l \in Q$, the evolution of the continuous variables is governed by the differential equation $\dot{\boldsymbol{x}} = f_l(\boldsymbol{x})$. The differential equation is called the dynamics of location l.
(4) An initial condition $I_0 : Q \to 2^{\mathcal{X}}$ assigning initial values to variables in each location.
(5) An invariant Inv: $Q \to 2^{\mathcal{X}}$. For each $l \in Q$, the continuous variables must satisfy the condition Inv(l) in order to remain in location l, otherwise it must make a discrete transition.
(6) A set of transitions δ. Every $tr \in \delta$ is of the form $tr = (l, g, \gamma, l')$, where $l, l' \in Q$, $g \subset \mathcal{X}$ is called the guard, defining when the discrete transition can occur, $\gamma \subset \mathcal{X} \times \mathcal{X}$ is called the reset relation applied after the transition from l to l'.

An HA is *deterministic* if it has exactly one solution for its differential equation in each location and the guards for the outgoing edges of locations are mutually exclusive. A configuration of an HA is a pair from $Q \times \mathcal{X}$. A *trajectory* of a hybrid automaton $\mathcal{H}$ starting from configuration $(l_0, \mathbf{x_0})$ where $l_0 \in Q, \mathbf{x_0} \in \mathcal{X}$ is a pair of functions $\pi_{l_0,\mathbf{x_0}} = (\lambda_{l_0,\mathbf{x_0}}(t), \xi_{l_0,\mathbf{x_0}}(t))$ such that

(1) $\lambda_{l_0,\mathbf{x_0}}(t) : [0, +\infty) \to Q$ is a piecewise function constant on every interval $[t_i, t_{i+1})$.
(2) $\xi_{l_0,\mathbf{x_0}}(t) : [0, +\infty) \to \mathbb{R}^n$ is a piecewise differentiable function and in each piece $\xi_{l_0,\mathbf{x_0}}$ is càdlàg (right continuous with left limits everywhere).

(3) On any interval $[t_i, t_{i+1})$ where $\lambda_{l_0,\mathbf{x_0}}$ is constant and $\xi_{l_0,\mathbf{x_0}}$ is continuous,

$$\xi_{l_0,\mathbf{x_0}}(t) = \xi_{l_0,\mathbf{x_0}}(t_i) + \int_{t_i}^{t} f_{\lambda_{l_0,\mathbf{x_0}}(t_i)}(\xi_{l_0,\mathbf{x_0}}(\tau))d\tau$$

for all $\tau \in [t_i, t_{i+1})$.

(4) For any t_i, there exists a transition $(l, g, \gamma, l') \in \delta$ such that
   (i) $\lambda_{l_0,\mathbf{x_0}}(t_i) = l$ and $\lambda_{l_0,\mathbf{x_0}}(t_{i+1}) = l'$;
   (ii) $\xi^-_{l_0,\mathbf{x_0}}(t_{i+1}) \in g(l, l')$ where $\xi^-_{l_0,\mathbf{x_0}}(t)$ means the left limit of $\xi_{l_0,\mathbf{x_0}}$ at t;
   (iii) $(\xi^-_{l_0,\mathbf{x_0}}(t_{i+1}), \xi_{l_0,\mathbf{x_0}}(t_{i+1})) \in \gamma$.

Definition 2. (n-PCD) *An n-dimensional Piecewise Constant Derivative (n-PCD) system [2] is a pair $\mathcal{H} = (\mathbb{P}, \mathbb{F})$ such that:*

(1) $\mathbb{P} = \{P_s\}_{1\le s\le k}$ is a finite family in $\mathbb{R}^n$, where $P_s \subseteq \mathbb{R}^n$ are non-overlapping convex polygonal sets.
(2) $\mathbb{F} = \{\boldsymbol{c}_s\}_{1\le s\le k}$ is a family of vectors in $\mathbb{R}$.
(3) The dynamics are given by $\dot{\boldsymbol{x}} = \boldsymbol{c}_s$ for $\boldsymbol{x} \in P_s$.

An n-PCD is called bounded if for its regions $\mathbb{P} = \{P_s\}_{1\le s\le k}$, there exists $r \in \mathbb{Q}^+$, such that for all P_s, we have that $P_s \subseteq B_{\mathbf{0}}(r)$, where $B_{\mathbf{0}}(r)$ is an origin-centered open ball of radius r and appropriate dimension. We define the *support set* of a PCD $\mathcal{H}$ as $\mathrm{Supp}_{\mathrm{PCD}}(\mathcal{H}) = \bigcup_{1\le s\le k} P_s$.

For full definitions of Hybrid Automata and their trajectories, see [5]. In the following we slightly modify the definition of HPCD [3] to allow different dimensions to be studied.

Definition 3. (n-HPCD) *A n-dimensional Hierarchical Piecewise Constant Derivative (n-HPCD) system is a hybrid automaton $\mathcal{H} = (\mathcal{X}, Q, f, l_0, Inv, \delta)$ such that Q and l_0 are defined as in Definition 1, with the dynamics at each $l \in Q$ given by an n-PCD and each transition $tr = (l, g, \gamma, l')$ is such that: (1) Its guard g is a convex region such that $g \subseteq \mathbb{R}^{n-1}$; and (2) The reset relation γ is an affine function of the form: $\boldsymbol{x}' = \gamma(\boldsymbol{x}) = A\boldsymbol{x} + \boldsymbol{b}$, where $A \in \mathbb{R}^{n\times n}$ and $\boldsymbol{b} \in \mathbb{R}^n$. We denote the* internal guards *of an HPCD location to be the guards of the underlying PCD regions which cause a change of region when they are reached. The* transition guards *are the guards used in transitions between locations. The Invariant (Inv) for a location l is defined to be $Supp_{PCD}(\mathcal{H}) \setminus \mathcal{G}_l$, where $Supp_{PCD}(\mathcal{H})$ is the support set of the underlying PCDs of the HPCD and $\mathcal{G}_l$ is the set of transition guards in location l. If all the PCDs are bounded, then the n-HPCD is said to be bounded.*

In this paper, we are interested in a *restricted* form of n-HPCD.

(I) Under the HPCD model, when transitioning between locations, we may apply an affine reset to non-continuously modify the current point. An n-HPCD has identity (or no) resets if for every transition $tr = (l, g, \gamma, l')$, $\gamma(x) = x$ for all points $x \in \mathbb{R}^n$. This means that starting from any initial configuration $(l_0, \mathbf{x_0})$, for the trajectory $\pi_{l_0,\mathbf{x_0}} = (\lambda_{l_0,\mathbf{x_0}}(t), \xi_{l_0,\mathbf{x_0}}(t))$ we have that $\xi_{l_0,\mathbf{x_0}}(t)$ is a continuous function of t. Note that the trajectory for a PCD is also continuous, and thus this seems to be a natural restriction.

(II) An n-HPCD system has *elementary flows* if the derivatives of all variables in each location are from $\{0, \pm 1\}$, otherwise it has *arbitrary constant flows.*

(III) Guards are used to change the derivative being applied within a location (internal guards), or to change which location we are in (transition guards). They can be described by Boolean combinations of atomic formulae (linear inequalities). If each atomic formula contains only one variable, then the guard is called non-comparative (meaning the guard is aligned with ones of the axes). An n-HPCD has *non-comparative guards* if all guards (both internal and transition) are non-comparative, e.g., for a 3-RHPCD, $\frac{3}{2} \leq x \leq 7 \wedge y = -1 \wedge 2 \leq z \leq 7$ is a non-comparative guard, but $0 \leq x \leq 1 \wedge 0 \leq y \leq \frac{1}{2} \wedge z = 5 \wedge x = 2y$ is a comparative guard (due to the term $x = 2y$).

Definition 4. (n-RHPCD) *An* n-dimensional Restricted Hierarchical Constant Derivative System (RHPCD) *is a bounded n-HPCD with identity resets, non-comparative guards, elementary flows and a finite number of PCD regions. See Fig. 2a and Fig. 2b for an example of a* 3*-RHPCD.*

Finally, we will also require the following *simultaneous incongruences problem*, which is known to be NP-complete [8].

Problem 1. (Simultaneous incongruences) *Given a set* $\{(a_1, b_1), \ldots, (a_n, b_n)\}$ *of ordered pairs of positive integers with* $a_i \leq b_i$ *for* $1 \leq i \leq n$*. Does there exist an integer* k *such that* $k \not\equiv a_i \pmod{b_i}$ *for every* $1 \leq i \leq n$*?*

3 Reachability and Mortality for n-RHPCDs

The following lemma shows that if an instance of the simultaneous incongruences problem has a solution, then there must be a solution less than a particular bound.

Lemma 1. *There exist solutions for the simultaneous incongruences problem with a collection* $\{(a_1, b_1), \ldots, (a_n, b_n)\}$ *if and only if there exists a solution* k *such that* $0 < k \leq \rho$*, where*

$$\rho = lcm(b_1, \ldots, b_n)$$

and $lcm(b_1, \ldots, b_n)$ *is the least common multiple of* $b_1, \ldots, b_n$*.*

Proof. The sufficient part is trivial. We show the necessary part. Given an instance $\{(a_1, b_1), \ldots, (a_n, b_n)\}$, let $\rho = \text{lcm}(b_1, \ldots, b_n)$. Then for every $1 \leq i \leq n$, $\rho \equiv 0 \pmod{b_i}$.

For every integer $k > \rho$, we can rewrite k as $k = k_0 + m\rho$, where $0 < k_0 \leq \rho$ and $m \in \mathbb{N}$. Suppose there exists a solution $k_s > \rho$. According to the simultaneous incongruences problem, we know that $k_s \not\equiv a_i \pmod{b_i}$ for all i, where $1 \leq i \leq n$. So we can find a k_0, where $0 < k_0 \leq \rho$, and a positive integer m such that

$$k_s \equiv k_0 + m\rho \not\equiv a_i \pmod{b_i},$$

for every i, where $1 \le i \le n$. But $\rho \equiv 0 \pmod{b_i}$ for all $1 \le i \le n$, which implies that

$$k_0 \not\equiv a_i \pmod{b_i}$$

for all $1 \le i \le n$, thus k_0 is the solution we want. □

Theorem 1. *The reachability problem for bounded* 3*-RHPCD systems is NP-hard.*

Proof. Consider an instance of the simultaneous incongruences problem with n pairs. We will encode the instance into a reachability problem for a 3-RHPCD. Starting from $k = 1$, we test whether $k \bmod b_i \neq a_i$ holds for each pair (a_i, b_i). If it does hold for every i, then the current value of k is the solution. If for some i we find $k \bmod b_i = a_i$, then the current value of k is not a potential solution. We increase the value of k by 1 and start the testing all over again. By Lemma 1 there are at most ρ integers to test.

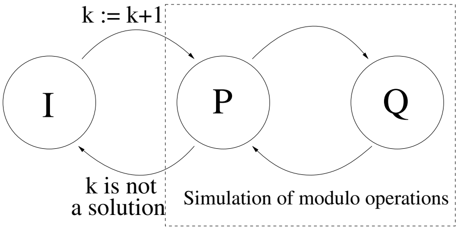


Fig. 1. Reachability problem for 3-RHPCD (Location I actually represents 3 locations I_1, I_2 and I_3)

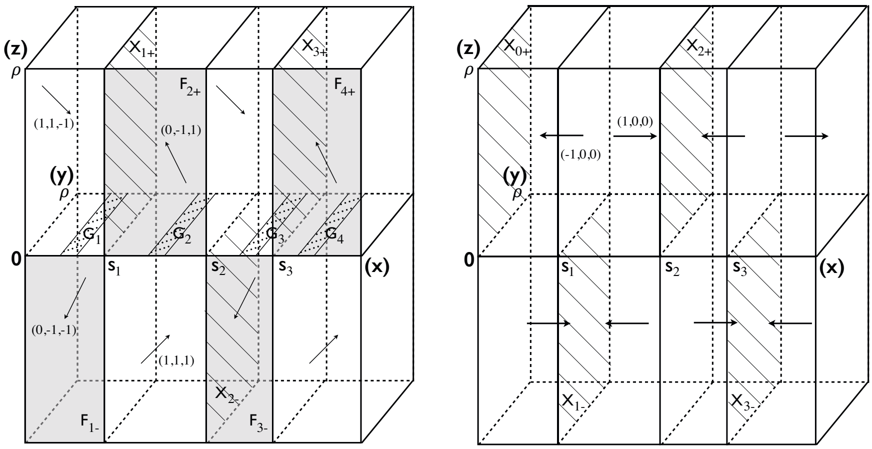
We construct the corresponding 3-RHPCD in the following way. We define 5 locations P, Q, I_1, I_2 and I_3. Locations P and Q together can simulate the modulo operation test for a certain value of k and every pair of (a_i, b_i). Locations I_1, I_2 and I_3 can increase the value of k by 1 when we find the current k is not a potential solution. See Fig. 1. Define regions A_i and B_i in locations P and Q :

$$A_i = (s_{i-1}, s_i) \times (0, \rho) \times (0, \rho);$$
$$B_i = (s_{i-1}, s_i) \times (0, \rho) \times (-\rho, 0),$$

where $i \in \{1, 2, ..., n\}$, $s_0 = 0$, $s_i = \sum_1^i b_i$ for $1 \le i \le n$, and $\rho = \mathrm{lcm}(b_1, \ldots, b_n)$. We call a region *odd* (resp. *even*) A_i or B_i if i is odd (resp. *even*). We also define surface O :

$$O = [0, s_n] \times [0, \rho] \times \{0\}.$$

To simulate the modulo operation for a certain pair (a_i, b_i), we use the regions odd $\overline{A}_i$ and even $\overline{B}_i$ in both locations P and Q. Define the derivative to be $(1, 1, -1)$ in odd $\overline{A}_i$ in P ($(1, 1, 1)$ in even $\overline{B}_i$ in P) and $(-1, 0, 0)$ in both odd $\overline{A}_i$ and even $\overline{B}_i$ in Q. See Fig. 2. Intuitively, we arrange the regions alternately



(a) Location P (b) Location Q

Fig. 2. 3-RHPCD simulating simultaneous incongruences problem (only location P and location Q are shown)

above and below the O surface instead of stacking them together. This is to avoid them sharing a common surface, which may cause nondeterminism when we define a (transition) guard on that surface.

For a point (x, y, z), we use the z coordinate to represent the current value of k and the y coordinate as a memory. Assuming i is odd (see Table. 1 for full details of both odd and even cases), we start at point $\boldsymbol{x}_0 = (s_{i-1}, 0, k)$ in P and move according to the flow $\dot{\boldsymbol{x}} = (1, 1, -1)$. While $|z| > 0$, every time when $x = b_i + s_{i-1} = s_i$, we jump to Q. In Q we keep variables y and z unchanged, simply reset x to 0 by the flow $\dot{\boldsymbol{x}} = (-1, 0, 0)$ and jump back to P. Each time the trajectory goes from P to Q and jumps back to P, the absolute value of variable z will be subtracted by b_i. So when the trajectory hits the O surface (i.e., $z = 0$), the value of x will be equal to $s_{i-1} + (k \bmod b_i)$. Since y and z in P change at the same rate, when the absolute value of z drops from k to 0, the value of y will increase from 0 to k.

If $k \bmod b_i \neq a_i$, we reset y to 0 and $|z|$ to k by switching the value of these two variables, and enter region $\overline{B}_{(i+1)}$ to test whether $k \bmod b_{i+1} \neq a_{i+1}$. To do this, we use the regions odd $\overline{B}_i$ and even $\overline{A}_i$ in both locations P and Q. Define the derivative to be $(0, -1, -1)$ in odd $\overline{B}_i$ in P ($(0, -1, 1)$ in even $\overline{A}_i$ in P) and $(1, 0, 0)$ in both odd $\overline{B}_i$ and even $\overline{A}_i$ in Q. By the flows in P the value of y and $|z|$ are switched. When $y = 0$ we jump to Q and reset x to s_i, and then jump back to P to start testing the case of pair (a_{i+1}, b_{i+1}).

If $k \bmod b_i = a_i$, which means that the current value of k is not a potential solution, we jump to locations I_1, and then I_2 and I_3, (defined in Table 1) which moves the trajectory to point $(0, 0, k+1)$ and 'restarts' in location P to test whether the new value $k+1$ is a correct solution [1]. A correct solution k should satisfy that the trajectory starts from point $(0,0,k)$ in location P and can finally reach some point (in location P) on the surface $(s_{n-1}, s_n) \times (0,\rho) \times \{0\}$ with $x \notin (s_{n-1} + a_n - \frac{\varepsilon}{2}, s_{n-1} + a_n + \frac{\varepsilon}{2})$.

Table 1. Reachability problem for 3-RHPCD

<table>
<tr><th>Location</th><th>Region</th><th>Flows</th><th>Guards</th></tr>
<tr><td rowspan="2">P</td><td rowspan="2">$\overline{A} \cup \overline{B}$</td><td rowspan="2">$\overline{A}_i$ (i is odd): $(1,1,-1)$
$\overline{A}_i$ (i is even): $(0,-1,1)$
$\overline{B}_i$ (i is odd): $(0,-1,-1)$
$\overline{B}_i$ (i is even): $(1,1,1)$</td><td>$\overline{X}_{i+}$ $(i = 1,3,...,n-1)$,
$\overline{X}_{i-}$ $(i = 2,4,...n)$,
$\overline{F}_{i+}$ $(i = 2,4,...,n)$,
$\overline{F}_{i-}$ $(i = 1,3,...,n-1)$:
jump to Q</td></tr>
<tr><td>G_i :
jump to I_1</td></tr>
<tr><td>Q</td><td>$(\overline{A} \cup \overline{B}) \setminus C$</td><td>$\overline{A}_i$ (i is odd): $(-1,0,0)$
$\overline{A}_i$ (i is even): $(1,0,0)$
$\overline{B}_i$ (i is odd): $(1,0,0)$
$\overline{B}_i$ (i is even): $(-1,0,0)$</td><td>$\overline{X}_{i+}$ $(i = 0,2,...,n-2)$,
$\overline{X}_{i-}$ $(i = 1,3,...,n-1)$:
jump to P</td></tr>
<tr><td>I_1</td><td>$\overline{A}$</td><td>$(-1,0,0)$</td><td>$x = 0$
jump to I_2</td></tr>
<tr><td>I_2</td><td>$\overline{A}$</td><td>$(0,0,1)$</td><td>$z = 1$
jump to I_3</td></tr>
<tr><td>I_3</td><td>$\overline{A}$</td><td>$(0,-1,1)$</td><td>$y = 0$
jump to P</td></tr>
</table>

We now give the formal details of this construction. Without loss of generality, we assume n is even. Define 2 regions A and B :

$$A = \cup_1^n A_i;$$
$$B = \cup_1^n B_i.$$

Also define four types of surfaces F_{i+}, F_{i-}, X_{i+} and X_{i-} :

$$\begin{aligned}
F_{i+} &= (s_{i-1}, s_i) \times \{0\} \times (0,\rho), && i = 1,2,...,n; \\
F_{i-} &= (s_{i-1}, s_i) \times \{0\} \times (-\rho,0), && i = 1,2,...,n; \\
X_{i+} &= \{s_i\} \times (0,\rho) \times (0,\rho), && i = 0,1,2,...,n; \\
X_{i-} &= \{s_i\} \times (0,\rho) \times (-\rho,0), && i = 0,1,2,...,n.
\end{aligned}$$

[1] Note that here in the guards we do not require exactly $x = a_i + s_{i-1}$, but allow some error ε, so tiny perturbations will not affect our result. The same analysis can be applied to Theorem 2. This seems to imply that our system has robust reachability and mortality problems, but we do not expand on the details in this paper. See more details about robustness in [10]

Finally, we define a set of ε-width strips G_i and a set of ε-width cubes C :

$$G_i = (s_{i-1} + a_i - \tfrac{\varepsilon}{2}, s_{i-1} + a_i + \tfrac{\varepsilon}{2}) \times [0, \rho] \times \{0\}, \qquad i = 1, 2, ..., n;$$
$$C = \cup_1^{n-1} C_i,$$

where

$$C_i = \begin{cases} (s_i, s_i + \varepsilon) \times (0, \rho) \times (0, \rho), & \text{if} \quad i = 1, 3, ..., n-1; \\ (s_i, s_i + \varepsilon) \times (0, \rho) \times (-\rho, 0), & \text{if} \quad i = 2, 4, ..., n-2. \end{cases}$$

The set C is to prevent nondeterminism in location Q. With the help of these notations, we construct the 3-RHPCD in Table. 1.

The number of regions and guards in the constructed 3-RHPCD is clearly polynomial in the number of pairs of the simultaneous incongruences problem. Furthermore, the points defining each such region can be represented in binary and are therefore polynomial in the description size of the simultaneous incongruences problem. Therefore the constructed 3-RHPCD has a polynomial description size. □

Theorem 2. *The mortality problem for a bounded* 3*-RHPCD system is NP-hard.*

Proof. We simulate a simultaneous incongruences problem by a bounded 3-RPHCD. The mortality problem asks whether for a certain system, starting from every initial configuration, the trajectory will eventually reach some fixed-point configuration, which we call the mortal configuration (in this case, the system is called mortal). Once we reach the mortal configuration, since it is a fixed point of the system, we assume the simulation halts since the point itself never changes. We construct our 3-RHPCD in such a way that the system is mortal if and only if there is no solution for the corresponding simultaneous incongruences problem, otherwise the system is immortal (i.e., starting from some configurations the system never reaches the mortal configuration).

For a pair (a_i, b_i) in the simultaneous incongruences problem, the derivatives of the associated regions $\overline{A}_i$ and $\overline{B}_i$ in locations P and Q are defined the same as in the proof of Theorem 1. In contrast to Theorem 1, in the mortality problem, we are not only concerned about some trajectories starting from certain points $(0, 0, k), 0 < k \leq \rho$, but want to know whether *all* the trajectories reach the mortal configuration.

In the following part we assume i is odd, similar analysis can be applied to the case when i is even. According to the flow $\dot{\boldsymbol{x}} = (1, 1, -1)$ of an odd region $\overline{A}_i$ in location P, there are 2 boundaries the trajectories will eventually reach: the O surface and the $y = \rho$ surface (some trajectories may also reach the $\overline{X}_{i+}$ or $\overline{X}_{i-}$ surface, but they will jump to location Q and jump back, then reach either one of the above two surfaces at the end). In odd $\overline{A}_i$ in P, all the trajectories which reach the $y = \rho$ surface or reach the strip G_i on the O surface are considered as mortal trajectories and will jump to location M_1, in which all the trajectories will eventually reach the mortal configuration of point $(0, 0, 0)$ in locations $\{M_1, M_2, M_3\}$. The trajectories which reach the O surfaces but do not reach the strip G_i are considered as the potential solution trajectories and move on by following the flows for a further check.

In contrast to the proof of Theorem 1, in region $\overline{A}_n$ (or $\overline{B}_n$ depending on if i is odd or even) if any trajectory reaches the surface O but does not reach the strip G_n, we do not conclude that we find a solution k. Instead, we keep moving in P until we hit the guard, jump to location T, reset the trajectory to the point $(0, 0, k)$ and go to location P to start the test again. If k indeed is a correct solution to the corresponding simultaneous incongruences problem, the system will loop forever; otherwise the trajectory will go to location M_1 at some region odd $\overline{A}_i$ or even $\overline{B}_i$ in location P. Full details are shown in Table. 2. □

Table 2. Mortality problem for 3RHPCD

Location	Region	Flows	Guards
P	$\overline{A} \cup \overline{B}$	$\overline{A}_i$ (i is odd): $(1, 1, -1)$ $\overline{A}_i$ (i is even): $(0, -1, 1)$ $\overline{B}_i$ (i is odd): $(0, -1, -1)$ $\overline{B}_i$ (i is even): $(1, 1, 1)$	$\overline{X}_{i+}$ $(i = 1, 3, ..., n-1)$, $\overline{X}_{i-}$ $(i = 2, 4, ...n)$, $\overline{F}_{i+}$ $(i = 2, 4, ..., n)$, $\overline{F}_{i-}$ $(i = 1, 3, ..., n-1)$: jump to Q
			$(y = \rho), G_i$: jump to M_1
Q	$(\overline{A} \cup \overline{B}) \setminus C$	$\overline{A}_i$ (i is odd): $(-1, 0, 0)$ $\overline{A}_i$ (i is even): $(1, 0, 0)$ $\overline{B}_i$ (i is odd): $(1, 0, 0)$ $\overline{B}_i$ (i is even): $(-1, 0, 0)$	$\overline{X}_{i+}$ $(i = 0, 2, ..., n-2)$, $\overline{X}_{i-}$ $(i = 1, 3, ..., n-1)$: jump to P
			$\overline{X}_{n+}$: jump to T
T	$\overline{A} \cup \overline{B}$	$(-1, 0, 0)$	$x = 0$: jump to P
M_1	$\overline{A} \cup \overline{B}$	$\overline{A} : (0, 0, -1)$ $\overline{B} : (0, 0, 1)$	z=0: jump to M_2
M_2	$\overline{A} \cup \overline{B}$	$(-1, 0, 0)$	x=0: jump to M_3
M_3	$\overline{A} \cup \overline{B}$	$(0, -1, 0)$	y=0: jump to M_1

Theorem 3. *Reachability and mortality are undecidable for unbounded 3-RHPCD systems.*

Proof. Both problems can be shown to be undecidable via a simulation of a two counter (Minsky) machine which represents a universal model of computation [12]. However we omit the details here due to page limit. □

The following proposition gives an upper bound of the complexity for both the reachability and mortality problems for bounded n-RHPCDs.

Proposition 1. *The reachability and mortality problems for bounded n-RHPCDs are in PSPACE.*

Proof. The proof is similar to that used to show that reachability for a 2-RHPCD is decidable, as was shown in [5]. Given an n-RHPCD $\mathcal{H}$, an initial configuration $(q_0, \boldsymbol{x}_0)$ and a target configuration $(q_f, \boldsymbol{x}_f)$, we show that starting from $(q_0, \boldsymbol{x}_0)$, the trajectory will hit the internal and transition guards finitely many times before either reaching $(q_f, \boldsymbol{x}_f)$, or detecting a cycle, or hitting some endpoints (at which the calculation halts), thus 'convergence' to a point is possible.

By the definition of n-RHPCD, the guards of $\mathcal{H}$ are of the form

$$\left(\bigwedge_{1 \le i \le n \,\wedge\, i \ne j} (a_i \prec x_i \prec' b_i) \right) \wedge (x_j = c_j)$$

where $j \in \{1, \ldots, n\}, x_i, x_j, a_i, b_i, c_i \in \mathbb{Q}$, and $\prec, \prec' \in \{<, \le\}$.

By definition, the components of $\boldsymbol{x}_0 = (x_{0_1}, \ldots, x_{0_n})$ and $\boldsymbol{x}_f = (x_{f_1}, \ldots, x_{f_n})$ are rational numbers, i.e., $\boldsymbol{x}_0, \boldsymbol{x}_f \in \mathbb{Q}^n$. Define

$$\gamma = \mathrm{lcd}(a_1, \ldots, a_n, b_1, \ldots, b_n, c_j, x_{0_1}, \ldots, x_{0_n}, x_{f_1}, \ldots, x_{f_n}),$$

where lcd denotes the *least common denominator*, and define

$$A_i = \gamma a_i, B_i = \gamma b_i, C_j = \gamma c_j, X_0 = \gamma \boldsymbol{x}_0, X_f = \gamma \boldsymbol{x}_f.$$

Thus, $A_i, B_i, C_j \in \mathbb{Z}$ and $\boldsymbol{X}_0, \boldsymbol{X}_f \in \mathbb{Z}^n$. Define a new n-RHPCD $\mathcal{H}'$ with initial configuration $(q_0, \boldsymbol{X}_0)$ and target configuration $(q_f, \boldsymbol{X}_f)$ by replacing a_i, b_i, c_j, $\boldsymbol{x}_0, \boldsymbol{x}_f$ by $A_i, B_i, C_j, \boldsymbol{X}_0, \boldsymbol{X}_f$. Clearly, $\mathcal{H}$ reaches $\boldsymbol{x}_f$ iff $\mathcal{H}'$ reaches $\boldsymbol{X}_f$.

Because all the flows of $\mathcal{H}'$ are chosen from the set $\{0, 1, -1\}$, when one variable x_i changes its value from one integer to another integer, any other variable x_j remains an integer. As the trajectory starts at integer point X_0, and the guards of $\mathcal{H}'$ are defined by integers, every time the trajectory hits a guard, it will have integer components.

We now prove that the problem can be solved in PSPACE. Note that the representation size of γ is clearly polynomial in the representation size of $\mathcal{H}$, thus so is the size of $\mathcal{H}'$. We now show that the representation size of the number of possible transition configurations (the configuration when the trajectory hits the guard and takes transition) of $\mathcal{H}'$ is also polynomial in the size of $\mathcal{H}$.

Let $k > 0$ be the number of locations of $\mathcal{H}'$. Since $\mathcal{H}$ is bounded, we can calculate $\tau \in \mathbb{N}$ to be the maximal absolute value of the endpoint of any invariant of $\mathcal{H}$ over all locations. Thus the range of variables of $\mathcal{H}'$ is contained within $[-\gamma\tau, \gamma\tau]$. Since we have n variables, the maximal number of transition configurations of $\mathcal{H}'$, starting at initial configuration $(q_0, \boldsymbol{X}_0)$ is thus $k(2\gamma\tau)^n$, which can be represented in size polynomial in the size of $\mathcal{H}$, since it requires at least $k \log(\gamma\tau)^n = nk \log(\gamma\tau)$ space to store $\mathcal{H}$ and

$$\frac{\log(k(2\gamma\tau)^n)}{nk \log(\gamma\tau)} = \frac{\log(k) + n \log(2\gamma\tau)}{nk \log(\gamma\tau)} < c$$

for some computable constant $c > 0$. We can use a counter to keep track of the number of transitions the trajectory of $\mathcal{H}'$ makes, starting from $(q_0, \boldsymbol{X}_0)$. As each transition is taken, we can determine if the final configuration was reached since the last transition. Otherwise, we increment the counter and proceed. If the counter reaches $k(2\gamma\tau)^n$, then the configurations must be periodic and we can halt. Using a similar approach, we can also show that the mortality problem for n-RHPCDs is also in PSPACE, however we omit the details here. □

4 Conclusions

We showed that for bounded three-dimensional Restricted Hierarchical Piecewise Constant Derivative systems (3-RHPCDs), the reachability and mortality problems are NP-hard (using the simultaneous incongruences problem) but also in PSPACE, even in the n-dimensional case. For unbounded 3-RHPCDs, we showed that both problems are undecidable by an encoding of a Minsky machine. Clearly there is still a gap left for the complexity of the reachability and mortality problems for bounded n-RHPCDs. To close the gap we need to answer some interesting open problems:

- Is there a large n for which both problems for n-RHPCD are PSPACE-hard?
- Can both problems be solved in NP in dimension three?
- Can both problems be solved in P in dimension two?

The model of RHPCD restricts various components of the hybrid automaton in ways which have parallels to other models, such as stopwatch automata, rectangular automata and PCDs. RHPCDs have decidable reachability problems for them but endowing them with small additional powers renders them much more powerful. Therefore they seem a useful tool in studying the frontier of undecidability and tractability, in a similar way to the model of HPCD which inspired them.

Acknowledgements: We would like to thank the anonymous referees for their very useful suggestions and comments.

References

1. Alur, R., Courcoubetis, C., Halbwachs, N., Henzinger, T.A., Ho, P.H., Nicollin, X., Olivero, A., Sifakis, J., Yovine, S.: The algorithmic analysis of hybrid systems. Theoretical Computer Science 138(1), 3–34 (1995)
2. Asarin, E., Maler, O., Pnueli, A.: Reachability analysis of dynamical systems having piecewise constant derivatives. Theoretical Computer Science 138, 35–65 (1995)
3. Asarin, E., Mysore, V., Pnueli, A., Schneider, G.: Low dimensional hybrid systems - decidable, undecidable, don't know. Information and Computation 211, 138–159 (2012)
4. Asarin, E., Schneider, G.: Widening the boundary between decidable and undecidable hybrid systems. In: Brim, L., Jančar, P., Křetínský, M., Kučera, A. (eds.) CONCUR 2002. LNCS, vol. 2421, pp. 193–208. Springer, Heidelberg (2002)
5. Bell, P.C., Chen, S.: Reachability problems for hierarchical piecewise constant derivative systems. In: Abdulla, P.A., Potapov, I. (eds.) RP 2013. LNCS, vol. 8169, pp. 46–58. Springer, Heidelberg (2013)

6. Blondel, V.D., Bournez, O., Koiran, P., Papadimitriou, C., Tsitsiklis, J.N.: Deciding stability and mortality of piecewise affine dynamical systems. Theoretical Computer Science 255(1-2), 687–696 (2001)
7. Bouyer, P., Dufourd, C., Fleury, E., Petit, A.: Updatable timed automata. Theoretical Computer Science 321(2), 291–345 (2004)
8. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman and Co., New York (1979)
9. Henzinger, T., Kopka, P., Puri, A., Varaiya, P.: What's decidable about hybrid automata? In: 27th ACM STOC, pp. 373–382. ACM Press (1995)
10. Henzinger, T.A., Raskin, J.-F.: Robust undecidability of timed and hybrid systems. In: Lynch, N.A., Krogh, B.H. (eds.) HSCC 2000. LNCS, vol. 1790, pp. 145–159. Springer, Heidelberg (2000)
11. Maler, O., Pnueli, A.: Reachability analysis of planar multi-linear systems. In: Courcoubetis, C. (ed.) CAV 1993. LNCS, vol. 697, pp. 194–209. Springer, Heidelberg (1993)
12. Minsky, M.: Computation: Finite and Infinite Machines. Prentice-Hall International, Englewood Cliffs (1967)