Metrization and Simulation of Controlled Hybrid Systems

Samuel Burden, Student, IEEE, Humberto Gonzalez, Member, IEEE, Ramanarayan Vasudevan, Member, IEEE, Ruzena Bajcsy, Fellow, IEEE, and S. Shankar Sastry, Fellow, IEEE

Abstract

The study of controlled hybrid systems requires practical tools for approximation and comparison of system behaviors. Existing approaches to these problems impose undue restrictions on the system’s continuous and discrete dynamics. Metrization and simulation of controlled hybrid systems is considered here in a unified framework by constructing a state space metric. The metric is applied to develop a numerical simulation algorithm that converges uniformly to orbitally stable executions of controlled hybrid systems, up to and including Zeno events. Benchmark examples of novel hybrid phenomena illustrate the utility of the proposed tools.

I. INTRODUCTION

For continuous–state dynamical systems and finite–state automata there separately exist rich sets of tools for metrization and simulation. The interaction of discrete transitions with continuous dynamics that is characteristic of controlled hybrid systems requires a new approach. To address this limitation, in this paper we construct a distance metric over the state space of a controlled hybrid system and apply this metric to develop a provably–convergent numerical simulation algorithm. Our framework imposes mild restrictions, enabling formal investigation of a wide range of systems: the dynamics may be nonlinear, the continuous dynamics may be controlled, and multiple discrete transitions may occur simultaneously, so long as executions are orbitally stable.

Efforts to topologize and subsequently metrize controlled hybrid systems have been significant but fragmented. Nerode and Kohn [1] consider state–space topologies induced by the finite–state automaton associated with the hybrid system. We propose a metric topology over the state space of controlled hybrid systems, effectively metrizing the hybrifold proposed by Simic et al. [2], as well as the colimit topology constructed by Ames and Sastry [3] over the regularization proposed by Johansson et al. [4]. In contrast, Tavernini [5] directly metrized the space of executions of hybrid systems, and Gokhman [6] later demonstrated the equivalence of the resulting topology with that generated by the Skorokhod trajectory metric (see Chapter 6 in [7]). We highlight the technical and practical limitations imposed by metrizing the trajectory space rather than state space in Section V-C.

The notion of regularization or relaxation of a hybrid system should not be confused with the “relaxation” of hybrid inclusions described by Cai et al. [8]. Since interpreting our controlled hybrid systems as hybrid inclusions yields singleton–valued “flow” and “jump” maps, relaxation in this sense does not yield a distinct hybrid system. Sanfelice and Teel [9] subsequently prove existence of approximating executions for a given “simulation” of a hybrid inclusion. In this paper we consider the opposite problem of proving convergence of approximating simulations for a given execution of a controlled hybrid system.

S. Burden was supported in part by an NSF Graduate Research Fellowship. Part of this research was sponsored by the Army Research Laboratory under Cooperative Agreement W911NF-08-2-0004.

S. Burden, R. Bajcsy, and S. S. Sastry are with the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley. Email: sburden, bajcsy, sastry@eecs.berkeley.edu.

H. Gonzalez is with the Department of Electrical & Systems Engineering, Washington University in St. Louis. Email: hgonzale@ese.wustl.edu.

R. Vasudevan is with the Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology. Email: ramv@csail.mit.edu.
The literature on numerical simulation of deterministic hybrid systems may be broadly partitioned into two groups: practical algorithm development focused on obtaining high-precision estimates of discrete event times, and theoretical proofs of convergence for simulations of certain classes of hybrid systems. Practical algorithms aim to place time-steps close to discrete event times using root-finding [10], [11], [12]. Theoretical proofs of convergence have generally required restrictive assumptions. Esposito et al. [13], for instance, apply feedback linearization to asymptotically guarantee event detection for semi-algebraic guards, while Paoli and Schatzman [14] develop a provably-convergent simulation algorithm for second-order mechanical states undergoing impact specified by a unilateral constraint. The most general convergence results relax the requirement that discrete transition times be determined accurately [5], [15], [16], and consequently can accommodate arbitrary nonlinear transition surfaces, Lipschitz continuous vector fields, and continuous discrete transition maps. We extend this approach using our proposed metric topology to prove convergence of simulations to executions of controlled hybrid systems that satisfy an orbital stability property described in Section [V] Our simulation algorithm may be applied to hybrid systems possessing control inputs and overlapping guards, representing a substantial contribution beyond our previous efforts [16] and those of others [5], [15].

The remainder of our paper is organized as follows. Section [II] contains definitions of the topological, metric, and dynamical system concepts used throughout the paper. We present our technique for metrization and relaxation of controlled hybrid systems in Section [III] and apply these constructions to define a metric for comparing trajectories in controlled hybrid systems. We then develop our algorithm for numerical simulation of controlled hybrid system executions in Section [IV] where we apply our trajectory metric to prove uniform convergence of simulations to orbitally stable executions. The technique is illustrated in Section [V] using examples for accuracy, verification, and novel controlled hybrid system behavior.

II. PRELIMINARIES

We begin with the definitions and assumptions used throughout the paper.

A. Topology

The 2-norm is our finite-dimensional norm of choice unless otherwise specified. Given $P$, the set of all finite partitions of $\mathbb{R}$, and $n \in \mathbb{N}$, we define the total variation of $f \in L^\infty(\mathbb{R}, \mathbb{R}^n)$ by:

$$V(f) = \sup \left\{ \sum_{j=0}^{m-1} \|f(t_{j+1}) - f(t_j)\|_1 | \{t_k\}_{k=0}^m \in P, m \in \mathbb{N} \right\},$$

(1)

where $L^\infty(\mathbb{R}, \mathbb{R}^n)$ is the set of all almost everywhere bounded functions from $\mathbb{R}$ to $\mathbb{R}^n$. The total variation of $f$ is a semi-norm, i.e. it satisfies the Triangle Inequality, but does not separate points. $f$ is of bounded variation if $V(f) < \infty$, and we define $BV(\mathbb{R}, \mathbb{R}^n)$ to be the set of all functions of bounded variation from $\mathbb{R}$ to $\mathbb{R}^n$.

Given $n \in \mathbb{N}$ and $D \subset \mathbb{R}^n$, $\partial D$ is the boundary of $D$, and int($D$) is the interior of $D$. Recall that given a collection of sets $\{S_\alpha\}_{\alpha \in \mathcal{A}}$, where $\mathcal{A}$ might be uncountable, the disjoint union of this collection is $\coprod_{\alpha \in \mathcal{A}} S_\alpha = \bigcup_{\alpha \in \mathcal{A}} S_\alpha \times \{\alpha\}$, a set that is endowed with the piecewise-defined topology. Throughout the paper we will abuse notation and say that given $\tilde{\alpha} \in \mathcal{A}$ and $x \in S_\tilde{\alpha}$, then $x \in \coprod_{\alpha \in \mathcal{A}} S_\alpha$, even though we should write $\iota_\tilde{\alpha}(x) \in \coprod_{\alpha \in \mathcal{A}} S_\alpha$, where $\iota_\tilde{\alpha}: S_\tilde{\alpha} \rightarrow \coprod_{\alpha \in \mathcal{A}} S_\alpha$ is the canonical identification $\iota_\tilde{\alpha}(x) = (x, \tilde{\alpha})$.

In this paper we make extensive use of the concept of a quotient topology induced by an equivalence relation defined on a topological space. We regard a detailed exposition of this important concept as outside the scope of this paper, and refer the reader to Chapter 3 in [17] or Section 22 in [18] for more details. The next definition formalizes equivalence relations in topological spaces induced by functions. If $f: A \rightarrow B$, $V \subset A$, and $V' \subset B$, then we let $f(V) = \{f(a) \in B | a \in V\}$ denote the image of $V$ under $f$, and $f^{-1}(V') = \{a \in A | f(a) \in V'\}$ denote the pre-image of $V$ under $f$. 
Definition 1. Let $S$ be a topological space, $A, B \subset S$ two subsets, and $f: A \to B$ a function. An $f$–induced equivalence relation is defined as:

$$\Lambda_f = \{(a, b) \in S \times S \mid a \in f^{-1}(b), \text{ or } b \in f^{-1}(a), \text{ or } a = b\},$$

(2)

$a, b \in S$ are $f$–related, denoted by $a \sim_f b$, if $(a, b) \in \Lambda_f$. Moreover, the equivalence class of $x \in S$ is defined as $[x]_f = \{a \in S \mid a \sim_f x\}$, and the set of equivalence classes is defined as $\frac{S}{\Lambda_f} = \{[x]_f \mid x \in S\}$.

We endow the quotient $\frac{S}{\Lambda_f}$ with the quotient topology.

Note that $\Lambda_f$ is reflexive, symmetric, and transitive, i.e. an equivalence relation. An important application of the function–induced quotient is the construction of a single topological space out of several disconnected sets. Indeed, given a collection of sets $\{S_\alpha\}_{\alpha \in \mathcal{A}}$, where $\mathcal{A}$ is some index set, and a function $f: U \to \coprod_{\alpha \in \mathcal{A}} S_\alpha$, where $U \subset \coprod_{\alpha \in \mathcal{A}} S_\alpha$, then $\hat{S} = \frac{\coprod_{\alpha \in \mathcal{A}} S_\alpha}{\Lambda_f}$ is a topological space.

Next, we present a useful concept from graph theory that simplifies our ensuing analysis.

Definition 2. Let $(\mathcal{J}, \Gamma)$ be a directed graph, where $\mathcal{J}$ is the set of vertices and $\Gamma \subset \mathcal{J} \times \mathcal{J}$ is the set of edges. Then, given $j \in \mathcal{J}$, define the neighborhood of $j$, denoted $N_j$, by:

$$N_j = \{e \in \Gamma \mid \exists j' \in \mathcal{J} \text{ s.t. } e = (j, j')\}.$$

(3)

B. Length Metrics

Every metric space has an induced length metric, defined by measuring the length of the shortest curve between two points. Throughout this paper, we use induced length metrics to metrize the function–induced quotients of disjoint unions of sets. To formalize this approach, we begin by defining the length of a curve in a metric space; the following definition is equivalent to Definition 2.3.1 in [19].

Definition 3. Let $(S, d)$ be a metric space, $I \subset [0, 1]$ be an interval, and $\gamma: I \to S$ be a continuous function. Define the length of $\gamma$ under the metric $d$ by:

$$L_d(\gamma) = \sup \left\{ \sum_{i=0}^{k-1} d(\gamma(t_i), \gamma(t_{i+1})) \mid k \in \mathbb{N}, \{t_i\}_{i=0}^k \subset I, \ t_0 < t_1 < \ldots < t_k \right\}.$$

(4)

We now define a generalization of continuous curves for quotiated disjoint unions.

Definition 4. Let $\{S_\alpha\}_{\alpha \in \mathcal{A}}$ be a collection of sets and $f: U \to \coprod_{\alpha \in \mathcal{A}} S_\alpha$, where $U \subset \coprod_{\alpha \in \mathcal{A}} S_\alpha$. $\gamma: [0, 1] \to \coprod_{\alpha \in \mathcal{A}} S_\alpha$ is $f$–connected if there exists $k \in \mathbb{N}$ and $\{t_i\}_{i=0}^k \subset [0, 1]$ with $0 = t_0 \leq t_1 \leq \ldots \leq t_k = 1$ such that $\gamma|_{[t_i, t_{i+1}]}$ is continuous for each $i \in \{0, 1, \ldots, k-2\}$, $\gamma|_{[t_{k-1}, t_k]}$ is continuous, and $\lim_{t \uparrow t_i} \gamma(t) \overset{f}{\sim} \gamma(t_i)$ for each $i \in \{0, 1, \ldots, k-1\}$. Moreover, in that case $\{t_i\}_{i=0}^k$ is called a partition of $\gamma$.

Note that, since each section $\gamma|_{[t_i, t_{i+1}]}$ is continuous, it must necessarily belong to a single set $S_\alpha$ for some $\alpha \in \mathcal{A}$ because the disjoint union is endowed with the piecewise–defined topology. In the case when $\mathcal{A} = \{\alpha\}$ is a singleton, then every $\text{id}_{S_\alpha}$–connected curve is simply a continuous curve over $S_\alpha$, where $\text{id}_{S_\alpha}$ denotes the identity function in $S_\alpha$. Figure 1a shows an example of a connected curve over a collection of two sets.

Using the concept of connected curves, we now define the induced length distance of a collection of metric spaces. The induced length distance is a generalization of the induced metric defined in Chapter 2 in [19].

Definition 5. Let $\{(S_\alpha, d_\alpha)\}_{\alpha \in \mathcal{A}}$ be a collection of metric spaces, and let $\{X_\alpha\}_{\alpha \in \mathcal{A}}$ be a collection of sets such that $X_\alpha \subset S_\alpha$ for each $\alpha \in \mathcal{A}$. Furthermore, let $f: U \to \coprod_{\alpha \in \mathcal{A}} X_\alpha$, where $U \subset \coprod_{\alpha \in \mathcal{A}} X_\alpha$, and let
This implies

points of
d to note that although

We invoke this definition to metrize both subsets and disjoint unions of metric spaces. It is important

(a) Let

Lemma 6.

straightforward consequence of Proposition 2.3.12 in [19].

(b) Illustration of a controlled hybrid system with three modes.

Fig. 1. An $g$–connected curve, Fig. 1a and an example of a controlled hybrid system, Fig. 1b.

\[ \hat{X} = \bigcup_{\alpha \in A} X_{\alpha}, \quad d_{i, \hat{X}} : \hat{X} \times \hat{X} \to [0, \infty] \text{ is the } f–\text{induced length distance of } \hat{X}, \text{ defined by:} \]

\[ d_{i, \hat{X}}(p, q) = \inf \left\{ \sum_{i=0}^{k-1} L_{d_{\alpha_i}}(\gamma|_{[t_i, t_{i+1}]}) \right\} \bigg| \gamma : [0, 1] \to \bigcup_{\alpha \in A} X_{\alpha}, \quad \gamma(0) = p, \quad \gamma(1) = q, \]

\[ \gamma \text{ is } f–\text{connected}, \quad \{t_i\}_{i=0}^{k} \text{ is a partition of } \gamma, \quad \{\alpha_i\}_{i=0}^{k-1} \text{ s.t. } \gamma([t_i, t_{i+1}]) \subset X_{\alpha_i} \quad \forall i \bigg\}. \]

We invoke this definition to metrize both subsets and disjoint unions of metric spaces. It is important
to note that although $d_{i, \hat{X}}$ is non–negative, symmetric, and subadditive, it does not necessarily separate
points of $\hat{X}$ (see Section 2.3 in [19]), and hence generally only defines a \textit{pseudo–metric}. In the special
case where no function $f$ is supplied, then by convention we let $f = t_{i, X}$, the identity function on $X$.
This implies $\hat{X} = X$ and the induced metric coincides with the given metric. The following Lemma is a
straightforward consequence of Proposition 2.3.12 in [19].

\textbf{Lemma 6.} Let $(S, d)$ be a metric space and $X \subset S$. Then $d_{i, X}$ is a metric. Moreover, the topology on $X$
induced by $d_{i, X}$ is equivalent to the topology on $X$ induced by $d$.

\textbf{C. Controlled Hybrid Systems}

Motivated by the definition of hybrid systems presented in [2], we define the class of hybrid systems
of interest in this paper.

\textbf{Definition 7.} A controlled hybrid system is a tuple $\mathcal{H} = (\mathcal{J}, \Gamma, \mathcal{D}, \mathcal{U}, \mathcal{F}, \mathcal{G}, \mathcal{R})$, where:

- $\mathcal{J}$ is a finite set indexing the discrete states of $\mathcal{H}$;
- $\Gamma \subset \mathcal{J} \times \mathcal{J}$ is the set of edges, forming a directed graph structure over $\mathcal{J}$;
- $\mathcal{D} = \{D_j\}_{j \in \mathcal{J}}$ is the set of domains, where each $D_j$ is a subset of $\mathbb{R}^{n_j}$, $n_j \in \mathbb{N}$;
- $\mathcal{U} \subset \mathbb{R}^m$ is the range space of control inputs, $m \in \mathbb{N}$;
- $\mathcal{F} = \{f_j\}_{j \in \mathcal{J}}$ is the set of vector fields, where each $f_j : \mathbb{R} \times D_j \times U \to \mathbb{R}^{n_j}$ is a vector field defined
on $D_j$;
- $\mathcal{G} = \{G_{(j,j')}\}_{j \in \mathcal{J}}$ is the set of guards, where each $G_{(j,j')} \subset \partial D_j$ is a guard in mode $j \in \mathcal{J}$ that defines
a transition to mode $j' \in \mathcal{J}$; and,
- $\mathcal{R} = \{R_{(j,j')}\}_{j \in \mathcal{J}}$ is the set of reset maps, where each $R_{(j,j')} : G_{(j,j')} \to D_j$ defines the transition from
guard $G_{(j,j')}$. 

For convenience, we sometimes refer to controlled hybrid systems as just hybrid systems, and we refer to the distinct vertices within the graph structure associated with a controlled hybrid system as modes. Each domain in the definition of a controlled hybrid system is a metric space with the Euclidean distance metric. A three–mode autonomous hybrid system, which is a particular case of Definition 7 where none the vector fields \( \{f_j\}_{j \in J} \) depend on the control input, is illustrated in Fig. 10. Note that we restrict control inputs to the continuous flow, hence inputs do not have an effect during discrete transitions.

Next, we impose several technical assumptions that support existence and uniqueness of trajectories on hybrid domains.

**Assumption 8.** Let \( \mathcal{H} \) be a controlled hybrid system. Then the following statements are true:

1. For each \( j \in J \), \( D_j \) is a compact \( n_j \)-dimensional manifold with boundary.
2. \( U \) is compact.
3. For each \( e \in \Gamma \), \( G_e \) is a closed, embedded, codimension 1, submanifold with boundary.
4. For each \( e \in \Gamma \), \( R_e \) is continuous.

**Assumption 9.** For each \( j \in J \), \( f_j \) is Lipschitz continuous. That is, there exists \( L > 0 \) such that for each \( j \in J \), \( t_1, t_2 \in \mathbb{R} \), \( x_1, x_2 \in D_j \), and \( u_1, u_2 \in U \):

\[
\|f_j(t_1, x_1, u_1) - f_j(t_2, x_2, u_2)\| \leq L (|t_1 - t_2| + \|x_1 - x_2\| + \|u_1 - u_2\|). \tag{6}
\]

Assumption 9 guarantees the existence and uniqueness for ordinary differential equations in individual domains. In the sequel we will consider control inputs of bounded variation \( u \in BV(\mathbb{R}, U) \). Note that without loss of generality we take 0 as the initial time in the following Lemma; a general initial time can be accommodated by a straightforward change of variables.

**Lemma 10.** Let \( \mathcal{H} \) be a controlled hybrid system. Then for each \( j \in J \), each initial condition \( p \in D_j \), and each control \( u \in BV(\mathbb{R}, U) \), there exists an interval \( I \subset \mathbb{R} \) with 0 \( \in I \) such that the following differential equation has a unique solution:

\[
\dot{x}(t) = f_j(t, x(t), u(t)), \quad t \in I, \quad x(0) = p. \tag{7}
\]

\( x \) is called the integral curve of \( f_j \) with initial condition \( p \) and control \( u \). Moreover, \( x|_I \) is absolutely continuous.

**Proof:** Let \( \tilde{f}_j : \mathbb{R} \times \mathbb{R}^{n_j} \times U \to \mathbb{R}^{n_j} \) be any globally Lipschitz continuous extension to \( f_j \) (guaranteed to exist by Theorem 1 in [20]). Given any \( p \in D_j \subset \mathbb{R}^{n_j} \) and \( u \in BV(\mathbb{R}, U) \), Proposition 5.6.5 in [21] guarantees the existence of an integral curve \( \tilde{x} : \tilde{I} \to \mathbb{R}^{n_j} \) for \( \tilde{f}_j \) with initial condition \( \tilde{x}(0) = p \). Note that \( \tilde{x} \) is absolutely continuous by Theorem 3.35 in [22]. Let \( I \subset \tilde{I} \) be the connected component of \( \tilde{x}^{-1}(D_j) \) containing 0. Then \( x = \tilde{x}|_I \) is an absolutely continuous integral curve of \( f_j \) and \( x(I) \subset D_j \). Note that \( x \) is unaffected by the choice of extension \( \tilde{f}_j \).

The following definition is used to construct executions of a controlled hybrid system.

**Definition 11.** Let \( \mathcal{H} \) be a controlled hybrid system, \( j \in J \), \( p \in D_j \), and \( u \in BV(\mathbb{R}, U) \). \( x : I \to D_j \) is the maximal integral curve of \( f_j \) with initial condition \( p \) and control \( u \) if, given any other integral curve with initial condition \( p \) and control \( u \), such as \( \tilde{x} : \tilde{I} \to D_j \), then \( \tilde{I} \subset I \).

Given a maximal integral curve \( x : I \to D_j \), a direct consequence of Definition 11 and Assumption 8 is that either \( \sup I = +\infty \), or \( \sup I = t' < \infty \) and \( x(t') \in \partial D_j \). This fact is critical during the definition of executions of a controlled hybrid systems in Section IV.

\(^1\)This follows from continuity of integral curves and compactness of hybrid domains.
III. METRIZATION AND RELAXATION OF CONTROLLED HYBRID SYSTEMS

In this section, we metrize a unified family of spaces containing all the domains of a controlled hybrid system $\mathcal{H}$. The constructed metric space has three appealing properties:

1. the distance between a point in a guard and its image via its respective reset map is zero;
2. the distance between points in different domains are properly defined and finite; and,
3. the distance between points is based on the Euclidean distance metric from each domain.

A. Hybrid Quotient Space

Using Definitions 5 and 7, we construct a metric space where the executions of a controlled hybrid system reside. The result is a metrization of the *hybrifold* [2].

**Definition 12.** Let $\mathcal{H}$ be a controlled hybrid system, and let $\hat{R}: \coprod_{e \in \Gamma} G_e \to \coprod_{j \in J} D_j$ be defined by $\hat{R}(p) = R_e(p)$ for each $p \in G_e$. Then the hybrid quotient space of $\mathcal{H}$ is:

$$\mathcal{M} = \left( \coprod_{j \in J} D_j \right) / \Lambda_{\hat{R}}.$$  

Fig. 2 illustrates the details about the construction described in Definition 12.

The induced length distance on $\mathcal{M}$ is in fact a distance metric:

**Theorem 13.** Let $\mathcal{H}$ be a controlled hybrid system, and let $d_{i,\mathcal{M}}$ be the $\hat{R}$–induced length distance of $\mathcal{M}$, where $\hat{R}$ is defined in (8). Then $d_{i,\mathcal{M}}$ is a metric on $\mathcal{M}$, and the topology it induces is equivalent to the $\hat{R}$–induced quotient topology.

**Proof:** We provide the main arguments of the proof, omitting the details in the interest of brevity. First, note that each domain is a normal space, i.e. every pair of disjoint closed sets have disjoint neighborhoods. Second, note that each reset map is a closed map, i.e. the image of closed sets under the reset map are closed. This fact follows by Condition 3 in Assumption 8, since each guard is compact, thus reset maps are closed by the Closed Map Lemma (Lemma A.19 in [23]).

Let $\hat{D} = \coprod_{j \in J} D_j$ and $p, q \in \hat{D}$. We aim to show that if $p$ and $q$ yield distinct equivalence classes (i.e. $(p, q) \notin \Lambda_{\hat{R}}$) then the induced distance between them is strictly positive. Note that the equivalence classes $[p]_{\hat{R}}$ and $[q]_{\hat{R}}$ are each a finite collection of closed sets. Moreover, since we can construct disjoint neighborhoods around each of these closed sets, then we can conclude that there exists $\delta > 0$ such that $d_{i,\mathcal{M}}([p]_{\hat{R}}, [q]_{\hat{R}}) > \delta$. The proof concludes by following the argument in Exercise 3.1.14 in [19], i.e. since each connected component in $\hat{D}$ is bounded, then $\mathcal{M}$ is also bounded (in the quotient topology).
Then, using a simple extension of Theorem 5.8 in [17], we get that the identity map from \( M \) to the space constructed by taking the quotient of all the points in \( \tilde{D} \) such that \( d_{i,M} \) has zero distance is a homeomorphism, thus they have the same topology.

It is crucial to note that all \( \tilde{R} \)-connected curves are continuous in the topology induced by the metric \( d_{i,M} \) on the hybrid quotient space \( M \). This implies in particular that executions of controlled hybrid systems (to be defined in Section IV[A]) are continuous in \( M \) since the endpoint of the segment of an execution that lies in a guard \( G_e \) will be \( \tilde{R} \)-related to the startpoint of the subsequent segment of the execution; alternately, this follows from Theorem 3.12(b) in [2] since \( M \) is equivalent to the “hybrifold” construction in that paper. This important property is foundational to the convergence results for sequences of (relaxed) executions and their simulations derived in Section IV. For further details, we refer the interested reader to Examples 3.2 and 3.3 in [2] where continuity is clearly discussed for simple examples.

**B. Relaxation of a Controlled Hybrid System**

To construct a numerical simulation scheme which does not require the exact computation of the time instant when an execution intersects a guard, we require a method capable of introducing some slackness within the computation. This is accomplished by relaxing each domain along its guard and then relaxing each vector field and reset map accordingly in order to define a relaxation of a controlled hybrid system.

To formalize this approach, we begin by defining the relaxation of each domain of a controlled hybrid system, which is accomplished by first attaching an \( \varepsilon \)-sized strip to each guard.

**Definition 14.** Let \( \mathcal{H} \) be a controlled hybrid system. For each \( e \in \Gamma \), let \( S_e^\varepsilon = G_e \times [0, \varepsilon] \) be the strip associated to guard \( G_e \). For each \( j \in J \), let

\[
\chi_j : \bigoplus_{e \in N_j} G_e \to \bigoplus_{e \in N_j} S_e^\varepsilon,
\]

be the canonical identification of each point in a guard with its corresponding strip defined for each \( p \in G_e \) as \( \chi_j(p) = (p, 0) \in S_e^\varepsilon \). Then, the relaxation of \( D_j \) is defined by:

\[
D_j^\varepsilon = \frac{D_j \bigsqcup \left( \bigoplus_{e \in N_j} S_e^\varepsilon \right)}{\Lambda_{\chi_j}}.
\]

By Condition 3 in Assumption 8, each point on a strip \( S_e^\varepsilon \) of \( D_j \) is defined using \( n_j \) coordinates \((\xi_1, \ldots, \xi_{n_j-1}, \tau)\), shortened \((\zeta, \tau)\), where \( \tau \) is called the transverse coordinate and is the distance along the interval \([0, \varepsilon]\). An illustration of Definition 14 together with the coordinates on each strip is shown in Fig. 3a.

We endow each \( S_e^\varepsilon \) with a distance metric in order to define an induced length metric on a relaxed domain \( D_j^\varepsilon \).

**Definition 15.** Let \( j \in J \) and \( e \in N_j \). Endow \( D_j^\varepsilon \) with \( d_{i,D_j} \) as its metric, and \( d_{S_e^\varepsilon} : S_e^\varepsilon \times S_e^\varepsilon \to [0, \infty) \) as the metric on \( S_e^\varepsilon \), defined for each \( \zeta, \zeta' \in G_e \) and \( \tau, \tau' \in [0, \varepsilon] \) by:

\[
d_{S_e^\varepsilon}((\zeta, \tau), (\zeta', \tau')) = d_{i,G_e}(\zeta, \zeta') + |\tau - \tau'|.
\]

We now define a length metric on relaxed domains using Definitions 4 and 5.

**Theorem 16.** Let \( j \in J \), and let \( d_{i,D_j} \) be the \( \chi_j \)-induced length distance on \( D_j^\varepsilon \), where \( \chi_j \) is as defined in (10). Then \( d_{i,D_j}^\varepsilon \) is a metric on \( D_j^\varepsilon \), and the topology it induces is equivalent to the \( \chi_j \)-induced quotient topology.

\(^2\)The extension aims to allow the domain of the map to be bounded instead of compact. The new proof follows step–by–step the argument in [17].
Proof: Since \( d_{i, D_j}^e \) is non–negative, symmetric, and subadditive, it remains to show that it separates points. Let \( p, q \in D_j^e \). First, we want to show that \([p]_{\chi_j} = [q]_{\chi_j}\) whenever \( d_{i, D_j}^e(p, q) = 0\). Note that for each \( e \in \mathcal{N}_j \) and each pair \( p, q \in G_e \), and by the Definition 5 and 15, \( d_{i, D_j}^e(p, q) \geq d_{i, D_j}(p, q) \), hence no connected curve that “jumps” to a strip can be shorter than a curve that stays in \( D_j \). This fact immediately shows that for \( p, q \in D_j \), \( d_{i, D_j}(p, q) = 0 \) implies that \([p]_{\chi_j} = [q]_{\chi_j}\). The case when one of the points is in \( G_e \times (0, \varepsilon) \subset S_e^\varepsilon \) follows easily by noting that those points can be separated by a suitably–sized \( d_{S_e} \)–ball. The proof concludes by following the argument in Exercise 3.1.14 in [19], as we did in the proof of Theorem 13. □

Refer to \( d_{i, D_j}^e \) as the relaxed domain metric. Note that Theorem 16 can be proved using the same argument as in the proof of Theorem 13 but we prove Theorem 16 to emphasize the utility of the inequality relating the induced metric on a domain and the metric on each strip.

Next, we define a vector field over each relaxed domain.

**Definition 17.** Let \( j \in \mathcal{J} \). For each \( e \in \mathcal{N}_j \), let the vector field on the strip \( S_e^\varepsilon \), denoted \( f_e \), be the unit vector pointing outward along the transverse coordinate. In coordinates, \( f_e(t, (\zeta, \tau), u) = (0, \ldots, 0, 1)^T \). Then, the relaxation of \( f_j \) is:

\[
\begin{align*}
    f_j^e(t, x, u) = \begin{cases}
        f_j(t, x, u) & \text{if } x \in D_j; \\
        f_e(t, x, u) & \text{if } x \in G_e \times (0, \varepsilon) \subset S_e^\varepsilon, \text{ for some } e \in \mathcal{N}_j.
    \end{cases}
\end{align*}
\]

(13)

Note that the relaxation of the vector field is generally not continuous along each \( G_e \), for \( e \in \mathcal{N}_j \). As we show in the algorithm in Fig. 7, this discontinuous vector field does not lead to sliding modes on the guards [24], [25], since the vector field on the strips always points away from the guard. An illustration of the relaxed vector field \( f_j^e \) on \( D_j^e \) is shown in Fig. 3b.

The definitions of relaxed domains and relaxed vector fields allow us to construct a relaxation of the controlled hybrid systems as follows:

**Definition 18.** Let \( \mathcal{H} \) be a controlled hybrid system. We say that the relaxation of \( \mathcal{H} \) is a tuple \( \mathcal{H}^e = (\mathcal{J}, \Gamma, D^e, U, \mathcal{F}^e, G^e, \mathcal{R}^e) \), where:

- \( D^e = \{ D_j^e \}_{j \in \mathcal{J}} \) is the set of relaxations of the domains in \( D \), and each \( D_j^e \) is endowed with its induced length distance metric \( d_{i, D_j^e} \);
- \( \mathcal{F}^e = \{ f_j^e \}_{j \in \mathcal{J}} \) is the set of relaxations of the vector fields in \( \mathcal{F} \);
- \( G^e = \{ G_e^e \}_{e \in \Gamma} \) is the set of relaxations of the guards in \( G \), where \( G_e^e = G_e \times \{ \varepsilon \} \subset S_e^\varepsilon \) for each \( e \in \Gamma \); and,
C. Relaxed Hybrid Quotient Space

Let \( \mathcal{H}^\varepsilon \) be the relaxation of the controlled hybrid system \( \mathcal{H} \). Also, let

\[
\hat{R}^\varepsilon : \bigoplus_{e \in \Gamma} G_e^\varepsilon \to \bigoplus_{j \in \mathcal{J}} D_j^\varepsilon
\]

be defined by \( \hat{R}^\varepsilon(p) = R^\varepsilon_e(p) \) for each \( p \in G_e^\varepsilon \). Then the relaxed hybrid quotient space of \( \mathcal{H}^\varepsilon \) is:

\[
\mathcal{M}^\varepsilon = \bigoplus_{j \in \mathcal{J}} D_j^\varepsilon \sslash \Lambda_{\hat{R}^\varepsilon}.
\]

The construction in Definition 19 is illustrated in Fig. 4.

We now show that the induced length distance on \( \mathcal{M}^\varepsilon \) is indeed a metric. We omit this proof since it is identical to the proof of Theorem 13.

**Theorem 20.** Let \( \mathcal{H} \) be a controlled hybrid system, let \( \mathcal{H}^\varepsilon \) be its relaxation, and let \( d_{i,\mathcal{M}^\varepsilon} \) be the \( \hat{R}^\varepsilon \)–induced length distance of \( \mathcal{M}^\varepsilon \), where \( \hat{R}^\varepsilon \) is defined in (14). Then \( d_{i,\mathcal{M}^\varepsilon} \) is a metric on \( \mathcal{M}^\varepsilon \), and the topology it induces is equivalent to the \( \hat{R}^\varepsilon \)–induced quotient topology.

All \( \hat{R}^\varepsilon \)–connected curves are continuous under the metric topology induced by \( d_{i,\mathcal{M}^\varepsilon} \) which is important when we study executions of hybrid systems in Section 17.

As expected, the metric on \( \mathcal{M}^\varepsilon \) converges pointwise to the metric on \( \mathcal{M} \).

**Theorem 21.** Let \( \mathcal{H} \) be a controlled hybrid system, and let \( \mathcal{H}^\varepsilon \) be its relaxation. Then for all \( p, q \in \mathcal{M} \),

\[
d_{i,\mathcal{M}^\varepsilon}(p, q) \rightarrow d_{i,\mathcal{M}}(p, q)
\]

as \( \varepsilon \rightarrow 0 \).

**Proof:** Abusing notation, let \( L(\gamma) \) denote the length of any connected curve \( \gamma \), defined as the sum of the lengths of each of its continuous sections under the appropriate metric. First, note that \( d_{i,\mathcal{M}}(p, q) \leq d_{i,\mathcal{M}^\varepsilon}(p, q) \). This inequality follows since, as we argued in the proof of Theorem 16, given an edge \((j, j') \in \Gamma, (p', 0), (q', 0)) \geq d_{i,\mathcal{D}}(p', q') \) for any pair of points \( p', q' \in G_{(j, j')} \). Thus, adding the strips \( \{S_e\}_{e \in \gamma} \) in \( \mathcal{M}^\varepsilon \) only make the length of a connected curve longer.

Now let \( \hat{D} = \bigoplus_{j \in \mathcal{J}} D_j \) and \( \hat{D}^\varepsilon = \bigoplus_{j \in \mathcal{J}} D_j^\varepsilon \). Given \( \delta > 0 \), there exists \( \gamma : [0, 1] \rightarrow \hat{D} \), an \( \hat{R} \)–connected curve with partition \( \{t_i\}_{i=0}^k \), such that \( \gamma(0) = p, \gamma(1) = q, \) and \( d_{i,\mathcal{M}}(p, q) \leq L(\gamma) \leq d_{i,\mathcal{M}^\varepsilon}(p, q) + \delta \). Moreover, without loss of generality let \( \gamma^\varepsilon : [0, 1] \rightarrow \hat{D}^\varepsilon \) be an \( \hat{R}^\varepsilon \)–connected curve that agrees with \( \gamma \) on...
Requirement: \( t = 0, j \in \mathcal{J}, p \in D_j, \) and \( u \in BV(\mathbb{R}, U) \).

1. Set \( x(0) = p \).
2. \textbf{loop}
3.   \textbf{if} \( t' = \sup I \) and \( x(s) = \gamma(s) \) for each \( s \in [t, t') \).
4.   \textbf{end if}
5.   \textbf{if} \( t' = \infty \), or \( \exists e \in N_j \) such that \( \gamma(t') \in G_e \)
6.   \textbf{then}
7.   \textbf{Stop}.
8. \textbf{end if}
9. Set \( x(t') = R_{j,j'}(\gamma(t')) \), \( t = t' \), and \( j = j' \).
10. \textbf{end loop}

Fig. 5. Algorithm to construct an execution of a controlled hybrid system \( \mathcal{H} \).

\( \mathcal{D} \), i.e. each section of \( \gamma^\varepsilon \) on \( \mathcal{D} \) is identical, up to time scaling, to a section of \( \gamma \) Thus \( \gamma^\varepsilon \) has at most \( k \varepsilon \)–length extra sections, and \( L(\gamma) \leq L(\gamma^\varepsilon) \leq L(\gamma) + k \varepsilon \). Thus, \( d_{i,\mathcal{M}^\varepsilon}(p, q) \leq L(\gamma^\varepsilon) \leq d_{i,\mathcal{M}}(p, q) + k \varepsilon + \delta \). The result follows after noting this inequality is valid for all \( \delta > 0 \), thus \( d_{i,\mathcal{M}^\varepsilon}(p, q) \leq d_{i,\mathcal{M}}(p, q) \).

Note that Theorem 21 does not imply that the topology of \( \mathcal{M}^\varepsilon \) converges to the topology of \( \mathcal{M} \). On the contrary, \( \mathcal{M}^\varepsilon \) is homotopically equivalent to the graph \( (\mathcal{J}, \Gamma) \) for each \( \varepsilon > 0 \) [3], whereas the topology of \( \mathcal{M} \) may be different [2].

We conclude this section by introducing metrics between curves on \( \mathcal{M}^\varepsilon \).

**Definition 22.** Let \( I \subset [0, \infty) \) a bounded interval. Given any two curves \( \gamma, \gamma': I \to \mathcal{M}^\varepsilon \), we define:

\[
\rho^I_f(\gamma, \gamma') = \sup \{ d_{i,\mathcal{M}^\varepsilon}(\gamma(t), \gamma'(t)) \mid t \in I \}.
\]  
(16)

Our choice of the supremum among point–wise distances in Definition 22 is inspired by the sup–norm for continuous real–valued functions.

**IV. RELAXED EXECUTIONS AND DISCRETE APPROXIMATIONS**

This section contains our main result: discrete approximations of trajectories of controlled hybrid systems, constructed using any variable step size numerical integration algorithm, converge uniformly to the actual trajectories. This section is divided into three parts. First, we define a pair of algorithms that construct executions of controlled hybrid systems and their relaxations, respectively. Next, we develop a discrete approximation scheme for executions of relaxations of controlled hybrid systems. Finally, we prove that these discrete approximations converge to the executions of the original, non–relaxed, controlled hybrid system using the metric topologies developed in Section III.

A. Execution of a Hybrid System

We begin by defining an execution of a controlled hybrid system. This definition agrees with the traditional intuition about executions of controlled hybrid systems which describes an execution as evolving as a standard control system until a guard is reached, at which point a discrete transition occurs to a new domain using a reset map. We provide an explicit definition to clarify technical details required in the proofs below. Given a controlled hybrid system, \( \mathcal{H} \), as in Definition 7 the algorithm in Fig. 5 defines an execution of \( \mathcal{H} \) via construction. A resulting execution, denoted \( x \), is an \( \mathcal{R} \)–connected curve from some interval \( I \subset [0, \infty) \) to \( \prod_{j \in \mathcal{J}} D_j \). Thus, abusing notation, we regard \( x \) as a continuous curve on \( \mathcal{M} \). Abusing notation again, we regard \( x \) as a piece–wise continuous curve on \( \mathcal{M}^\varepsilon \) for each \( \varepsilon > 0 \). Fig. 6a shows an execution undergoing a discrete transition.

Note that executions constructed using the algorithm in Fig. 5 are not necessarily unique. Indeed, Definition 11 implies that once a discrete transition has been performed, the execution is unique until a
new transition is made; however, the choice in Step 8 is not necessarily unique if the maximal integral curve passes through the intersection of multiple guards. It is not hard to prove that a sufficient condition for uniqueness of executions is that all the guards are disjoint, even though, as we show in Section V-C, uniqueness of the executions can be obtained for some cases where guards do intersect.

With the definition of execution of a controlled hybrid system, we can define a class of executions unique to controlled hybrid systems.

**Definition 23.** An execution is Zeno when it undergoes an infinite number of discrete transitions in a finite amount of time. Hence, there exists \( T > 0 \), called the Zeno Time, such that the execution is only defined on \( I = [0, T) \).

Zeno executions are hard to simulate since they apparently require an infinite number of reset map evaluations, an impossible task to implement on a digital computer. A consequence of the algorithm in Fig. 3 is that if \( x : I \to \mathcal{M} \) is an execution such that \( T = \sup I < \infty \), then either

1. \( x \) has a finite number of discrete transitions on \( I = [0, T] \), and \( x(T) \in \partial D_j \) for some \( j \in \mathcal{J} \), or
2. \( x \) is a Zeno execution and \( I = [0, T) \).

We now introduce a property of Zeno executions of particular interest in this paper:

**Definition 24.** Let \( \mathcal{H} \) be a controlled hybrid system, \( p \in \mathcal{M} \), \( u \in BV(\mathbb{R}, U) \), and \( x : [0, T) \to \mathcal{M} \) be a Zeno execution with initial condition \( p \), control \( u \), and Zeno Time \( T \). \( x \) accumulates at \( p' \in \mathcal{M} \) if

\[
\lim_{t \to T} d_{i,\mathcal{M}}(x(t), p') = 0.
\]

Examples of Zeno executions that do not accumulate can be found in [26]. Fig. 6b shows a Zeno execution that accumulates at \( p' \). Note that for \( p' \) to be a Zeno accumulation point, it must belong to a guard of a controlled hybrid system.

Since \( \mathcal{M} \) is a metric space, we can introduce the concept of continuity of a hybrid execution with respect to its initial condition and control input in a straightforward way. Employing this definition, we can define the class of executions that are numerically approximable:

**Definition 25.** Let \( \mathcal{H} \) be a controlled hybrid system, and assume that all the executions of \( \mathcal{H} \) are unique. Denote by \( x_{(p,u)} : I_{(p,u)} \to \mathcal{M} \) the hybrid execution of \( \mathcal{H} \) with initial condition \( p \in \mathcal{M} \) and control \( u \in BV(\mathbb{R}, U) \). Given \( T > 0 \), we say that the map \( (p, u) \mapsto x_{(p,u)} \) is orbitally stable in \([0, T] \) at \((p', u') \in \mathcal{M} \times BV(\mathbb{R}, U) \) if there exists a neighborhood of \((p', u') \), say \( N_{(p', u')} \subset \mathcal{M} \times BV(\mathbb{R}, U) \), such that the following conditions are satisfied:

1. \([0, T] \subseteq I_{(p,u)} \) for each \((p, u) \in N_{(p', u')} \).
2. The map \( (p, u) \mapsto x_{(p,u)}(t) \) is continuous at \((p', u') \) for each \( t \in [0, T] \).

As observed in Section III.B in [27], executions that are not orbitally stable are difficult to approximate.
with a general algorithm. Figure 6c shows a non–orbitally stable execution that intersects the guard
tangentially, and note that executions initialized arbitrarily close to \( p' \in D_1 \) undergo different sequences
of transitions. Unfortunately, there is presently no general test (analytical or numerical) that ensures a
given execution is orbitally stable. Theorem III.2 in [27] provides one set of sufficient conditions ensuring
orbital stability.

B. Relaxed Execution of a Hybrid System

Next, we define the concept of relaxed execution for a relaxation of a controlled hybrid system. The main
idea is that, once a relaxed execution reaches a guard, we continue integrating over the strip with the
relaxed vector field, \( f_\varepsilon \), as in Definition 17. Given the controlled hybrid system, \( \mathcal{H} \), its relaxation, \( \mathcal{H}_\varepsilon \),
for some \( \varepsilon > 0 \), the algorithm in Fig. 7 defines a relaxed execution of \( \mathcal{H}_\varepsilon \) via construction. The resulting
relaxed execution, denoted \( x_\varepsilon \), is a continuous function defined from an interval \( I \subset [0, \infty) \) to \( \mathcal{M}_\varepsilon \). Note
that this algorithm is only defined for initial conditions belonging to \( D_j \) for some \( j \in \mathcal{J} \) since the strips are
artificial objects that do not appear in \( \mathcal{H} \). The generalization to all initial conditions is straightforward;
we omit it to simplify the presentation.

Step 9 of the algorithm in Fig. 7 relaxes each instantaneous discrete transition by integrating over the
vector field on a strip, hence forming a continuous curve on \( \mathcal{M}_\varepsilon \). Also note that our definition for the
relaxed execution over each strip \( S_\varepsilon \), also in Step 9, is exactly equal to the maximal integral curve of \( f_\varepsilon \).
Fig. 8a shows an example of a relaxed mode transition produced the algorithm in Fig. 7. Given a hybrid
system \( \mathcal{H} \) and its relaxation \( \mathcal{H}_\varepsilon \), the relaxed execution of \( \mathcal{H}_\varepsilon \) produced by the algorithm in Fig. 7 is a
delayed version of the execution of \( \mathcal{H} \) produced by the algorithm in Fig. 5 since the relaxed version has
to expend \( \varepsilon \) time units during each discrete transition. In that sense, our definition of relaxed execution is
equivalent to an execution of a regularized hybrid systems [4].

Note that if a relaxed execution is unique for a given initial condition and input, then the corresponding
hybrid execution is also unique, but not vice versa. Indeed, consider the case of a hybrid execution
performing a single discrete transition at a point, say \( p \), where two guards intersect, i.e. \( p \in G_e \) and
\( p \in G'_e \), such that \( R_e(p) = R'_e(p) \). In this case the hybrid execution is unique, but its relaxed counterpart
either evolves via \( S_e \) or \( S'_e \), hence obtaining 2 different executions. Nevertheless, both relaxed executions
reach the same point after evolving over the strip.

Next, we state our first convergence theorem.

Theorem 26. Let \( \mathcal{H} \) be a controlled hybrid system and \( \mathcal{H}_\varepsilon \) be its relaxation. Let \( p \in \mathcal{M}_\varepsilon, u \in BV(\mathbb{R}, U) \),
\( x: I \rightarrow \mathcal{M}_\varepsilon \) be an execution of \( \mathcal{H} \) with initial condition \( p \) and control \( u \), and let \( x_\varepsilon : I_\varepsilon \rightarrow \mathcal{M}_\varepsilon \) be a
corresponding relaxed execution of \( x \). Assume that the following conditions are satisfied:

Require: \( t = 0, j \in \mathcal{J}, p \in D_j, \) and \( u \in BV(\mathbb{R}, U) \).
1: Set \( x(0) = p \).
2: loop
3: Let \( \gamma: I \rightarrow D_j \), the maximal integral curve of \( f_j \) with control \( u \) such that \( \gamma(t) = x_\varepsilon(t) \).
4: Let \( t' = \sup I \) and \( x_\varepsilon(s) = \gamma(s) \) for each \( s \in [t, t') \). \( \triangleright \) Note if \( t' < \infty \), then \( \gamma(t') \in \partial D_j \).
5: if \( t' = \infty \) or \( \nexists \varepsilon \in N_j \) such that \( \gamma(t') \in G_e \) then
6: Stop.
7: end if
8: Let \( (j, j') \in N_j \) such that \( \gamma(t') \in G_{(j,j')} \), thus \( (\gamma(t'), 0) \in S_{(j,j')}^{\varepsilon} \).
9: Set \( x_\varepsilon(t' + \tau) = (\gamma(t'), \tau) \) for each \( \tau \in [0, \varepsilon) \).
10: Set \( x_\varepsilon(t' + \varepsilon) = R_{(j,j')}^\varepsilon(\gamma(t'), \varepsilon) \), \( t = t' + \varepsilon \), and \( j = j' \). \( \triangleright \) Note \( (\gamma(t'), \varepsilon) \sim x_\varepsilon(t' + \varepsilon) \).
11: end loop

Fig. 7. Algorithm to construct a relaxed execution of a relaxation of a controlled hybrid system, \( \mathcal{H}_\varepsilon \).
A relaxed execution (left) and its discrete approximation (right).

\( \gamma \) where we have used a standard property of the functions of bounded variation (Exercise 5.1 in [28]). Thus, \( C_{\epsilon} \parallel x_{\tau} - x_{\tau + \epsilon} \parallel \leq K(\epsilon) \).

\( 0 \leq T < \infty \) such that for each \( \epsilon \) small enough, \([0, T] \subset I \cap I^\epsilon\) if \( x \) has a finite number of discrete transitions, and \([0, T] \subset I \cap I^\epsilon\) if \( x \) is Zeno.

Then, \( \lim_{\epsilon \to 0} \rho_{[0, T]}^{\epsilon}(x, x^\epsilon) = 0 \).

**Proof:** We provide the main arguments of the proof, omitting some details in the interest of brevity. First, given \( j \in J \) and \([\tau, \tau') \subset [0, T]\) such that \( x(t) \in D_j \) for each \( t \in [\tau, \tau') \), then, since \( x_{|_{\tau, \tau'}} \) is absolutely continuous, for each \( t, t' \in [\tau, \tau') \),

\[
d_i, M^\epsilon(x(t), x(t')) \leq L_{d, D_j}(x_{|_{\tau, \tau'}}) = \int_t^{t'} \| f_j(s, x(s), u(s)) \| \, ds \leq K(t' - t),
\]

where \( K = \sup\| f_j(t, x, u) \|_j \in J \), \( t \in [0, T] \), \( x \in M^\epsilon \), \( u \in U \) < \infty.

Second, let \( k \in \mathbb{N} \) and \( \{\lambda_i\}_{i=0}^k \subset [0, 1] \) be a sequence such that \( 0 = \lambda_0 \leq \lambda_1 \leq \ldots \leq \lambda_k = 1 \). Given \( \epsilon > 0 \), let \( \gamma: [0, 1] \to M^\epsilon \) be defined by \( \gamma(t) = x^{\epsilon}_{\tau}(t) \). Thus, by Theorem 21 and the algorithm in Figure 7, \( \gamma(0) = x^{0\epsilon}(t) = x(t) \) and \( \gamma(1) = x^{\epsilon}(t) \). Assume that \( x^\epsilon(t) \in D_j \) for each \( t \in [\tau + \epsilon, \tau' + \epsilon) \), where \([\tau, \tau')\) is as defined above. Using Picard’s Lemma (Lemma 5.6.3 in [21]), for each \( t \in [\tau + \epsilon, \tau') \),

\[
\| x^\epsilon(t + \epsilon) - x(t) \| \leq e^{L(t-\tau)} \left( \| x^\epsilon(\tau + \epsilon) - x(\tau) \| + \right.
\]

\[
+ \int_\tau^t \| f_j(s, x(s), u(s)) - f_j(s + \epsilon, x(s), u(s + \epsilon)) \| \, ds \right)
\]

\[
\leq e^{L(t-\tau)} \left( \| x^\epsilon(\tau + \epsilon) - x(\tau) \| + L \int_\tau^t \| x + u(s) - u(s + \epsilon) \| \, ds \right)
\]

\[
\leq e^{L(t-\tau)} \left( \| x^\epsilon(\tau + \epsilon) - x(\tau) \| + (L + V(u))(t - \tau) \epsilon \right),
\]

where we have used a standard property of the functions of bounded variation (Exercise 5.1 in [28]). Thus, if we assume that \( \| x^\epsilon(\tau + \epsilon) - x(\tau) \| = O(\epsilon) \), i.e. that there exists \( C > 0 \) such that \( \| x^\epsilon(\tau + \epsilon) - x(\tau) \| \leq C \epsilon\), then \( \| x^\epsilon(t + \epsilon) - x(t) \| = O(\epsilon) \) for each \( t \in [\tau + \epsilon, \tau') \). Using the same argument as above \( \| x^{\lambda_{i+1}}(t + \epsilon) - x^{\lambda_i}(t) \| = O((\lambda_{i+1} - \lambda_i) \epsilon) \), which implies that \( \gamma \) is continuous for each \( t \in [\tau + \epsilon, \tau') \), and that \( L(\gamma) = O(\epsilon) \), hence \( d_i, D_j(x^\epsilon(t + \epsilon), x(t)) = O(\epsilon) \).

Assuming now that \( x \) performs 2 discrete transitions at times \( \tau, \tau' \in [0, T] \), such that \( \tau + \epsilon < \tau' \), transitioning from mode \( j \) to \( j' \), and the from mode \( j' \) to \( j'' \). Note that, by definition, \( x_{|[\tau, \tau']} = x^{\epsilon}_{|[\tau, \tau']} \). Moreover, since \( x \) is orbitally stable, we know that \( x^\epsilon \) performs the same 2 discrete transitions for \( \epsilon \) small enough. Let \( \tau'' = \tau + \epsilon \in [0, T] \) be such that \( x^\epsilon(\tau'' + \epsilon) \in G_{(j', j'')} \). Note that \( |\tau'' - \tau'| = O(\epsilon) \) since \( x^\epsilon \to x \)
uniformly and \( x \) is Lipschitz continuous (both propositions shown above). Assume that \( \tau' \leq \tau^\varepsilon + \varepsilon \) and consider the following upper bounds:

1. If \( t \in [\tau, \tau + \varepsilon) \), then \( x(t) \in D_J' \) and \( x^\varepsilon(t) \in S_{(j,J')}^\varepsilon \), thus:
   \[
   d_i,M^\varepsilon (x(t), x^\varepsilon(t)) \leq d_i,D_J' (x(t), x(\tau)) + d_{S_{(j,J')}^\varepsilon} (x(\tau), x^\varepsilon(t)) = O(\varepsilon). \tag{19}
   \]

2. If \( t \in [\tau + \varepsilon, \tau') \), then \( x(t), x^\varepsilon(t) \in D_J' \), thus, using the bound obtained above:
   \[
   d_i,M^\varepsilon (x(t), x^\varepsilon(t)) \leq d_i,D_J' (x(t), x(t - \varepsilon)) + d_i,D_J' (x(t - \varepsilon), x^\varepsilon(t)) = O(\varepsilon). \tag{20}
   \]

3. If \( t \in [\tau', \tau^\varepsilon + \varepsilon) \), then \( x(t) \in D_J'' \) and \( x^\varepsilon(t) \in D_J' \), thus, denoting \( \lim_{t \rightarrow \tau'} x(t) = x(\tau' -) \):
   \[
   d_i,M^\varepsilon (x(t), x^\varepsilon(t)) \leq d_i,D_J'' (x(t), x(\tau')) + d_{S_{(j,J'')}^\varepsilon} (x(\tau'), x(\tau' -)) + d_i,D_J' (x(\tau' -), x^\varepsilon(\tau^\varepsilon + \varepsilon)) + d_i,D_J' (x^\varepsilon(\tau^\varepsilon + \varepsilon), x^\varepsilon(t)) \tag{21}
   \]
   \[
   \leq O(\varepsilon).
   \]

4. If \( t \in [\tau^\varepsilon + \varepsilon, \tau^\varepsilon + 2\varepsilon) \), then \( x(t) \in D_J'' \) and \( x^\varepsilon(t) \in S_{(j,J'')}^\varepsilon \), thus:
   \[
   d_i,M^\varepsilon (x(t), x^\varepsilon(t)) \leq d_i,D_J'' (x(t), x(\tau')) + d_{S_{(j,J'')}^\varepsilon} (x(\tau'), x^\varepsilon(t)) \leq O(\varepsilon). \tag{22}
   \]

5. If \( t \in [\tau^\varepsilon + 2\varepsilon, T] \), then \( x(t), x^\varepsilon(t) \in D_J'' \), thus we get the same bound as in case 2.

Therefore, \( \rho^\varepsilon_{[0,T]} (x, x^\varepsilon) = O(\varepsilon) \) as desired. Note that the general case, with an arbitrary number of discrete transitions and where the discrete transitions of \( x^\varepsilon \) occur before the discrete transitions of \( x \), follows by using a similar argument as above by properly considering the time intervals and then applying the upper bounds inductively.

Next, let us consider the case when \( x \) is a Zeno execution that accumulates on \( p' \). Let \( \delta > 0 \), then \( x|_{[T - \delta, T]} \) has a finite number of discrete transitions, and as shown above, \( d_{i,M^\varepsilon} (x(T - \delta), x^\varepsilon(T - \delta)) = O(\varepsilon) \). Moreover, \( d_{i,M^\varepsilon} (x(T - \delta), x(T)) = O(\delta) \) and \( d_{i,M^\varepsilon} (x^\varepsilon(T - \delta), x^\varepsilon(T)) = O(\delta) \) for each \( t \in [T - \delta, T] \). The conclusion follows by noting that these bounds are valid for each \( \delta > 0 \).

### C. Discrete Approximations

Finally, we are able to define the discrete approximation of a relaxed execution, which is constructed as an extension of any existing ODE numerical integration algorithm. Given a controlled hybrid system \( \mathcal{H}, \mathcal{A}^h_j : \mathbb{R} \times \mathbb{R}^n \times U \rightarrow \mathbb{R}^n \), where \( h > 0 \) and \( j \in J \), is a numerical integrator of order \( \omega \), if given \( p \in D_J \), \( u \in BV(\mathbb{R}, U) \), \( x \) the maximal integral curve of \( f_j \) with initial condition \( p \) and control \( u \), \( N = \lfloor \frac{T}{h} \rfloor \), and a sequence \( \{z_k\}_{k=0}^N \) with \( z_0 = p \) and \( z_{k+1} = \mathcal{A}^h_j (kh, z_k, u(kh)) \), then \( \sup \{ \|x(kh) - z_k\| : k \in \{0, \ldots, N\} \} = O(h^\omega) \). This definition of numerical integrator is compatible with commonly used algorithms, including Forward and Backward Euler algorithms and the family of Runge–Kutta algorithms (Chapter 7 in [29]). The algorithm in Fig. 9 defines a discrete approximation of a relaxed execution of \( \mathcal{H}^\varepsilon \). The resulting discrete approximation, for a step size \( h > 0 \), denoted by \( z^{\varepsilon,h} \), is a function from a closed interval \( I \in [0, \infty) \) to \( \mathcal{M}^\varepsilon \).

We now make several remarks about the algorithm in Fig. 9. First, the condition in Step 4 can only be satisfied, i.e. the Algorithm only stops, if \( z^{\varepsilon,h}(t_k) \in \partial D_j \) and \( f_j(t_k, z^{\varepsilon,h}(t_k), u(t_k)) \) is outward–pointing, since otherwise a smaller step–size would produce a valid point. Second, the function \( z^{\varepsilon,h} \) is continuous on \( \mathcal{M}^\varepsilon \). Third, and most importantly, similar to the algorithm in Fig. 7 the curve assigned to \( z^{\varepsilon,h} \) in Step 11 is exactly the maximal integral curve of \( f_\varepsilon \) while on the strip. By relaxing the guards using strips, and then endowing the strips with a trivial vector field, we avoid having to find the exact point where the trajectory intersects a guard. Our relaxation does introduce an error in the approximation, but as we show in Theorem 27, the error is of order \( \varepsilon \). Fig. 85 shows a discrete approximation produced by the algorithm in Fig. 9 as it performs a mode transition.
Require: \( h > 0, k = 0, j \in J, \) and \( p \in D_j \).

1. Set \( t_0 = 0 \) and \( z^{\varepsilon,h}(0) = p \).
2. \textbf{loop}
3. Find \( n' = \inf\{ n \in \mathbb{N} \mid \mathcal{A}^{h_{2^n}}_{J_j}(t_k, z^{\varepsilon,h}(t_k), u(t_k)) \in D_j^\varepsilon \} \)
4. if \( n' = \infty \) then
   \[ \text{return } z^{\varepsilon,h} |_{[0,t_k]} \]
5. end if
6. Set \( t_{k+1} = t_k + h2^{-n'} \), and \( z^{\varepsilon,h}(t_{k+1}) = \mathcal{A}^{h_{2^n'}}_{J_j}(t_k, z^{\varepsilon,h}(t_k), u(t_k)) \).
7. Set \( z^{\varepsilon,h}(t) = \frac{t_{k+1} - t_k}{t_{k+1} - t_{k-1}} z^{\varepsilon,h}(t_k) + \frac{t_{k-1} - t_k}{t_{k+1} - t_{k-1}} z^{\varepsilon,h}(t_{k+1}) \) for each \( t \in [t_k, t_{k+1}] \).
8. if \( \exists (j,j') \in \mathcal{N}_j \) such that \( z^{\varepsilon,h}(t_{k+1}) \in S^{\varepsilon}_{(j,j')} \) then
   \[ \text{Set } (q, \tau) = z^{\varepsilon,h}(t_{k+1}) \in S^{\varepsilon}_{(j,j')}, \text{ and } t_{k+2} = t_{k+1} + \varepsilon - \tau. \]
9. else
   \[ \text{Set } z^{\varepsilon,h}(t_{k+2}) = R^{\varepsilon}_{(j,j')}(q, \varepsilon), k = k + 2, \text{ and } j = j'. \]
\[ \text{\textbf{loop}} \]
10. \[ \text{Set } k = k + 1. \]
11. \[ \text{end if} \]

Fig. 9 Discrete approximation of a relaxed execution of the relaxation of a controlled hybrid system \( \mathcal{H}^\varepsilon \).

**Theorem 27.** Let \( \mathcal{H} \) be a controlled hybrid system and \( \mathcal{H}^\varepsilon \) its relaxation. Let \( p \in \mathcal{M}, u \in BV(\mathbb{R}, U) \), and let \( x^\varepsilon : I^\varepsilon \to M^\varepsilon \) be a orbitally stable execution of \( \mathcal{H} \) with initial condition \( p \) and control \( u \). Furthermore, let \( x^\varepsilon : I^\varepsilon \to M^\varepsilon \) be a relaxed execution with initial condition \( p \) and control \( u \), and let \( z^{\varepsilon,h} : I^{\varepsilon,h} \to M^{\varepsilon} \) be its discrete approximation. If \( [0,T] \subset I^{\varepsilon,h} \cap I^\varepsilon \) for each \( \varepsilon \) and \( h \) small enough, then there exists \( C > 0 \) such that \( \lim_{h \to 0} \rho_{[0,T]}(x^\varepsilon, z^{\varepsilon,h}) \leq C \varepsilon \).

Proof: As we have done with the previous proofs, we only provide a sketch of the argument in the interest of brevity. Assume that \( x^\varepsilon \) performs a single discrete transition in the interval \( [0,T] \) for each \( \varepsilon \) small enough, crossing the guard \( G_{(j,j')} \) at time \( \tau^\varepsilon \). Then, since \( x \) is orbitally stable and \( \mathcal{A}^h \) is convergent with order \( \omega \), for \( \varepsilon \) and \( h \) small enough \( z^{\varepsilon,h} \) also crosses guard \( G_{(j,j')} \) at time \( \tau^\varepsilon \) for each \( \varepsilon \) small enough, crossing the guard \( G_{(j,j')} \) at time \( \tau_{k'}^\varepsilon \) for some \( k' \in \mathbb{N} \) is the set of time samples associated to \( z^{\varepsilon,h} \). Moreover, since \( x^\varepsilon(0) = z^{\varepsilon,h}(0) \), then for each \( \delta > 0, |t_{k+1} - t_{k'}| \leq \delta + O(h^\omega) \) and \( |t_{k+2} - \tau^\varepsilon + \varepsilon| = O(h^\omega) \).

Define the following times:
\[
\sigma_m = \min\{t_{k' + 1}, \tau^\varepsilon \}, \quad \sigma_M = \max\{t_{k' + 1}, \tau^\varepsilon \}, \quad \nu_m = \min\{t_{k' + 2}, \tau^\varepsilon + \varepsilon \}, \quad \nu_M = \max\{t_{k' + 2}, \tau^\varepsilon + \varepsilon \},
\] (23)
and, in order to simplify our argument, assume that \( \sigma_M \leq \nu_m \). Then on the interval \([0, \sigma_m]\) we get convergence due to \( \mathcal{A}^h \). On the interval \([\sigma_m, \sigma_M]\) one execution has transitioned into a strip, while the other is still governed by the vector field on \( D_j \). On the interval \([\sigma_M, \omega_m]\) both executions are inside the strip, and on the interval \([\omega_m, \omega_M]\) one execution has transitioned to a new domain, while the second is still on the strip. After time \( \omega_M \) both executions are in a new domain, and we can repeat the process.

Consider the following cases:

1. By the convergence of algorithm \( \mathcal{A}^h \), \( d_{i,M^\varepsilon}(x^\varepsilon(\sigma_m), z^{\varepsilon,h}(\sigma_m)) = O(h^\omega) \).
2. Using (17) from the proof of Theorem 26
\[
d_{i,M^\varepsilon}(x^\varepsilon(\sigma_M), z^{\varepsilon,h}(\sigma_M)) \leq d_{i,M^\varepsilon}(x^\varepsilon(\sigma_M), x^\varepsilon(\sigma_m)) + d_{i,M^\varepsilon}(x^\varepsilon(\sigma_m), z^{\varepsilon,h}(\sigma_m)) + \quad d_{i,M^\varepsilon}(z^{\varepsilon,h}(\sigma_m), z^{\varepsilon,h}(\sigma_M)) = O(h^\omega). \tag{24}
\]
3. Using the same argument as in the inequality above,
\[
d_{i,M^\varepsilon}(x^\varepsilon(\nu_m), z^{\varepsilon,h}(\nu_m)) \leq d_{i,M^\varepsilon}(x^\varepsilon(\sigma_M), z^{\varepsilon,h}(\sigma_M)) + 2\varepsilon. \tag{25}
\]
(4) Finally, again using the same argument as in case 2
\[ d_{i,M^e} \left(x^e(\nu_M), z^{e,h}(\nu_M)\right) \leq d_{i,M^e} \left(x^e(\nu_m), z^{e,h}(\nu_m)\right) + O(h^\omega). \]  
(26)

The generalization to any relaxed execution defined on \( M^e \) and its discrete approximation follows by noting that they perform a finite number of discrete jumps on any bounded interval and that \( \delta' \) can be chosen arbitrarily small.

Next, we state the main result of this Section, which is a result of Theorems 26 and 27.

**Corollary 28.** Let \( H \) be a hybrid dynamical system and \( H^e \) be its relaxation. Let \( p \in M, u \in BV(\mathbb{R}, U) \), \( x : I \rightarrow M^e \) be an execution of \( H \) with initial condition \( p \) and control \( u \), \( x^e : I^e \rightarrow M^e \) be its corresponding relaxed execution, and \( z^{e,h} : I^{e,h} \rightarrow M^e \) be its corresponding discrete approximation. If the following conditions are satisfied:

1. \( x \) has a finite number of mode transitions or is a Zeno execution that accumulates,
2. \( x \) is orbitally stable,
3. \( [0,T] \subset I \cap I^e \cap I^{e,h} \) for each \( \varepsilon \) and \( h \) small enough,

then \( \lim_{\varepsilon,h \rightarrow 0} \rho^e_{[0,T]}(x, z^{e,h}) = 0. \)

**Proof:** Note that, by Theorem 26 together with the Triangle Inequality, this corollary is equivalent to proving that \( \rho^e_{[0,T]}(x^e, z^{e,h}) \rightarrow 0 \) as both \( \varepsilon, h \rightarrow 0 \). Hence we show that \( \rho^e_{[0,T]}(x^{e}, z^{e,h}) \) converges uniformly on \( h \) as \( \varepsilon \rightarrow 0 \). Using an argument similar to the one in the proof of Theorem 7.9 in [30], proving the uniform convergence on \( h \) is equivalent to showing that \( \lim_{h \rightarrow 0} \limsup_{\varepsilon \rightarrow 0} \rho^e_{[0,T]}(x^{e}, z^{e,h}) = 0 \), but this is clearly true by Theorem 27 therefore obtaining our desired result.

V. Examples

We present three examples where our discrete approximation algorithm is applied, first comparing its performance to existing state-of-the-art algorithms in Section V-A, then showing its use while computing the trajectories of a legged-locomotion model in Section V-C, and finally implementing a benchmark hybrid system verification example in Section V-B. It is important to note that, to the best of our knowledge, there are no results in the literature that can allow us to prove the orbital stability of these examples. We intuitively expect the examples in Section V-A and V-C to be orbitally stable, and the example in Section V-B to be non-orbitally stable for some initial conditions (i.e. initial conditions whose trajectories hit corners between different regions). In spite of our inability to formally check the orbital stability of the examples, we experimentally checked that the trajectories in each case were continuous with respect small variations of the initial conditions and control inputs.

A. Forced Linear Oscillator with Stop

We consider a single degree-of-freedom oscillator consisting of a mass that is externally forced and can impact a plane fixed rigid stop, as in Fig. 10a. The state of the oscillator is the position, \( x(t) \in \mathbb{R} \), and velocity, \( \dot{x}(t) \in \mathbb{R} \), of the mass. The oscillator is forced with a control \( u \in BV(\mathbb{R}, \mathbb{R}) \). The oscillator is modeled as a controlled hybrid system with a single mode, denoted \( D \), and single guard corresponding to the mass impacting the stop with non-negative velocity, denoted \( G \):

\[ D = \{ (x(t), \dot{x}(t)) \in \mathbb{R}^2 \mid x(t) \leq x_{\text{max}} \} \]
\[ G = \{ (x(t), \dot{x}(t)) \in \mathbb{R}^2 \mid x(t) = x_{\text{max}}, \dot{x}(t) \geq 0 \} \]

Upon impact, the state is updated using the reset map \( R(x, \dot{x}) = (x, -c \dot{x}) \), where \( c \in [0,1] \) is the coefficient of restitution. Within the single domain, the dynamics of the system are governed by \( \ddot{x}(t) + 2a \dot{x}(t) + \omega^2 x(t) = m^{-1} u(t) \), where \( \omega = \sqrt{m^{-1}k} \), \( a = 0.5 m^{-1} \mu \), \( k \) is the spring constant, and \( \mu \) is the damping coefficient.
Given an initial condition \((x(t_0), \dot{x}(t_0)) = (x_0, \dot{x}_0)\) \(\in D\), the oscillator’s motion is analytically determined by \(x(t) = e^{-at} (A_n \cos(\omega t) + B_n \sin(\omega t)) + \tilde{\omega}^{-1} \int_0^t u(s) e^{-a(t-s)} \sin(\tilde{\omega}(t-s)) \, ds\) for each \(t \in [t_{n-1}, t_n]\), where \(\tilde{\omega} = \sqrt{\omega^2 - \alpha^2}\) (assuming that the damping is sub–critical), with \(t_n\) such that \(x(t_n^-) = x_{\text{max}}\) for each \(n \in \mathbb{N}\), and \(A_n\) and \(B_n\) are determined by the given initial conditions when \(n = 0\), or those determined by applying the reset map to \(x(t_n^-)\) when \(n \geq 1\). Note that determining the impact times can be done analytically. The analytical solution holds provided that the mass does not stick to the stop, since in that case the dynamics are given by \(\ddot{x}(t) + 2a\dot{x}(t) + \omega^2 x(t) = m^{-1} (u(t) + \lambda(t))\), where \(\lambda(t) \in \mathbb{R}\) denotes the force generated by the stop to prevent movement. This equation holds as long as \(x(t) = x_{\text{max}}\), \(\dot{x}(t) = \ddot{x}(t) = 0\), and the reaction of the stop is negative, i.e. \(\lambda(t) \geq m\omega^2 x_{\text{max}}\). For the contact to cease, \(\lambda(t) = m\omega^2 x_{\text{max}}\) must become zero and change sign. Once this happens, the analytical solution can be used again to construct the motion of the mass with the initial condition \((x_{\text{max}}, 0)\).

Assuming that the forcing \(u\) is continuous (an assumption that is violated by many control schemes such as ones generated via optimal control) a convergent numerical simulation scheme, which we call the PS Method, to determine the position of a mechanical system with unilateral constraints was proposed in [14]. Fixing a step–size \(h > 0\), their approach is a two–step method that for a set of time instances,
\{t_k\}_{k \in \mathbb{N}}, computes a set of positions, \(z_{PS}: \{t_k\}_{k \in \mathbb{N}} \rightarrow \mathbb{R}\), by:

\[
\begin{align*}
    z_{PS}(t_0) &= x_0, \\
    z_{PS}(t_1) &= x_0 + \dot{x}_0 h + \frac{h^2}{2}(u(0) - 2a\dot{x}_0 - \omega^2 x_0), \\
    z_{PS}(t_{k+1}) &= -c z_{PS}(t_{k-1}) + \min \{y_{PS}(t_k), (1 + c)x_{max}\}, \\
    y_{PS}(t_k) &= \frac{1}{1 + ah}(h^2 u(t_k) + (2 - h^2 \omega^2) z_{PS}(t_k) - ((1 - c) - (1 + c)a h) z_{PS}(t_{k-1})),
\end{align*}
\]

(28)

where \(t_{k+1} = t_k + h\) for each \(k \in \mathbb{N}\).

We illustrate the performance of our approach by considering the two examples described in Table I whose solutions, which are defined for all \(t \in [0, t_{max}]\), can be computed analytically. The position component of the analytical trajectory of each example is plotted in Figs. 10b and 10c. The evaluation metric at much faster speeds. In Example 1, for step–sizes \(h = 5 \times 10^{-2}\) and relaxation parameter \(\varepsilon = 2 \times 10^{-7}\), our numerical simulation achieves a \(\hat{\rho}\) value of approximately \(10^{-4}\) while taking approximately 0.1 seconds, whereas the PS method requires a step–size of \(h = 5 \times 10^{-4}\) which takes approximately 5 seconds in order to achieve the same level of accuracy.

### B. Navigation Benchmark for Hybrid System Verification

Next, we illustrate the utility of our discrete approximation algorithm in Fig. 9 by considering three instances of a navigation benchmark proposed in [32] for hybrid system verification tools. The benchmark considers an object moving in the plane while following a set of desired velocities, \(v_{d_j} = (\sin(\frac{j\pi}{4}), \cos(\frac{j\pi}{4}))\), for \(j \in \{0, \ldots, 7\}\) where \(j\) is attributed to unit–sized squares according to a labeling map. Special symbols denoted “Goal” and “Obstacle” are reserved for a set of target and forbidden states, respectively. The labeling map for the three instances considered within this subsection are illustrated in Figure 11 where the label \(j\) in each cell refers to the desired velocity, target, or forbidden states. If the trajectory leaves the grid, the desired velocity is the velocity of the closest cell.

The dynamics of the four dimensional state, \(x(t) \in \mathbb{R}^4\), are given by \(\dot{x}(t) = v(t)\), and \(\dot{v}(t) = A(\nu(t) - v_{d_j})\), where \(A = \begin{pmatrix} -1.2 & 0.1 \\ 0.1 & -1.2 \end{pmatrix}\) for the instances illustrated in Figs. 11a and 11b and \(A = \begin{pmatrix} -0.8 & -0.2 \\ -0.2 & -0.8 \end{pmatrix}\) for the instance illustrated in Fig. 11c. For each instance, we attempt to verify that for all trajectories beginning from a set of initial conditions there exists some finite time at which the “Goal” set is reached.

| Example | \(a\) | \(c\) | \(t_{max}\) | \(u(t)\) | \(x_0\) | \(\dot{x}_0\) | \(x_{max}\) | \(\omega\) |
|---------|------|-----|---------|------|-------|-------|-------|-----|
| Example 1 | 0.05 | 0.9 | 40\pi | \(20 \cos(2.5 t)\) | 11.36263 | 31.40358 | 14 | 2.5 |
| Example 2 | 0.95 | 0.5 | 4\pi  | \(\cos(t)\) | -0.8   | 0     | -0.8 | 1   |
while avoiding the “Obstacle” set. We perform this verification by discretizing over the given set of initial conditions.

For the instance illustrated in Fig. 11a, we select a set of initial conditions $x \in [0, 1] \times [0, 1]$ and $v \in [0.1, 0.5] \times [0.05, 0.25]$. By choosing 10,000 uniformly spaced points over the set of initial conditions, a step–size of $10^{-1}$, and relaxation size of $10^{-3}$, we are able to verify this system in approximately 100 seconds. For the instance illustrated in Fig. 11b, we select a set of initial conditions $x \in [3, 4] \times [3, 4]$ and $v \in [-1, 1] \times [-1, 1]$. This instance fails the verification task as trajectories are unable to reach the “Goal” set, but do not ever intersect with the “Obstacle” set. By choosing 10,000 uniformly spaced points over the set of initial conditions, a step–size of $10^{-1}$, and relaxation size of $10^{-3}$, we discover that for this system the task is not verifiable in approximately 85 seconds. For the instance illustrated in Fig. 11a, we select a set of initial conditions $x \in [3, 3.5] \times [3, 3.5]$ and $v \in [0.5] \times [-0.5, 0.5]$. In this instance, verification is possible, but trajectories get close to the “Obstacle” set. By choosing 10,000 uniformly spaced points over the set of initial conditions, a step–size of $10^{-1}$, and relaxation size of $10^{-3}$, we are able to verify this system in approximately 210 seconds.

C. Simultaneous Transitions in Models of Legged Locomotion

As a terrestrial agent traverses an environment, its appendages intermittently contact the terrain. Since the equations governing the agent’s motion change with each limb contact, the dynamics are naturally modeled by a controlled hybrid system with discrete modes corresponding to distinct contact configurations. Because the dynamics of dexterous manipulation are equivalent to that of legged locomotion [33], such controlled hybrid systems model a broad and important class of dynamic interactions between an agent and environment.

Legged animals commonly utilize gaits that, on average, involve the simultaneous transition of multiple limbs from aerial motion to ground contact [34], [35]. Similarly, many multi–legged robots enforce simultaneous leg touchdown via virtual constraints implemented algorithmically [36], [37] or physical constraints implemented kinematically [38], [39]. Trajectories modeling such gaits pass through the intersection of multiple transition surfaces in the corresponding controlled hybrid system models. Therefore simulation of this frequently–observed behavior requires a numerical integration scheme that can accommodate overlapping guards. The algorithm in Fig. 9 has this capability, and to the best of our knowledge is the only existing algorithm possessing this property. We demonstrate this advanced capability using a pronking gait in a sagittal–plane locomotion model.

Fig. 12a contains an extension of the “Passive RHEx–runner” in [40] that allows pitching motion. A rigid body with mass $m$ and moment–of–inertia $I$ moves in the sagittal plane under the influence of
Fig. 12. Schematic and discrete mode diagram for the sagittal–plane locomotion model.

Fig. 13. Snapshots of pronk at discrete transition times from initial condition \((x_0, z_0, \theta_0, \dot{x}_0, \dot{z}_0, \dot{\theta}_0) = (0, 1, 1, 0, 3, 4, 0, 0)\), parameters \((m, I, k, \ell, d, g, \psi) = (1, 1, 30, 1, 1, 9.81, \pi/5)\), step size \(h = 10^{-3}\), relaxation parameter \(\varepsilon = 10^{-2}\) (left). Same as before, but with \(\theta_0 = -0.4\) (right).

Gravity \(g\). Linear leg–springs are attached to the body via a frictionless pin joint located symmetrically at distance \(d/2\) from the center–of–mass. The leg–springs are massless with linear stiffness \(k\), rest length \(\ell\), and make an angle \(\psi\) with respect to the body while in the air. When a foot touches the ground it attaches via a frictionless pin joint, and it detaches when the leg extends to its rest length.

A pronk is a gait wherein all legs touch down and lift off from the ground at the same time \([34], [35]\). Due to symmetries in our model, motion with pitch angle \(\theta = 0\) for all time is invariant. Therefore periodic orbits for the spring–loaded inverted pendulum model in \([41]\) correspond exactly to pronking gaits for our model. Fig. 12b contains a projection of the guards \(G_{(a,l)}, G_{(a,r)}, G_{(l,g)}, G_{(r,g)}\) in \((\theta, z)\) coordinates for the transition from the aerial domain \(D_a\) to the ground domain \(D_g\) through left stance \(D_l\) and right stance \(D_r\). The pronking trajectory is illustrated by a downward–pointing vertical arrow, and a nearby trajectory initialized with negative rotational velocity is illustrated by a dashed line. Fig. 13 contains snapshots from these simulations.

The \(\theta_0 = 0\) trajectory in Fig. 12b clearly demonstrates the need for a simulation algorithm that allows the intersection of multiple transition surfaces. We emphasize that our state–space metric was necessary to derive a convergent numerical approximation for this execution: since the discrete mode sequence differs for any pair of trajectories arbitrarily close to the \(\theta_0 = 0\) execution that pass through the interior of \(D_l\) and \(D_r\), respectively, a naïve application of the trajectory–space metric in \([15]\) would yield a distance larger than unity between the pair. Consequently no numerical simulation algorithm can be shown to converge to the \(\theta_0 = 0\) execution using a trajectory–space metric.

Another interesting property of this example is that it is possible to show (by carefully studying the transitions between vector fields through the guards) that the hybrid quotient space \(\mathcal{M}\) is a smooth 6–dimensional manifold near the pronk execution, and that the piecewise–defined dynamics yield a continuously–differentiable vector field on this quotient.

As a final note to practitioners, we remark that our algorithm does not require a specialized mechanism...
to handle overlapping guards or control inputs: a single code will accurately simulate any orbitally stable execution of the hybrid system under investigation, dramatically simplifying practical implementation.

VI. CONCLUSION

We developed an algorithm for the numerical simulation of controlled hybrid systems and proved the uniform convergence of our approximations to executions using a novel metrization of the controlled hybrid system’s state space. The metric and the algorithm impose minimal assumptions on the hybrid system beyond those required to guarantee existence and uniqueness of executions. Beyond their immediate utility, it is our conviction that these tools provide a foundation for formal analysis and computational controller synthesis in hybrid systems.

REFERENCES

[1] A. Nerode and W. Kohn, “Models for hybrid systems: Automata, topologies, controllability, observability,” in Proceedings of the Workshop on the Theory of Hybrid Systems, 1993, pp. 317–356.
[2] S. N. Simic, K. H. Johansson, J. Lygeros, and S. S. Sastry, “Towards a geometric theory of hybrid systems,” Dynamics of Continuous Discrete and Impulsive Systems Series B: Applications & Algorithms, vol. 12, no. 5/6, pp. 649–688, 2005.
[3] A. D. Ames and S. S. Sastry, “A homology theory for hybrid systems: Hybrid homology,” in Proceedings of the 8th International Workshop on Hybrid Systems: Computation and Control, 2005, pp. 86–102.
[4] K. H. Johansson, M. Egerstedt, J. Lygeros, and S. S. Sastry, “On the regularization of Zeno hybrid automata,” Systems & Control Letters, vol. 38, no. 3, pp. 141–150, 1999.
[5] L. Tavernini, “Differential automata and their discrete simulators,” Nonlinear Analysis: Theory, Methods & Applications, vol. 11, no. 6, pp. 665–683, 1987.
[6] D. Gokhman, “Topologies for hybrid solutions,” Nonlinear Analysis: Hybrid Systems, vol. 2, no. 2, pp. 468–473, 2008.
[7] D. Pollard, Convergence of Stochastic Processes, ser. Springer Series in Statistics. Springer, 1984.
[8] C. Cai, R. Goebel, and A. R. Teel, “Relaxation results for hybrid inclusions,” Set-Valued Analysis, vol. 16, pp. 733–757, 2008.
[9] R. G. Sanfelice and A. R. Teel, “Dynamical properties of hybrid systems simulators,” Automatica, vol. 46, no. 2, pp. 239–248, 2010.
[10] M. Carver, “Efficient integration over discontinuities in ordinary differential equation simulations,” Mathematics and Computers in Simulation, vol. 20, no. 3, pp. 190–196, 1978.
[11] J. Guckenheimer and A. Nerode, “Simulation for hybrid systems and nonlinear control,” in Proceedings of the 31st IEEE Conference on Decision and Control, 1992, pp. 2980–2981.
[12] J. Esposito, V. Kumar, and G. J. Pappas, “Accurate event detection for simulating hybrid systems,” in Proceedings of the 4th International Workshop on Hybrid Systems: Computation and Control, 2001, pp. 204–217.
[13] E. Polak, Optimization: Algorithms and Consistent Approximation, ser. Applied Mathematical Sciences. Springer, 1997.
[14] G. Folland, Real Analysis: Modern Techniques and Their Applications, 2nd ed., ser. Pure and Applied Mathematics. John Wiley & Sons, 1999.
[15] J. M. Lee, Introduction to Smooth Manifolds, ser. Graduate Texts in Mathematics. Springer, 2003.
[16] J. Munkres, Topology. Prentice-Hall, 2000.
[17] E. J. McShane, “Extension of range of functions,” Bulletin of the American Mathematical Society, vol. 40, no. 12, pp. 837–843, Jan 1934.
[18] A. D. Ames, “Generic asymptotic error estimates for the numerical simulation of hybrid systems,” Nonlinear Analysis: Hybrid Systems, vol. 3, no. 2, pp. 108–123, 2009.
[19] M. Carver, “Dynamical properties of hybrid systems simulators,” in Proceedings of the 30th IEEE Conference on Decision and Control, 2011, pp. 3958–3965.
[20] J. L. Kelley, General Topology, ser. Graduate Texts in Mathematics. Springer, 1955.
[21] E. Polak, Optimization: Algorithms and Consistent Approximation, ser. Applied Mathematical Sciences. Springer, 1997.
[22] G. Folland, Real Analysis: Modern Techniques and Their Applications, 2nd ed., ser. Pure and Applied Mathematics. John Wiley & Sons, 1999.
[31] O. Janin and C. Lamarque, “Comparison of several numerical methods for mechanical systems with impacts,” *International Journal for Numerical Methods in Engineering*, vol. 51, no. 9, pp. 1101–1132, 2001.

[32] A. Fehnker and F. Ivančić, “Benchmarks for hybrid systems verification,” in *Proceedings of the 7th International Workshop on Hybrid Systems: Computation and Control*, 2004, pp. 326–341.

[33] A. M. Johnson, G. C. Haynes, and D. E. Koditschek, “Standing self–manipulation for a legged robot,” in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, 2012.

[34] R. M. Alexander, “The gaits of bipedal and quadrupedal animals,” *The International Journal of Robotics Research*, vol. 3, no. 2, pp. 49–59, 1984.

[35] M. Golubitsky, I. Stewart, P. L. Buono, and J. J. Collins, “Symmetry in locomotor central pattern generators and animal gaits,” *Nature*, vol. 401, no. 6754, pp. 693–695, 1999.

[36] M. Raibert, M. Chepponis, and H. Brown Jr., “Running on four legs as though they were one,” *IEEE Journal of Robotics and Automation*, vol. 2, no. 2, pp. 70–82, 1986.

[37] U. Saranli, M. Buehler, and D. E. Koditschek, “RHex: A simple and highly mobile hexapod robot,” *The International Journal of Robotics Research*, vol. 20, no. 7, pp. 616–631, 2001.

[38] S. Kim, J. Clark, and M. Cutkosky, “iSprawl: Design and tuning for high–speed autonomous open–loop running,” *The International Journal of Robotics Research*, vol. 25, no. 7, pp. 616–631, 2001.

[39] A. Hoover, S. Burden, X. Fu, S. S. Sastry, and R. Fearing, “Bio–inspired design and dynamic maneuverability of a minimally actuated six–legged robot,” in *Proceedings of the 3rd IEEE International Conference on Biomedical Robotics and Biomechatronics*, 2010, pp. 869–876.

[40] J. Seipel and P. Holmes, “Three–dimensional translational dynamics and stability of multi–legged runners,” *The International Journal of Robotics Research*, vol. 25, no. 9, pp. 889–902, 2006.

[41] R. M. Ghiglizia, R. Allendorfer, P. Holmes, and D. E. Koditschek, “A simply stabilized running model,” *SIAM Journal on Applied Dynamical Systems*, vol. 2, no. 2, pp. 187–218, 2003.