A New Class of Decidable Hybrid Systems

Gerardo Lafferriere[1], George J. Pappas[2], and Sergio Yovine[3]

[1] Department of Mathematical Sciences
Portland State University, PO Box 751
Portland, OR 97207
gerardo@mth.pdx.edu
[2] Department of Electrical Engineering and Computer Sciences
University of California at Berkeley
Berkeley, CA 94720
gpappas@eecs.berkeley.edu
[3] VERIMAG
Centre Equation, 2 Avenue de Vignate
38610 Gieres, France
Sergio.Yovine@imag.fr

Abstract. One of the most important analysis problems of hybrid systems is the reachability problem. State of the art computational tools perform reachability computation for timed automata, multirate automata, and rectangular automata. In this paper, we extend the decidability frontier for classes of *linear hybrid systems*, which are introduced as hybrid systems with linear vector fields in each discrete location. This result is achieved by showing that any such hybrid system admits a finite bisimulation, and by providing an algorithm that computes it using decision methods from mathematical logic.

1   Introduction

Hybrid systems are roughly discrete event systems with differential equations in each discrete location. A modeling approach to hybrid systems extends finite state machines to incorporate simple dynamics. One of the most important problems for hybrid systems is the *reachability problem* which asks whether some unsafe region is reachable from an initial region. For purely continuous systems, the reachability problem is known to be a very difficult problem with a few exceptions. Optimal control [14] and game theoretic [11] approaches have been used among others to calculate reachable sets for some systems.

The main tool for obtaining classes of hybrid system for which the reachability problem is decidable, is given by the concept of *bisimulation*. Bisimulations are simply reachability preserving quotient systems. If an infinite state hybrid system has a finite state bisimulation, then checking reachability for the hybrid system can be equivalently performed on the finite, discrete, quotient graph. Since the quotient graph is finite, the algorithm will terminate. If in addition, each step of the algorithm can be encoded and implemented by a computer

program, then the problem is decidable. The first successful application of this approach was for the model of timed automata [2], and was then extended to multirate [1] and rectangular [6,9] automata.

Unfortunately, the above decidable classes of hybrid systems have limited modeling power for most control applications, where systems with complicated continuous dynamics are frequently encountered. *In this paper, we extend the decidability frontier to capture classes of hybrid systems with linear dynamics in each discrete location*. This broadens the applicability potential of the approach given the wide use of linear systems in control theory. In addition, this result is achieved by using new mathematical and computational techniques which may bring other benefits to control theorists and practitioners. The notion of *o-minimality* [12] from *model theory* is used to define a class of hybrid systems, called *o-minimal hybrid systems*. In [7], it is shown that all o-minimal hybrid systems admit finite bisimulations. In order to make the bisimulation algorithm computationally feasible, we use the framework of *mathematical logic* as the main tool to symbolically represent and manipulate sets. The main computational tool for symbolic set manipulation in this context is *quantifier elimination*. Since quantifier elimination is possible for the theory of reals with addition and multiplication [10,15], we either find or transform subclasses of o-minimal hybrid systems which are definable in this theory. This immediately leads to new decidability results. In addition, the framework presented in this paper, provides a unifying platform for further studies along this direction.

In Section 2 we review bisimulations of transition systems which are applied in Section 3 to transition systems generated by a class of hybrid systems. After a brief introduction to mathematical logic and model theory in Section 4, Section 5 transforms the reachability problem for various classes of linear vector fields into a quantifier elimination problem in the decidable theory of the reals as an ordered field. This leads to a computational bisimulation algorithm whose termination is guaranteed in Section 6, using the notion of o-minimality. The main decidability result is contained in Theorem 3.

2   Bisimulations And Decidability

A transition system $T = (Q, \Sigma, \rightarrow, Q_O, Q_F)$ consists of a (not necessarily finite) set Q of states, an alphabet Σ of events, a transition relation $\rightarrow \subseteq Q \times \Sigma \times Q$, a set $Q_O \subseteq Q$ of initial states, and a set $Q_F \subseteq Q$ of final states. A transition $(q_1, \sigma, q_2) \in \rightarrow$ is denoted as $q_1 \overset{\sigma}{\rightarrow} q_2$. The transition system is finite if the cardinality of Q and $\rightarrow$ is finite, and it is infinite otherwise. A region is a subset $P \subseteq Q$. Given $\sigma \in \Sigma$ we define the predecessor $Pre_\sigma(P)$ of a region P as

$$Pre_\sigma(P) = \{q \in Q \mid \exists p \in P : q \overset{\sigma}{\rightarrow} p\} \tag{1}$$

One of the main problems for transition systems is the reachability problem which can be used to formulate many safety verification problems.

Problem 1 (Reachability Problem). Given a transition system T, is a state $q_f \in Q_F$ reachable from a state $q_0 \in Q_O$ by a sequence of transitions?

The complexity of the reachability problem is reduced using special quotient transition systems. Given an equivalence relation $\sim \subseteq Q \times Q$ on the state space one can define a quotient transition system as follows. Let $Q/\sim$ denote the quotient space. For a region P we denote by $P/\sim$ the collection of all equivalence classes which intersect P. The transition relation $\rightarrow_\sim$ on the quotient space is defined as follows: for $Q_1, Q_2 \in Q/\sim$, $Q_1 \xrightarrow{\sigma}_\sim Q_2$ iff there exist $q_1 \in Q_1$ and $q_2 \in Q_2$ such that $q_1 \xrightarrow{\sigma} q_2$. The quotient transition system is then $T/\sim = (Q/\sim, \Sigma, \rightarrow_\sim, Q_O/\sim, Q_F/\sim)$.

Given an equivalence relation $\sim$ on Q, we call a set a $\sim$-block if it is a union of equivalence classes. The equivalence relation $\sim$ is a *bisimulation* of T iff Q_O, Q_F are $\sim$-blocks and for all $\sigma \in \Sigma$ and all $\sim$-blocks P, the region $Pre_\sigma(P)$ is a $\sim$-block. In this case the systems T and $T/\sim$ are called *bisimilar*. We will also say that a partition is a bisimulation when its induced equivalence relation is a bisimulation. A bisimulation is called finite if it has a finite number of equivalence classes. Bisimilar transition systems generate the same language [5].

Recently, the above bisimulation methodology has been applied to hybrid systems which combine discrete and continuous dynamics. The main reason for doing this is that if T is a transition system with an infinite state space and $T/\sim$ is a finite bisimulation, then the reachability problem for hybrid systems can be converted to an equivalent reachability problem on a finite graph. If, in addition, this can be performed in a computationally feasible way, then one obtains classes of hybrid systems for which the reachability problem is *decidable*. This approach has successfully resulted in various decidable classes of hybrid systems, including timed automata [2], multirate automata [1], and initialized rectangular automata [6,9].

These results are based on the following geometric characterization of bisimulations. If $\sim$ is a bisimulation, it can be easily shown that if $p \sim q$ then

B1 $p \in Q_F$ iff $q \in Q_F$, and $p \in Q_O$ iff $q \in Q_O$
B2 if $p \xrightarrow{\sigma} p'$ then there exists q' such that $q \xrightarrow{\sigma} q'$ and $p' \sim q'$

Based on the above characterization, given a transition system T, the following algorithm computes increasingly finer partitions of the state space Q. If the algorithm terminates, then the resulting quotient transition system is a finite bisimulation. The state space $Q/\sim$ is called a bisimilarity quotient.

Algorithm 1 (Bisimulation Algorithm for Transition Systems)
Set $Q/\sim = \{Q_O, Q_F, Q \setminus (Q_O \cup Q_F)\}$
while $\exists P, P' \in Q/\sim$ and $\sigma \in \Sigma$ such that $\emptyset \neq P \cap Pre_\sigma(P') \neq P$
    set $P_1 = P \cap Pre_\sigma(P')$, $P_2 = P \setminus Pre_\sigma(P')$
    refine $Q/\sim = (Q/\sim \setminus \{P\}) \cup \{P_1, P_2\}$
end while

In order for a transition system to have a finite bisimulation, the above algorithm must terminate after a finite number of iterations. If, in addition, each step of the algorithm is constructive, then the reachability problem for the transition system is *decidable*. This requires that we have computational methods to represent sets, perform set intersections and complements, check whether a set is empty, and compute $Pre_\sigma(P)$ for any set P and any $\sigma \in \Sigma$.

3   Hybrid Systems

In this paper, we focus on transition systems generated by the following class of hybrid systems.

Definition 1. *A* hybrid system *is a tuple* $H = (X, X_O, X_F, F, E, I, G, R)$ *where*

- $X = X_D \times X_C$ *is the state space with* $X_D = \{q_1, \dots, q_m\}$ *and* X_C *a manifold.*
- $X_O \subseteq X$ *is the set of initial states.*
- $X_F \subseteq X$ *is the set of final states.*
- $F : X \longrightarrow TX_C$ *assigns to each discrete location* $q \in X_D$ *a vector field* $F(q, \cdot)$.
- $E \subseteq X_D \times X_D$ *is the set of discrete transitions.*
- $I : X_D \longrightarrow 2^{X_C}$ *assigns to* $q \in X_D$ *an invariant of the form* $I(q) \subseteq X_C$.
- $G : E \longrightarrow X_D \times 2^{X_C}$ *assigns to* $e = (q_1, q_2) \in E$ *a guard of the form* $\{q_1\} \times U$, $U \subseteq I(q_1)$.
- $R : E \longrightarrow X_D \times 2^{X_C}$ *assigns to* $e = (q_1, q_2) \in E$ *a reset of the form* $\{q_2\} \times V$, $V \subseteq I(q_2)$.

Trajectories of the hybrid system H originate at any $(q, x) \in X_O$ and consist of concatenations of continuous evolutions and discrete jumps. Continuous trajectories keep the discrete part of the state constant, and the continuous part evolves according to the continuous flow $F(q, \cdot)$ as long as the flow remains inside the invariant set $I(q)$. If the flow exits $I(q)$, then a discrete transition is *forced*. If, during the continuous evolution, a state $(q, x) \in G(e)$ is reached for some $e \in E$, then discrete transition e is *enabled*. The hybrid system may then instantaneously jump from (q, x) to any $(q', x') \in R(e)$ and the system then evolves according to the flow $F(q', \cdot)$. Notice that even though the continuous evolution is deterministic, the discrete evolution may be nondeterministic. The discrete transitions allowed in our model are slightly more restrictive than those in initialized rectangular automata.

Every hybrid system H generates a transition system $T = (Q, \Sigma, \rightarrow, Q_O, Q_F)$ by setting $Q = X$, $Q_O = X_O$, $Q_F = X_F$, $\Sigma = E \cup \{\tau\}$, and $\rightarrow = (\cup_{e \in E} \xrightarrow{e}) \cup \xrightarrow{\tau}$ where

Discrete Transitions $(q, x) \xrightarrow{e} (q', x')$ for $e \in E$ iff $(q, x) \in G(e)$ and $(q', x') \in R(e)$

Continuous Transitions $(q_1, x_1) \xrightarrow{\tau} (q_2, x_2)$ iff $q_1 = q_2$ and there exist $\delta \geq 0$ and a curve $x : [0, \delta] \longrightarrow X_C$ with $x(0) = x_1$, $x(\delta) = x_2$ and for all $t \in [0, \delta]$ it satisfies $\frac{dx}{dt} = F(q_1, x(t))$ and $x(t) \in I(q_1)$.

The continuous τ transitions are time-abstract transitions, in the sense that the time it takes to reach one state from another is ignored. Having defined the continuous and discrete transitions $\xrightarrow{\tau}$ and $\xrightarrow{e}$ allows us to formally define $Pre_\tau(P)$ and $Pre_e(P)$ for $e \in E$ and any region $P \subseteq X$ using (1). Furthermore, the structure of the discrete transitions allowed in our hybrid system model results in

$$Pre_e(P) = \begin{cases} \emptyset & \text{if } P \cap R(e) = \emptyset \\ G(e) & \text{if } P \cap R(e) \neq \emptyset \end{cases} \tag{2}$$

for all discrete transitions $e \in E$ and regions P. Therefore, if the sets $R(e)$ and $G(e)$ are blocks of any partition of the state space, then no partition refinement is necessary in the bisimulation algorithm due to any discrete transitions $e \in E$. This fact, in a sense, decouples the continuous and discrete components of the hybrid system. In turn, this implies that the initial partition in the bisimulation algorithm should contain the invariants, guards and reset sets, in addition to the initial and final sets. This allows us to carry out the algorithm independently for each location.

More precisely, define for any region $P \subseteq X$ and $q \in X_D$ the set $P_q = \{x \in X_C \mid (q, x) \in P\}$. For each location $q \in X_D$ consider the finite collection of sets

$$\mathcal{A}_q = \{I(q), (X_O)_q, (X_F)_q\} \cup \{G(e)_q, R(e)_q \mid e \in E\} \tag{3}$$

which describes the initial and final states, guards, invariants and resets associated with location q. Let $\mathcal{S}_q$ be the coarsest partition of X_C compatible with the collection $\mathcal{A}_q$ (by compatible we mean that each set in $\mathcal{A}_q$ is a union of sets in $\mathcal{S}_q$). The (finite) partition $\mathcal{S}_q$ can be easily computed by successively finding the intersections between each of the sets in $\mathcal{A}_q$ and their complements. We define $(q, \mathcal{S}_q)$ to be the set $\{\{q\} \times P \mid P \in \mathcal{S}_q\}$. These collections $(q, \mathcal{S}_q)$ will be the starting partitions of the bisimulation algorithm. In addition, since by definition $Pre_\tau(P)$ applies to regions $P \subseteq X$, but not to its continuous projection P_q, we define for $Y \subseteq X_C$ the operator $Pre_q(Y) = (Pre_\tau(\{q\} \times Y))_q$. The general bisimulation algorithm for transition systems then takes the following form for the class of hybrid systems that are considered in this paper.

Algorithm 2 (Bisimulation Algorithm for Hybrid Systems)
Set $X/\sim = \bigcup_q (q, \mathcal{S}_q)$
for $q \in X_D$
    while $\exists\ P, P' \in \mathcal{S}_q$ such that $\emptyset \neq P \cap Pre_q(P') \neq P$
        Set $P_1 = P \cap Pre_q(P')$; $P_2 = P \setminus Pre_q(P')$
        refine $\mathcal{S}_q = (\mathcal{S}_q \setminus \{P\}) \cup \{P_1, P_2\}$
    end while
end for

It is clear from the structure of the bisimulation algorithm that, the iteration is carried out independently for each discrete location. In order for the above algorithm to terminate, the partition refinement process must terminate for each discrete location $q \in X_D$. It therefore suffices to look at one continuous slice of the hybrid system at a time and see whether we can construct a finite bisimulation that is consistent with all relevant sets of each location q as well as with the continuous flows of the vector field $F(q, \cdot)$. Since we focus on each continuous slice at a time, we will drop the q subscript from $Pre_q(Y)$, which will be denoted from now on by $Pre(Y)$.

It is now clear that the decidability of the reachability problem amounts to solving the following two problems.

Problem 1 (Computability) In order for the bisimulation algorithm to be *computational*, we need to effectively

1. Represent sets,
2. Perform set intersection and complement,
3. Check emptiness of sets,
4. Compute $Pre(Y)$ of a set Y.

Problem 2 (Finiteness) Determine whether the bisimulation algorithm terminates in a finite number of steps.

A natural platform for solving the above computational issues is provided by mathematical logic where sets would be represented as formulas of first-order logic over the real numbers. In the next section we introduce the necessary notions of mathematical logic and model theory that will provide the means for representing and manipulating sets defined by first-order formulas (Section 5) as well as for ensuring the termination of the algorithm (Section 6).

4 Mathematical Logic and Model Theory

4.1 Languages and formulas

A *language* is a set of symbols separated in three groups: relations, functions and constants. The sets $\mathcal{P} = \{<, +, -, 0, 1\}$, $\mathcal{R} = \{<, +, -, \cdot, 0, 1\}$, and $\mathcal{R}_{\exp} = \{<, +, -, \cdot, 0, 1, \exp\}$ are examples of languages where $<$ (less than) is the relation, $+$ (plus), $-$ (minus), $\cdot$ (product) and $\exp$ (exponentiation) are the functions, and 0 (zero) and 1 (one) are the constants.

Let $\mathcal{V} = \{x, y, z, x_0, x_1, \ldots\}$ be a countable set of *variables*. The set of *terms* of a language is inductively defined as follows. A term θ is a variable, a constant, or $F(\theta_1, \ldots, \theta_m)$, where F is a m-ary function and θ_i, $i = 1, \ldots, m$ are terms. For instance, $x - 2y + 3$ and $x + yz^2 - 1$ are terms of $\mathcal{P}$ and $\mathcal{R}$, respectively. In other words, terms of $\mathcal{P}$ are linear expressions and terms of $\mathcal{R}$ are polynomials with integer coefficients. Notice that integers are the only numbers allowed in expressions (they can be obtained by adding up the constant 1).

The *atomic formulas* of a language are of the form $\theta_1 = \theta_2$, or $R(\theta_1, \ldots, \theta_n)$, where θ_i, $i = 1, \ldots, n$ are terms and R is an n-ary relation. For example, $xy > 0$ and $x^2 + 1 = 0$ are terms of $\mathcal{R}$.

The set of *(first-order) formulas* is recursively defined as follows. A formula ϕ is an atomic formula, $\phi_1 \wedge \phi_2$, $\neg\phi_1$, $\forall x : \phi_1$ or $\exists x : \phi_1$, where ϕ_1 and ϕ_2 are formulas, x is a variable, $\wedge$ (conjunction) and $\neg$ (negation) are the boolean connectives, and $\forall$ (for all) and $\exists$ (there exists) are the quantifiers.

Examples of $\mathcal{R}$-formulas are $\forall x \; \forall y : xy > 0$, $\exists x : x^2 - 2 = 0$, and $\exists w : xw^2 + yw + z = 0$. The occurrence of a variable in a formula is *free* if it is not inside the scope of a quantifier; otherwise, it is *bound*. For example, x, y, and z are free and w is bound in the last example. We often write $\phi(x_1, \ldots, x_n)$ to indicate that $x_1, \ldots, x_n$ are the free variables of the formula ϕ. A *sentence* of $\mathcal{R}$ is a formula with no free variables. The first two examples are sentences.

4.2   Models

A *model* of a language consists of a non-empty set S and an interpretation of the relations, functions and constants. For example, $(\mathbb{R}, <, +, -, \cdot, 0, 1)$ and $(\mathbb{Q}, <, +, -, \cdot, 0, 1)$, are *models* of $\mathcal{R}$ with the usual meaning of the symbols.

We say that a set $Y \subseteq S^n$ is *definable* in a language if there exists a formula $\phi(x_1, \ldots, x_n)$ such that $Y = \{(a_1, \ldots, a_n) \in S^n \mid \phi(a_1, \ldots, a_n)\}$. For example, over $\mathbb{R}$, the formula $x^2 - 2 = 0$ defines the set $\{\sqrt{2}, -\sqrt{2}\}$. Two formulas $\phi(x_1, \ldots, x_n)$ and $\psi(x_1, \ldots, x_n)$ are *equivalent* in a model, denoted by $\phi \equiv \psi$, if for every assignment $(a_1, \ldots, a_n)$ of $(x_1, \ldots, x_n)$, $\phi(a_1, \ldots, a_n)$ is true if and only if $\psi(a_1, \ldots, a_n)$ is true. Equivalent formulas define the same set.

4.3   Theories

A *theory* is a subset of sentences. Any model of a language defines a theory: *the set of all sentences which hold in the model.* We denote by $\mathsf{Lin}(\mathbb{R})$ the theory defined as the formulas of $\mathcal{P}$ that are true over $(\mathbb{R}, <, +, -, 0, 1)$, i.e., $\mathsf{Lin}(\mathbb{R})$ is the theory of linear constraints (polyhedra). We denote by $\mathsf{OF}(\mathbb{R})$ the theory obtained by interpreting $\mathcal{R}$ over $(\mathbb{R}, <, +, -, \cdot, 0, 1)$. In other words, $\mathsf{OF}(\mathbb{R})$ is the set of all true assertions about the set of real numbers when viewed as an *ordered field.* The theory $\mathsf{OF}_{\exp}(\mathbb{R})$ is the extension of the ordered field of real numbers with the exponentiation.

4.4   Decidability and quantifier elimination

Given a theory, it is important to determine the sentences of the language that belong to the theory. Tarski [10] showed that $\mathsf{OF}(\mathbb{R})$ is *decidable*, i.e., there is a computational procedure that, given any $\mathcal{R}$-sentence ϕ, decides whether ϕ belongs to $\mathsf{OF}(\mathbb{R})$. The decision procedure is based on the elimination of the quantifiers. Over $\mathbb{R}$, every formula $\phi(x_1, \ldots, x_n)$ of $\mathcal{R}$ is equivalent to a formula $\psi(x_1, \ldots, x_n)$ without quantifiers. Moreover, there is an algorithm that transforms ϕ into ψ by *eliminating the quantifiers.* For example, the formula $\exists w : xw^2 + yw + z = 0$ is equivalent to $4xz - y^2 \leq 0$.

Quantifier elimination implies that every $\mathcal{R}$-definable set $Y \subseteq \mathbb{R}^n$ is definable without quantifiers. Moreover, the decidability of $\mathsf{OF}(\mathbb{R})$ implies that the algorithm for eliminating the quantifiers also provides a computational procedure (that terminates in a finite number of steps) for checking whether Y is empty: $Y = \{(y_1, \ldots, y_n) \in \mathbb{R}^n \mid \phi(y_1, \ldots, y_n)\} = \emptyset$ if and only if the sentence $\exists y_1 \ldots \exists y_n : \phi(y_1, \ldots, y_n)$ is equivalent to the (quantifier-free) formula *false.* There are different methods to perform quantifier elimination, e.g., [3,15]. All the examples considered in this paper have been solved using the tool REDLOG [4].

Therefore, the theory $\mathsf{OF}(\mathbb{R})$ provides the means for representing sets as well as performing boolean operations and checking for emptiness. All that remains in order to make the bisimulation algorithm computational, is to compute

$Pre(Y)$ for any definable set Y. Computing $Pre(Y)$ for a linear vector field generally includes formulas involving the exponential function which are definable in $\mathsf{OF}_{\exp}(\mathbb{R})$. This theory does not admit elimination of quantifiers. Nevertheless, in the next section, we identify several subsets of $\mathcal{R}_{\exp}$ where quantifiers can be eliminated and the resulting quantifier-free formula is in $\mathcal{R}$, yielding a decision procedure.

5   Linear Hybrid Systems

For subclasses of hybrid systems, like multirate automata and rectangular automata [1], where the subsets of $\mathbb{R}^n$ obtained by the application of the bisimulation algorithm are polyhedral sets, i.e., sets definable in the language of linear constraints $\mathcal{P}$, the computation of $Pre(Y)$ relies on the decidability of the theory $\mathsf{Lin}(\mathbb{R})$ via the elimination of the quantifiers.

In this section we identify several classes of hybrid systems with linear vector fields where the ability of computing $Pre(Y)$ depends on the decidability of $\mathsf{OF}(\mathbb{R})$.

Definition 2 (Linear Hybrid Systems). *A hybrid system $H = (X, X_O, X_F, F, E, I, G, R)$ is a linear hybrid system if*

- $X_C = \mathbb{R}^n$.
- *for each $q \in X_D$ the vector field $F(q, x) = A_q x$, where $A_q \in \mathbb{Q}^{n \times n}$.*
- *for each $q \in X_D$ the family of sets $\mathcal{A}_q = \{I(q), (X_O)_q, (X_F)_q\} \cup \{G(e)_q, R(e)_q \mid e \in E\}$ is definable in $\mathsf{OF}(\mathbb{R})$.*

As indicated previously, having a computational bisimulation algorithm requires having a procedure for computing $Pre(Y)$ for a definable set Y for each discrete location q. Therefore, we only need to investigate a single location and a single linear vector field $F(x) = Ax$ where the subscript q is dropped for notational convenience. In addition, since the invariant $I(q)$ is a definable set, there exists an $\mathcal{R}$-formula $I(x)$ such that $I(q) = \{x \in \mathbb{R}^n \mid I(x)\}$.

Now let $Y \triangleq \{y \in \mathbb{R}^n \mid P(y)\}$. Then we can write explicitly

$$Pre(Y) = \{x \in \mathbb{R}^n \mid \exists y \, \exists t : P(y) \wedge t \geq 0 \wedge x = e^{-tA}y \wedge$$
$$\forall t' : 0 \leq t' \leq t \implies I(e^{-t'A}y)\}$$

In order to simplify the following presentation, we will assume that $I(x)$ is *true*. In this case, the above definition reduces to

$$Pre(Y) = \{x \in \mathbb{R}^n \mid \exists y \, \exists t : P(y) \wedge t \geq 0 \wedge x = e^{-tA}y\} = \{x \in \mathbb{R}^n \mid \eta(x)\} \, (4)$$

It will be clear from the following results that more complicated invariant sets can be dealt with by the same techniques. Our goal in this section is to transform formula $\eta(x)$ to an equivalent formula in $\mathsf{OF}(\mathbb{R})$, which is indeed decidable. Based on the eigenstructure of A, we identify several classes of linear vector fields for which this transformation is feasible.

5.1   Nilpotent matrices

We consider first the special case when the vector field is linear with a nilpotent matrix A, that is, $A^n = 0$. Recall that nilpotent matrices can only have zero as an eigenvalue. Another important property of nilpotent matrices is that we can express e^{-tA} explicitly as a finite sum

$$e^{-tA} = \sum_{k=0}^{n-1} (-1)^k \frac{t^k}{k!} A^k \tag{5}$$

Thus, the formula $\eta(x)$ can be rewritten as:

$$\eta(x) \triangleq \exists y \; \exists t : P(y) \wedge t \geq 0 \wedge x = \sum_{k=0}^{n-1} (-1)^k \frac{t^k}{k!} A^k y$$

$$\triangleq \exists y : P(y) \wedge \mu(x, y)$$

Clearly, $\mu(x, y)$ is an $\mathcal{R}$-formula, and so is $\eta(x)$, which implies that the following proposition holds.

Proposition 1. *Let $F(x) = Ax$ be a linear vector field and $A \in \mathbb{Q}^{n \times n}$ a nilpotent matrix, and $Y \subseteq \mathbb{R}^n$ definable in $\mathcal{R}$. Then $Pre(Y)$ is definable in $\mathcal{R}$.*

Therefore, based on the computational procedure for eliminating quantifiers in $\mathsf{OF}(\mathbb{R})$, we can compute $Pre(Y)$ for linear vector fields with nilpotent matrices. Note that nilpotent linear vector fields capture integrators which are an extremely important class of linear systems.

5.2   Diagonalizable matrices with rational eigenvalues

In this case we can write $A = TDT^{-1}$ where D is a diagonal matrix with the eigenvalues of A along the diagonal and both T and T^{-1} have rational entries. Then $e^{-tA} = [f_{ij}(t)]$ where $f_{ij}(t) = \sum_{k=1}^{n} a_{ijk} e^{-\lambda_k t}$ with $a_{ijk} \in \mathbb{Q}$ for all i, j, k, and $\{\lambda_k\}$ are the eigenvalues of A. Moreover, $x = e^{-tA} y$ can be written component-wise as $x_i = \sum_{k=1}^{n} \psi_{ik}(y) e^{-\lambda_k t}$, with ψ_{ik} being $\mathcal{R}$-formulas. Therefore, $\eta(x)$ can be rewritten as

$$\eta(x) \triangleq \exists y \; \exists t : P(y) \wedge t \geq 0 \wedge \bigwedge_{i=1}^{n} x_i = \sum_{k=1}^{n} \psi_{ik}(y) e^{-\lambda_k t}$$

$$\triangleq \exists y : P(y) \wedge \varphi(x, y)$$

Since the formula for Y, $P(y)$, is already in $\mathcal{R}$, we will concentrate on studying $\varphi(x, y)$. First we reparameterize the time t to reduce the problem to integers in the exponent. More precisely, for each $k = 1, \ldots, n$ let d_k denote the denominator of λ_k and let $d_0 = \prod d_k$. We assume that the λ_k are in reduced form, with positive

denominators. Then $d_0 > 0$ and for each $k = 1, \ldots, n$ we write $r_k = \lambda_k d_0$. Then we have that $\varphi(x, y) \equiv \varphi_{\mathbb{Z}}(x, y)$ where

$$\varphi_{\mathbb{Z}}(x, y) \triangleq \exists s : s \geq 0 \wedge \bigwedge_{i=1}^{n} x_i = \sum_{k=1}^{n} \psi_{ik}(y) \, e^{-r_k s} \tag{6}$$

Still, $\varphi_{\mathbb{Z}}$ is an $\mathcal{R}_{\exp}$-formula. We consider a second formula $\zeta(x, y)$ which does not involve the exponential function:

$$\zeta(x, y) \triangleq \exists z : 0 < z \leq 1 \wedge \bigwedge_{i=1}^{n} x_i = \sum_{k=1}^{n} \psi_{ik}(y) \, z^{r_k} \tag{7}$$

The following lemma holds.

Lemma 1. *Formulas $\varphi_{\mathbb{Z}}(x, y)$ and $\zeta(x, y)$ are equivalent.*

Proof. The equivalence follows from the change of variables $z = e^{-s}$.

The third step eliminates negative polynomial powers. It consists of grouping the indices $1, \ldots, n$ according to the sign of the corresponding eigenvalue. Let $I^+ = \{k \mid r_k > 0\}$, $I^- = \{k \mid r_k < 0\}$, and $I^0 = \{k \mid r_k = 0\}$. Consider now the following formula:

$$\nu(x, y) \triangleq \exists w_1 \, \exists w_2 : \tag{8}$$
$$w_1 > 0 \wedge w_2 > 0 \wedge w_1 w_2 = 1$$
$$\wedge \bigwedge_{i=1}^{n} x_i = \sum_{k \in I^+} \psi_{ik}(y) \, w_1^{r_k} + \sum_{k \in I^-} \psi_{ik}(y) \, w_2^{-r_k} + \sum_{k \in I^0} \psi_{ik}(y)$$

Clearly, $\nu(x, y)$ is an $\mathcal{R}$-formula. The following lemma holds.

Lemma 2. *The formulas $\zeta(x, y)$ and $\nu(x, y)$ are equivalent.*

Proof. The equivalence follows from the change of variables $w_1 = z$, $w_2 = 1/z$.

The combination of the above lemmas gives the following proposition which implies that we have a computational procedure for computing reachable sets for diagonalizable linear vector fields with rational eigenvalues.

Proposition 2. *Let $F(x) = Ax$ be a linear vector field and $A \in \mathbb{Q}^{n \times n}$ a diagonalizable matrix with rational eigenvalues, and $Y \subseteq \mathbb{R}^n$ definable in $\mathcal{R}$. Then $Pre(Y)$ is definable in $\mathcal{R}$.*

Example 1. Consider the linear vector field

$$F(x) = \begin{bmatrix} 2 & 0 \\ 0 & -1 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{9}$$

Let $Y = \{(y_1, y_2) \in \mathbb{R}^2 \mid y_1 = 4 \wedge y_2 = 3\}$. Let ψ be such that $Pre(Y) = \{(x_1, x_2) \in \mathbb{R}^2 \mid \psi(x_1, x_2)\}$. Applying the previous lemmas we have that

$$
\begin{aligned}
\psi(x_1, x_2) &\triangleq \exists y_1 \, \exists y_2 \, \exists t : y_1 = 4 \wedge y_2 = 3 \wedge t \geq 0 \wedge x_1 = y_1 e^{-2t} \wedge x_2 = y_2 e^t \\
&\equiv \exists y_1 \, \exists y_2 \, \exists z : y_1 = 4 \wedge y_2 = 3 \wedge 0 < z \leq 1 \wedge x_1 = y_1 z^{-2} \wedge x_2 = y_2 z \\
&\equiv \exists y_1 \, \exists y_2 \, \exists w_1 \, \exists w_2 : y_1 = 4 \wedge y_2 = 3 \wedge w_1 > 0 \wedge w_2 > 0 \wedge w_1 w_2 = 1 \\
&\qquad \wedge x_1 = y_1 w_1^2 \wedge x_2 = y_2 w_2 \\
&\equiv x_1 x_2^2 - 36 = 0 \wedge x_2 > 0
\end{aligned}
$$

5.3   Pure imaginary eigenvalues

In this case, we assume the matrix A is similar to a matrix in a special block-diagonal form, a real Jordan form. First, the number of rows (and columns) of A, is even. Second, there exist D and T such that $A = TDT^{-1}$, T invertible, and D is block diagonal with each block of size 2×2 and of the form

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

where b is the imaginary part of an eigenvalue of A. In this case we say that A has a *diagonal real Jordan form.* Moreover, if each eigenvalue is of the form $\mathbf{i}r$ with $r \in \mathbb{Q}$, then the entries of D, T, and T^{-1} are all rational.

Using a similar approach as in the case of real eigenvalues, and relying on the fact that $\cos^2 s + \sin^2 s = 1$ and $\cos(ms)$, $\sin(ms)$ can be written as polynomials in $\cos(s)$, $\sin(s)$, results in the following proposition.

Proposition 3. *Let $F(x) = Ax$ be a linear vector field and $A \in \mathbb{Q}^{n \times n}$ a matrix with pure imaginary eigenvalues of the form $\mathbf{i}r$ with $r \in \mathbb{Q}$, and with a real Jordan form. If $Y \subseteq \mathbb{R}^n$ is definable in $\mathcal{R}$, then $Pre(Y)$ is definable in $\mathcal{R}$.*

Proposition 3 implies that we have a computational procedure for the reachability problem of a class of linear vector fields with pure imaginary eigenvalues of the form $\mathbf{i}r$ with $r \in \mathbb{Q}$.

We have presented above three classes of linear vector fields for which $Pre(Y)$ can be computed for sets Y definable in $\mathsf{OF}(\mathbb{R})$. The computational results obtained in the section are now summarized by the following theorem.

Theorem 1. *Let H be a linear hybrid system where for each discrete location $q \in X_D$ the vector field is of the form $F(q, x) = Ax$ where*

- *$A \in \mathbb{Q}^{n \times n}$ is nilpotent or*
- *$A \in \mathbb{Q}^{n \times n}$ is diagonalizable with rational eigenvalues or*
- *$A \in \mathbb{Q}^{n \times n}$ has pure imaginary eigenvalues of the form $\mathbf{i}r$, $r \in \mathbb{Q}$, with diagonal real Jordan form.*

Then the reachability problem for H is semidecidable.

To obtain a decision procedure, we need to guarantee that the bisimulation algorithm terminates. In the next section we use the notion of o-minimality in order to guarantee termination of the bisimulation algorithm.

6   O-Minimal Hybrid Systems And Decidability

In order for the bisimulation algorithm to terminate, the partition of the state space resulting from the bisimulation algorithm should have a finite number of equivalence classes. It is therefore important that during the partition refinement process, the intersection of the predecessor of an equivalence class with any other equivalence class has a finite number of connected components. The search for such desirable finiteness properties of definable sets has lead to the notion of *o-minimality*. While this concept applies to any theory, we consider here only theories over the real numbers. Let $\mathcal{L}$ be a language and $Th(\mathbb{R})$ be a theory of the reals.

Definition 3. *$Th(\mathbb{R})$ is* o-minimal *("order minimal") if every definable subset of $\mathbb{R}$ is a finite union of points and intervals (possibly unbounded).*

The class of o-minimal theories is quite rich. Quantifier elimination implies that $\mathsf{Lin}(\mathbb{R})$ and $\mathsf{OF}(\mathbb{R})$ are o-minimal. In addition, even though $\mathsf{OF}_{\exp}(\mathbb{R})$ does not admit elimination of quantifiers, such theory is indeed o-minimal (see [16]). Another extension of $\mathsf{OF}(\mathbb{R})$ is obtained by adding to $\mathcal{R}$ a symbol $\hat{f}$ for every function defined by

$$\hat{f}(x) = \begin{cases} f(x) & \text{if } x \in [-1, 1]^n \\ 0 & \text{otherwise} \end{cases}$$

where f is a real-analytic function in a neighborhood of the cube $[-1, 1]^n \subset \mathbb{R}^n$. The resulting theory denoted $\mathsf{OF}_{\mathrm{an}}(\mathbb{R})$ is then an extension of $\mathsf{OF}(\mathbb{R})$ which is also o-minimal. The theory $\mathsf{OF}_{\mathrm{an}}(\mathbb{R})$ includes subanalytic sets as definable sets. In addition, it captures periodic trajectories of linear systems as the sine function restricted to a period is definable. Finally, the theory $\mathsf{OF}_{\mathrm{an},\exp}(\mathbb{R})$ obtained by adding both symbols $\hat{f}$ and exp is also o-minimal (see [13]). The following table summarizes the o-minimal theories that are of interest in this paper along with examples of sets and flows that are definable in these theories.

Table 1 : O-Minimal Theories			
Theory	Model	Definable Sets	Definable Flows
$\mathsf{Lin}(\mathbb{R})$	$(\mathbb{R}, +, -, <, 0, 1)$	Polyhedral sets	Linear flows
$\mathsf{OF}(\mathbb{R})$	$(\mathbb{R}, +, -, \cdot, <, 0, 1)$	Semialgebraic sets	Polynomial flows
$\mathsf{OF}_{\mathrm{an}}(\mathbb{R})$	$(\mathbb{R}, +, -, \cdot, <, 0, 1, \{\hat{f}\})$	Subanalytic sets	Polynomial flows
$\mathsf{OF}_{\exp}(\mathbb{R})$	$(\mathbb{R}, +, -, \cdot, <, 0, 1, \exp)$	Semialgebraic sets	Exponential flows
$\mathbb{R}_{\exp,\mathrm{an}}$	$(\mathbb{R}, +, -, \cdot, <, 0, 1, \exp, \{\hat{f}\})$	Subanalytic sets	Exponential flows

Based on the notion of o-minimality, the concept of o-minimal hybrid systems is introduced as hybrid systems whose relevant sets and flows are definable in an o-minimal theory.

Definition 4. *A hybrid system $H = (X, X_O, X_F, F, E, I, G, R)$ is said to be* o-minimal *if*

- $X_C = \mathbb{R}^n$.
- *for each $q \in X_D$ the flow of $F(q, \cdot)$ is complete (exists for all time).*
- *for each $q \in X_D$ the family of sets $\mathcal{A}_q = \{I(q), (X_O)_q, (X_F)_q\} \cup \{G(e)_q, R(e)_q \mid e \in E\}$ and the flow of $F(q, \cdot)$ are definable in the same o-minimal theory.*

For various classes of o-minimal hybrid systems, the reader is referred to [7], where the following property of o-minimal hybrid systems is proven.

Theorem 2. *Every o-minimal hybrid system admits a finite bisimulation. In particular, the bisimulation algorithm terminates for o-minimal hybrid systems.*

We can now combine the semidecision result of Theorem 1 and the termination result of Theorem 2 in order to obtain the desired decidability result.

Theorem 3. *Let H be a linear hybrid system where for each discrete location $q \in X_D$ the vector field is of the form $F(q, x) = Ax$ where*

- *$A \in \mathbb{Q}^{n \times n}$ is nilpotent or*
- *$A \in \mathbb{Q}^{n \times n}$ is diagonalizable with rational eigenvalues or*
- *$A \in \mathbb{Q}^{n \times n}$ has purely imaginary eigenvalues of the form $\mathbf{i}r$, $r \in \mathbb{Q}$, with diagonal real Jordan form.*

Then the reachability problem for H is decidable.

Proof. All relevant sets of linear hybrid systems are by definition definable in $\mathsf{OF}(\mathbb{R})$ and the flows of linear vector fields are complete. Therefore, given the semidecision result of Theorem 1, all we have to show is that the flow of the linear vector field Ax is definable in an o-minimal theory. Then Theorem 2 would guarantee termination of the bisimulation algorithm. If A is nilpotent then the flow is also definable in $\mathsf{OF}(\mathbb{R})$ which is o-minimal. If A is diagonalizable then the flow is definable in $\mathsf{OF}_{\exp}(\mathbb{R})$ which is also o-minimal. If A has purely imaginary eigenvalues, then the flow contains the functions sin and cos which are not definable in any of the o-minimal theories of Table 1. However, o-minimality of the flow is only used in the proof of Theorem 2 to show o-minimality of the *Pre* operator. Even though the flow of this vector field is not definable, the *Pre* operator corresponding to these periodic flows is still definable, as all we need is the restriction of sin and cos on $[0, 2\pi]$. These restrictions are indeed definable in $\mathsf{OF}_{\mathrm{an}}(\mathbb{R})$ which is also o-minimal. Therefore in all cases the relevant objects are definable in the same o-minimal theory, $\mathsf{OF}_{\mathrm{an,exp}}(\mathbb{R})$

Theorem 3 is the first decidability result in the area of hybrid systems that provides the modeling expressiveness to capture relatively complex continuous dynamics. In addition, Theorem 3 contains in it a purely continuous version of reachability analysis for linear systems under state constraints, a problem which is known to be very difficult. As a result, its potential application to analyze various realistic hybrid systems using computational methods is significant.

7   Conclusions

In this paper, we presented a new class of hybrid system with a decidable reachability problem. This new class captures classes of linear vector fields in each discrete location. In addition, this extension is obtained using techniques from mathematical logic and model theory. The mathematical machinery presented in this paper provides a natural and unified platform for pursuing further research along this direction.

Issues for further research include the incorporation of linear vector fields with inputs in each discrete location. This will allow to model significant modeling disturbances as well as provide us with a framework for doing symbolic controller synthesis. Preliminary results along this direction indicate very delicate nonresonance conditions between the control inputs and the eigenvalues of the systems [8]. In addition, more complicated discrete transitions are also of interest.

Another direction of research includes complexity analysis and reduction of the proposed algorithms as well as their implementation into a computational tool whose kernel will be a quantifier elimination engine.

References

1. R. Alur, C. Coucoubetis, N. Halbwachs, T.A. Henzinger, P.H. Ho, X. Nicolin, A. Olivero, J. Sifakis, and S. Yovine. The algorithmic analysis of hybrid systems. *Theoretical Computer Science*, 138:3–34, 1995.
2. R. Alur and D.L. Dill. A theory of timed automata. *Theoretical Computer Science*, 126:183–235, 1994.
3. D. S. Arnon, G. E. Collins, and S. McCallum. Cylindrical algebraic decomposition I: The basic algorithm. *SIAM Journal on Computing*, 13(4):865–877, November 1984.
4. A. Dolzman and T. Sturm. REDLOG : Computer algebra meets computer logic. *ACM SIGSAM Bulletin*, 31(2):2–9, June 1997.
5. T.A. Henzinger. Hybrid automata with finite bisimulations. In Z. Fülöp and F. Gécseg, editors, *ICALP 95: Automata, Languages, and Programming*, pages 324–335. Springer-Verlag, 1995.
6. T.A. Henzinger, P.W. Kopke, A. Puri, and P. Varaiya. What's decidable about hybrid automata? In *Proceedings of the 27th Annual Symposium on Theory of Computing*, pages 373–382. ACM Press, 1995.
7. G. Lafferriere, G. J. Pappas, and S. Sastry. O-minimal hybrid systems. Technical Report UCB/ERL M98/29, University of California at Berkeley, Berkeley, CA, April 1998.
8. G. Lafferriere, G.J. Pappas, and S. Yovine. Reachability computation of linear hybrid systems. In *Proc. of 14th IFAC World Congress*. Elsevier Science Ltd., 1999. To appear.

9. A. Puri and P. Varaiya. Decidability of hybrid systems with rectangular differential inclusions. In *Computer Aided Verification*, pages 95–104, 1994.
10. A. Tarski. *A decision method for elementary algebra and geometry*. University of California Press, second edition, 1951.
11. C. Tomlin, G. J. Pappas, and S. Sastry. Conflict resolution for air traffic management : A study in multi-agent hybrid systems. *IEEE Transactions on Automatic Control*, 43(4):509–521, April 1998.
12. L. van den Dries. *Tame Topology and o-minimal structures*. Cambridge University Press, 1998.
13. L. van den Dries and C. Miller. On the real exponential field with restricted analytic functions. *Israel Journal of Mathematics*, 85:19–56, 1994.
14. P. Varaiya. Reach set computation using optimal control. 1997. preprint.
15. V. Weispfenning. A new approach to quantifier elimination for real algebra. Technical Report MIP-9305, Universität Passau, Germany, July 1993.
16. A. J. Wilkie. Model completeness results for expansions of the ordered field of real numbers by restricted pfaffian functions and the exponential function. *Journal of the American Mathematical Society*, 9(4):1051–1094, Oct 1996.