

On the Existence of Solutions to Controlled Hybrid Automata

Michael Lemmon

University of Notre Dame
Dept. of Elect. Eng., Notre Dame, IN 46556, USA
lemmon@maddog.ee.nd.edu

Abstract. This paper studies the existence of solutions to a class of hybrid automata in which the underlying continuous dynamics are represented by inhomogeneous linear time-invariant systems whose inputs are *controls* that can be determined by the user. The principal result of the paper is a procedure that searches for global periodic non-terminating solutions of systems having a single cycle.

1 Introduction

A *controlled hybrid automaton* is a hybrid automaton [Alu93] [Lyn96] whose underlying continuous-state dynamics are modeled as inhomogeneous differential equations. In particular, we restrict our attention to continuous-dynamics represented by linear time-invariant (LTI) systems of the form $\dot{x}(t) = Ax(t) + Bu(t)$ where $A \in \mathbb{R}^{n \times n}$, $x(t) \in \mathbb{R}^n$, $B \in \mathbb{R}^n$, and $u(t) \in \mathbb{R}$. The scalar $u(t)$ is the *control input* at time $t \in \mathbb{R}$ and it is selected by the system designer. In this paper, we further restrict our attention to systems with only a single cycle. This paper presents preliminary work examining conditions under which non-chattering and non-terminating solutions exist for the controlled hybrid automaton. The principal result is a gradient-following algorithm that provides a systematic means of searching for global periodic non-terminating solutions of systems with single cycles.

The remainder of the paper is organized as follows. Section 2 defines the controlled hybrid automaton and defines the sense in which a hybrid trajectory satisfies such a system. Section 3 outlines conditions for the existence of local non-chattering solutions. Section 4 outlines conditions for global periodic non-terminating system trajectories. Final remarks are found in section 5.

2 Controlled Hybrid Automata

A *controlled hybrid automaton* is a labeled digraph characterized by the 4-tuple (N, A, ℓ_N, ℓ_A) . N is a set of *nodes* in the directed graph (represented graphically as open circles). The set of nodes is usually taken as a subset of the positive integers. $A \subset N \times N$ is a set of *directed arcs* between nodes. The arc (i, j) from node i to node j is graphically represented as an arrow that starts at node i and

terminates at node j . The ordered pair (N, A) is the *finite automaton* associated with the hybrid system. The map $\ell_N : N \times \mathfrak{R}^{n \times n} \times \mathfrak{R}^n$ associates a pair of real vectors with the node. In particular, the label $\ell_N(i) = (A_i, B_i)$ associates a real matrix $A_i \in \mathfrak{R}^{n \times n}$ and a real matrix (vector) $B_i \in \mathfrak{R}^n$ with the i th node. Associated with node i is the following inhomogeneous differential equation,

$$\dot{x}(t) = A_i x(t) + B_i u(t) \tag{1}$$

where $x(t) \in \mathfrak{R}^n$ and $u(t) \in \mathfrak{R}$. Equation 1 is called the *modal equation* of the i th node. The map $\ell_A : A \rightarrow \mathcal{P}(\mathfrak{R}^n)$ maps an arc $a_1 \in A$ onto a collection of vectors in \mathfrak{R}^n . In particular, if arc a_1 is labeled as

$$\ell_A(a_1) = \{v_{11}, v_{12}, \dots, v_{1p_1}\}$$

then we can associate with a_1 a special subset $\Gamma(a_1) \subset \mathfrak{R}^n$ that is called the *guard* of the arc. The *guard* is defined to be the convex hull of the points in the collection $\ell_A(a_1)$. By the standard representation theorems for convex sets, we therefore know that $\Gamma(a_1)$ can be characterized as

$$\Gamma(a_1) = \left\{ x = \sum_{i=1}^{p_1} \lambda_{1i} v_{1i} \ : \ \sum_{i=1}^{p_1} \lambda_{1i} = 1 \ , \ \lambda_{1i} \geq 0 \ , \ v_{1i} \in \ell_A(a_1) \right\}$$

From the above equation it should be clear that the vectors in $\ell_A(a_1)$ are the extreme points (vertices) for convex polytope $\Gamma(a_1)$.

Remark: In this paper we've adopted the convention of representing guards as convex combinations of vertices, rather than as feasible regions bounded by hypersurfaces.

A *controlled hybrid trajectory* $z : \mathfrak{R} \rightarrow X \times N \times U$ is a function mapping a real number $\tau \in \mathfrak{R}$ onto the ordered triple $(x(\tau), i(\tau), u(\tau))$ where $x(\tau) \in X \subset \mathfrak{R}^n$ is called the *continuous state*, $i(\tau) \in N$ is called the *discrete state*, and $u(\tau) \in U \subset \mathfrak{R}$ is the *control*. It is assumed that X is a closed connected subset of \mathfrak{R}^n and it is assumed that U is a compact subset of \mathfrak{R} .

A time instant $\tau \in \mathfrak{R}$ is said to be *regular* if z is continuous at τ . (In this case, we assume that N is equipped with a discrete metric $d(i, j) = 1$ if $i \neq j$ and is zero if $i = j$). If τ is not a regular point, then it is called a *switching instant*. Controlled hybrid trajectories with a finite number of switching instants in any closed time interval are said to be *non-chattering*. A controlled trajectory with an infinite number of switching instants is said to be *non-terminating*. The trajectory is said to be *local* if its maximum interval of existence has the form $[\tau_0, \tau_0 + T)$ and T is finite. The trajectory is said to be *global* if its maximum interval of existence of $[\tau_0, \infty)$.

A controlled hybrid trajectory $z : [\tau_0, \tau_0 + T) \rightarrow X \times N \times U$ is said to *satisfy* the controlled hybrid automaton (N, A, ℓ_N, ℓ_A) with initial condition $x_0 \in X$ and $i_0 \in N$ at time $\tau_0 \in \mathfrak{R}$ if and only if

- $x(\tau_0) = x_0, i(\tau_0) = i_0,$ and $u(\tau_0) \in U.$
- For all closed intervals $[\tau_a, \tau_b]$ containing *no* switching instant, there exists a $j \in N,$ an absolutely continuous [Aub84] trajectory $x : [\tau_a, \tau_b] \rightarrow X,$ and a measurable control $u : [\tau_a, \tau_b] \rightarrow U$ such that $i(\tau) = j$ and $\dot{x}(\tau) = A_j x(\tau) + B_j u(\tau)$ for all $\tau \in [\tau_a, \tau_b].$
- At any switching instant, $\tau_s \in \mathfrak{R},$ there exists a j and k in N such that $(j, k) \in A,$ $\lim_{\tau \rightarrow \tau_s^-} i(\tau) = k,$ and $x(\tau_s) \in \Gamma((j, k)).$

Such trajectories are also said to be *solutions* of the hybrid automaton. A system that can generate non-chattering solutions will be said to be *non-Zeno*. A system that can generate non-terminating solutions will be said to be *deadlock-free*.

Remark: Note that switching can occur anywhere within the guard set.

Consider a controlled trajectory z defined over $[\tau_0, \tau_0 + T]$ with discrete state trajectory $i : [\tau_0, \tau_0 + T] \rightarrow N.$ The sequence of discrete states associated with i can be denoted by the string $\sigma \in N^*$ where N^* is the Kleene closure of $N.$ We refer to σ as the trajectory’s *event sequence*. By the pumping lemma [Dav83], we know any finite length event sequence can be decomposed as $\sigma = usv$ such that the event sequence $us^n v$ (for any positive n) is accepted by the *finite automaton* (N, A) associated with our system. This means that the sequence s represents a *cycle* of events. If there exist trajectories such that the *hybrid automaton* can execute this cycle repeatedly, then we say that the hybrid automaton is deadlock-free with respect to $s.$ A key issue in the study of hybrid automata (whether or not they are controlled) concerns the deadlock-freedom of such systems. This issue is, in essence, a question concerns the existence of global non-terminating solutions to hybrid automata.

3 Local Non-chattering Solutions

Figure 1 shows a cyclic controlled hybrid automaton. Assume that the initial continuous state at time τ_0 is x_0 and that the initial discrete state is $i_0 = 1.$ In this section, we briefly examine conditions ensuring the existence of a $T > 0$ such that there exists a controlled hybrid trajectory z over the interval $[\tau_0, \tau_0 + T]$ that is a solution to the controlled hybrid automaton. In this section, we consider two distinct cases. The first case occurs when x_0 is not in $\Gamma((1, 2)).$ The second case occurs when $x_0 \in \Gamma((1, 2)).$

The following results are a routine application of *viability theory* [Aub84] and are presented here for the sake of completeness. See [Aub84] for a precise statement of the definitions and theorems cited below.

Let’s assume that $x_0 \notin \Gamma((1, 2)).$ Since the guards are closed sets, this means that x_0 belongs to an open set so we can enclose x_0 in an open neighborhood $B_\epsilon(x_0)$ that is contained completely within the complement of the two guards. Over this neighborhood we can define a set valued mapping $F : X \rightarrow \mathcal{P}(\mathfrak{R}^n)$ that takes the value

$$F(x) = \{A_1 x + B_1 u, u \in U\} \tag{2}$$

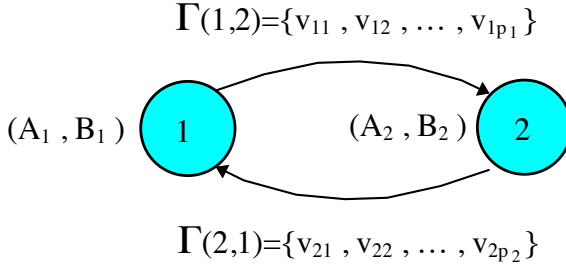


Fig. 1. cyclic controlled hybrid automata

at x . Since U is compact, we know that $F(x)$ will be an upper semi-continuous (USC) convex-valued map and we can use the USC Existence Theorem to infer that there exists a $\delta > 0$ and an absolutely continuous $x : [\tau_0, \tau_0 + \delta) \rightarrow X$ that satisfies the differential inclusion $\dot{x} \in F(x)$. By the Measurable Selection Theorem we can infer the existence of a measurable u over the same interval. Since there are no switching instants over this time interval, we know that $i(\tau) = 1$ for all $\tau \in [\tau_0, \tau_0 + \delta)$. This particular case, therefore, has a hybrid trajectory satisfying the system.

The other major case of interest occurs when $x_0 \in \Gamma((1, 2))$. Let's first consider the case when $x_0 \in \Gamma((1, 2))$ and $x_0 \notin \Gamma((2, 1))$. We consider the set valued map, $F(x)$, of equation 2. Since x_0 may lie on the boundary of $\Gamma((1, 2))$, we cannot enclose x_0 in an open neighborhood over which F is defined. However, F is upper semicontinuous and provided we can ensure that F satisfies the *tangential condition* [Aub84], then we can use the Viability Theorem to ensure the existence of an absolutely continuous solution to the differential inclusion $\dot{x} \in F(x)$ that is viable in $\Gamma((1, 2))$. Finally, the Measurable Selection Theorem ensures the existence of the desired Lebesgue measurable control, u .

Now let's consider the case when $x \in K = \Gamma((1, 2)) \cap \Gamma((2, 1))$. Consider a set valued map, F , that takes the value

$$F(x) = \{A_i x + B_i u : u \in U, i = \{1, 2\}\}$$

at point $x_0 \in K$. Since K is compact, then x_0 may lie on the boundary of K and cannot be enclosed in an open neighborhood over which F is defined. Moreover, F may not be upper semicontinuous over K . Therefore we cannot use the USC Existence Theorem to establish the existence of local trajectories. However, the convex hull $\overline{\text{co}}(F(x))$ of $F(x)$ is clearly convex valued and Lipschitzean on K . Moreover, if we can ensure that the tangential condition holds, then the Viability Theorem ensures the existence of an absolutely continuous solution to $\dot{x} \in \overline{\text{co}}(F(x))$ that is viable in K . The Relaxation Theorem can then be used to infer the existence of absolutely continuous solutions to the original differential inclusion $\dot{x} \in F(x)$. As before, an application of the Measurable Selection Theorem ensures the existence of a Lebesgue measurable u for this selected tra-

jectory x . Finally, since x is absolutely continuous, we know that each closed interval has a finite number of switching instants in which the discrete state changes value thereby establishing that the trajectory is non-chattering. We've therefore established the existence of a local non-chattering solution provided the tangential condition found in the viability theorem is satisfied.

4 Global Nonterminating Solutions

A global nonterminating solution to a controlled hybrid automaton is a hybrid trajectory that exists over $[\tau_0, \infty)$ and that generates an infinite number of switching instants. This section studies the existence of global non-terminating periodic trajectories for the cyclic controlled hybrid automaton in figure 1. This hybrid automaton consists of two nodes (1 and 2) and two arcs. The i th node is labeled with the system (A_i, B_i) and arcs (1, 2) and (2, 1) are labeled with vertex collections $\mathbf{V}_1 = \{v_{11}, v_{12}, \dots, v_{1p_1}\}$ and $\mathbf{V}_2 = \{v_{21}, v_{22}, \dots, v_{2p_2}\}$, respectively. The i th guard associated with the arc entering the i th node denoted as $\Gamma_i = \overline{co}(\mathbf{V}_i)$.

Consider one of the modal systems ($i = 1$ or 2)

$$\dot{x}(t) = A_i x(t) + B_i u(t) \tag{3}$$

where $x(t) \in \mathbb{R}^n$, $A_i \in \mathbb{R}^{n \times n}$, $B_i \in \mathbb{R}^n$, and $u(t) \in \mathbb{R}$. We say a state $v \in \mathbb{R}^n$ is *reachable* from a state $w \in \mathbb{R}^n$ if there exists a time $T > 0$ and a measurable control $u : [\tau_0, \tau_0 + T] \rightarrow U$ such that the controlled trajectory $x : [\tau_0, \tau_0 + T] \rightarrow X$ satisfies equation 3 with $x(\tau_0) = w$ and $x(\tau_0 + T) = v$. The set of all points from which v is reachable is called the *preset* of v and will be denoted as $\text{pre}(v)$. The preset of a subset $\Gamma \subset X$ is denoted as $\text{pre}(\Gamma)$ and is defined by the equation $\text{pre}(\Gamma) = \bigcup_{v \in \Gamma} \text{pre}(v)$. A necessary and sufficient condition [Ant97] for w to lie in the preset of v is that there exist a $T > 0$ such that

$$e^{A_i T} w - v \in \mathcal{R}(\mathcal{C}_i) \tag{4}$$

where

$$\mathcal{C}_i = [B_i \ A_i B_i \ A_i^2 B_i \ \dots \ A_i^{n-1} B_i] \tag{5}$$

is called the *controllability matrix* for the i th modal system. The range space of \mathcal{C}_i is denoted as $\mathcal{R}(\mathcal{C}_i)$ and we assume it has a dimension of r_i . In the following discussion, \mathbf{E}_i is a matrix of dimension $n \times r_i$ ($i = 1, 2$) whose columns are standard basis vectors for the subspace $\mathcal{R}(\mathcal{C}_i)$.

Remark: Note that this reachability condition applies when the control u can be unbounded (as is the case in so-called impulsive controls).

Remark: Note that the term *reachability* is used in a somewhat different sense than what is found in traditional algorithmic verification [Alu95]. Traditional hybrid automata have homogeneous modal equations and as a result $\text{pre}(v)$ for a fixed transition time T consists of a single point. In view of equation

4, it is apparent that the introduction of the control extends the preset of v to a set formed from affine varieties of the controllability subspace.

We assume an initial condition $x_0 \in X$ and $i_0 = 1$. The question to be answered is whether there exists a control u and a pair of switching times T_1 and T_2 such that a hybrid trajectory z is a solution of the system over the interval $[\tau_0, \infty)$ and such that z generates an infinite sequence of switching instants

$$\tau_0, \tau_{11}, \tau_{21}, \tau_{21}, \tau_{22}, \dots, \tau_{ij}, \dots$$

where τ_{ij} is the j th switching instant out of mode i , $\tau_{2j} - \tau_{1j} = T_1$, and $\tau_{1,j+1} - \tau_{2j} = T_2$. In other words, the hybrid trajectory z is nonterminating and periodic in time.

By our definition of a solution to a controlled hybrid automaton, we know that the continuous state at each switching instant must lie in the appropriate guard set. In other words $x(\tau_{ij}) \in \Gamma_i$ for all j and $i = 1, 2$. Since the guards are convex polytopes, the switching instants $x(\tau_{ij})$ can be represented as convex combination of the form

$$x(\tau_{1j}) = \sum_{i=1}^{p_1} \lambda_{1i} v_{1i}$$

$$x(\tau_{2j}) = \sum_{i=1}^{p_2} \lambda_{2i} v_{2i}$$

where $\lambda_{ij} \geq 0$ for $i = 1, 2$ and all j and where $\sum_{i=1}^{p_j} \lambda_{ij} = 1$ for $j = 1, 2$. Therefore if we are to have a nonterminating behavior, we know that $x(\tau_{2j})$ must be reachable from $x(\tau_{1j})$ in time T_1 and $x(\tau_{1,j+1})$ is reachable from $x(\tau_{2j})$ in time T_2 . From equation 4, this condition is satisfied if there exist vectors $\bar{\beta}_1 = [\beta_{11}, \beta_{12}, \dots, \beta_{1r_1}]^T$ and $\bar{\beta}_2 = [\beta_{21}, \beta_{22}, \dots, \beta_{2r_2}]^T$ such that

$$0 = \sum_{i=1}^{r_1} \beta_{1i} e_{1i} + e^{A_1 T_1} \sum_{i=1}^{p_1} \lambda_{1i} v_{1i} - \sum_{i=1}^{p_2} \lambda_{2i} v_{2i} \tag{6}$$

$$0 = \sum_{i=1}^{r_2} \beta_{2i} e_{2i} - \sum_{i=1}^{p_1} \lambda_{1i} v_{1i} + e^{A_2 T_2} \sum_{i=1}^{p_2} \lambda_{2i} v_{2i} \tag{7}$$

$$1 = \sum_{i=1}^{p_1} \lambda_{1i} \tag{8}$$

$$0 \geq \lambda_{1i} \quad , (i = 1, \dots, p_1) \tag{9}$$

$$1 = \sum_{i=1}^{p_2} \lambda_{2i} \tag{10}$$

$$0 \geq \lambda_{2i} \quad , (i = 1, \dots, p_2) \tag{11}$$

We reframe equations 6, 7, 8, and 10 as the matrix vector equation

$$c = S\bar{\eta} \tag{12}$$

$$\begin{bmatrix} \mathbf{0}_{2n \times 1} \\ 1 \\ 1 \end{bmatrix} = \left[\begin{array}{cc|cc} \mathbf{E}_1 & \mathbf{0}_{n \times r_2} & e^{A_1 T_1} \mathbf{V}_1 & -\mathbf{V}_2 \\ \mathbf{0}_{n \times r_1} & \mathbf{E}_2 & -\mathbf{V}_1 & e^{A_2 T_2} \mathbf{V}_2 \\ \hline \mathbf{0}_{1 \times r_1} & \mathbf{0}_{1 \times r_2} & \mathbf{1}_{1 \times p_1} & \mathbf{0}_{1 \times p_2} \\ \mathbf{0}_{1 \times r_1} & \mathbf{0}_{1 \times r_2} & \mathbf{0}_{1 \times p_1} & \mathbf{1}_{1 \times p_2} \end{array} \right] \begin{bmatrix} \bar{\beta}_1 \\ \bar{\beta}_2 \\ \lambda_1 \\ \lambda_2 \end{bmatrix} \tag{13}$$

$$= [\mathbf{G}^T | \mathbf{F}^T] \begin{bmatrix} \mathbf{z} \\ \mathbf{y} \end{bmatrix} \tag{14}$$

where $\mathbf{V}_1 = [v_{11}, v_{12}, \dots, v_{1p_1}]$, $\mathbf{V}_2 = [v_{21}, v_{22}, \dots, v_{2p_2}]$ (matrices whose columns are the guard vertices), $\mathbf{c} = [\mathbf{0}_{n \times 1} \mid \mathbf{1}_{2 \times 1}]^T$, $\bar{\eta} = [\mathbf{z}^T, \mathbf{y}^T]^T$, $\mathbf{z} = [\beta_1, \dots, \beta_r]^T$, and $\mathbf{y} = [\bar{\lambda}_1, \bar{\lambda}_2]^T$.

Remark: The vectors $\bar{\lambda}_1$, $\bar{\lambda}_2$, $\bar{\beta}_1$, and $\bar{\beta}_2$ satisfying equations 6 to 11 characterize affine spaces which are mutually reachable from each other. Note that these solutions provide an explicit characterization of mutually reachable presets in terms of the vertices of the guards. This explicit representation of the presets of the system is the reason why the guards were represented as convex combinations of vertices.

By the theorem of the alternative [Baz93], a necessary and sufficient condition for equations 6 to 11 to have a non-negative solution is that there exist no vector \mathbf{x} such that

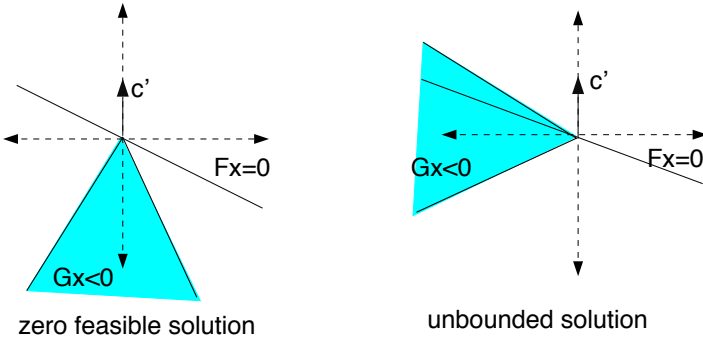
$$\mathbf{G}\mathbf{x} \leq 0, \quad \mathbf{F}\mathbf{x} = 0, \quad \mathbf{c}^T \mathbf{x} > 0 \tag{15}$$

The solution to equation 15 can be checked by solving the associated linear program

$$\begin{aligned} & \text{maximize : } \mathbf{c}^T \mathbf{x} \\ & \text{subject to : } \mathbf{G}\mathbf{x} \leq 0 \\ & \mathbf{F}\mathbf{x} = 0 \end{aligned} \tag{16}$$

Solutions to the above problem have a special form due to the fact that $\mathbf{G}\mathbf{x}$ forms a polytopic cone whose apex is at the origin. Figure 2 shows the possible situations that can occur with this linear program. The figure shows that solutions to this linear program are either unbounded and positive or bounded and equal to zero. If the solution is $\mathbf{x} = 0$, then the alternative problem in equation 15 has no solution since $\mathbf{c}^T \mathbf{x} = 0$. This means that equation 12 has a non-negative solution and we can infer that for the fixed time T that the guard Γ is reachable from $\overline{\text{co}}(\mathbf{W})$. If an unbounded solution occurs then equation 12 has no non-negative solutions and we can infer that for the given T , the guard Γ is not directly reachable from $\overline{\text{co}}(\mathbf{W})$.

Remark: A feasible solution at $\mathbf{x} = 0$ implies that the specified cycle exists between the two guard sets and an unbounded solution implies that a cycle does not exist with the specified transition times T_1 and T_2 . Note that the existence of an unbounded solution does not imply that the guards don't support a recurrent cycle, for there may be other transition times for which the cycle exists and it may be possible that the guards support a cycle in which the transitions are not necessarily periodic.



The alternative problem's linear program must have a solution lying in the null space of F will either be unbounded or lie at the apex of the cone formed by the equation $Gx < 0$.

Fig. 2. The Alternative Problem's Linear Program

If the linear program in equation 16 returns an unbounded solution, then it may be possible to adjust the transition times T_1 and T_2 to force a solution at $\mathbf{x} = 0$. From duality theory [Baz93], we know that if the primal problem in equation 16 is unbounded, then its dual is infeasible. The infeasibility of the dual can be readily checked by examining the Lagrange multipliers associated with the inequality constraints of the primal problem. These Lagrange multipliers are generated by any primal-dual linear programming algorithm. They represent part of the solution to the dual problem and are used to help assess how close a linear programming algorithm is to being finished. If these multipliers are negative, then the dual is infeasible and we can immediately conclude that the guard does not support a cycle at the specified transition times.

The preceding observation suggests a simple heuristic method for adjusting the times T_1 and T_2 in order to force the dual problem to be feasible. Let ν_k denote the k th Lagrange multiplier associated with the linear program's inequality constraints. We define a performance measure associated with a specific pair of times (T_1, T_2) as

$$J((T_1, T_2)) = \min_k \nu_k \tag{17}$$

This measure identifies the smallest Lagrange multiplier and uses it as a measure of how close the dual problem is to being feasible. The obvious strategy is to perturb the current transition times T_1 and T_2 , observe the change in J and then select a new set of times that will increase J . We continue in this manner until J becomes positive.

This idea was tested using the following, very simple, search strategy. First initialize the search by selecting a set of times T_1 and T_2 . The search is then executed by the following steps.

1. Perturb (T_1, T_2) by a small adjustment $\delta > 0$ and solve the linear program (equation 16) for points (T_1, T_2) , $(T_1, T_2 + \delta)$, and $(T_1 + \delta, T_2)$, .
2. If any of these linear programs are feasible, then the system supports a periodic solution and we're finished.
3. If all of these linear programs are infeasible, then the Lagrange multipliers for each problem are used to compute costs $J((T_1, T_2))$, $J((T_1 + \delta, T_2))$, and $J((T_1, T_2 + \delta))$.
4. Select a new set of times, (T'_1, T'_2) according to the following rule,

$$T'_1 = \begin{cases} T_1 + \delta, & \text{if } J((T_1, T_2)) < J((T_1 + \delta, T_2)) \\ T_1 - \delta, & \text{if } J((T_1, T_2)) > J((T_1 + \delta, T_2)) \\ T_1, & \text{otherwise} \end{cases}$$

$$T'_2 = \begin{cases} T_2 + \delta, & \text{if } J((T_1, T_2)) < J((T_1, T_2 + \delta)) \\ T_2 - \delta, & \text{if } J((T_1, T_2)) > J((T_1, T_2 + \delta)) \\ T_2, & \text{otherwise} \end{cases}$$

5. Set $T_1 = T'_1$, $T_2 = T'_2$ and return to step 1.

What this algorithm does is attempt to solve a nonlinear optimization problem using a gradient-following strategy. The preceding steps describe the master algorithm that uses the results of the linear program in equation 16 to select a set of better times.

Remark: In the procedure we've chosen, of course, there are no guarantees that this search will terminate as it is currently unclear how the times, T_1 and T_2 are related to the problem's Lagrange multipliers. Nonetheless, this search program provides what seems to be a very pragmatic method for testing for the existence of global solutions and if it does terminate, then we know for certain that the cycle is live.

The following example illustrates the proposed search algorithm. Consider the cyclic hybrid automaton shown in figure 1 where the nodes are labeled as

$$\ell_N(1) = \left(\begin{bmatrix} 0 & 4 \\ 1/4 & 0 \end{bmatrix}, \begin{bmatrix} 4 \\ 1 \end{bmatrix} \right)$$

$$\ell_N(2) = \left(\begin{bmatrix} 0 & -10 \\ -1/10 & 0 \end{bmatrix}, \begin{bmatrix} 10 \\ 1 \end{bmatrix} \right)$$

The arcs are labeled with vertex collections

$$\mathbf{V}_1 = \left\{ \begin{bmatrix} -2 \\ 0 \end{bmatrix}, \begin{bmatrix} -3 \\ 0 \end{bmatrix}, \begin{bmatrix} -3 \\ 1 \end{bmatrix} \right\}$$

$$\mathbf{V}_2 = \left\{ \begin{bmatrix} 2 \\ 0 \end{bmatrix}, \begin{bmatrix} 3 \\ 0 \end{bmatrix}, \begin{bmatrix} 3 \\ 1 \end{bmatrix} \right\}$$

A MatLab script was written to implement the master program given above and this script was used to search for a global non-terminating solution of our hybrid automaton. The lefthand plot in figure 3 illustrates the results of this search. The x -axis shows the times T_1 and T_2 whereas the y -axis shows the value of the

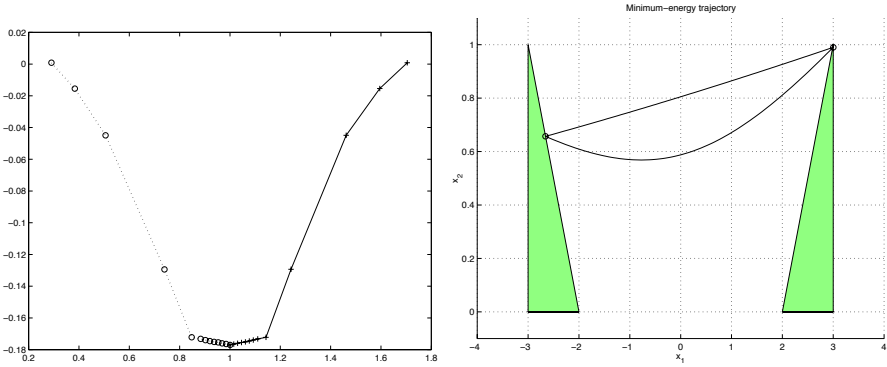


Fig. 3. Performance Measure J versus Switching Intervals and Simulation Results

cost $J((T_1, T_2))$. The search starts with $T_1 = T_2 = 1$. This starting point is in the middle of the x -axis. The intermediate values of J computed by the master algorithm are shown by the solid and dotted line trajectories (the dotted line for T_1 and the solid line for T_2). We see that the master algorithm computes a monotone sequence of times in which T_2 is decreasing and T_1 is increasing. After a finite number of iterations the master program has identified the times $T_2 = 0.29$ and $T_1 = 1.7$ as points whose linear programs have non-negative Lagrange multipliers. These points, therefore, are feasible and characterize a global non-terminating periodic cycle for this system.

The master algorithm allows us to assert that a global periodic solution to this system exists. The intermediate results of the algorithm also allow us to characterize the switching sets and we can actually identify some of the control strategies, u , that enforce this periodic solution. This additional information is contained in all non-negative solution vectors $\bar{\lambda}_1$ and $\bar{\lambda}_2$ satisfying our system $\mathbf{S}\bar{\eta} = \mathbf{c}$. The set of all solutions can be parameterized as $\bar{\eta} \in \bar{\eta}_{p0} + \text{null}(\mathbf{S})$ where η_{p0} is a particular solution to the inhomogeneous equation $\mathbf{c} = \mathbf{S}\bar{\eta}$. Note that this implies that the mutually reachable sets in the guards are affine sets. It was our parametrization of the guard as a convex combination of vertices that allowed us to obtain such a simple and explicit representation of these sets. For the example above, we can readily identify these sets in which the particular solution is $\bar{\eta}_{p0} = [0 \ -1.94 \ 1.39 \ -1.10 \ 0.71 \ 0 \ 0.048 \ 0.95]^T$ and the null space of \mathbf{S} is spanned by the columns of the matrix

$$N = \begin{bmatrix} -0.9237 & 0 \\ 0.1380 & -0.4162 \\ 0.1820 & -0.3445 \\ -0.2321 & 0.1920 \\ 0.0501 & 0.1525 \\ -0.1536 & -0.6431 \\ 0.1144 & 0.4389 \\ 0.0393 & 0.2042 \end{bmatrix}$$

Let's now look at the controls required to enforce the nonterminating cycle. We first look at a pair of specific switching points in the guards and then identify an open loop control enforcing a periodic trajectory between these points. One specific set of switching points for our example system is

$$\begin{aligned} x(\tau_1) &= \mathbf{V}_1 \bar{\lambda}_1 = [-2.6569 \ 0.6569] \\ x(\tau_2) &= \mathbf{V}_2 \bar{\lambda}_2 = [3 \ 0.9903] \end{aligned}$$

The existence of a control driving the system between these two points is guaranteed by the termination of our master program. What is this open loop control? We have many choices and one obvious choice is the minimum energy control strategy. The minimum energy control $u(t)$ that transfers the first modal system from the initial state $x(\tau_1)$ to target state $x(\tau_2)$ satisfies the condition

$$x(\tau_2) - e^{A_1 T_1} x(\tau_1) \in \mathcal{R}(\mathcal{C}_1)$$

is given by

$$u_1(t) = B_1^T e^{A_1^T (T_1 - t)} \eta_1$$

where η_1 is the solution of the equation

$$W_1(0, T_1) \eta_1 = x(\tau_2) - e^{A_1 T_1} x(\tau_1)$$

$W_1(0, T_1)$ is the controllability Gramian of (A_1, B_1) . For the system at hand the solution is

$$u_1(t) = \frac{2e^{T_1}}{e^{2T_2} - 1} \beta_1 e^{-t} = 0.3349e^{-t}$$

Similarly the minimum energy solution for the second mode is

$$u_2(t) = \frac{2e^{-T_2}}{1 - e^{-2T_2}} \beta_2 e^t = -0.9758e^t$$

The hybrid automaton's trajectory with this minimum-energy control is shown in the righthand plot in figure 3. This figure shows the state space for our system. The two triangular regions in this plot represent the guards. For the specific choice of points $x(\tau_1)$ and $x(\tau_2)$, we use the control $u(t)$ identified above to compute the state trajectory between these points. The solid line in figure 3 shows the resulting controlled trajectory.

It is, of course, possible to obtain other controls realizing this cycle. For instance, an "impulsive" control strategy can be employed, in which we impulsively drive the system state along an affine variety of the controllable subspace and then allow the system to relax into the guard. (In other words we let $u(t)$ be an impulse function of specified magnitude). The lefthand plot of figure 4 illustrates the state trajectory generated by this control law. As in figure 3, we are looking at the system's state space. The triangular regions represent the guards and the solid lines denote the state trajectory generated by the impulsive control.

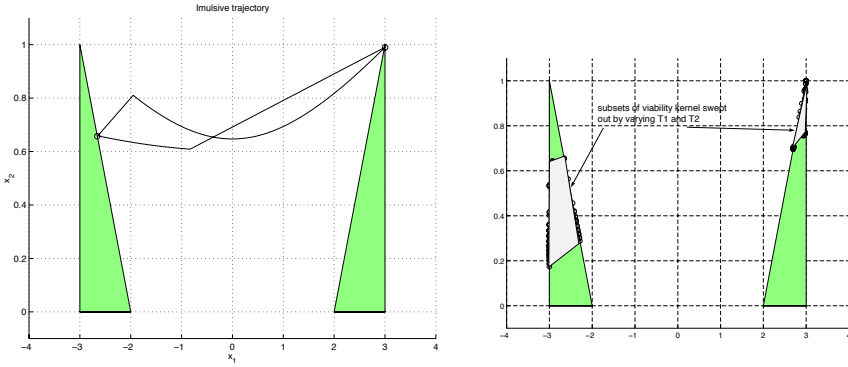


Fig. 4. Impulsive simulation results and sweeping out the viability kernel

For a specific pair of times, our master algorithm identifies all states in the guards that are mutually reachable from each other. If we were able to identify a *range* of feasible switching times, then it should be possible to identify larger subsets of the guard that are mutually reachable from each other. The complete set of states in the guards that are mutually reachable from each other (under any control strategy) is sometimes referred to as the *viability kernel*. Our algorithm, therefore, provides a means of approximating the viability kernel. Note that this is an under approximation to the viable set (as opposed to the over-approximation computed by model checking algorithms [Alu95]). For our specific example, we were able to identify a range of times over which periodic solutions could be guaranteed. This range was computed to be $1.7 < T_1 < 3.75$ and $.1059 < T_2 < .29$. The set of points swept out by these various times is shown in the righthand plot of figure 4. We’ve compared this set to the actual viability kernel for this system and the specified bounds appear to provide a close approximation to the actual viability kernel.

Remark: The failure of the master program to find any feasible solution does not guarantee that a global solution doesn’t exist. How quickly we find a feasible solution clearly depends upon the type of search strategy the master program uses and depends on our initial guess.

Remark: Our approach focuses on identifying *periodic* global solutions and obviously it may be possible that this is overly restrictive. For instance, it may be possible that only chaotic trajectories exist between the two guards, or that a more complex periodic behavior exists between the two guards.

Remark: The preceding discussion focused on establishing non-terminating solutions to a rather simple hybrid automaton. This problem was chosen as a *canonical problem* in the sense that its solution may provide a foundation upon which to establish the existence of global solutions to more complex systems. How might this be done? This is the topic of another paper, but we can speculate on a possible strategy based on prior results on the role of cycles in hybrid automata [He98] [Zhi98]. Essentially, the argument runs as follows. From the pumping

lemma, we know that the logical behavior generated by any automaton can be broken down into a concatenation of *fundamental cycles*. This paper, essentially, is proposing a pragmatic way for determining whether a given fundamental cycle is viable. Moreover, our algorithm computes an under-approximation to the cycle's viability kernel that can be very good (as shown in our example). Let's assume we can determine controls guaranteeing all fundamental cycles are viable. Given a specific concatenation of cycles in the system, we then look at the intersection of viable sets of contiguous cycles (actually look at the approximations computed using the methods in this paper). If this intersection is non-empty, it should be possible to determine control strategies enforcing the viability of arbitrary concatenations of fundamental cycles and thereby ensure the viability of the entire complex system. As noted above, whether or not this approach will work is still under study.

5 Conclusions

Controlled hybrid automata are automata in which a user-determined input control signal can be used to help supervise overall system behavior. In this paper, we assumed the modal systems were linear and time invariant with polytopic guards formed from the convex combination of vertices. This paper studied the existence of solutions to this class of hybrid system. A routine application of Viability theory was used to characterize the existence of local trajectories. This paper presented a necessary and sufficient condition for the existence of a global periodic non-terminating trajectory with specified switching intervals. This result was used to propose a gradient following search strategy for determining a set of switching intervals ensuring a global nonterminating trajectory. The proposed method also provides an under-approximation of the cycle's viability kernel that could be used in extending this work to more complex switching systems. A distinguishing feature of this study is the explicit use of the open loop control signal $u(t)$ as a means of enforcing a cycle's viability.

This work is preliminary in that there are still a number of open questions that need to be answered. There is uncertainty over the performance of the proposed search algorithm. It should be noted, however, that such gradient following heuristics often work extremely well on real-life problems, so this approach may still be a pragmatic approach to hybrid system verification. Another open issue concerns the conservatism imposed by confining our search to periodic non-terminating solutions. While this might appear to be very restrictive on the surface, it must be realized that the proposed approach can actually identify a *set* of periodic solutions and that other non-periodic solutions might be seen as limiting points of this set. Another interesting issue brought up by this paper is the explicit use of control. Traditional analyses of hybrid systems assume no control and the verification process can be seen as a "take it or leave it" analysis that provides little guidance on determining how "close" a system is to being viable. The use of control advocated in this paper may provide the system designer with a more sophisticated approach to verification in which control becomes a

necessary component in system design. Finally, this paper has focused on hybrid systems containing only one cycle. This simple problem is viewed as a necessary starting place for the analysis of more complex hybrid systems and the details of this later analysis will be the subject of future papers.

References

- Alu93. R. Alur, C. Courcoubetis, T.A. Henzinger, and P.-H. Po, "Hybrid Automata: an Algorithmic Approach to the Specification and Verification of Hybrid Systems", Robert L. Grossman, Anil Nerode, Anders P. Ravn, and Hans Rischel, editors, *Hybrid Systems*, Lecture Notes in Computer Science, vol. 736, Springer-Verlag, pp. 209-229, 1993.
- Alu95. R. Alur, C. Courcoubetis, Halbwachs, T.A. Henzinger, P-H Ho , X Nicollin, Olivero, J. Sifjakis, and S. Yovine, "The Algorithmic Analysis of Hybrid Systems", *Theoretical Computer Science* Vol. 138:, pp. 3-34, 1995
- Ant97. P.J. Antsaklis, A.N. Michel, *Linear Systems*, MacGraw-Hill, 1997.
- Aub84. J.P. Aubin and A. Cellina, *Differential Inclusions*, Springer Verlag, 1984.
- Baz93. M. Bazaraa, H.D. Sherali, and C.M. Shetty, *Nonlinear Programming: theory and algorithms*, 2nd Edition, John-Wiley, 1993.
- Dav83. M.D. Davis and E.J. Weyuker, *Computability, Complexity, and Languages*, Academic Press, 1983.
- He98. K.X. He and M.D. Lemmon, Lyapunov stability of continuous valued systems under the supervision of discrete event transition systems, in *Proceedings of Hybrid Systems: Control and Computation*, lecture notes in computer science Vol. 1386, Springer Verlag, 1998.
- Lyn96. N. Lynch, R. Segala, F. Vaandrager, and H.B. Weinberg, "Hybrid I/O Automata", In R. Alur, T. Henzinger, and E. Sontag, editors, *Hybrid Systems III: Verification and Control* (DIMACS/SYCON Workshop on Verification and Control of Hybrid Systems), New Brunswick, New Jersey, October 1995), volume 1066 of Lecture Notes in Computer Science, pages 496-510. Springer-Verlag 1996.
- Zhi98. P. Zhivoglyadov and R.H. Middleton, On Stability in Hybrid Systems, Proceedings off 37th IEEE Conference on Decision and Control, Tampa Florida, USA, pp. 3687-3692, December 1998.