Proceedings of the 39th IEEE Conference on Decision and Control
Sydney, Australia • December, 2000

Semidecidable Controller Synthesis for Classes of Linear Hybrid Systems

Omid Shakernia† George J. Pappas‡ Shankar Sastry†

†Department of EECS
University of California at Berkeley
Berkeley, CA 94720
{omids,sastry}@eecs.berkeley.edu

‡Department of EE
University of Pennsylvania
Philadelphia, PA 19104
pappasg@ee.upenn.edu

Abstract

A problem of great interest in the control of hybrid systems is the design of least restrictive controllers for reachability specifications. Controller design typically uses game theoretic methods to compute the region of the state space for which there exists a control such that for all disturbances, an unsafe set is not reached. In general, the computation of the controllers requires the steady state solution of a Hamilton-Jacobi partial differential equation which is very difficult to compute, if it exists. In this paper, we show that for special classes of hybrid systems where the continuous vector fields are linear, the controller synthesis problem is *semi-decidable*: There exists a computational algorithm which, if it terminates in a finite number of steps, will exactly compute the least restrictive controller. This result is achieved by a very interesting interaction of results from mathematical logic and optimal control.

1 Introduction

Reachability specifications for hybrid systems require that trajectories of a hybrid system avoid an undesirable region of the state space. One of the most important problems in the control of hybrid systems is the design of least restrictive controllers which satisfy reachability specifications. This problem has been considered in the context of classical discrete automata [2, 12], timed automata [1], linear hybrid automata [16], and general hybrid systems [7]. The framework presented in [7] has been applied to automated vehicles [6], and air traffic management systems [15].

Designing least restrictive controllers for reachability specifications requires computing the set of all initial states for which there exists a control such that for all disturbances, the system will avoid the undesirable region. The least restrictive controller is then a static feedback controller which allows any control value outside this set of initial conditions while allowing all safe control values on the boundary of this set [7].

The computation of the safe set of initial states for general hybrid systems leads to game theoretic methods, and in particular to the steady state solution of Hamilton-Jacobi equations [7, 14]. In general, these partial differential equations are very difficult to solve. In addition, due to discontinuities in the optimal control policy, steady state solutions, if they exist, may be discontinuous even if the initial problem data is continuous. Recent results in [8] provide a new formulation of the Hamilton-Jacobi partial differential equations for the *Reach* operator, and numerical techniques based on level set methods for it's computation. Despite the considerable progress in this area, there remain difficult issues that must be resolved: Existence and uniqueness of steady-state solutions to Hamilton-Jacobi equations, *shocks* (non-smooth solutions to smooth problems), *convergence* of numerical algorithms, and *leaking-corners* (points of discontinuity in the computed safe set where regardless of the control action the state can "leak" into the unsafe set).

The above difficulties in the computation of least restrictive controllers naturally raise the following question : *Can we find classes of systems where the game theoretic approach does not require the solution of the Hamilton-Jacobi equation?* Recently in [10], we showed that for certain classes of continuous, *normal* linear control systems, the controller synthesis problem is *decidable.* That is, there exists an algorithm which in a finite number of steps will *exactly* compute the least restrictive controller. In this paper, we show that for classes of hybrid systems whose continuous dynamics are linear, normal, and the dynamic matrices are nilpotent or have real, rational eigenvalues, the controller synthesis problem is *semi-decidable.* Therefore there exists an algorithm which, if it terminates, will exactly compute the least restrictive controller.

In optimal control theory [9], the normality condition requires complete controllability from *each* input and disturbance. This condition ensures that the optimal control and disturbance are well defined, and unique. If in addition the dynamic matrix has real eigenvalues, then normality also ensures that the optimal control and disturbance have a finite number of switchings [9]. Our framework first applies Pontryagin's Maximum Principle to synthesize the optimal control and worst disturbance. By combining the recent decidability results in [4, 5], with the normality condition which guarantees finite number of switchings of piecewise constant inputs [9], we show that the least restrictive controller for this class of hybrid systems can be *semi-decidably* computed. This interesting interplay of results from mathematical logic and optimal control presents us with the first semi-decidable controller synthesis result for classes of hybrid systems which include linear control systems.

In Section 2 we introduce the notation and definitions of the hybrid system model. In Section 3 we briefly review the definitions and the controller synthesis methodology proposed in [7]. In Section 4 we present the main result of the paper. Finally in Section 5 we give concluding remarks and directions for future research.

2 Hybrid System Model

Here we briefly review the definitions of a hybrid system following [7, 14]. For a finite collection V of variables, let $\mathbf{V}$ denote the set of valuations of these variables, *i.e.* the set of all possible assignments of the variables in V. For example if x is a state variables taking values in $\mathbb{R}^n$ we write $X = \{x\}$ with $\mathbf{X} = \mathbb{R}^n$.

Definition 2.1 (Hybrid system) *A hybrid system H is a collection (X, V, I, f, E, ϕ), with:*

- ***State and input variables:*** *X and V are disjoint collections of state and input variables. We assume that $X = X_D \cup X_C$ and $V = V_D \cup V_C$, where X_C and V_C contain continuous, and X_D and V_D discrete variables. We refer to valuations $x \in \mathbf{X}$ and $v \in \mathbf{V}$ as the state and the input of the hybrid system. To fix notation, we have $\mathbf{X_C} = \mathbb{R}^n$.*
- ***Initial states:*** *$I \subseteq \mathbf{X}$ is a set of initial valuations of the state variables.*
- ***Continuous evolution:*** *$f : \mathbf{X} \times \mathbf{V} \to T\mathbf{X_C}$ is a vector field.*
- ***Discrete transitions:*** *$E \subseteq \mathbf{X} \times \mathbf{V} \times \mathbf{X}$ is a set of discrete transitions.*
- ***Admissible inputs:*** *$\phi : \mathbf{X} \to 2^{\mathbf{V}}$ gives the set of admissible inputs at a given state $x \in \mathbf{X}$.*

By abuse of notation, we sometimes use $(q, x) = (x|_{X_D}, x|_{X_C}) \in \mathbf{X}$. The meaning of the variable x will be clear from the context.

Definition 2.2 (Hybrid time trajectory) *A hybrid time trajectory τ, is a finite or infinite sequence of intervals $\tau = \{I_i\}$ of the real line, starting with I_0 and satisfying:*

- *I_i is closed unless τ is a finite sequence and I_i is the last interval, in which case it is left closed but can be right open.*
- *Let $I_i = [\tau_i, \tau_i']$. Then for all i $\tau_i \leq \tau_i'$ and for $i > 0, \tau_i = \tau_{i-1}'$.*

We denote by $\mathcal{T}$ the set of all hybrid time trajectories.

Definition 2.3 (Execution) *An execution of a hybrid system H is a collection (τ, x, v) with $\tau \in \mathcal{T}$, $x : \tau \to \mathbf{X}$ and $v : \tau \to \mathbf{V}$ which satisfies:*

- ***Initial condition:*** *$x(\tau_0) \in I$.*
- ***Discrete evolution*** *: $x(\tau_{i-1}', v(\tau_{i-1}'), x(\tau_i)) \in E$ for all i.*
- ***Continuous evolution:*** *for all i with $\tau_i < \tau_i'$, x is continuous and v is piecewise continuous in $[\tau_i, \tau_i']$ and for all $t \in [\tau_i, \tau_i')$, $(x(t), v(t), x(t)) \in E$. Moreover, for all $t \in [\tau_i, \tau_i']$ where v is continuous $\dot{x}(t)_{|X_C} = f(x(t), v(t))$.*
- ***Input constraints:*** *for all $t \in \tau$, $v(t) \in \phi(x(t))$.*

We use χ to denote and execution of H, and $\mathcal{H}$ to denote the set of all executions of H. A *property* P of a hybrid system H is a map $P : \mathcal{H} \to \{\mathsf{True}, \mathsf{False}\}$. An execution $\chi \in \mathcal{H}$ satisfies property P if $P(\chi) = \mathsf{True}$, and the hybrid system satisfies property P if $P(\chi) = \mathsf{True}$ for all $\chi \in \mathcal{H}$. Given a set $F \subseteq \mathbf{X}$ we define a *safety property*, denoted by $\Box F$, by:

$$\Box F(\chi) = \begin{cases} \mathsf{True} & \text{if } \forall t \in \tau, x(t) \in F \\ \mathsf{False} & \text{otherwise.} \end{cases}$$

3 Controller Synthesis for Hybrid Systems

Given a hybrid system H, we are asked to control it using its input variables so that its executions satisfy certain properties. The input variables of the hybrid system are partitioned into two classes: *controls* and *disturbances*. We have $V = U \cup D$ where U and D are the control and disturbance variables. The controls can be influenced using a *controller* to guide the system, while the disturbances are determined by the environment, which may included uncertainties and modeling errors. In this paper, we concentrate on the *controller* *synthesis problem* $(H, \Box F)$: the problem of synthesizing a feedback controller $g : \mathbf{X} \to 2^{\mathbf{U}}$ such that for some *safe set* F all closed loop trajectories of the hybrid system H satisfy the property $\Box F$.

We now review the controller synthesis methodology for general hybrid systems as presented in [7, 13, 14]. A set $W \subseteq \mathbf{X}$ is *controlled invariant* if the controller synthesis problem $(H, \Box W)$ can be solved when $I = W$. In [7], it was shown that the controller synthesis problem $(H, \Box F)$ can be solved iff there exists a unique *maximal controlled invariant* subset $W^* \subseteq F$. Once the maximal controlled invariant set W^* is found the goal is to find the unique *least restrictive controller* g that renders W^* invariant.

For any input $v = (u, d) \in \mathbf{V}$ define the set:

$$Inv(v) = \{x \in \mathbf{X} \mid v \in \phi(x) \wedge (x, v, x) \in E\}. \quad (1)$$

For a state $x \in \mathbf{X}$ and input $v = (u, d)$ define:

$$Next(x, v) = \begin{cases} \{y \in \mathbf{X} \mid (x, v, y) \in E\} & \text{if } v \in \phi(x) \\ \emptyset & \text{if } v \notin \phi(x). \end{cases} \quad (2)$$

$Inv(v)$ is the set of states from which continuous evolution is possible under input v, while $Next(x, v)$ is the set of states that can be reached from x under input v through a discrete transition. For any set $K \subseteq \mathbf{X}$ and input $v = (u, d)$ the *successor* of K under v is given by:

$$Next(K, v) = \bigcup_{x \in K} Next(x, v) \quad (3)$$

For any set $K \subseteq \mathbf{X}$ define the *controllable predecessor* of K, $Pre_u(K)$, and the *uncontrollable predecessor* of K, $Pre_d(K)$, by:

$$\begin{aligned} Pre_u(K) = \{x \in \mathbf{X} \mid\ & \exists u \in \mathbf{U}\ \forall d \in \mathbf{D}\ x \notin Inv(v) \\ & \wedge Next(K, v) \subseteq K\} \cap K \\ Pre_d(K) = \{x \in \mathbf{X} \mid\ & \forall u \in \mathbf{U}\ \exists d \in \mathbf{D}\ Next(K, v) \\ & \cap K^c \neq \emptyset\} \cup K^c. \end{aligned} \quad (4)$$

where $v = (u, d)$. $Pre_u(K)$ contains all states in K for which u can force a transition back into K. $Pre_d(K)$ contains all states outside K together with those states for which it is possible to transition outside K regardless of the action of u. As discussed in [14], it is direct to see that for any set K, $Pre_u(K) \cap Pre_d(K) = \emptyset$.

Whereas Pre_u and Pre_d capture information about regions of the state space we can reach using discrete transitions of the system, the following operator captures continuous reachability information.

Definition 3.1 (Reach-Avoid [14]) *For two disjoint sets $K, G \subseteq \mathbf{X}$, the Reach-Avoid operator $Reach : 2^{\mathbf{X}} \times 2^{\mathbf{X}} \to 2^{\mathbf{X}}$ is defined as:*

$$\begin{aligned} Reach(K, G) = \ & \{x_0 \mid \forall u \in \mathcal{U}\ \exists d \in \mathcal{D}\ \exists t \geq 0 : \\ & x(t) \in K \wedge \forall s \in [0, t]\ x(s) \notin G\}. \end{aligned} \quad (5)$$

Here $\mathcal{U}, \mathcal{D}$ denote the set of piecewise continuous functions from the $\mathbb{R}$ to $\mathbf{U}, \mathbf{D}$ respectively, and $x(\cdot)$ is the unique state trajectory starting from initial condition $x(0) = x_0$ under the input (u, d). The set $Reach(K, G)$ contains the states from which for all controls there exists a disturbance such that the state trajectory can be driven to K while avoiding the escape set G.

The following algorithm uses the *Reach* operator to compute the maximal controlled invariant subset of F (see [14]).

Algorithm 3.2 (Max Controlled Invariant Set)

initialize
$W^0 = F$
$W^{-1} = \emptyset$
$i = 0$
while $W^i \neq W^{i-1}$
$W^{i-1} = W^i \setminus Reach(Pre_d(W^i), Pre_u(W^i))$
$i = i - 1$
end while

Algorithm 3.2 iteratively removes from the safe set F all states for which there is a disturbance which either through continuous evolution or discrete transition can bring the system outside F regardless of the control action. In general one can not expect the algorithm to converge in a finite number of iterations. However, if the algorithm terminates, then the algorithm computes the unique maximal controlled invariant set $W^* \subseteq F$.

In order to implement Algorithm 3.2, one needs to encode sets of states, perform set intersection, union, test for emptiness, and *exactly* compute $Reach(\cdot, \cdot)$. If all these conditions hold for a class of systems, then the problem is *semi-decidable* for that class of systems. Even though there is no *guarantee* of termination, if the algorithm terminates, then it exactly computes the maximal controlled invariant set W^*. If in addition, Algorithm 3.2 is guaranteed to terminate after a finite number of iterations, then we say the problem is *decidable.*

The main difficulty in the implementation of the above algorithm is the computation of the *Reach* operator. For general nonlinear hybrid system, the computation of *Reach* relies on the numerical solution of a pair of coupled Hamilton-Jacobi partial differential equations [14]. Recent results in [8] provide a new formulation of the Hamilton-Jacobi PDEs and level set methods for the approximate computation of *Reach*. These new results have improved the state of the art in computation for hybrid systems, however there remain very difficult issues that must be resolved: Existence and uniqueness of such solutions, *shocks* (non-smooth solutions to smooth problems), *convergence* of numerical algorithms, and *leaking-corners* (points of discontinuity in the computed safe set where regardless of the control action the state can "leak" into the unsafe set).

In the following section, we extend the results on decidable controller synthesis for linear systems in [10] to introduce a semi-decision procedure for controller synthesis for classes of linear hybrid systems where the solution of the Hamilton-Jacobi PDE is not necessary.

4 Semidecidable Controller Synthesis

The controller synthesis Algorithm 3.2 requires methods to represent sets, perform unions, intersections, complements, check emptiness of sets, and compute Pre_u, Pre_d, and *Reach*. A natural platform for solving these computational issues is provided by first order logic where sets would be represented as quantifier free formulas. By restricting ourselves to the theory of reals $(\mathbb{R}, <, +, -, \cdot, 0, 1)$ which is known to admit quantifier elimination [11], we can represent semi-algebraic sets (boolean combinations of sets of the form $\{x \in \mathbb{R}^n \mid f(x) < 0\}$ and $\{x \in \mathbb{R}^n \mid g(x) = 0\}$ where $f, g \in \mathbb{Q}[x_1, \ldots, x_n]$). In this setting, the computation of intersection, union, and tests for emptiness is provided by mathematical logic and quantifier elimination [11].

For a linear system $\dot{x} = Ax + Bu$, the pair (A, B) is called *normal* if for each column b_i of B, the pair (A, b_i) is completely controllable. This motivates the following definition.

Definition 4.1 (Normal Linear Hybrid System) *Hybrid system $H = (X, V, I, f, E, \phi)$ is called a* normal *linear hybrid system if $\forall q \in \mathbf{X}_D$ the set of feasible inputs $\phi(q, x)|_{V_C} = \mathbf{U}_q \times \mathbf{D}_q$ where $\mathbf{U}_q$ and $\mathbf{D}_q$ are compact rectangles with rational vertices, the reset relation $E \subseteq \mathbf{X} \times \mathbf{V} \times \mathbf{X}$ is semi-algebraic, and the vector field is given by $f(q, x, u, d) = A_q x + B_q u + E_q d$, where A_q, B_q, E_q, have rational coefficients and the pairs (A_q, B_q) and (A_q, E_q) are* normal.

As we will see, the normality condition and the structure of the feasible inputs are necessary for proving the computability of the *Reach* operator.

The semi-algebraic condition on the discrete transition relation E is used in proving computability of the Pre_u, Pre_d operators. In fact, by defining the sets $Inv(v)$ and $Next(x, v)$ as first-order formulas in the theory of reals, one can see that the following lemma is a result of the decidability of quantifier elimination on first order formula in $(\mathbb{R}, <, +, -, \cdot, 0, 1)$.

Lemma 4.2 (Computable Discrete Transitions) *Given a normal linear hybrid system H, for any semi-algebraic set $K \subseteq \mathbf{X}$ operators $Pre_u(K)$ and $Pre_d(K)$ given in equation (4) are computable.*

Since the discrete state remains constant along continuous evolution, in Algorithm 3.2 the set $Reach(Pre_d(W^i), Pre_u(W^i))$ can be computed separately for each discrete state. Hence, for each discrete location we need to solve a *continuous* controller synthesis problem to solve for all states such that the disturbance can drive the system to a bad state while avoiding an escape set regardless of the actions of the controller. The following theorem, which is a generalization of the main result in [10], provides conditions on when the above *continuous* controller synthesis problem is decidable.

Theorem 4.3 (Computable Reach) *Consider the linear differential game*

$$\dot{x} = Ax + Bu + Ed$$

with controls $u \in U$ and disturbances $d \in D$, which satisfies the following properties:

- *$A \in \mathbb{Q}^{n \times n}$, $B \in \mathbb{Q}^{n \times n_u}$, $E \in \mathbb{Q}^{n \times n_d}$,*
- *the pairs $(A, B), (A, E)$ are normal,*
- *the sets U, D are compact rectangles with rational vertices.*

If A is nilpotent or diagonalizable with real rational eigenvalues, then given disjoint semi-algebraic sets $K, G \subseteq \mathbb{R}^n$, the set $Reach(K, G)$ is a computable *semi-algebraic set.*

Proof: We provide a sketch of the proof due to space limitations. From the definition of *Reach* it is direct to check that $Reach(\cup_i K_i, G) = \cup_i Reach(K_i, G)$. Hence it is sufficient to show that $Reach(K, G)$ is computable if K is a *basic* semi-algebraic set of the form

$$K = \{x \in \mathbb{R}^n \mid f_1(x) < 0, \ldots, f_p(x) < 0, \\ g_1(x) = 0, \ldots g_q(x) = 0\},$$

and the escape set is a general semi-algebraic set defined by $G = \{x \in \mathbb{R}^n \mid \psi(x)\}$.

1. Construct the Hamiltonian:

$$H(p, x, u, d) = p^T Ax + p^T Bu + p^T Ed.$$

The state and co-state dynamics are given by:

$$\dot{x} = \frac{\partial H}{\partial p}, \qquad \dot{p} = -\frac{\partial H}{\partial x}^T = -A^T p.$$

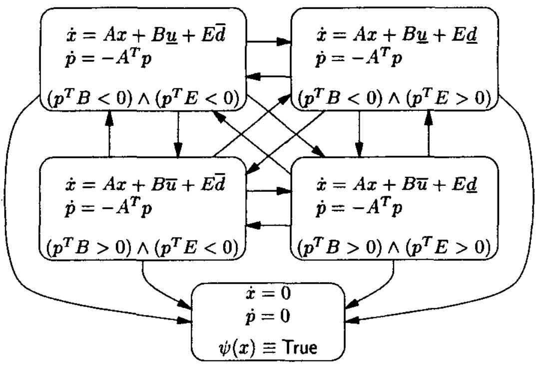

Figure 1: The hybrid system constructed for *Reach* computation for a linear differential game with one input $u \in [\underline{u}, \overline{u}]$, one disturbance $d \in [\underline{d}, \overline{d}]$ and an escape set $G = \{x \mid \psi(x)\}$.

2. Apply the Pontryagin Maximum Principle [9] to obtain the *saddle solution* of optimal (u^*, d^*):

$$u^* = \arg\max_{u \in U} p^T Bu, \quad d^* = \arg\min_{d \in D} p^T Ed$$

By the normality condition, the Maximum Principle provides that the optimal controls and disturbances are unique and switch on the vertices of the feasible control and disturbance sets. Since the matrix A has purely real eigenvalues, the Maximum principle provides that there exists a uniform upper bound on the number of switchings of (u^*, d^*).

3. Construct a hybrid system from the switching logic of (u^*, d^*). The continuous state of the hybrid system consists of $(x,p)^T \in \mathbb{R}^{2n}$. Add a "stop" discrete state which stops the computation when the condition $x \notin G$ is violated (see Definition 3.1 and Figure 1). The hybrid system constructed for *Reach* computation of a system has 2 discrete states for each input and disturbance plus the stopping state, yielding a total of $2^{n_u+n_d}+1$ discrete states. Please refer to [10] for details on the construction of this hybrid system.

   Each discrete state of the hybrid system has a linear vector field with a constant input. Proposition 4 in [10], which is builds on the results in [3, 4, 5], provides that for a linear system with constant input, if the dynamic matrix A is nilpotent or diagonalizable with real rational eigenvalues, then the problem of computing the set of states which can reach a semi-algebraic set is decidable.

4. Compute the Usable Part and the co-Usable Part of K:

$$\partial K \triangleq \bigcup_{j=1}^{p} \{x \in \mathbb{R}^n \mid g_1(x) = 0, \ldots, g_q(x) = 0, f_1(x) \le 0, \ldots, f_j(x) = 0, \ldots, f_p(x) \le 0\}.$$

$$\begin{aligned}\mathrm{UP} \triangleq \{&x \in \partial K \mid \exists d \in D\ \forall u \in U : \\ &\textstyle\bigwedge_{i=1}^{p}[(f_i(x) = 0) \Rightarrow (\frac{\partial f_i(x)}{\partial x})^T(Ax+Bu+Ed) < 0] \\ &\textstyle\bigwedge_{j=1}^{q}[(g_j(x) = 0) \wedge (\frac{\partial g_j(x)}{\partial x})^T(Ax+Bu+Ed) = 0] \\ &\textstyle\bigvee[\exists j : g_j(x) = 0 \wedge (\frac{\partial g_j(x)}{\partial x})^T(Ax+Bu+Ed) \neq 0]\}.\end{aligned}$$

$$\begin{aligned}\mathrm{coUP}(x) \triangleq \{&v \in \mathbb{R}^n \mid \\ &\textstyle\bigwedge_{i=1}^{p}[(f_i(x) = 0) \Rightarrow (\frac{\partial f_i(x)}{\partial x})^T v > 0] \\ &\textstyle\bigwedge_{j=1}^{q}[(g_j(x) = 0) \Rightarrow (\frac{\partial g_j(x)}{\partial x})^T v \neq 0]\}.\end{aligned}$$

   UP is the subset of ∂K for which an input exists that can instantaneously drive x into K^o. For each $x \in \mathrm{UP}$, $\mathrm{coUP}(x)$ is the set of vectors such that if the system were to flow along any of these directions the state would instantaneously enter K^c.

5. Initialize the hybrid system for Pre computation:

$$\widetilde{\mathrm{UP}} = \{(x,p) \in \mathbb{R}^{2n} \mid x \in \mathrm{UP}, p \in \mathrm{coUP}(x)\}$$

Proposition 3 in [10] can be applied here to show that the constructed hybrid system is *non-blocking, deterministic, and non-Zeno.* Since in each discrete state of the hybrid system, the input is constant we can apply Proposition 4 in [10] to decidably compute the set of states that can reach $\widetilde{\mathrm{UP}}$. The predecessor operator of the discrete jumps of the hybrid system is trivial since the reset map of the jumps is the identity map. Since the dynamic matrix A has real eigenvalues, there is a finite number of switchings of the optimal control uniformly in $(x,p) \in \widetilde{\mathrm{UP}}$. Hence, after a finite number of jumps of the hybrid system, we may apply Proposition 4 in [10] one last time.

For the constructed hybrid system, let $M \subseteq \mathbb{R}^{2n}$ be the set of all points that can reach $\widetilde{\mathrm{UP}}$. The computation of M requires a finite number of discrete transitions, and for each discrete state the continuous predecessor is computable. Hence M is a computable semi-algebraic set. Then we have $Reach(K,G) = \{x \in \mathbb{R}^n \mid (\exists p : (x,p) \in M) \vee x \in K\}$. ■

The following corollary is a direct result of Lemma 4.2 and Theorem 4.3.

Corollary 4.4 *Given a normal linear hybrid system H and a semi-algebraic set F, if the dynamic matrices in the linear vector fields A_q are either nilpotent or diagonalizable with real rational eigenvalues, then each iteration of Algorithm 3.2 is computable. Hence hence for normal linear hybrid systems, problems of computing the maximum controlled invariant set $W^* \subseteq F$ is* semidecidable.

Having computed the maximum controlled invariant set $W^* \subseteq F$, the following lemma computes the least restrictive controller which renders W^* invariant. Computability of the least restrictive controller comes directly from the fact that the theory of reals admits quantifier elimination.

Lemma 4.5 (Least Restrictive Controller) *Given the normal linear hybrid system H and a semi-algebraic maximal controlled invariant set*

$$W^* = \left\{x \in \mathbb{R}^n \mid \bigvee_{j=1}^{K} \left(\bigwedge_{k=1}^{L_j} h_{j_k}(x) \;\leq\; 0\right)\right\},$$

the least restrictive controller $g(x) : \mathbf{X} \to 2^{\mathbf{U}}$ that renders W^* invariant is computable and is given by:

$$g(x) = \begin{cases} \{u \in \phi(x)|_U \mid \forall d \in \phi(x)|_D : \\ \quad Next(x,(u,d)) \subseteq W^*\}, \text{ if } x \in (W^*)^o \\ \{u \in \phi(x)|_U \mid [\bigvee_{j=1}^{K}(\bigwedge_{k=1}^{L_j}(h_{j_k}(x) = 0) \Rightarrow \\ \quad \forall d \in \phi(x)|_D : (\frac{\partial h_{j_k}(x)}{\partial x})^T f(x,(u,d)) \leq 0) \\ \quad \wedge x \in Inv(u,d)] \\ \quad \vee [\forall d \in \phi(x)|_D : Next(x,(u,d)) \subseteq W^* \\ \qquad \wedge x \notin Inv(u,d)]\}, \quad \text{if } x \in \partial W^* \\ \phi(x)|_U, \qquad\qquad \text{if } x \in (W^*)^c. \end{cases}$$

Corollary 4.4 and 4.5 together give us the main result:

Theorem 4.6 (Semidecidable Synthesis) *Given a normal linear hybrid system H and a semi-algebraic set F, if the dynamic matrices in the linear vector fields A_q are either nilpotent or diagonalizable with real rational eigenvalues, the controller synthesis problem $(H, \Box F)$ is* semi-decidable.

5 Conclusions

In this paper we have shown that controller synthesis for classes of linear hybrid systems with semi-algebraic reachability specifications is semi-decidable. In further research, we will investigate conditions for semi-decidability in the absence of the normality condition, and the case where the dynamic matrices have purely imaginary eigenvalues. In the case of purely imaginary eigenvalues, the problem becomes quickly undecidable unless one remains in a compact region of the state space. The observation along with the results of this paper have a clear and natural connection with o-minimal theories of the reals from mathematical logic.

Acknowledgments

The authors would like to thank John Lygeros for his valuable comments. The research by Omid Shakernia was supported by DARPA under grant F33615-98-C-3614. The work of George Pappas was supported by supported by DARPA Grant N66001-99-C-8510, DARPA Grant F33615-00-C-1707, and by the University of Pennsylvania Research Foundation.

References

[1] E. Asarin, O. Maler, and A. Pnueli. Symbolic controller synthesis for discrete and timed systems. In P. Antsaklis, W. Kohn, A. Nerode, and S. Sastry, editors, *Hybrid Systems II*, volume 999 of *Lecture Notes in Computer Science*. Springer-Verlag, 1995.

[2] A. Church. Logic, arithmetic, and automata. In *Proceedings of the International Congress of Mathematics*, pages 23–35, 1962.

[3] G. Lafferriere, G. J. Pappas, and S. Sastry. O-minimal hybrid systems. *Mathematics of Control, Signals, and Systems.* To appear.

[4] G. Lafferriere, G. J. Pappas, and S. Yovine. A new class of decidable hybrid systems. In *Hybrid Systems : Computation and Control*, volume 1569 of *Lecture Notes in Computer Science*, pages 137–151. Springer Verlag, 1999.

[5] G. Lafferriere, G. J. Pappas, and S. Yovine. Reachability computation for linear hybrid systems. In *Proceedings of the 14th IFAC World Congress*, volume E, pages 7–12, Beijing, P.R. China, July 1999.

[6] J. Lygeros, D.N. Godbole, and S. Sastry. Verified hybrid controllers for automated vehicles. *IEEE Transactions on Automatic Control*, 43(4):522–539, April 1998.

[7] J. Lygeros, C. Tomlin, and S.S. Sastry. Controllers for reachability specifications for hybrid systems. *Automatica*, 35(3):349–370, March 1999.

[8] I. Mitchell and C. Tomlin. Level set methods for computation in hybrid systems. In *Proceedings of Hybrid Systems: Computation and Control*, LNCS 1790, pages 310–323. Springer-Verlag, March 2000.

[9] L.S. Pontryagin, V. Boltyanskii, R. Gamkrelidze, and E. Mischenko. *The Mathematical Theory of Optimal Processes.* John Wiley & Sons, 1962.

[10] O. Shakernia, G. Pappas, and S. Sastry. Decidable controller synthesis for classes of linear systems. In *Proceedings of Hybrid Systems: Computation and Control*, LNCS 1790, pages 407–420. Springer-Verlag, March 2000.

[11] A. Tarski. *A decision method for elementary algebra and geometry.* University of California Press, second edition, 1951.

[12] W. Thomas. On the synthesis of strategies in infinite games. In Ernst W. Mayr and Claude Puech, editors, *Proceedings of STACS 95, Volume 900 of LNCS*, pages 1–13. Springer Verlag, Munich, 1995.

[13] C. Tomlin, J. Lygeros, and S. Sastry. Synthesizing controllers for nonlinear hybrid systems. In *Proceedings of Hybrid Systems: Computation and Control*, LNCS 1386, pages 360–373. Springer-Verlag, April 1998.

[14] C. Tomlin, J. Lygeros, and S. Sastry. Computing controllers for nonlinear hybrid systems. In *Proceedings of Hybrid Systems: Computation and Control*, LNCS 1569. Springer-Verlag, March 1999.

[15] C. Tomlin, G. J. Pappas, and S. Sastry. Conflict resolution for air traffic management : A study in muti-agent hybrid systems. *IEEE Transactions on Automatic Control*, 43(4):509–521, April 1998.

[16] H. Wong-Toi. The synthesis of controllers for linear hybrid automata. In *Proceedings of the 36th IEEE CDC*, San Diego, CA, December 1997.