The Importance of Being Bounded *

Alberto Casagrande[1,2], Venkatesh Mysore[3], Carla Piazza[2], and Bud Mishra[3,4]

[1] PARADES, Via S.Pantaleo, 66, 00186 Roma, Italy
[2] DIMI, Università di Udine, Via delle Scienze, 206, 33100 Udine, Italy
[3] Courant Institute, NYU, New York, USA
[4] NYU School of Medicine, NYU, New York, U.S.A.

Abstract. In this paper we introduce and study a new class of hybrid automata, *Independent Dynamics Hybrid Automata* (IDA). IDA are an extension of decidable O-minimal automata in which also identity resets are allowed. We define the conditions under which reachability is decidable over IDA. These conditions involve the satisfiability of first-order formulæ that limit the interval of time we need to consider to study reachability. In order to prove the decidability of reachability we mainly exploit the decidability of the first-order formulæ which define IDA. Then we introduce the subclass ∞IDA of IDA over which reachability is always decidable. An interesting subclass of ∞IDA is the class of IDA whose flows are non-constant polynomials. IDA and ∞IDA are usefull in the modeling of biological systems where it is possible to have variables which continue their flows independently (e.g., input reactants coming from other systems). We briefly comment on how to model bacterial chemotaxis using IDA.

1 Introduction

Hybrid automata have emerged as the predominant formalism for modeling and analyzing biochemical systems [5]. The different discrete states naturally capture the different regimes of cellular behavior, while the flow equations in each state correspond to the time-variation of their concentrations based on the laws of chemical kinetics. Guards are necessary to capture the fact that the biochemical system changes its state only when certain criteria are met, while reset relations are used to capture the new conditions from where the system begin its evolution in the new mode/phase.

The notion of *Hybrid Automata* was first introduced in [2] as a model and specification language for hybrid systems, i.e., systems consisting of a discrete program within a continuously changing environment. Since their introduction

* This work is partially done in the framework of the HYCON Network of Excellence, contract number FP6-IST-511368, and supported by NSF's ITR program, Defense Advanced Research Projects Agency (DARPA), the US Air Force (AFRL), National Institutes of Health (NIH) and New York State Office of Science, Technology & Academic Research (NYSTAR), the European Community Project IST-2001-33520 CC, the MIUR FIRB grant RBAU018RCZ, and the MIUR PRIN'04 grant 2004013015.

they have been widely used for the automatic verification of both natural and engineered systems. It is only more recently that their utility in Systems Biology is is being appreciated. However, the omniprescent tussle between complexity of the system and the ease of analysis continues to dictate which hybrid automaton subclass is suitable for which application. For example, though remarkably efficient verification techniques have been perfected for Timed Automata [1], the very idea of modeling a biochemical system as a set of well-behaved clocks is quite outrageous. At the other end of the spectrum, detailed spatio-temporal models at the atomic level, though perhaps extremely accurate, will leave simulation-based analysis as the only feasible option. As a reasonable intermediate, *Semi-Algebraic Hybrid Automata* [25, 24] were identified as the largest subclass amenable to algebraic analysis, with their suitability for Systems Biology well described.

In [10], we introduced the *Semi-Algebraic Constant reset Hybrid Automata* (SACoRe), which extended O-minimal automata over the reals, in the case of flows obtained from non-autonomous systems of differential inclusions. SACoRe automata were shown to admit decision procedures for reachability and model checking for a limited fragment of CTL, by combining Tarski's decidability result over the reals and Michael's selection theorem. However, this formalism is still quite restrictive to the biochemical domain as the constant reset requirement is very constraining. This is because, when a biochemical system changes its "discrete" state, it is very unnatural for the concentrations to be reset to constant values. In fact, the most common result of a state change is no change, because of the continuous nature of chemical concentrations and other biochemical variables. In other words, identity resets are necessary to capture this fundamental aspect of most biological state transitions.

In this paper, we introduce and study a new class of hybrid automata – *Independent Dynamics Hybrid Automata* (IDA), whose characterizing conditions are based upon a decidable first-order theory over the reals (e.g., $(\mathbb{R}, 0, 1, +, *, =, <)$). In particular, an hybrid automaton of dimension k can be defined only using formulæ over k dimensional vectors of reals. The dynamics are solutions of autonomous systems of differential equations. The reset conditions can be either constants as in the case of O-minimal hybrid automata [19] or the identity function. In particular, we distinguish *independent variables*, whose resets are the identity function, from *dependent variables* whose resets are constant functions. The flows and the reset functions of the dependent variables can depend on the independent ones, but not vice-versa.

Our motivation for defining this new subclass has two sources: (1) Extend O-minimal automata to make them suitable for Systems Biology applications; (2) Restrain Semi-Algebraic Hybrid Automata and make them more amenable to analysis. We exploit the decidability of the first-order theory over which an IDA is defined, both to bound the time interval we need to consider to solve a reachability problem, and to prove the decidability of reachability. The bounds on the time interval do not always exists on IDA, but we can prove that they are always defined on an interesting subclass of IDA which we call ∞IDA. As a consequence, reachability is always decidable on ∞IDA. It is important to observe that we do not explicitly compute these time bounds, but we check their existence, again, by solving a satisfiability problem.

When the IDA we consider are defined on the first-order theory ($\mathbb{R}$, 0, 1, +, $*$, =, <), our approach exploits Tarski's result and quantifier elimination to study reachability. Similar approaches have been under investigation, over the last decade or so. For instance, Jirstrand [17] demonstrated, in the context of non-linear control system design, the use of `Qepcad` for the problems of computing reachability, stationarizable sets, range of controllable output, and curve-following. Subsequently, Anai [4] and Franzle [22] independently suggested the use of quantifier elimination for the verification of polynomial hybrid systems. Franzle went on to prove that progress, safety, state recurrence and reachability are semi-decidable using quantifier elimination [12], and developed "proof engines" for bounded model checking [13]. More recently, Lafferiere et al. [20] have again described a method based upon quantifier elimination for symbolic reachability computation of linear vector fields.

Lately, Ratschan and She [26] have suggested a new constraint propagation based abstraction refinement for the safety verification of hybrid systems with autonomous differential equations. Other recent developments include Becker et al.'s integration of bounded model checking and inductive verification [7]. Lanotte and Tini [21] have recently proved that the semi-algebraic hybrid automaton obtained by approximating each formula in any hybrid system definition with its Taylor polynomial is an over-approximation. The novelty of our approach mainly lies in the use of the formulæ to bound the interesting time interval a-priori, and in our observation that continuity of the dynamics and compactness of the invariants can be exploited to define the time-bound.

The paper is organized as follows. In Section 2, we introduce some basic notions about graphs and hybrid automata. In Section 3, we introduce our class of automata. In Section 4, we start considering the reachability problem from a general perspective, and then investigate the conditions under which we can prove the decidability of reachability on IDA. In Section 5, we identify an interesting subclass of IDA over which reachability is always decidable. We conclude in Section 7, with consideration of extensions and applications.

2 Preliminaries

2.1 Hybrid Automata

First, we introduce some notations and conventions. Capital letters Z_m, Z'_m, where $m \in \mathbb{N}$, denote variables ranging over $\mathbb{R}$. Analogously, Z denotes the vector of variables $\langle Z_1, \ldots, Z_k \rangle$ and Z' denotes the vector $\langle Z'_1, \ldots, Z'_k \rangle$; and Z^n denotes the vector $\langle Z^n_1, \ldots, Z^n_k \rangle$. Moreover, if $X = \langle X_1, \ldots, X_m \rangle$ is a vector of variables, Γ_X denotes the set $\{X_1, \ldots, X_m\}$. The temporal variables T and T' model time and range over $\mathbb{R}^+$. We use the small letters $p, q, r, s, \ldots$ to denote k-dimensional vectors of real numbers.

Occasionally, we will use the notation $\varphi[X_1, \ldots, X_m]$ to stress the fact that the set of free variables of the first-order formula φ, denoted by $Free(\varphi)$, is included in the set of variables $\{X_1, \ldots, X_m\}$. By extension, if $\{X^1, \ldots, X^n\}$ is a set of variable vectors, $\varphi[X^1, \ldots, X^n]$ indicates that the free variables of φ are included in the set of components of $X^1, \ldots, X^n$. Moreover, given a formula $\varphi[X^1, \ldots, X^i, \ldots, X^n]$ and a vector p of the same dimension as the variable vector X^i, the formula obtained by component-wise substitution of X^i with p is denoted by $\varphi[X^1, \ldots, X^{i-1}, p, X^{i+1}, \ldots, X^n]$. If in φ the only free variables were the components of X^i, after the substitution we can compute the truth value of $\varphi[p]$. Later on, given a formulæ $\psi[Z]$, we will denote the set of values satisfing ψ as $\mathrm{Sat}(\psi)$, i.e., $\mathrm{Sat}(\psi) = \{\, p \mid \psi[p]\}$.

We are now ready to formally introduce hybrid automata. For each node of a graph, we have an invariant condition and a dynamic law. This dynamic law may depend on the initial conditions, i.e., on the values of the continuous variables at the beginning of the evolution in the state. The jumps from one discrete state to another are regulated by the activation and reset conditions.

Definition 1 (Hybrid Automata - Syntax). *A* hybrid automaton $H = (Z, Z', \mathcal{V}, \mathcal{E}, Inv, \mathcal{F}, Act, Reset)$ *of dimension k consists of the following components:*

1. *$Z = \langle Z_1, \ldots, Z_k\rangle$ and $Z' = \langle Z'_1, \ldots, Z'_k\rangle$ are two vectors of variables ranging over the reals $\mathbb{R}$;*
2. *$\langle \mathcal{V}, \mathcal{E}\rangle$ is a graph; the objects, $v \in \mathcal{V}$, are called* locations*;*
3. *Each vertex $v \in \mathcal{V}$ is labeled by the formula $Inv(v)[Z]$;*
4. *$\mathcal{F}$ is a function assigning to each vertex $v \in \mathcal{V}$ a continuous vector field over $\mathbb{R}^k$; we will use $f_v : \mathbb{R}^k \times \mathbb{R}^+ \longrightarrow \mathbb{R}^k$ to indicate the solution of the vector field $\mathcal{F}(v)$ and $Dyn(v)[Z, Z', T]$ to identify the corresponding formula, i.e., $Dyn(v)[Z, Z', T] \equiv Z' = f_v(Z, T)$;*
5. *Each edge $e \in \mathcal{E}$ is labeled by the two formulæ $Act(e)[Z]$ and $Reset(e)[Z, Z']$; $\overline{Reset}(e)[Z'] \overset{\mathrm{def}}{=} \exists Z\ Reset(e)[Z, Z']$.*

To easy notation, later on, we will write $\mathcal{I}(v)$, $\mathcal{A}(e)$, and $\mathcal{R}(e)$ in place of $\mathrm{Sat}(Inv(v))$, $\mathrm{Sat}(Act(e))$, and $\mathrm{Sat}(\overline{Reset}(e))$ respectively.

Notice that it is not restrictive to consider only hybrid automata whose formulæ are satisfiable. In fact, if this is not the case, we can transform the automaton and eliminate the unsatisfiable formulæ. For instance, if there is an edge e such that $Reset(e)[Z, Z']$ is unsatisfiable we can simply delete the edge from the automaton.

Here we always use k to denote the dimensions of the hybrid automaton H. Moreover, when we refer to a hybrid automaton H, we implicitly assume that H is of the form $H = (Z, Z', \mathcal{V}, \mathcal{E}, Inv, \mathcal{F}, Act, Reset)$.

Definition 2 (Hybrid Automata - Semantics). *A* state *ℓ of H is a pair $\langle v, r\rangle$, where $v \in \mathcal{V}$ is a location and $r = \langle r_1, \ldots, r_k\rangle \in \mathbb{R}^k$ is an assignment of values for the variables of Z. A state $\langle v, r\rangle$ is said to be* admissible *if $Inv(v)[r]$ is true.*

The continuous reachability transition relations *$\xrightarrow{t}_C$, where $t > 0$ the transition elapsed time, between admissible states is defined as follows:*

$\langle v, r\rangle \xrightarrow{t}_C \langle v, s\rangle \iff$ *The equation $s = f_v(r, t)$ holds, and for each $t' \in [0, t]$ the formula $Inv(v)[f_v(r, t')]$ is true.*

The discrete reachability transition relation $\to_D$ *between admissible states is defined as follows:*

$\langle v, r\rangle \to_D \langle u, s\rangle \iff$ *The relation $\langle v, u\rangle \in \mathcal{E}$ holds, and the formulæ $Act(\langle v, u\rangle)[r]$ and $Reset(\langle v, u\rangle)[r, s]$ are true.*

Building upon continuous and discrete transitions, we can introduce the notions of *trace* and *reachability*. A trace is a sequence of continuous and discrete transitions. A point s is reachable from a point r, if there is a trace starting from r and ending in s. We use the notation $\ell \to \ell'$ to denote that either $\ell \xrightarrow{t}_C \ell'$, for some t, or $\ell \to_D \ell'$.

Definition 3 (Hybrid Automata - Reachability). *Let I be either $\mathbb{N}$ or an initial interval of $\mathbb{N}$. A* trace *of H is a sequence $\ell_0, \ell_1, \ldots, \ell_i$, with $i \in I$, of admissible states such that $\ell_{i-1} \to \ell_i$ holds for each $i \in I$ with $i > 0$. Such trace is also denoted by $(\ell_i)_{i\in I}$.*

A point $r \in \mathbb{R}^k$ reaches *a point $s \in \mathbb{R}^k$ (in time t), if there exists a trace $\ell_0, \ldots, \ell_n$ of H such that $\ell_0 = \langle v, r\rangle$ and $\ell_n = \langle u, s\rangle$, for some $v, u \in \mathcal{V}$ (and t is the sum of the elapsed continuous transition times).*

Given a trace of H, we can identify a path of $\langle \mathcal{V}, \mathcal{E}\rangle$ as follows.

Definition 4 (Corresponding Path). *Let H be a hybrid automaton and let $tr = \langle v_0, r_0\rangle, \ldots, \langle v_n, r_n\rangle$ be a trace of H. The* corresponding path *of tr is the path $ph = \langle v'_1, \ldots, v'_m\rangle$ of the graph $\langle \mathcal{V}, \mathcal{E}\rangle$ obtained by considering the discrete transitions occurring in tr. In this case, we also say that ph* corresponds *to tr.*

In this paper we are interested in the reachability problem for hybrid automata, i.e., given a hybrid automaton H and two formulæ, ι and τ, denoting an initial set of points $\mathrm{Sat}(\iota) \subseteq \mathbb{R}^k$ and a target set of points $\mathrm{Sat}(\tau) \subseteq \mathbb{R}^k$ respectively, we want to decide whether there exists a point in $\mathrm{Sat}(\iota)$ which reaches a point in $\mathrm{Sat}(\tau)$.

Though it has been proved that reachability is in general not decidable [16], many interesting classes of hybrid automata over which reachability is decidable have been characterized in the literature [18, 15, 19, 9]. A common approach in deciding reachability of hybrid automata is that of discretizing the automata, either using equivalence relations which strongly preserve reachability (e.g., bisimulation [19]), or using abstractions (e.g., predicate abstraction [30, 3]). In this paper, we study reachability of hybrid automata by adopting a different strategy: translating the reachability problem into first-order formulæ over the reals. The formulæ we get from the translation include the formulæ occurring in the automata, and hence we need to know the theory using which the automata was built.

Definition 5 ($\mathbb{T}$-Automata). *Let $\mathbb{T}$ be a theory over the reals. A* $\mathbb{T}$-automaton *H is a hybrid automaton such that, for each $v \in \mathcal{V}$ and for each $e \in \mathcal{E}$, the formulæ $Dyn(v)$, $Inv(v)$, $Act(e)$, $Reset(e)$ are formulæ of $\mathbb{T}$.*

An interesting class of hybrid automata is the class of *O-minimal hybrid automata* [19,9]. Such automata are defined using formulæ taken from an O-minimal theory, i.e., they are $\mathbb{T}$-automata in which $\mathbb{T}$ is an O-minimal theory. Moreover, their resets are constant, i.e., they do not depend on the point from which the edge is crossed. In the case of O-minimal automata, reachability, as well as other temporal logic properties, can be decided through bisimulation [19]. In fact, O-minimal automata always have a finite bisimulation quotient, whose computation is effective when the O-minimal theory is decidable. A theory which is both O-minimal and decidable is the first-order theory $(\mathbb{R}, 0, 1, +, *, <)$ [29].

In the following, we try to relax the constant reset condition used in O-minimal automata, maintaining the decidability of the reachability problem. In particular, we will introduce a class of hybrid automata, the *independent dynamics automata*, and we will show how we can decide reachability.

3 Independent Dynamics Hybrid Automata

In our class of hybrid automata the components of Z can be partitioned into two sets: the *independent* and the *dependent* variables. We denote by X the vector of independent variables which maintains the same component ordering of Z. Similarly we indicate with Y the vector of dependent variables and with X' and Y' the primed version of X and Y respectively. The independent variables are never reset and their dynamics are the same in all the locations. This condition is similar to that used in rectangular initialized hybrid automata (see [15,18]). Moreover, we impose conditions that will ensure the existence of a minimum amount of time, which has to be spent in a location between two jumps. In particular, we impose that the invariants are closed and bounded, and that the distance between reset and activation regions is greater than 0. For this last condition we need to consider a norm $||\cdot||$ on $\mathbb{R}^k$, and the induced distance $d(\cdot,\cdot)$ between subsets of $\mathbb{R}^k$ defined as $d(A, B) = \inf\{||a - b|| \mid a \in A \text{ and } b \in B\}$. From now on we say that two edges e and e' are *subsequent* if the target node of e is the source node of e'.

Definition 6 (Independent Dynamics Automata). *A hybrid automaton H is an* independent dynamics *automaton, or simply an* IDA*, if:*

1. *H is a $\mathbb{T}$-automaton, with $\mathbb{T}$ decidable;*
2. *For each pair of subsequent edges e and e' it holds $d(\mathcal{R}(e), \mathcal{A}(e')) > 0$;*
3. *The vector Z of variables can be partitioned into two vectors X (independent variables) and Y (dependent variables) such that:*
   (a) *for each $e \in \mathcal{E}$ $Reset(e)[Z, Z']$ is of the form $(X' = X) \wedge \sigma(e)[Z, Y']$;*
   (b) *for each $v \in \mathcal{V}$ $Dyn(v)[Z, Z', T]$ is of the form $(X' = fi(X, T)) \wedge (Y = fd(v)(Z, T))$.*
4. *The set of values $p \in \mathbb{R}^k$ satisfing $Inv(v)[Z]$ is closed and bounded for each $v \in \mathcal{V}$.*

In condition 3 of the definition, we impose that the independent variables which are not reset have the same dynamics in each location. The reset and the dynamics of the dependent variables can depend on the independent variables.

Example 1. Consider the hybrid automaton $H = (Z, Z', \mathcal{V}, \mathcal{E}, Inv, \mathcal{F}, Act, Reset)$ such that:

- The dimension, k of the automata is 2;
- The discrete projection $\langle \mathcal{V}, \mathcal{E} \rangle$ is reported in Figure 1;
- The function $\mathcal{F}$ induces the following dynamics:
  - $Dyn(v_1)[Z, Z', T] \equiv Z'_1 = 2(T)^3 + Z_1 \ \wedge \ Z'_2 = -3T + Z_2$;
  - $Dyn(v_2)[Z, Z', T] \equiv Z'_1 = -(T)^2 \ \wedge \ Z'_2 = -3T + Z_2$;
  - $Dyn(v_3)[Z, Z', T] \equiv Z'_1 = 2(T)^3 + Z_1 \ \wedge \ Z'_2 = -3T + Z_2$;
- The formulæ $Inv(v_1)[Z]$, $Inv(v_2)[Z]$ and $Inv(v_3)[Z]$ are equal to $Z_2 \geq (Z_1)^2 \ \wedge \ Z_2 \leq 100$;
- The function *Reset* is such that:
  - $Reset(e_1)[Z, Z'] \equiv Z'_1 \leq 8 \ \wedge \ Z'_2 = Z_2$;
  - $Reset(e_2)[Z, Z'] \equiv Z'_1 = Z_1 \ \wedge \ Z'_2 = Z_2$;
  - $Reset(e_3)[Z, Z'] \equiv Z'_1 = Z_1 \ \wedge \ Z'_2 = Z_2$;
- The function *Act* is such that:
  - $Act(e_1)[Z] \equiv Z_2 > 5$;
  - $Act(e_2)[Z] \equiv Z_1 \ \wedge \ Z_2 \geq 0$;
  - $Act(e_3)[Z] \equiv (Z_1)^2 + (Z_2 - 5)^2 \leq 8$;

The automaton H is an IDA.

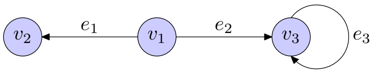

Fig. 1. The discrete component of the Example 1.

It is not difficult to extend our class of automata allowing different partitionings of the variables depending on the topology of the discrete structure. However, such an extension would require many technical details which we prefer to omit here.

4 IDA Reachability

In this section, we show that reachability over IDA is decidable, when there is a maximum time within which a set is reachable for the independent variables. This assumption, paired with the fact that in IDA we have to spend a minimum amount of time in each location, allows us to compute the maximum length of the corresponding paths that we need to consider.

4.1 Reachability Formulæ

We start introducing the first-order formulæ which encode the reachability along a path. First, we define the formulæ encoding the continuous reachability in a location and the discrete reachability through an edge.

Lemma 1. *Let H be a hybrid automaton, consider the first-order formulæ below:*

$$Reach(v)[Z, Z', T] \overset{\text{def}}{=} Z' = f_v(Z, T) \wedge \forall 0 \leq T' \leq T \; Inv(v)[f_v(Z, T')]$$

$$Reach(\langle v, u\rangle)[Z, Z'] \overset{\text{def}}{=} Act(\langle v, u\rangle)[Z] \wedge Reset(\langle v, u\rangle)[Z, Z']$$

Then $\langle v, r\rangle \xrightarrow{t}_C \langle v, s\rangle$, if and only if $Reach(v)[r, s, t]$ is true, and $\langle v, r\rangle \rightarrow_D \langle u, s\rangle$ if and only if $Reach(\langle v, u\rangle)[r, s]$ is true.

Proof. The thesis comes directly from the hybrid automata semantics (see Definition 2). □

Given a point $r \in \mathbb{R}^k$, the first-order formula $Reach(v)[r, Z', t]$ in Lemma 1, with free variables in Z', characterizes the set of points reachable from r in the node v using only t-timed continuous dynamics. Similarly, the first-order formula $Reach(e)[r, Z']$ defines the set of points reachable from r using the discrete transition e.

Now suppose that a point r reaches a point s in time t through a trace tr, whose corresponding path is $ph = \langle v, u\rangle$. Since our dynamics are solutions of autonomous differential equations, we see that $\langle v, r\rangle \xrightarrow{0}_C \langle v, r\rangle$ and $\langle u, s\rangle \xrightarrow{0}_C \langle u, s\rangle$. Hence, tr is equivalent to tr' of the form $\langle v, r\rangle \xrightarrow{t'}_C \langle v, r_1\rangle \rightarrow_D \langle u, s_1\rangle \xrightarrow{t''}_C \langle u, s\rangle$, where $t = t' + t''$. Thus, the reachability can always be expressed through a trace whose corresponding path is ph and results in the following first-order formula:

$$\begin{aligned} Reach(v, u)[Z^0, Z^1, Z^2, Z^3, T] \overset{\text{def}}{=} & \exists T_1 \geq 0 \; \exists T_2 \geq 0 \Big(Reach(v)[Z^0, Z^1, T_1] \wedge \\ & Reach(\langle v, u\rangle)[Z^1, Z^2] \wedge Reach(u)[Z^2, Z^3, T_2] \wedge \\ & T = T_1 + T_2) \end{aligned}$$

If we have a path $ph = \langle v_0, \ldots, v_h\rangle$ in the graph $\langle \mathcal{V}, \mathcal{E}\rangle$, then following two cases are possible: either it corresponds to a trace of H or it does not. In both cases, we can express the desired reachability relation with a first-order formula, which characterizes all the pairs of $\mathbb{R}^k$ that can be connected in H through a trace corresponding to path $ph = \langle v_0, \dots, v_h \rangle$:

$$Reach(ph)[Z^0, \dots, Z^{2h+1}, T] \overset{\text{def}}{=} \exists T_0 \geq 0 \dots \exists T_h \geq 0 \left(T = \sum_{i=0}^{h} T_i \wedge Reach(v_0)[Z^0, Z^1, T_0] \wedge \bigwedge_{i \in [0,h-1]} \Big(Reach(\langle v_i, v_{i+1} \rangle)[Z^{2i+1}, Z^{2i+2}] \wedge Reach(v_{i+1})[Z^{2i+2}, Z^{2i+3}, T_{i+1}] \Big) \right)$$

Notice that in the above formula we consider only traces in which continuous and discrete transitions are alternating. This is not restrictive, since our dynamics are solutions of autonomous systems of differential equations. Hence any trace can be mapped into a trace which satisfies the continuous/discrete alternation, with the same starting and finishing states. The following lemma proves that $Reach(ph)[Z^0, \dots, Z^{2h+1}, T]$ is correct and complete.

Lemma 2. *Let H be an automaton, and let $ph = \langle v_0, \dots, v_h \rangle$ be a path in $\langle \mathcal{V}, \mathcal{E} \rangle$. It holds that a point $r \in \mathbb{R}^k$ reaches a point $s \in \mathbb{R}^k$ in time t through a trace tr whose corresponding path is ph, if and only if $Reach(ph)[r, Z^1, \dots, Z^{2h}, s, t]$ is satisfiable.*

Proof. ($\Rightarrow$) Let $tr = \ell_0, \dots, \ell_n$ with $\ell_0 = \langle v_0, r \rangle$ and $\ell_n = \langle v_n, s \rangle$. If in tr there are two consecutive discrete transitions $\ell_i \to_D \ell_{i+1} \to_D \ell_{i+2}$ we can replace them by $\ell_i \to_D \ell_{i+1} \xrightarrow{0}_C \ell_{i+1} \to_D \ell_{i+2}$. Similarly, if in tr there are two consecutive continuous transitions $\ell_i \xrightarrow{t}_C \ell_{i+1} \xrightarrow{t'}_C \ell_{i+2}$ the fact that the flows are solutions of autonomous systems allows us to replace them with $\ell_i \xrightarrow{t''}_C \ell_{i+2}$, where $t'' = t + t'$. Hence, without loss of generality, we may assume that in tr discrete and continuous transitions are alternated. We may further assume that tr starts and ends with a continuous transition, since, otherwise, we may simply add either $\ell_0 \xrightarrow{0}_C \ell_0$ or $\ell_n \xrightarrow{0}_C \ell_n$ or both. Hence, without loss of generality, we have that $n = 2h$. Let $\ell_i = \langle v_i, r_i \rangle$ and consider the valuation, which replaces Z^i by r_i in the formula $Reach(ph)[r, Z^1, \dots, Z^{2h}, s, t]$. By induction on h, we can prove that this valuation satisfies $Reach(ph)[r, Z^1, \dots, Z^{2h}, s, t]$.

($\Leftarrow$) Since $Reach(ph)[r, Z^1, \dots, Z^{2h}, s, t]$ is satisfiable, there exists an assignment to the Z^i's which satisfies it by replacing Z^i with z_i. Consider the trace $tr = \ell_0, \ell_1, \dots, \ell_{2h}$ such that $\ell_0 = \langle v, r \rangle$, $\ell_{2h} = \langle v_h, s \rangle$, and for each $i \in [1, h-1]$, we have $\ell_{2i-1} = \langle v_{i-1}, z_{2i-1} \rangle$ and $\ell_{2i} = \langle v_i, z_{2i} \rangle$. By induction on the length of ph, we can prove that tr is a trace of H, which connects r to s in time t. □

Hence, r reaches s in time t if and only if there exists a path $ph = \langle v_0, \dots, v_h \rangle$ of $\langle \mathcal{V}, \mathcal{E} \rangle$ and has a formula $Reach(ph)[Z^0, \dots, Z^{2h+1}, T]$ as a witness to this fact. So, if we just considered the disjunction of all the formulæ for all the paths of $\langle \mathcal{V}, \mathcal{E} \rangle$, we would characterize reachability. Unfortunately, if $\langle \mathcal{V}, \mathcal{E} \rangle$ has a cycle, then it has an infinite number of paths. Later on, we will show that, under some assumptions, there is a path of maximum length that we need to consider to evaluate the reachability for IDA.

4.2 Reachability Decision Procedure

In this section, we consider the problem of limiting the length of the paths we need to consider in the study of reachability. This will lead us to the definition of a decision procedure for reachability.

We first show that elapsed time we need to spend inside a location is always greater than 0 in IDA automata.

Given two formulæ ι and τ, the time instants at which a point in $\mathrm{Sat}(\iota)$ can reach a point in $\mathrm{Sat}(\tau)$ can be characterized by the formula

$$\mathit{Flow}(v,\iota,\tau)[T] \overset{\mathrm{def}}{=} \exists Z, Z' \, (\iota[Z] \wedge \tau[Z'] \wedge \mathit{Reach}(v)[Z,Z',T])$$

On the one hand, if this set of time instants is empty, i.e., $\mathrm{Sat}(\iota)$ cannot reach $\mathrm{Sat}(\tau)$, then the following formula is true.

$$\mathit{NotFlow}(v,\iota,\tau) \overset{\mathrm{def}}{=} \forall T \geq 0 \, \neg \mathit{Flow}(v,\iota,\tau)[T]$$

On the other hand, if $\mathrm{Sat}(\iota)$ can reach $\mathrm{Sat}(\tau)$, the infimum of the time instants at which $\mathrm{Sat}(\iota)$ reaches $\mathrm{Sat}(\tau)$ is the solution of the formula:

$$\begin{aligned} \mathit{InfFlow}(v,\iota,\tau)[T] \overset{\mathrm{def}}{=} \; & \forall \epsilon > 0 \, \exists T' \; (|T' - T| < \epsilon \wedge \mathit{Flow}(v,\iota,\tau)[T']) \wedge \\ & \forall T' \geq 0 \, (\mathit{Flow}(v,\iota,\tau)[T'] \longrightarrow T \leq T') \end{aligned}$$

We prove now the correctness of the formula $\mathit{InfFlow}(v,\iota,\tau)[T]$. Moreover, we show that if it has a solution this is greater than 0.

Lemma 3. *Let H be a hybrid automaton and let v be a location of H. Moreover, let ι and τ be two formulæ such that there exists $\delta > 0$ such that for each p satisfying ι and q satisfying τ it holds $||p - q|| > \delta$. If $\mathcal{I}(v)$ is closed and bounded, then $\mathit{NotFlow}(v,\iota,\tau)$ does not hold if and only if there exists the infimum, $\bar{t}$, of the set $\{t \mid \mathit{Flow}(v,\iota,\tau)[t]\}$, $\bar{t} > 0$ and $\bar{t}$ is the unique value satisfying $\mathit{InfFlow}(v,\iota,\tau)[T]$.*

Proof. The set $\mathcal{I}(v) \subseteq \mathbb{R}^k$ is closed and bounded by hypothesis, hence $\mathcal{I}(v)$ is compact. Moreover, the function $\mathcal{F}(v)$ is continuous by Definition 1. Thus, since the image of a compact set over a continuous function is compact, the image, $f_v(\mathcal{I}(v))$, of $\mathcal{I}(v)$ over f_v, is compact and then bounded. It follows that there exists a $r > 0$ such that $||f_v(p)|| \leq r$ for each $p \in \mathcal{I}(v)$. Hence, if $q = f_v(p,t)$ and $\tilde{t}$ is the minimum time instant at which p can reach q, then the distance between p and q can be at most the distance which can be covered at maximum speed r in time $\tilde{t}$, i.e., $||p - q|| \leq r\tilde{t}$.

Let ϕ be the formula $\phi[Z,Z',T] \overset{\mathrm{def}}{=} \iota[Z] \wedge \tau[Z'] \wedge Z' = f_v(Z,T)$. By hypothesis, there exists a $\delta > 0$ such that for all p and q satisfying ι and τ respectively

$\|p-q\| > \delta$, it follows that if $\phi[p,q,t]$ holds then $\delta \leq \|p-q\| \leq rt$, but $r > 0$, and then $t \geq \frac{\delta}{r}$. Thus, since $\mathit{Flow}(v,\iota,\tau)[T]$ implies the formula $\exists Z, Z' \ \phi[Z, Z', T]$ by definition, if $\mathit{Flow}(v,\iota,\tau)[t]$ holds then $t \geq \frac{\delta}{r}$.

($\Rightarrow$) If the formula $\mathit{NotFlow}(v,\iota,\tau)$ does not hold, there exists $t \geq 0$ such that $\mathit{Flow}(v,\iota,\tau)[t]$. Thus there exists an infimum $\bar{t}$ of the set of $t \geq 0$ satisfying $\mathit{Flow}(v,\iota,\tau)[t]$. Hence for all $t \geq 0$, if $\mathit{Flow}(v,\iota,\tau)[t]$ holds then $t \geq \bar{t}$. Moreover for all $\epsilon > 0$ there exists a t' is such that $\|t' - \bar{t}\| < \epsilon$ and $\mathit{Flow}(v,\iota,\tau)[t']$ holds by definition of infimum. Thus $\bar{t}$ satisfies $\mathit{InfFlow}(v,\iota,\tau)$. Furthermore, by definition of infimum, $t' \geq \bar{t}$, and then for all $\epsilon > 0$ there exists a t' is such that $t' < \epsilon + \bar{t}$ and $t' \geq \frac{\delta}{r}$. It follows that $\frac{\delta}{r} < \epsilon + \bar{t}$ holds for all $\epsilon > 0$. Hence $\frac{\delta}{r} \leq \bar{t}$ and, since δ is greater than zero by hypothesis, $\bar{t} > 0$.

Now we will prove than exists an unique $\bar{t}$ satisfying $\mathit{InfFlow}(v,\iota,\tau)$. Let $\tilde{t}$ be such that $\mathit{InfFlow}(v,\iota,\tau)[\tilde{t}]$ holds. Thus $\tilde{t}$ is such that for all $\epsilon > 0$ there exists a t' is such that $\|t' - \tilde{t}\| < \epsilon$ and $\mathit{Flow}(v,\iota,\tau)[t']$ and $\tilde{t}$ is smaller or equal than each t such that $\mathit{Flow}(v,\iota,\tau)[t]$. Hence $\tilde{t}$ is an infimum for the set of t satisfying $\mathit{Flow}(v,\iota,\tau)$. But $\bar{t}$ is the infimum for such set and then $\bar{t}$ is the unique t such that $\mathit{InfFlow}(v,\iota,\tau)[t]$.

($\Leftarrow$) Let assume that there exists the infimum, $\bar{t}$, of the set $\{t \mid \mathit{Flow}(v,\iota,\tau)[t]\}$, that $\bar{t} > 0$ and that $\bar{t}$ is the unique value satisfying $\mathit{InfFlow}(v,\iota,\tau)[T]$. Thus, there exists a $\bar{t} > 0$ such that the formula $\mathit{InfFlow}(v,\iota,\tau)[\bar{t}]$ holds. It follows that it does not hold that, for all $t \geq 0$, the formula $\mathit{Flow}(\tau, v, \iota)[t]$ does not hold. Hence $\mathit{NotFlow}(v,\iota,\tau)$ does not hold by definition. □

We are interested in the infimum time we need to spend inside a location v reached through an edge e before we can cross another edge e'. The following formula characterizes the set of time instants at which the reset of $e = \langle v', v\rangle$ can reach the activation of e' subsequent to e.

$$\mathit{EFlow}(e,e')[T] \stackrel{\text{def}}{=} \mathit{Flow}(v, \overline{\mathit{Reset}}(e), \mathit{Act}(e'))[T]$$

In Lemma 4 we will prove that the reset of e cannot reach the activation of e' if and only if the following formula holds.

$$\mathit{NotEFlow}(e,e') \stackrel{\text{def}}{=} \mathit{NotFlow}(v, \overline{\mathit{Reset}}(e), \mathit{Act}(e'))$$

As a consequence we get that when the reset of e can reach the activation of e', the infimum of the instants at which the reset of e reaches the activation of e' is the unique solution of the formula

$$\mathit{InfEFlow}(e,e')[T] \stackrel{\text{def}}{=} \mathit{InfFlow}(v, \overline{\mathit{Reset}}(e), \mathit{Act}(e'))[T]$$

and it is greater than 0. We prove that the above formulæ are correct.

Lemma 4. *Let H be a IDA. If both $e = \langle v', v\rangle$ and $e' = \langle v, v''\rangle$ are edge of H, then $\mathcal{A}(e')$ is not reachable from $\mathcal{R}(e)$ with a flow in v if and only if the formula $\mathit{NotEFlow}(e,e)$ holds.*

Proof. The set $\mathcal{A}(e')$ is not reachable from $\mathcal{R}(e)$ with a flow in v if and only if $\langle v, r\rangle \rightarrow_C \langle v, s\rangle$ does not hold for all $r \in \mathcal{R}(e)$ and $s \in \mathcal{A}(e')$ and then if and only if, for all $r \in \mathcal{R}(e)$ and $s \in \mathcal{A}(e')$, there is no $t \geq 0$ such that $s = f_v(r, t)$, and for each $t' \in [0, t]$ the formula $Inv(v)[f_v(r, t')]$ is true. Thus $\mathcal{A}(e')$ is not reachable from $\mathcal{R}(e)$ with a flow in v if and only if for all $t \geq 0$ it does not hold that there exists $r \in \mathcal{R}(e)$ and $s \in \mathcal{A}(e')$ such that $s = f_v(r, t)$, and for each $t' \in [0, t]$ the formula $Inv(v)[f_v(r, t')]$ is true. Moreover, $s \in \mathcal{A}(e')$ if and only if $Act(e')[s]$ by definition and $r \in \mathcal{R}(e)$ if and only if $\overline{Reset}(e)[r]$. It follows that $\mathcal{A}(e')$ is not reachable from $\mathcal{R}(e)$ with a flow in v if and only if for all $t \geq 0$ it does hold not that there exist r and s such that $Act(e')[s]$, $\overline{Reset}(e)[r]$, and $s = f_v(r, t)$ holds and for each $t' \in [0, t]$ the formula $Inv(v)[f_v(r, t')]$ is true. Thus $\mathcal{A}(e')$ is not reachable from $\mathcal{R}(e)$ with a flow in v if and only if the formula $Flow(v, \overline{Reset}(e), Act(e'))[T]$ does not hold for any $t \geq 0$ and thus, by definition, if and only if $NotEFlow(e, e')$ holds. □

At this point, we have to characterize the supremum of the time instants at which, starting from a set of points we can reach another set of points. As already said, we will not consider the case in which there are no independent variables.

When some of the variables are not reset, their dynamics are preserved along a path. Hence we can try to compute the supremum on the projection of all the sets of interest (i.e., initial set, target set and invariants) on the independent variables. If this supremum is a real number, then, because of the Lemma 3, the length of the traces we have to consider to evaluate reachability is bounded.

The time instants at which a point in the initial set $\mathrm{Sat}(\iota)$ can reach a point in the target set $\mathrm{Sat}(\tau)$, considering only the dynamics of the independent variables, are characterized by the formula:

$$\begin{aligned} IFlow(\iota, \tau)[T] \overset{\text{def}}{=} \; & \exists X, Y, X', Y' \; ((\iota[X, Y] \wedge \tau[X', Y'] \wedge X = fi(X, T)) \wedge \\ & \forall 0 \leq T' \leq T \; \exists X'', Y'' \; (X'' = fi(X, T') \wedge \vee_{v \in \mathcal{V}} Inv(v)[X'', Y''])) \end{aligned}$$

The *I* in *IFlow* is to stress the fact that we consider only the independent variables flows. When the above formula is not satisfiable the following one is true.

$$NotIFlow(\iota, \tau) \overset{\text{def}}{=} \forall T \geq 0 \; \neg IFlow(\iota, \tau)[T]$$

Otherwise, the supremum of the time instants which satisfy $IFlow(\iota, \tau)[T]$ is defined by:

$$\begin{aligned} SupIFlow(\iota, \tau)[T] \overset{\text{def}}{=} \; & \forall \epsilon > 0 \; \exists T' \; (|T' - T| < \epsilon \wedge IFlow(\iota, \tau)[T']) \wedge \\ & \forall T' \geq 0 \, (IFlow(\iota, \tau)[T'] \longrightarrow T \geq T') \end{aligned}$$

We prove the correctness of our formulæ. In particular, we prove that if $SupIFlow(\iota, \tau)[T]$ has a solution, this solution is an upper bound for time reachability.

Lemma 5. *Let H be a IDA and let ι and τ be two formulæ. Either the formula $NotIFlow(\iota, \tau)$ holds or $\bar{t} \in \mathbb{R}$ satisfies $SupIFlow(\iota, \tau)[T]$ if and only if $\bar{t}$ is the supremum of the set $\{t \mid IFlow(\iota, \tau)[t]\}$.*

Proof. If the formula $\mathit{NotIFlow}(\iota,\tau)$ does not hold, there exists $t \geq 0$ such that $\mathit{IFlow}(\iota,\tau)[t]$. Thus the set of $t \geq 0$ satisfying $\mathit{IFlow}(\iota,\tau)[t]$ is not empty. Let assume that the supremum $\bar{t}$ of such set exists. Hence for all $t \geq 0$, if $\mathit{IFlow}(\iota,\tau)[t]$ holds then $t \geq \bar{t}$. Moreover for all $\epsilon > 0$ there exists a t' is such that $\|t' - \bar{t}\| < \epsilon$ and $\mathit{IFlow}(\iota,\tau)[t']$ holds by definition of superior. Thus $\bar{t}$ satisfies $\mathit{SupIFlow}(\iota,\tau)$.

Now we will prove than if $\mathit{SupIFlow}(\iota,\tau)[\bar{t}]$ holds, then $\bar{t}$ is the supremum of the set $\{t \mid \mathit{IFlow}(\iota,\tau)[t]\}$. Let $\bar{t}$ be such that $\mathit{SupIFlow}(\iota,\tau)[\bar{t}]$ holds. Thus $\bar{t}$ is such that for all $\epsilon > 0$ there exists a t' is such that $\|t' - \bar{t}\| < \epsilon$ and $\mathit{IFlow}(\iota,\tau)[t']$ and $\bar{t}$ is greater or equal than each t such that $\mathit{IFlow}(\iota,\tau)[t]$. Hence $\bar{t}$ is the supremum for the set of t satisfying $\mathit{IFlow}(\iota,\tau)$. □

To conclude we need to introduce a formula which expresses the fact that either H cannot reach $\mathrm{Sat}(\tau)$ from $\mathrm{Sat}(\iota)$ or it can reached it with less than $j+1$ discrete transitions.

In this formula we use Sb to denote set of pairs of edges $\langle e, e'\rangle$ of $\mathcal{E}$ such that e' is subsequent to e.

$$\theta(j,\iota,\tau) \overset{\text{def}}{=} \mathit{NotIFlow}(\iota,\tau) \vee \left(\bigwedge_{\langle e,e'\rangle \in Sb} \mathit{NotEFlow}(e,e') \right) \vee$$
$$\exists T \left(\mathit{SupIFlow}(\iota,\tau)[T] \wedge \left(\bigwedge_{\langle e,e'\rangle \in Sb} \forall T' \left(\mathit{InfEFlow}(e,e')[T'] \rightarrow T' j > T\right) \right)\right)$$

We prove the correctness of $\theta(j,\iota,\tau)$.

Theorem 1. *Let H be a IDA. Moreover, let ι and τ be two formulæ. If the formula $\exists T \geq 0\ \mathit{SupIFlow}(\iota,\tau)[T]$ holds, then there exists a $j \in \mathbb{N}$ such that $\theta(j,\iota,\tau)$ holds. Furthermore, if $\theta(j,\iota,\tau)$ holds and there exist $p \in \mathrm{Sat}(\iota)$ and $q \in \mathrm{Sat}(\tau)$ and q is reachable from p in H, then q is reachable from p in H with less than $j+1$ discrete transitions.*

Proof. By definition, $\theta(j,\iota,\tau)$ holds if and only if one of the followings hold:

- $\mathit{NotIFlow}(\iota,\tau)$
- $\left(\bigwedge_{\langle e,e'\rangle \in Sb} \mathit{NotEFlow}(e,e')\right)$
- $\exists T \left(\mathit{SupIFlow}(\iota,\tau)[T] \wedge \left(\bigwedge_{\langle e,e'\rangle \in Sb} \forall T' \left(\mathit{InfEFlow}(e,e')[T'] \rightarrow T' j > T\right)\right)\right)$

If either $\mathit{NotIFlow}(\iota,\tau)$ holds or $\left(\bigwedge_{e,e' \in \mathcal{E}'} \mathit{NotEFlow}(e,e')\right)$ holds then, $\theta(0,\iota,\tau)$ holds. Let assume that both $\mathit{NotIFlow}(\iota,\tau)$ and $\left(\bigwedge_{e,e' \in \mathcal{E}'} \mathit{NotEFlow}(e,e')\right)$ do not hold. If the formula $\left(\bigwedge_{e,e' \in \mathcal{E}'} \mathit{NotEFlow}(e,e')\right)$ does not hold, then there exist pairs of edges in $\mathcal{E}'$, $e = \langle v', v\rangle$ and $e' = \langle v, v''\rangle$, such that $\mathit{NotEFlow}(e,e')$ does not hold. Hence, by Lemma 3, for each of these pair, e and e', there exists a unique $t \in \mathbb{R}$ satisfying $\mathit{InfEFlow}(e,e')[T]$ and such t is greater than zero. Let $\bar{t}$ be the minimum t, over all these pairs, satisfying $\mathit{InfEFlow}(e,e')[T]$. Moreover, by hypothesis, there exists the supremum, $\tilde{t} \in \mathbb{R}$, of the set $\{t \mid \mathit{IFlow}(\iota,\tau)[t]\}$. Thus, since $\mathit{NotIFlow}(\iota,\tau)$ does not hold, $\tilde{t}$ is the unique satisfies $\mathit{SupIFlow}(\iota,\tau)[T]$ by

Lemma 5. Hence $\bar{j} = \lceil \tilde{t}/\bar{t} \rceil + 1$ is a natural number. Furthermore for any pair $\langle e, e' \rangle \in Sb$, if $\mathit{InfEFlow}(e, e')[t]$ then, $t \geq \bar{t}$ and thus:

$$t\bar{j} \geq t \left\lceil \frac{\tilde{t}}{\bar{t}} \right\rceil + 1 \geq t\frac{\tilde{t}}{\bar{t}} + 1 \geq \tilde{t} + 1 > \tilde{t}$$

Hence $\bigwedge_{e,e' \in Sb} \forall T\, (\mathit{InfEFlow}(e, e')[T] \longrightarrow T\bar{j} > \tilde{t})$ holds. It follows that the formula $\theta(\bar{j}, \iota, \tau)$ holds and hence if the set $\{t \mid \mathit{IFlow}(\iota, \tau)[t]\}$ has a supremum, there exists a $j \in \mathbb{N}$ such that $\theta(j, \iota, \tau)$ holds.

Now we will show that if $\theta(j, \iota, \tau)$ holds and there exist $p \in \mathrm{Sat}(\iota)$ and $q \in \mathrm{Sat}(\tau)$ such that q is reachable from p in H then, q is reachable from p in H with less than $j + 1$ resets.

Let assume that $\mathit{NotIFlow}(\iota, \tau)$ holds. Hence $\theta(0, \iota, \tau)$ holds too. By definitions of the formulæ $\mathit{NotIFlow}(\iota, \tau)$ and $\mathit{IFlow}(\iota, \tau)$:

$$\begin{aligned}
\mathit{NotIFlow}(\iota, \tau) &\equiv \forall T \geq 0\ \neg \mathit{IFlow}(\iota, \tau)[T] \\
&\equiv \forall T \geq 0\ \neg \Bigg(\exists X, Y, X', Y' \Bigg(\iota[X, Y] \wedge \tau[X', Y'] \wedge X = fi(X, T) \wedge \\
&\qquad \forall 0 \leq T' \leq T\ \exists X'', Y'' \Bigg(X'' = fi(X, T') \wedge \bigvee_{v \in \mathcal{V}} \mathit{Inv}(v)[X'', Y''] \Bigg)\Bigg)\Bigg) \\
&\equiv \forall T \geq 0\ \forall X, Y, X', Y' \Bigg(\neg\iota[X, Y] \vee \neg\tau[X', Y'] \vee \neg \Bigg(X = fi(X, T) \wedge \\
&\qquad \forall 0 \leq T' \leq T\ \exists X'', Y'' \Bigg(X'' = fi(X, T') \wedge \bigvee_{v \in \mathcal{V}} \mathit{Inv}(v)[X'', Y''] \Bigg)\Bigg)\Bigg)
\end{aligned}$$

Thus, by Lemma 2 and by IDA's definition, it holds that for any path ph in $\langle \mathcal{V}, \mathcal{E} \rangle$:

$$\begin{aligned}
\mathit{NotIFlow}(\iota, \tau) \implies &\forall T \geq 0\ \forall Z, Z'\ \ (\neg\iota[Z] \vee \neg\tau[Z'] \vee \\
&\neg \left(\exists Z^1 \ldots Z^{2|ph|}\ \mathit{Reach}(ph)[Z, Z^1, \ldots, Z^{2|ph|}, Z', T] \right))
\end{aligned}$$

By Lemma 2, it follows that there are no p and q such that $\iota[p]$ and $\tau[q]$ and q is reachable from p in H.

Let us assume that the formula $\left(\bigwedge_{\langle e,e' \rangle \in Sb} \mathit{NotEFlow}(e, e') \right)$ holds. Hence $\theta(0, \iota, \tau)$ holds too. Moreover, by Lemma 4, for all pair of edges $e = \langle v', v \rangle$ and $e' = \langle v, v'' \rangle$ in $\mathcal{E}$, $\mathcal{A}(e')$ is not reachable from $\mathcal{R}(e)$ with a flow in v. Furthermore it may exist an edge $e'' = \langle v, \tilde{v} \rangle \in \mathcal{E}$ and a $p \in \mathbb{R}^k$ such that $\iota[p]$ and p reaches $\mathcal{A}(e'')$. Thus for all p and q such that $\iota[p]$ holds, $\tau[q]$ holds, and q is reachable from p with at most one reset.

Let assume that neither $\mathit{NotIFlow}(\iota, \tau)$ nor $\left(\bigwedge_{\langle e,e' \rangle \in Sb} \mathit{NotEFlow}(e, e') \right)$ hold. As proved above, it follows that there exists $\tilde{t}$ such that $\mathit{SupIFlow}(\iota, \tau)[\tilde{t}]$. Moreover, for each $\langle e, e' \rangle \in Sb$ it holds that either $\mathit{InfEFlow}(e, e')[T]$ has not a solution, since it is not possible to reach the activation of e' from the reset of

e, or $\mathit{InfEFlow}(e,e')[T]$ is satisfied by $\bar{t}(e,e') \in \mathbb{R}$ such that $(\bar{t}(e,e')j > \tilde{t})$ holds. Moreover, by Lemma 5, $\tilde{t} \in \mathbb{R}$ satisfies $\mathit{SupIFlow}(\iota,\tau)[T]$ if and only if $\tilde{t}$ is the supremum of the set $\{t \mid \mathit{IFlow}(\iota,\tau)[t]\}$. Thus, for all $t > \tilde{t}$, the formula $\mathit{IFlow}(\iota,\tau)[t]$ does not hold. Hence, by IDA's definition, by Lemma 2 and by the definition of the formula *IFlow*, if there exist p, q and t such that $\iota[p]$ holds, $\tau[q]$ holds and p reaches q in H in time t, then $t \leq \tilde{t}$. Moreover by Lemma 3, $\bar{t}(e,e') \in \mathbb{R}$ satisfies $\mathit{InfEFlow}(e,e')[T]$ if and only if $\bar{t}(e,e')$ is the inferior of the set $\{t \mid \mathit{Flow}(v,\overline{\mathit{Reset}}(e),\mathit{Act}(e'))\}$, where v is the bridge vertex from e to e'. Thus for all $t < \bar{t}(e,e')$, the formula $\mathit{Flow}(v,\overline{\mathit{Reset}}(e),\mathit{Act}(e'))[t]$ does not hold. Hence, by Flow's definition and by Lemma 2, if there exist p', q' and t' such that $\mathit{Act}(e')[q']$ holds, $\overline{\mathit{Reset}}(e)[p']$ holds and p' reaches q' in v in time t', then $t' \geq \bar{t}(e,e')$. Furthermore $\mathit{Act}(e')[q']$ holds if and only if $q' \in \mathcal{A}(e')$ and $\overline{\mathit{Reset}}(e)[p']$ if and only if $p' \in \mathcal{R}(e)$. Thus for each pair of edges $e = \langle v', v\rangle$ and $e' = \langle v, v''\rangle$ in $\mathcal{E}$, if there exist p', q' and t' such that $p' \in \mathcal{R}(e)$, $q' \in \mathcal{A}(e')$ and p' reaches q' in v in time t', then $t' \geq \bar{t}(e,e')$. Hence if there exist p, q such that $\iota[p]$ holds, $\tau[q]$ holds and p reaches q through a trace tr in H, whose corresponding path is $ph = \langle v_0, \ldots, v_n\rangle$, the automaton stays in each location v_i of tr at least for time $\bar{t}(\langle v_{i-1}, v_i\rangle, \langle v_i, v_{i+1}\rangle)$. Let $\bar{t}$ be the minimum of the infimus $\bar{t}(e,e')$ with e and e' subsequent in $\mathcal{T}$. Thus the number of resets in the trace tr must be less than $\lceil \bar{t}/\tilde{t} \rceil + 1$. But since $\bar{t}$ is one of the infimum in our formula we have that $(\bar{t}j > \tilde{t})$, and hence $j + 1 > \lceil \bar{t}/\tilde{t} \rceil + 1$. It follows that if there exist p, q such that $\iota[p]$ holds, $\tau[q]$ holds and p reaches q through a trace tr in H, the number of resets in the trace tr must be less than $j + 1$. □

Exploiting Theorem 1 and Lemma 2, we can write Algorithm 1 which decides whether there exists p and q such that $\iota[p]$ holds, $\tau[q]$ holds and p reaches q in a IDA H. The algorithm works when the involved supremums are real numbers. The following formula is used to exit the repeat loop:

$$\mathrm{Test}(ph,\iota,\tau) \overset{\mathrm{def}}{=} \exists T \geq 0 \exists Z, Z' \left(\iota[Z] \wedge \tau[Z'] \wedge \overline{\mathit{Reach}}(ph)[Z,Z',T]\right)$$

The correctness of our algorithm follows by the second part of Theorem 1 and by Lemma 2. Termination is ensured by the first part of Theorem 1.

5 Decidable Classes of IDA

In this section, we define some conditions under which the existence of a maximum time of flow is always guaranteed. This automatically gives us the decidability of reachability for the IDA satisfying these assumptions.

We need to guarantee that either the formula $\mathit{NotIFlow}(\iota,\tau)$ is true or the formula $\mathit{SupIFlow}(\iota,\tau)[t]$ has a solution in $\mathbb{R}$. Let us assume that the formula $\mathit{NotIFlow}(\iota,\tau)$ is not true. This means that there exists $t > 0$ such that $\mathit{IFlow}(\iota,\tau)[t]$ is true. Let us assume that there are I independent variables. Unraveling the formula $\mathit{IFlow}(\iota,\tau)[t]$, we get that there exists $p \in \mathbb{R}^I$ satisfying $\exists Y\ \iota[X,Y]$ and

Algorithm 1 Check whenever there exist p and q such that $\iota[p]$ holds, $\tau[q]$ holds and p reaches q in a IDA H

```
Require: An IDA $H$ and two formulæ $\iota$ and $\tau$ such that the formula $\exists T \geq 0\ SupIFlow(\iota,\tau)[T]$ holds.
Ensure: Return TRUE if there exist $p$ and $q$, FALSE otherwise.
  $j \leftarrow 0$
  repeat
    for all $ph$ path of length $j$ do
      if Test($ph$,$\iota$,$\tau$) then
        Return TRUE
      end if
    end for
    $j \longleftarrow j + 1$
  until $\theta(j-1,\iota,\tau)$
  Return FALSE
```

$q \in \mathbb{R}^I$ satisfying $\exists Y\ \tau[X, Y]$ such that $q = fi(p, t)$ is true and for each $t' \leq t$ it holds that

$$\bigvee_{v \in \mathcal{V}} \exists Y Inv(v)[fi(p, t'), Y]$$

is true, i.e., $fi(p, t')$ belongs to the projection on the independent variables of one of the invariants. This simply means that p is in the projection of ι on the independent variables, q is in the projection of τ on the independent variables and p reaches q in time t using the dynamics of the independent variables in v. Under these assumptions, we have to find conditions which ensure that there exists a time $\tilde{t}$, such that for each time $t > \tilde{t}$ the projection of ι on the independent variables cannot reach the projection of τ on the independent variables using the dynamics of the independent variables. Hence, if we find a condition which ensures that there exists a time $\tilde{t}$ such that for each admissible state $\langle v, p\rangle$ after time $\tilde{t}$ we are outside the union of all the invariants of the automaton, then we have that all the supremums are bounded by $\tilde{t}$, hence they are real numbers.

Since in IDA the invariants are bounded, we try to ensure that we exit the invariants imposing that there exists at least one independent variable whose dynamic has an unbounded limit. Moreover, to exploit the standard compactness theorems, we have to require the continuity of f_v with respect to all its components. In fact, the fact that f_v is a solution of a vectorial field ensures only the continuity with respect to the time.

Definition 7 **(∞IDA).** *Let H be a IDA. We say that H is in the class ∞IDA if and only if for each edge $e = \langle v, v'\rangle$ of H it holds that f_v is continuous on $\mathbb{R}^k \times \mathbb{R}^+$ and for each $p \in \mathfrak{I}(v)$ it holds that*

$$\lim_{t \to +\infty} \|fi(p, t)\| = +\infty$$

Notice that only some of the components of p are used in $fi(p, t)$, i.e., only the components of p corresponding to the independent variables.

The dynamics of the independent variables are preserved in all the locations, hence it is sufficient to check the condition of Definition 7 in one location. Moreover, since the invariants in IDA are always bounded we get that in ∞IDA for each state $\langle v, p\rangle$ after a finite amount of time we exit the union of the invariants.

Lemma 6. *Let H be a ∞IDA and $\langle v, p\rangle$ be a state of H. There exists $t_{\langle v,p\rangle} \in \mathbb{R}^+$ such that for each $t' > t_{\langle v,p\rangle}$ it holds that $f_v(p, t')$ is not in the union of the invariants of H.*

Proof. We have that

$$\lim_{t\to+\infty} \|fi(p, t)\| = +\infty$$

Hence,

$$\lim_{t\to+\infty} \|f_v(p, t)\| = +\infty$$

From the fact that all the invariants are bounded we have that the set of points $I(\mathcal{V})$ inside the union of the invariants is bounded. Hence, there exists M such that for each $q \in I(\mathcal{V})$ it holds $\|q\| \leq M$. By definition of limit we have that there exists a time $t_{\langle v,p\rangle}$ such that for each $t' > t_{\langle v,p\rangle}$ it holds $\|f_v(p, t')\| > M$, from which we immediately get the thesis. □

The above lemma is not sufficient to conclude that there exists a time $\tilde{t}$ such that for each state $\langle v, p\rangle$ and for each $t' > \tilde{t}$ it holds that $f_v(p, t')$ is outside all the invariants.

Let M be such that for each q inside the union of the invariants it holds $\|q\| \leq M$. Consider the function $G_v : \mathcal{I}(v) \longrightarrow \mathbb{R}^+$ defined as

$$G_v(p) = \inf\{t \mid \|f_v(p, t')\| > M\}$$

By Lemma 6 and from the fact that $\|f_v(p, 0)\| = \|p\| \leq M$ we have that G_v is correctly defined. We prove that G_v is bounded.

Lemma 7. *There exists $t_v \in \mathbb{R}^+$ such that for each $p \in \mathcal{I}(v)$ it holds $G_v(p) \leq t_v$.*

Proof. Let us assume by contradiction that there the thesis does not hold. This means that the supremum $\sup_{p\in\mathcal{I}(v)} G_v(p)$ of $G_v(p)$ is $+\infty$. Hence for each $h \in \mathbb{N}$ there exists $p_h \in \mathcal{I}(v)$ such that $G_v(p_h) > h$. Consider the sequence $s = (p_h)_{h\in\mathbb{N}}$. Since $\mathcal{I}(v)$ is closed and bounded, $\mathcal{I}(v)$ is compact and s admits a convergent subsequence s'. With an opportune renaming of the indexes we have that s' is of the form $s' = (q_h)_{h\in\mathbb{N}}$ with $G_v(q_h) > h$. Let $\tilde{p} \in \mathcal{I}(v)$ be the limit of s'. From the fact that $\tilde{p}$ is the limit of s' we get that in each neighborhood $U_{\tilde{p}}$ of $\tilde{p}$ there exists j such that for each $m \geq j$ it holds $q_j \in U_{\tilde{p}}$. Let us call this property (*).

From Definition 7 we have that $\lim_{t\to+\infty} \|f_v(\tilde{p}, t)\| = +\infty$. Hence there exists $\bar{t}$ such that, for each $t \geq \bar{t}$ it holds $\|f_v(\tilde{p}, t)\| > M + 1$. Exploiting the continuity of $\|f_v(\tilde{p}, \bar{t})\|$ we obtain that for each ϵ-neighborhood $V(\epsilon)$ of $\|f_v(\tilde{p}, \bar{t})\|$ there exists a δ-neighborhood $W(\delta)$ of $\langle\tilde{p}, \bar{t}\rangle$ such that for each $\langle q, t\rangle \in W(\delta)$ it holds $\|f_v(q, t)\| \geq \|f_v(\tilde{p}, \bar{t})\| - \epsilon$. In particular, with $\epsilon < 1$ we have that for each $\langle q, t\rangle \in W(\delta)$ it holds $\|f_v(q, t)\| \geq \|f_v(\tilde{p}, \bar{t})\| - \epsilon > M$. Let W' be the neighborhood of $\tilde{p}$ obtained projecting $W(\delta)$ on the components of $\tilde{p}$. We have that for each $q \in W'$ it holds $\|f_v(q, \bar{t})\| > M$, i.e., $G(q) < \bar{t}$. This contradicts property (*), hence the thesis is true. □

Since the locations of a hybrid automaton are finite, we can now conclude that the maximum of the t_v's defined in the above lemma is the $\tilde{t}$ which ensures that the all the supremums are limited.

Corollary 1. *Let H be a ∞IDA, ι and τ be two formulæ. Algorithm 1 can be used to decide whether in H* Sat(ι) *can reach* Sat(τ).

Proof. We only have to check that the requirements of Algorithm 1 are satisfied. This is an immediate consequence of Lemma 7. □

Notice that if H is a IDA in which the flows of the independent variables are non constant polynomials, then H is also a ∞IDA.

Example 2. Let us consider the automaton H defined in Example 1. From the above consideration it follows that H is a ∞IDA. Hence, we can decide any reachability problem on it.

6 Example

Next the utility of an IDA model is demonstrated through a simple, yet biologically relevant example: namely, the bacterial chemotaxis [28, 8]. *Escherichia coli* has evolved an extremely effective strategy for responding to a chemical gradient in its environment, by detecting the concentration of ligands through a number of receptors, and then reacting to the input signal for driving its flagella motors to alter its path of motion. *E. coli* responds in one of two ways: either it "runs" – moves in a straight line by moving its flagella counterclockwise (CCW) (typically lasting 1000 ms), or it "tumbles" – randomly change its heading by moving its flagella clockwise (CW) (typically lasting 100 s). The response is mediated through the molecular concentration of CheY in a phosphorylated form (Y_P variable in figure ???), which in turn is determined by the bound ligands at the receptors that appear in several forms (LT variables in figure???). The ratio of $y = Y_P/Y_0$ (phosphorylated concentration of CheY to its concentration in the unphosphorylated form) determines a bias with an associated probability that flagella will exert a CW rotation; note in our example IDAmodel (fig ???) we have ignored this stochastic effect by modeling it deterministically. Thus the most important output variable is the angular velocity w that takes discrete values +1 for CW and -1 for CCW. The more detailed pathway involves other CheB (either with phosphorylation or without, B_p and B_0), CheZ (Z), bound receptors (LT) and unbound receptors (T), wile their continuous evolution is determined by a set of differential algebraic equations derived through kinetic mass action formulation.

This IDA model captures the essence of how an *E. coli* cell performs a biased random walk by transiently decreasing its tumbling frequency to move towards a region with greater ligand concentration. A second feature of this control is its sensitivity to concentration gradients and its observed dynamic range: rather than responding to absolute concentrations, the *E. coli* adapts quickly as it compares its environment during the immediate past to what existed a bit earlier. Further, it does so over a wide range of inpout concentrations.

As we model just the simplest aspect of this behavior of *E. coli* using an IDAautomaton 2, there are several futures that become notable. The concentrations of all signal transduction proteins evolve in a continuous manner, oblivious of the mode switches. But as certain functions of these concentrations (y) trigger various guard conditions the hybrid automaton switches discretely from one mode to the other ("run" or "tumble") through a dependent variable w. Note that it is rather straightforward to represent this system as a pair of communicating automata: one a single-mode hybrid automaton and the other a purely discrete two-state automaton.

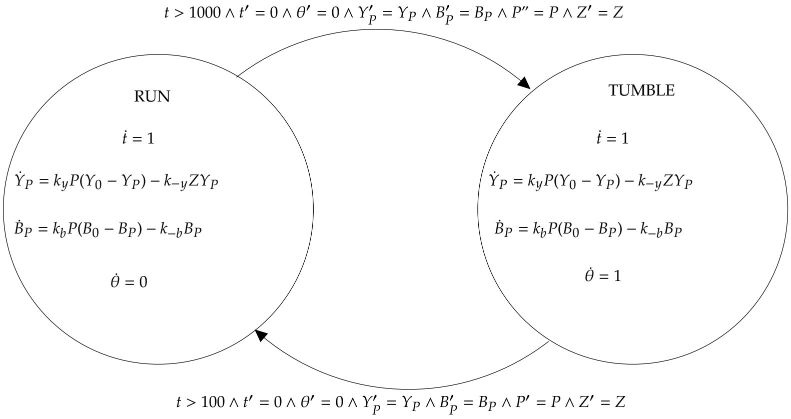
$t > 1000 \wedge t' = 0 \wedge \theta' = 0 \wedge Y_P' = Y_P \wedge B_P' = B_P \wedge P'' = P \wedge Z' = Z$
RUN
$\dot{t} = 1$
$\dot{Y}_P = k_y P(Y_0 - Y_P) - k_{-y} Z Y_P$
$\dot{B}_P = k_b P(B_0 - B_P) - k_{-b} B_P$
$\dot{\theta} = 0$
TUMBLE
$\dot{t} = 1$
$\dot{Y}_P = k_y P(Y_0 - Y_P) - k_{-y} Z Y_P$
$\dot{B}_P = k_b P(B_0 - B_P) - k_{-b} B_P$
$\dot{\theta} = 1$
$t > 100 \wedge t' = 0 \wedge \theta' = 0 \wedge Y_P' = Y_P \wedge B_P' = B_P \wedge P' = P \wedge Z' = Z$

Fig. 2. An IDA capturing the run-tumble mechanism of *E. coli*.

7 Conclusions

In this paper we introduced IDA hybrid automata. IDA are a generalization of decidable O-minimal automata in which non-constant resets are allowed. The resets which are not constant are identities.

We introduced the assumptions under which we can prove that reachability is decidable on IDA automata. In particular, we translated the reachability problem into an infinite set first-order formulæ over a decidable theory. Then, we characterized a finite subset of formulæ and proved that the original reachability problem corresponds to the validity of a formula in this subset.

To conclude we presented a subclass ∞IDA of IDA in which all the assumptions are guaranteed. Hence, on these automata reachability is always decidable. All the IDA automata whose dynamics are non constant polynomials are in the class ∞IDA. We demonstrated a typical Systems Biology application, by presenting a simplistic characterization of the run-tumble chemotaxis mechanism of E. Coli.

The complexity of our algorithm strongly depends on the complexity of the decidable theory used to define the automaton. In the case of formulæ over ($\mathbb{R}$, 0, 1, +, $*$, =, <), decidability has been proved by Tarski [29], and one of the first double-exponential algorithms has been proposed by Collins [11]. Later Hoon Hong, using many useful and practical heuristics, created the first practical quantifier elimination software `Qepcad`. Alternative CAD-based methods have been proposed Grigoriev [14] and Renegar [27], that are doubly exponential in the number of quantifier alternations rather than the number of variables. New quantifier elimination approaches have been proposed by Basu [6]. Another line of investigation is to characterize the simplifications that result for IDAin the approximate algorithms developed for semi-algebraic hybrid automta [23]. More importantly, symbolic algebraic geometry holds many other powerful tools such as Groebner bases and characteristic sets in its arsenal, whose utility is just beginning to be examined.

References

1. Alur, R., Courcoubetis, C., Halbwachs, N., Henzinger, T. A., Ho, P.-H., Nicollin, X., Olivero, A., Sifakis, J., and Yovine, S. The Algorithmic Analysis of Hybrid Systems. *Theoretical Computer Science 138* (1995), 3–34.
2. Alur, R., Courcoubetis, C., Henzinger, T. A., and Ho, P. H. Hybrid Automata: An Algorithmic Approach to the Specification and Verification of Hybrid Systems. In *Hybrid Systems* (1992), R. L. Grossman, A. Nerode, A. P. Ravn, and H. Richel, Eds., LNCS, Springer, pp. 209–229.
3. Alur, R., Dang, T., and Ivancic, F. Progress on Reachability Analysis of Hybrid Systems Using Predicate Abstraction. In *Hybrid Systems: Computation and Control (HSCC'03)* (2003), O. Maler and A. Pnueli, Eds., vol. 2623 of *LNCS*, Springer, pp. 4–19.
4. Anai, H. Algebraic Approach to Analysis of Discrete-Time Polynomial Systems. In *European Control Conference (ECC'99)* (1999).
5. Antoniotti, M., Mishra, B., Piazza, C., Policriti, A., and Simeoni, M. Modeling cellular behavior with hybrid automata: Bisimulation and collapsing. In *CMSB '03: Proceedings of the First International Workshop on Computational Methods in Systems Biology* (London, UK, 2003), Springer-Verlag, pp. 57–74.
6. Basu, S. An Improved Algorithm for Quantifier Elimination Over Real Closed Fields. In *IEEE Symposium on Foundations of Computer Science (FOCS'97)* (1997), pp. 56–65.
7. Becker, B., Behle, M., Eisenbrand, F., Fränzle, M., Herbstritt, M., Herde, C., Hoffmann, J., Kroening, D., Nebel, B., Polian, I., and Wimmer, R. Bounded model checking and inductive verification of hybrid discrete-continuous systems. In *GI/ITG/GMM Workshop* (2004).
8. Berg, H. Motile behavior of bacteria. *Physics Today 53*, 1 (2000), 24–29.
9. Brihaye, T., Michaux, C., Rivière, C., and Troestler, C. On O-Minimal Hybrid Systems. In *Hybrid Systems: Computation and Control (HSCC'04)* (2004), R. Alur and G. J. Pappas, Eds., vol. 2993 of *LNCS*, Springer, pp. 219–233.

10. Casagrande, A., Piazza, C., and Mishra, B. Semi-algebraic constant reset hybrid automata - sacore. In *In Proc. of IEEE Int. Conference on Decision and Control (CDC'05) , IEEE, 2005, To appear.* (2005).
11. Collins, G. E. Quantifier Elimination for the Elementary Theory of Real Closed Fields by Cylindrical Algebraic Decomposition. In *Proceedings of the Second GI Conference on Automata Theory and Formal Languages* (1975), vol. 33 of *LNCS*, Springer, pp. 134–183.
12. Fränzle, M. What Will Be Eventually True of Polynomial Hybrid Automata? In *Theoretical Aspects of Computer Software (TACS'01)* (2001), N. Kobayashi and B. C. Pierce, Eds., vol. 2215 of *LNCS*, Springer, pp. 340–359.
13. Fränzle, M., and Herde, C. Efficient Proof Engines for Bounded Model Checking of Hybrid Systems. In *FMICS* (2004).
14. Grigoriev, D. Complexity of Deciding Tarski Algebra. *Journal of Symbolic Computation 5* (1988), 65–108.
15. Henzinger, T. A., and Kopke, P. W. State Equivalences for Rectangular Hybrid Automata. In *Proc. of Int. Conference on Concurrency Theory (Concur'96)* (1996), U. Montanari and V. Sassone, Eds., vol. 1119 of *LNCS*, Springer, pp. 530–545.
16. Henzinger, T. A., Kopke, P. W., Puri, A., and Varaiya, P. What's decidable about hybrid automata? In *Proc. of ACM Symposium on Theory of Computing (STOCS'95)* (1995), pp. 373–382.
17. Jirstrand, M. Nonlinear Control System Design by Quantifier Elimination. *J. Symb. Comput. 24*, 2 (1997), 137–152.
18. Kopke, P. *The Theory of Rectangular Hybrid Automata*. PhD thesis, Cornell University, 1996.
19. Lafferriere, G., Pappas, G. J., and Sastry, S. O-minimal Hybrid Systems. *Mathematics of Control, Signals, and Systems 13* (2000), 1–21.
20. Lafferriere, G., Pappas, G. J., and Yovine, S. Symbolic Reachability Computation for Families of Linear Vector Fields. *J. Symb. Comput. 32*, 3 (2001), 231–253.
21. Lanotte, R., and S.Tini. Taylor Approximation for Hybrid Systems. In *Hybrid Systems: Computation and Control (HSCC'05)* (2005), M. Morari and L. Thiele, Eds., vol. 3114 of *LNCS*, Springer, pp. 402–416.
22. Martin, F. Analysis of Hybrid Systems: An ounce of realism can save an infinity of states. In *Computer Science Logic (CSL'99)* (1999), J. Flum and M. Rodríguez-Artalejo, Eds., vol. 1683 of *LNCS*, Springer, pp. 126–140.
23. Mysore, V., and Mishra, B. Algorithmic Algebraic Model Checking III: Approximate Methods. In *Infinity – The 7th International Workshop on Verification of Infinite-State Systems* (2005).
24. Mysore, V., Piazza, C., and Mishra, B. Algorithmic Algebraic Model Checking II: Decidability of Semi-Algebraic Model Checking and its Applications to Systems Biology. In *Automated Technology for Verification and Analysis (ATVA)* (2005).
25. Piazza, C., Antoniotti, M., Mysore, V., Policriti, A., Winkler, F., and Mishra, B. Algorithmic Algebraic Model Checking I: The Case of Biochemical Systems and their Reachability Analysis. In *Computer Aided Verification (CAV)* (2005).
26. Ratschan, S., and She, Z. Safety verification of hybrid systems by constraint propagation based abstraction refinement. In *Hybrid Systems: Computation and Control (HSCC'05)* (2005), M. Morari and L. Thiele, Eds., vol. 3114 of *LNCS*, Springer, pp. 573–589.
27. Renegar, J. On the Computational Complexity and Geometry of the First-order Theory of the Reals, parts I-III. *Journal of Symbolic Computation 13* (1992), 255–352.
28. Spiro, P., Parkinson, J., and Othmer, H. A model of excitation and adaptation in bacterial chemotaxis. *Proc Natl Acad Sci, U S A 94*, 14 (Jul 1997), 7263–8.

29. Tarski, A. *A Decision Method for Elementary Algebra and Geometry*. Univ. California Press, 1951.

30. Tiwari, A., and Khanna, G. Series of Abstraction for Hybrid Automata. In *Hybrid Systems: Computation and Control (HSCC'02)* (2002), C. J. Tomlin and M. Greenstreet, Eds., vol. 2289 of *LNCS*, Springer, pp. 465–478.