Proceedings of the 8th IEEE Mediterranean Conference
on Control and Automation (MED 2000)
Rio, Patras, GREECE 17-19 July, 2000

Fundamental Properties of Hybrid Dynamical Systems [1]

John Lygeros, Karl Henrik Johansson, Jun Zhang, and Shankar Sastry
Department of Electrical Engineering and Computer Sciences
University of California, Berkeley, CA 94720-1770
{johans,lygeros,zhangjun,sastry}@eecs.berkeley.edu

Abstract

Basic dynamical features of hybrid systems are reviewed in this paper. Some results on existence and uniqueness of executions for hybrid automata are obtained. Continuous dependence on initial states are shown for a class of hybrid automata. Zeno hybrid automata, i.e., hybrid automata that exhibit infinitely many discrete transitions in finite time, are also discussed.

1 Introduction

The importance of systems with interacting digital and analog computations is increasing dramatically. Areas such as aeronautics, automotive vehicles, bioengineering, embedded software, process control, and transportation are growing tremendously [14, 2, 3, 7, 8, 16]. There is a large number of new applications, where computers are coupled to physical environment. This has led to a need for better understanding of the behavior of these hybrid systems with linked continuous-time and discrete-time dynamics, in order to guarantee design performance.

Hybrid automata have proved to be an efficient way to model systems with both continuous and discrete dynamics. Their rich structure allow them to accurately predict the behavior of quite complex systems. Based on computer science and control theory, tools are now evolving for analyzing and designing hybrid systems within the hybrid automata framework. The work presented in the paper is part of this activity.

The main contribution of the paper is to present some results on the fundamental properties of hybrid automata. We investigate the existence and uniqueness of executions of hybrid automata. Although such results form the foundation for analysis and design methods, these problems have only recently been addressed [15, 9]. Continuous dependence on initial conditions is another important issue in the analysis of hybrid automata [4, 13]. It is for instance often a desirable property in computer simulations. In the paper, we show a new result in this area. Zeno hybrid systems are systems that exhibit infinitely many discrete transitions in a finite time interval. This type of behavior does only occur in systems with interacting continuous and discrete dynamics. Physical systems are of course not Zeno, but a model of a physical system may however be Zeno due to a too high level of abstraction. It is therefore important to characterize Zeno hybrid automata and in applicable cases modify the model to avoid Zenoness [5]. In the paper, we are able to completely characterize the set of Zeno states for a couple of quite broad classes of Zeno hybrid automata.

The outline of the paper is as follows. Section 2 introduces notation and the definitions of hybrid automata and executions. Some recent results on existence and uniqueness of executions for classes of hybrid automata are given in Section 3. A result on continuous dependence on initial conditions is presented in Section 4. Finally, Zeno hybrid automata are discussed in Section 5 and some conclusions are given in Section 6.

2 Hybrid Automata and Executions

The following definitions are based on [10, 5, 18]. For a finite collection V of variables, let $\mathbf{V}$ denote the set of valuations of these variables. We use lower case letter to denote both a variable and its valuation. We refer to variables whose set of valuations is finite or countable as *discrete* and to variables whose set of valuations is a subset of a Euclidean space as *continuous*. For a set of continuous variables X with $\mathbf{X} = \mathbb{R}^n$ for $n \geq 0$, we assume that $\mathbf{X}$ is given the Euclidean metric topology, and use $\|\cdot\|$ to denote the Euclidean norm. For a set of discrete variables Q, we assume that $\mathbf{Q}$ is given the discrete topology (every subset is an open set), generated by the metric $d_D(q,q') = 0$ if $q = q'$ and $d_D(q,q') = 1$ if $q \neq q'$. We denote the valuations of the union $Q \cup X$ by $\mathbf{Q} \times \mathbf{X}$, which is given the product topology generated by the metric $d((q,x),(q',x')) = d_D(q,q') + \|x - x'\|$. We assume that a subset U of a topological space is given the induced topology, and we use $\overline{U}$ to denote its

[1]This work was supported by ARO under the MURI grant DAAH04-96-1-0341, the Swedish Foundation for International Cooperation in Research and Higher Education, Telefonaktiebolaget LM Ericsson's Foundation, ONR under grant N00014-97-1-0946, and DARPA under contract F33615-98-C-3614.

closure, U^o its interior, $\partial U = \overline{U} \setminus U^o$ its boundary, U^c its complement, $|U|$ its cardinality, and $P(U)$ the set of all subsets of U.

Definition 1 (Hybrid Automaton)
A hybrid automaton H is a collection $H = (Q, X, \mathrm{Init}, f, \mathrm{Dom}, \mathrm{Reset})$, where

- *Q is the finite collection of discrete variables with values in $\mathbf{Q}$;*
- *X is the finite collection of continuous variables with values in $\mathbf{X} = \mathbb{R}^n$;*
- *$\mathrm{Init} \subseteq \mathbf{Q} \times \mathbf{X}$ is the set of initial states;*
- *$f : \mathbf{Q} \times \mathbf{X} \to T\mathbf{X}$ is the vector field;*
- *$\mathrm{Dom} \subseteq \mathbf{Q} \times \mathbf{X}$ is the domain of H;*
- *$\mathrm{Reset} : \mathbf{Q} \times \mathbf{X} \to P(\mathbf{Q} \times \mathbf{X})$ is the reset relation.*

We refer to $(q, x) \in \mathbf{Q} \times \mathbf{X}$ as the *state* of H. We make the standing assumption that $|\mathbf{Q}| < \infty$ and that f is Lipschitz continuous in its second argument. A hybrid automaton can be represented by a directed graph $(\mathbf{Q}, E)$, with vertices $\mathbf{Q}$ and edges $E = \{(q, q') \in \mathbf{Q} \times \mathbf{Q} : \exists x, x' \in \mathbf{X}, (q', x') \in \mathrm{Reset}(q, x)\}$. With each vertex $q \in \mathbf{Q}$, we associate a set of continuous initial states $\mathrm{Init}(q) = \{x \in \mathbf{X} : (q, x) \in \mathrm{Init}\}$, a vector field $f(q, \cdot)$, and a set $D(q) = \{x \in \mathbf{X} : (q, x) \in \mathrm{Dom}\}$. With each edge $e = (q, q') \in E$, we associate a guard $G(e) = \{x \in \mathbf{X} : \exists x' \in \mathbf{X}, (q', x') \in \mathrm{Reset}(q, x)\}$, and a reset map $R(e, x) = \{x' \in \mathbf{X} : (q', x') \in \mathrm{Reset}(q, x)\}$.

Definition 2 (Hybrid Time Trajectory)
A hybrid time trajectory τ is a finite or infinite sequence of intervals $\{I_i\}_{i=0}^N$, such that

- *$I_i = [\tau_i, \tau_i']$ for $i < N$, and, if $N < \infty$, $I_N = [\tau_N, \tau_N']$ or $I_N = [\tau_N, \tau_N')$,*
- *$\tau_i \leq \tau_i' = \tau_{i+1}$ for $i \geq 0$.*

Hybrid time trajectories can extend to infinity if τ is an infinite sequence or if it is a finite sequence ending with an interval of the form $[\tau_N, \infty)$. Since the hybrid automaton is time invariant, we assume that $\tau_0 = 0$.

For a hybrid time trajectory $\tau = \{I_i\}_{i=0}^N$, let $\langle \tau \rangle$ denote the set $\{0, 1, \ldots, N\}$ if N is finite and $\{0, 1, \ldots\}$ if N is infinite. We use q and x to denote the time evolution of the discrete and continuous state, respectively (with a slight abuse of notation). Here q is a map from $\langle \tau \rangle$ to $\mathbf{Q}$ and $x = \{x^i : i \in \langle \tau \rangle\}$ is a collection of C^1 maps. An execution is now defined as a triple $\chi = (\tau, q, x)$ in the following way.

Definition 3 (Execution)
An execution of a hybrid automaton H is a collection $\chi = (\tau, q, x)$ with τ being a hybrid time trajectory, $q : \langle \tau \rangle \to \mathbf{Q}$ a map, and $x = \{x^i : i \in \langle \tau \rangle\}$ a collection of C^1 maps $x^i : I_i \to D(q(i))$, such that

- *$(q(0), x^0(0)) \in \mathrm{Init}$,*
- *for all $t \in I_i$, $\dot{x}^i(t) = f\big(q(i), x^i(t)\big)$,*
- *for all $i \in \langle \tau \rangle$, $e = (q(i), q(i+1)) \in E$, $x^i(\tau_i') \in G(e)$, and $x^{i+1}(\tau_{i+1}) \in R(e, x^i(\tau_i'))$.*

We say a hybrid automaton *accepts* an execution χ or not. The *execution time* $\mathcal{T}(\chi)$ is defined as $\mathcal{T}(\chi) = \sum_{i=0}^N (\tau_i' - \tau_i) = \tau_N'$, where $N + 1$ is the number of intervals in the hybrid time trajectory. The argument χ will sometimes be left out. An execution is *finite* if τ is a finite sequence ending with a compact interval, it is called *infinite* if τ is either an infinite sequence or if $\mathcal{T}(\chi) = \infty$, and it is called *Zeno* if it is infinite but $\mathcal{T}(\chi) < \infty$. The execution time of a Zeno execution is called the *Zeno time*. We use $\mathcal{E}_H(q_0, x_0)$ to denote the set of all executions of H with initial state $(q_0, x_0) \in \mathrm{Init}$, $\mathcal{E}_H^M(q_0, x_0)$ to denote the set of all maximal executions (i.e., executions that are not strict prefix of any other executions [9]), and $\mathcal{E}_H^\infty(q_0, x_0)$ to denote the set of all infinite executions. We define $\mathcal{E}_H = \bigcup_{(q_0, x_0) \in \mathrm{Init}} \mathcal{E}_H(q_0, x_0)$ and $\mathcal{E}_H^\infty = \bigcup_{(q_0, x_0) \in \mathrm{Init}} \mathcal{E}_H^\infty(q_0, x_0)$. To simplify the notation, we will drop the subscript H whenever the automaton is clear from the context.

3 Existence and Uniqueness

The notation previously introduced gives a convenient way to express existence and uniqueness of executions.

Definition 4 (Non-Blocking)
A hybrid automaton H is non-blocking if $\mathcal{E}_H^\infty(q_0, x_0)$ is non-empty for all $(q_0, x_0) \in \mathrm{Init}$.

Definition 5 (Deterministic)
A hybrid automaton H is deterministic if $\mathcal{E}_H^M(q_0, x_0)$ contains at most one element for all $(q_0, x_0) \in \mathrm{Init}$.

Note that if a hybrid automaton is both non-blocking and deterministic, then it accepts a unique infinite execution for each initial state. In [9] conditions were established that determine whether an automaton is non-blocking and deterministic. These are reviewed next, but first we need to introduce some more notation.

The set of states reachable by H is defined as

$$\begin{aligned}\mathrm{Reach}_H = \{&(\hat{q}, \hat{x}) \in \mathbf{Q} \times \mathbf{X} : \exists\, \chi = (\tau, q, x) \in \mathcal{E}_H, \\ &\big(q(N), x^N(\tau_N')\big) = (\hat{q}, \hat{x}), N < \infty\}.\end{aligned}$$

Note that $\text{Reach}_H \supset \text{Init}$, since we may choose $\tau'_N = \tau_N$ and $N = 0$. Let $\phi(t, a)$ denote the conventional solution to the differential equation $\dot{x} = f(q, x)$ with $x(0) = a$. The set of continuous states from which continuous evolution is impossible when in discrete state q is then given by

$$\begin{aligned}\text{Out}_H =&\{(\bar{q}, \bar{x}) \in \mathbf{Q} \times \mathbf{X} : \forall \epsilon > 0,\ \exists t \in [0, \epsilon), \\ &(\bar{q}, \phi(t, \bar{x})) \notin \text{Dom}\},\end{aligned}$$

Note that if Dom is an open set, then Out is simply Dom^c. If Dom is closed, then Out may also contain parts of the boundary of Dom. In [9] methods for computing Out were proposed, under appropriate smoothness assumptions on f and the boundary of Dom. As before, we will use $\text{Out}_H(q)$ to denote the projection of Out to discrete state q, and drop the subscript H whenever the automaton is clear from the context. With these two pieces of notation one can show the following two results [9]. The first gives a condition that guarantees that there always exist infinite executions.

Proposition 1 (Non-Blocking)
A (deterministic) hybrid automaton is non-blocking if (and only if) for all $(q, x) \in \text{Out} \cap \text{Reach}$, $\text{Reset}(q, x) \neq \emptyset$.

Note that the condition is necessary and sufficient if the hybrid automaton is deterministic. A condition for determinism is given next.

Proposition 2 (Deterministic)
A hybrid automaton is deterministic if and only if for all $(q, x) \in \text{Reach}$, $|\text{Reset}(q, x)| \leq 1$ and, if $\text{Reset}(q, x) \neq \emptyset$, then $(q, x) \in \text{Out}$.

4 Continuous Dependence on Initial States

Continuity of solutions with respect to initial states is a desirable property of many dynamical systems. For a conventional continuous-time dynamical system, a Lipschitz condition on the vector field guarantees this property. For hybrid systems, however, it is not sufficient to require that the vector field in each discrete state is Lipschitz continuous. In this section, we show what extra assumptions that may be needed to guarantee continuous dependence on initial states. Continuity is interpreted in the metric $d\big((q, x), (q', x')\big) = d_D(q, q') + \|x - x'\|$.

We study a particular class of hybrid automata, which we refer to as having *transverse domain*. A hybrid automaton H is said to have transverse domain if there exists a function $\sigma : \mathbf{Q} \times \mathbf{X} \to \mathbb{R}$ continuously differentiable in its second argument, such that

$$\text{Dom} = \{(q, x) \in \mathbf{Q} \times \mathbf{X} :\ \sigma(q, x) \geq 0\}$$

and for all (q, x) with $\sigma(q, x) = 0$, $L_f\sigma(q, x) \neq 0$. Here $L_f\sigma : \mathbf{Q} \times \mathbf{X} \to \mathbb{R}$ denotes the Lie derivative of σ along f defined as $L_f\sigma(q, x) = \partial\sigma/\partial x(q, x) \cdot f(q, x)$. In other words, an automaton has transverse domain if the set Dom is closed, its boundary is differentiable, and the vector field f is pointing either inside or outside of Dom along the boundary. A hybrid automaton is called *domain preserving* if $\text{Reach} \subseteq \overline{\text{Dom}}$, i.e., if the states remain in the closure of the domains along all executions. The following example show a hybrid automaton that has transverse domain and is domain preserving.

Example 1
Consider the hybrid automaton

- $\mathbf{Q} = \{q_1, q_2\}$ and $\mathbf{X} = \mathbb{R}^2$;
- $\text{Init} = \{q_1\} \times \mathbb{R}^2$;
- $f(\cdot, \cdot) \equiv (1, 0)^T$;
- $\text{Dom} = \{(q_1, x) :\ x_1 \leq 0\} \cup \{(q_2, x) :\ x_1 \geq 0\}$;
-
$$\text{Reset}(q, x_1, x_2) = \begin{cases} (q_2, x_1, 1), & \text{if } q = q_1,\ x_1 \geq 0,\ x_2 > 0 \\ (q_2, x_1, 0), & \text{if } q = q_1,\ x_1 \geq 0,\ x_2 \leq 0 \\ \emptyset, & \text{otherwise.} \end{cases}$$

It is easy to check (for example, by using Propositions 1 and 2) that the hybrid automaton is deterministic and non-blocking, and has thus a unique infinite execution for every initial state. It shows, however, in general not continuous dependence on the initial state as illustrated next. Consider two executions $\chi = (\tau, q, x)$ and $\hat{\chi} = (\hat{\tau}, \hat{q}, \hat{x})$ with initial states $\big(q_1, (0, 0)\big)$ and $\big(q_1, (0, \epsilon)\big)$, respectively. For every $\epsilon > 0$, it holds that for $\langle\tau\rangle = \langle\tau'\rangle = \{0, 1\}$, $\|x_2^1(t) - \hat{x}_2^1(\hat{t})\| = 1$ for all $t \in I_1$ and $\hat{t} \in \hat{I}_1$.

The reason for the absence of continuous dependence in the example is of course due to the discontinuous reset relation. The following theorem gives sufficient conditions for continuous dependence on initial states for a class of hybrid automata. The result is proved in [17]. Some other work on continuity in hybrid systems can be found in [4, 13].

Theorem 1
Consider a deterministic hybrid automaton H and assume

- *H has transverse domain and is domain preserving;*
- *for all $q \in \mathbf{Q}$, $f(q, \cdot)$ is C^1;*

- *for all $e \in E$, $R(e, \cdot)$ is a continuous function;*
- *for all $e = (q, q') \in E$, $G(e) \cap D(q)$ is an open subset of $\partial D(q)$.*

Consider a finite execution $\chi = (\tau, q, x) \in \mathcal{E}_H(q_0, x_0)$ with $\tau = \{I_i\}_{i=0}^N$. For every $\epsilon > 0$ there exists $\delta > 0$ such that for all $(\tilde{q}_0, \tilde{x}_0) \in \mathrm{Init}$ with $d((\tilde{q}_0, \tilde{x}_0), (q_0, x_0)) < \delta$, there exists $T(\tilde{x}_0) > 0$ such that the execution $\tilde{\chi} = (\tilde{\tau}, \tilde{q}, \tilde{x}) \in \mathcal{E}_H(\tilde{q}_0, \tilde{x}_0)$ with $\tilde{\tau} = \{\tilde{I}_i\}_{i=0}^N$ and $\tilde{\tau}'_N = T(\tilde{x}_0)$ satisfies

- $|\mathcal{T}(\tilde{\chi}) - \mathcal{T}(\chi)| < \epsilon$;
- $d((\tilde{q}(N), \tilde{x}^N(\tilde{\tau}'_N)), (q(N), x^N(\tau'_N))) < \epsilon$.

Remark 1
The result says that for a given execution χ, any execution $\tilde{\chi}$ starting close enough to χ will stay close at the end point with some appropriate execution time. Note that for a given initial state and execution time, the execution $\tilde{\chi}$ is unique by assumption. Also note that it is in general not possible to guarantee the same execution time for $\tilde{\chi}$ and χ.

Remark 2
If there is only one discrete state and no reset relations, the hybrid automaton, of course, defines a continuous dynamical system. It is easy to see that all assumptions are satisfied for this case. By setting $N = 0$ and $\mathcal{T}(\tilde{\chi}) = \mathcal{T}(\chi)$, we obtain the traditional result of continuous dependence on initial states.

Note that Example 1 satisfies all conditions in Theorem 1 except for the requirement on continuous reset map. An example that violates the last assumption of the theorem is the following.

Example 2
Consider the hybrid automaton

- $\mathbf{Q} = \{q_1, q_2, q_3\}$ and $\mathbf{X} = \mathbb{R}^2$;
- $\mathrm{Init} = \{q_1\} \times \mathbb{R}^2$;
- $$f(q,x) = \begin{cases} (1,0)^T, & \text{if } q = q_1 \\ (1,1)^T, & \text{if } q = q_2 \\ (1,-1)^T, & \text{if } q = q_3; \end{cases}$$
- $$\begin{aligned} \mathrm{Dom} = &\{(q_1, x) : \ x_1 \le 0\} \\ &\cup \{(q_2, x) : \ x_2 \ge 0\} \cup \{(q_3, x) : \ x_2 \le 0\}; \end{aligned}$$
- $$\begin{aligned} &\mathrm{Reset}(q,x) \\ &= \begin{cases} (q_2, x), & \text{if } q = q_1,\ x_1 \ge 0,\ x_2 \ge 0 \\ (q_3, x), & \text{if } q = q_1,\ x_1 \ge 0,\ x_2 < 0 \\ \emptyset, & \text{otherwise.} \end{cases} \end{aligned}$$

Here $G(q_1, q_2) \cap D(q_1) = \{x \in \mathbb{R}^2 : \ x_1 = 0, \ x_2 \ge 0\}$, which is hence not an open subset of $\partial D(q_1) = \{x \in \mathbb{R}^2 : \ x_1 = 0\}$, so Theorem 1 is not applicable. To see that the hybrid automaton in general does not show continuous dependence on the initial state, consider initial states in a neighborhood of $(q_1, (0,0))$.

5 Zeno Hybrid Automata

Zeno hybrid automata accept executions with infinitely many discrete transitions within a finite time interval. Such systems are hard to analyze and simulate in a way that gives constructive information about the behavior of the real system. It is therefore important to be able to determine if a model is Zeno and in applicable cases remove Zenoness. However, for models composed of several hybrid subsystems, this is in general a non-trivial task. These problems have been discussed in [5, 6]. In this section, some further characterization of Zeno executions is presented. First, we illustrate Zenoness by an example of Alur and Henzinger [1].

Example 3
Consider the water tank system in Figure 1. Here x_i denotes the volume of water in Tank i, and $v_i > 0$ denote the constant flow of water out of Tank i. Let w denote the constant flow of water into the system, directed exclusively to either Tank 1 or Tank 2 at each point in time. The objective is to keep the water volumes above r_1 and r_2, respectively (assuming that $x_1(0) > r_1$ and $x_2(0) > r_2$). This is to be achieved by a switched control strategy that switches the inflow to Tank 1 whenever $x_1 \le r_1$ and to Tank 2 whenever $x_2 \le r_2$. A hybrid automaton modeling the described system is shown in Figure 1. It is straightforward to show that the unique infinite execution the hybrid automaton accepts for each initial state is Zeno, if $\max\{v_1, v_2\} < w < v_1 + v_2$. The Zeno time is $\big(x_1(0) + x_2(0) - r_1 - r_2\big)/(v_1 + v_2 - w)$. Of course, a real implementation of the water tank system cannot be Zeno, but instead one or both of the tanks will drain. The Zeno model does not capture this. The actual scenario depends on the dynamics of the switch, which in the model was assumed to be instantaneous.

For further discussions on this example, see [5].

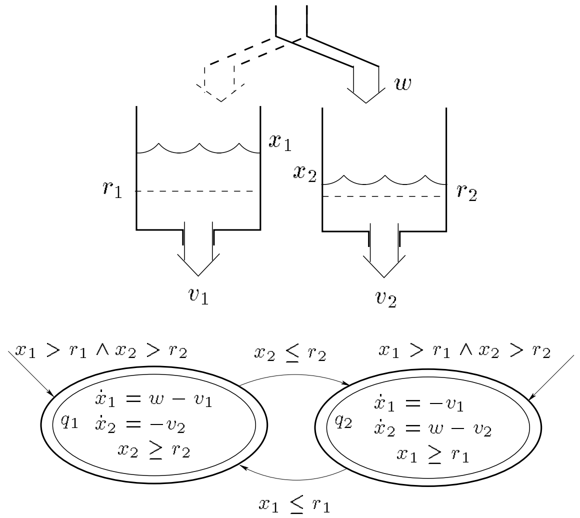

Figure 1: Water tank system and corresponding hybrid automaton.

Definition 6 (Zeno Hybrid Automaton)
A hybrid automaton H is Zeno if there exists $(q_0, x_0) \in$ Init such that all executions in $\mathcal{E}_H^\infty(q_0, x_0)$ are Zeno.

We make a straightforward generalization from dynamical systems and define the ω limit point $(\hat{q}, \hat{x}) \in \mathbf{Q} \times \mathbf{X}$ of an execution $\chi = (\tau, q, x) \in \mathcal{E}_H^\infty$ as a point for which there exists a sequence $\{\theta_n\}_{n=0}^\infty$, $\theta_n \in I_{ni}$, $ni \in \langle\tau\rangle$ such that as $n \to \infty$, $\theta_n \to \mathcal{T}(\chi)$ and $(q(ni), x^{ni}(\theta_n)) \to (\hat{q}, \hat{x})$. The set of ω limit points is the ω limit set. When the continuous part of the Zeno execution is bounded, the Bolzano-Weierstrass Property implies that there exists at least an ω limit point. We introduce the term Zeno state for such a point.

Definition 7 (Zeno State)
The ω limit point of a Zeno execution is called the Zeno state.

We use $Z_\infty \subset \mathbf{Q} \times \mathbf{X}$ to denote the set of Zeno states, so that Z_∞ is the ω limit set of the Zeno execution. We write $\mathbf{Q}_\infty$ for the discrete part and $\mathbf{X}_\infty$ for the continuous part of Z_∞.

For a Zeno execution, the Zeno set can be empty, finite, countable, or even uncountable, see [19] for examples. Figure 2 shows a Zeno execution of the water tank hybrid automaton, for which $Z_\infty = \{q_1, q_2\} \times \{0\}$. Note that $\mathbf{X}_\infty$ is a point for this example. This holds in general if the continuous part of the reset relation is the identity map.

Proposition 3
Consider a hybrid automaton such that $(q', x') \in$ Reset(q, x) implies $x' = x$. Then, for every Zeno execution $\chi = (\tau, q, x)$, it holds that $|\mathbf{X}_\infty| = 1$.

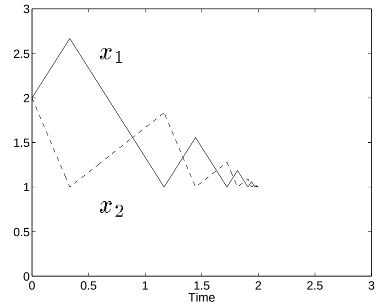

Figure 2: Continuous part of a Zeno execution for the water tank hybrid automaton.

Note that Proposition 3 gives the structure of the Zeno state for the large class of hybrid systems called switched systems [11], since these systems can be modeled as hybrid automata with identity reset relation.

A reset relation Reset is *contracting*, if there exists $\delta \in [0, 1)$ such that $(q', x') \in \text{Reset}(q, x)$ and $(q', y') \in \text{Reset}(q, y)$ imply $\|x' - y'\| \le \delta\|y - x\|$. If the reset relation is contracting and $(q', x') \in \text{Reset}(q, 0)$ implies that x' is the origin, then the continuous part of the Zeno state is also the origin.

Proposition 4
Consider a Zeno hybrid automaton with contracting reset relation and such that $(q', x') \in$ Reset$(q, 0)$ implies $x' = 0$. Then, for every Zeno execution $\chi = (\tau, q, x)$, it holds that $\mathbf{X}_\infty = \{0\}$.

6 Conclusions

In this paper we have highlighted hybrid automata as a tool for modeling heterogeneous systems. Important properties of these systems, such as well-posedness, are not immediate. In the paper, however, we reviewed ongoing activities on establishing a formal framework for analysis and design of hybrid systems modeled as hybrid automata. Local conditions for existence and uniqueness of executions were presented together with a new result about continuous dependence on initial states. We also illustrated some of the nature of Zeno hybrid automata by characterizing Zeno executions for a couple of quite broad classes of hybrid systems.

Ongoing work include the generalization of LaSalle's principle to hybrid systems [18], geometric theory for hybrid systems [12], and optimal control with applications to real-time scheduling.

References

[1] R. Alur and T. A. Henzinger. Modularity for timed and hybrid systems. In *CONCUR 97: Concurrency Theory*, Lecture Notes in Computer Science 1243, pages 74–88. Springer-Verlag, 1997.

[2] R. Balluchi, M. Di Benedetto, C. Pinello, C. Rossi, and A. Sangiovanni-Vincentelli. Hybrid control for automotive engine management: The cut-off case. In T. A. Henzinger and S. Sastry, editors, *Hybrid Systems: Computation and Control*, volume 1386 of *Lecture Notes in Computer Science*. Springer-Verlag, 1998.

[3] R. W. Brockett. Hybrid models for motion control systems. In H. Trentelman and J. Willems, editors, *Essays in Control: Perspectives in the Theory and Its Applications*, pages 29–53. Birkhäuser, Boston, 1993.

[4] M. Broucke. Regularity of solutions and homotopy equivalence for hybrid systems. In *IEEE Conference on Decision and Control*, Tampa, FL, 1998.

[5] K. H. Johansson, M. Egerstedt, J. Lygeros, and S. Sastry. On the regularization of Zeno hybrid automata. *Systems & Control Letters*, 38:141–150, 1999.

[6] K. H. Johansson, J. Lygeros, S. Sastry, and M. Egerstedt. Simulation of Zeno hybrid automata. In *IEEE Conference on Decision and Control*, Phoenix, AZ, 1999.

[7] E. A. Lee. Embedded software—An agenda for research. Technical Report ERL Technical Report UCB/ERL No. M99/63, Dept. EECS, University of California, Berkeley, CA, Dec 1999.

[8] B. Lennartsson, M. Tittus, B. Egardt, and S. Pettersson. Hybrid systems in process control. *Control Systems Magazine*, 16(5):45–56, 1996.

[9] J. Lygeros, K. H. Johansson, S. Sastry, and M. Egerstedt. On the existence of executions of hybrid automata. In *IEEE Conference on Decision and Control*, Phoenix, AZ, 1999.

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

[11] A. S. Morse. Control using logic-based switching. In Alberto Isidori, editor, *Trends in Control. A European Perspective*, pages 69–113. Springer, 1995.

[12] S. Simić, K. H. Johansson, S. Sastry, and J. Lygeros. Towards a geometric theory of hybrid systems. In B. Krogh and N. Lynch, editors, *Hybrid Systems: Computation and Control*, Lecture Notes in Computer Science. Springer-Verlag, New York, 2000.

[13] L. Tavernini. Differential automata and their discrete simulators. *Nonlinear Analysis, Theory, Methods & Applications*, 11(6):665–683, 1987.

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

[15] A. J. van der Schaft and J. M. Schumacher. Complementarity modeling of hybrid systems. *IEEE Transactions on Automatic Control*, 43(4):483–490, April 1998.

[16] P. Varaiya. Smart cars on smart roads: Problems of control. *IEEE Transactions on Automatic Control*, 38(2):195–207, 1993.

[17] J. Zhang. Dynamical systems revisited: Hybrid systems with Zeno executions. Master's thesis, Dept of EECS, University of California, Berkeley, 1999.

[18] J. Zhang, K. H. Johansson, J. Lygeros, and S. Sastry. Dynamical systems revisited: Hybrid systems with Zeno executions. In *Hybrid Systems: Computation and Control*, Pittsburgh, PA, 2000.

[19] J. Zhang, K. H. Johansson, J. Lygeros, and S. Sastry. Some properties of Zeno hybrid systems. *submitted for journal publication*, 2000.