On the Relaxation of Hybrid Dynamical Systems

Tyler Westenbroek
S. Shankar Sastry
Humberto Gonzalez

ABSTRACT

Hybrid dynamical systems have proven to be a powerful modeling abstraction, yet fundamental questions regarding the dynamical properties of these systems remain. In this paper, we develop a novel class of relaxations which we use to recover a number of classic systems theoretic properties for hybrid systems, such as existence and uniqueness of trajectories, even past the point of Zeno. Our relaxations also naturally give rise to a class of provably convergent numerical approximations, capable of simulating through Zeno. Using our methods, we are also able to perform sensitivity analysis about nominal trajectories undergoing a discrete transition – a technique with many practical applications, such as assessing the stability of periodic orbits.

ACM Reference format:
Tyler Westenbroek, S. Shankar Sastry, and Humberto Gonzalez . 2016. On the Relaxation of Hybrid Dynamical Systems . In *Proceedings of ACM Conference, Washington, DC, USA, July 2017 (Conference'17),* 13 pages.
DOI: 10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

While hybrid dynamical systems have proven to be a highly expressive modeling framework, the flexibility they provide does not come without its challenges. Despite considerable efforts to extract classic systems theoretic properties from hybrid systems in works such as [16] and [13], fundamental questions regarding even the existence and uniqueness of their executions abound, as the interplay between their discrete and continuous dynamics is not fully understood.

Perhaps the most notable phenomena unique to hybrid systems, Zeno executions [18] arise when an infinite number of discrete transitions occur in a finite amount of time. In order to accommodate Zeno trajectories into theoretical and computational frameworks, a number of techniques have been proposed. In [8], the authors propose techniques to regularize hybrid systems in time or space, which prevent an infinite number of transitions from occurring. Yet they are able to prove convergence for their relaxations only for Zeno executions which accumulate to a single point. In [3], the authors extend this proof of convergence to the numerical setting. Alternatively, the authors in [14] go to great lengths to identify Zeno executions, and replace them with executions of a reduced order dynamical system, in order to avoid directly handling Zeno. The authors are able to extend simulations past the point of Zeno in some cases, but the results only hold for mechanical systems.

Even if we disregard the pathologies introduced by Zeno executions, a number of theoretical and practical challenges remain to fully understand the executions of hybrid systems. The trajectories of hybrid systems may be discontinuous with respect to inputs and initial conditions [13], and in such cases may not be faithfully approximated in the numerical setting. Indeed, many works focussed on numerical integration for hybrid systems such as [3] and [5] make restrictive assumptions about the trajectories being simulated to accommodate this obstacle, and require that timesteps be placed in small neighborhoods around discrete events.

Taking steps to overcome these limitations, we introduce a novel relaxation scheme for hybrid dynamical systems. First, we demonstrate how to reduce a discrete jump of a hybrid system to the execution of a switched system. This enables us to use the solution concept of Filippov [6] to define closed form solutions for some Zeno executions of our hybrid systems. We then extend the procedure presented in [11] to regularize this collection of switched systems, which recover the sliding solutions of Filippov in the limit, and use the resulting vector fields to construct trajectories over our *relaxed hybrid systems.* To construct these relaxations we take the approach of adding an epsilon-thick strip to each of our guard sets as in [8], and endow our relaxed hybrid systems with the topology from [3].

Using this framework we extend the state of the art in several directions. Firstly, we use the limit of our relaxations to construct a novel solution concept for hybrid systems, wherein the trajectory generated by ever pair of initial conditions and inputs is unique and well defined, even past the point of Zeno. Secondly, our relaxation yields a provably convergent numerical approximations, which can approximate solutions past Zeno. Finally, we are able to perform sensitivity analysis on the trajectories of our relaxations – which has numerous practical applications, which we discuss in our closing remarks.

2 FILIPPOV SOLUTIONS

We briefly introduce the solution concept of Filippov [6] for switched systems. Consider the bimodal switched system depicted in Figure 1, where the domains $D_1, D_2 \subset \mathbb{R}^n$ are separated by the plane $G = \left\{x \in \mathbb{R}^n : g(x) = \hat{g}^T x - c = 0\right\}$,

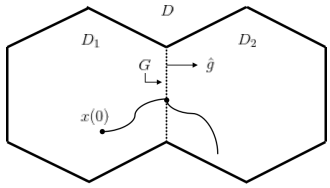

Figure 1: A bimodal switched system with a representative trajectory x.

where $\hat{g} \in \mathbb{R}^n$ is a unit vector and c is a scalar. Defining $D = D_1 \cup D_1$, and an allowable set of inputs U, the dynamics of the system are governed by $f\colon D \times U \to \mathbb{R}^n$ where

$$f(x,u) = \begin{cases} f_1(x,u), & \text{if } x \in D_1 \\ f_2(x,u), & \text{if } x \in D_2, \end{cases} \tag{1}$$

and $f_1\colon D_1 \times U \to \mathbb{R}^n$ and $f_2\colon D_2 \times U \to \mathbb{R}^n$ are both continuous but f may be discontinuous along G. In particular, we align $\hat{g}$ such that $g_e(x) \leq 0, \forall x \in D_1$, and $g_e(x) \geq 0, \forall x \in D_2$.

We may partition the set $H = G \times U$ into three distinct regions: the *crossing region* (Σ_c), the *sliding region* (Σ_s), and the *escaping region* (Σ_{es}). We characterize these regions by $\Sigma_c = \left\{(x,u) \in H\colon \left(\hat{g}^T f_1(x,u)\right)\left(\hat{g}^T f_2(x,u)\right) > 0\right\}$, $\Sigma_s = \left\{(x,u) \in H\colon \left(\hat{g}^T f_1(x,u)\right) > 0, \left(\hat{g}^T f_2(x,u)\right) < 0\right\}$, and $\Sigma_{es} = \left\{(x,u) \in H\colon \hat{g}^T f_1(x,u) \leq 0, \hat{g}^T f_2(x,u) \geq 0\right\}$.

When $(x,u) \in \Sigma_c$, the trajectory x simply crosses from one domain to the other. When $(x,u) \in \Sigma_s$, both f_1 and f_2 are pointing into the surface G, confining the trajectory to this set. One way to model trajectories in this regime is to switch between the vector fields f_1 and f_2 infinitely fast – i.e as a Zeno execution. However, Filippov solutions offer us another route to understand such systems. The *Filippov sliding vector field* for this system, $f^s\colon \Sigma_s \to \mathbb{R}^n$, is given by $f^s(x,u) = (1-\alpha(x,u))f_1(x,u) + \alpha(x,u)f_2(x,u)$, where $\alpha\colon \Sigma_s \to [0,1]$ is defined by $\alpha(x,u) = \frac{\hat{g}^T f_1(x,u)}{\hat{g}^T f_1(x,u) - \hat{g}^T f_2(x,u)}$. In particular, α is constructed such that $\forall (x,u) \in \Sigma_s$, we have $\hat{g}^T f^s(x,u) = 0$, confining the solution to G as desired. Thus, we may use the solution concept of Filippov replace some Zeno trajectories with well defined vector fields. However, when $(x,u) \in \Sigma_{es}$, the solution concept of Filippov breaks down, and there are Zeno trajectories that are left ill-defined.

3 HYBRID SYSTEMS

We introduce our class of hybrid systems, inspired by [3].

Definition 3.1. A hybrid system is a seven-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 graphical structure over $\mathcal{J}$, where edge $e = (j, j') \in \Gamma$ corresponds to a transition from j to j';
- $\mathcal{D} = \left\{D_j\right\}_{j \in \mathcal{J}}$ is the set of domains, where D_j is a compact n-dimensional polytope in $\mathbb{R}^n$, $n \in \mathbb{N}$;
- $U \subset \mathbb{R}^m$ is a compact set of inputs, $m \in \mathbb{N}$;
- $\mathcal{F} = \left\{f_j\right\}_{j \in \mathcal{J}}$ is the set of vector fields, where each $f_j\colon D_j \times U \to \mathbb{R}^n$ is continuously differentiable [1] and defines the continuous dynamics of the system on D_j[2] ;
- $\mathcal{G} = \{G_e\}_{e=(j,j') \in \Gamma}$ is the set of guards, where each $G_{(j,j')} \subset \partial D_j$ is a codimension 1 plane with corner; that is, there exists a unit vector $\hat{g}_e \in \mathbb{R}^n$ and a scalar c_e such that $G_e \subset \left\{x \in \partial D_j\colon g_e(x) = \hat{g}_e^T x - c_e = 0\right\}$[3]; and,
- $\mathcal{R} = \{R_e\}_{e=(j,j') \in \Gamma}$ is the set of reset maps where, for each $e \in \Gamma$, $R_e\colon G_e \to \partial D_{j'}$ is defined by $R_e(x) = A_e x + b_e$, where $A_e \in \mathbb{R}^{n \times n}$ and $b_e \in \mathbb{R}^n$.

When the guard $G_{(j,j')}$ is crossed, a discrete transition from mode j to j' occurs, and the continuous state is instantaneously rest by $R_{(j,j')}$. We unify our continuous and discrete state spaces using the concept of a *disjoint union*. That is, we embed our continuous domains in the space $\coprod_{j \in J} D_j = \bigcup_{j \in \mathcal{J}} D_j \times \{j\}$. By an abuse of notation, throughout the paper we shall simply use D_j to refer to $D_j \times \{j\}$. For each $j \in \mathcal{J}$, we let $\mathcal{N}_j = \{e \in \Gamma\colon \exists j' \in \mathcal{J}\ s.t.\ e = (j,j')\}$ be the *neighborhood* of $\mathcal{J}$. We let $PC([0,T],U)$ denote the class of piecewise continuous functions from the interval $[0,T]$ to U. For each $j \in \mathcal{J}$, let $f_j\colon \mathbb{R}^n \times U \to \mathbb{R}^n$ be any continuously differentiable extension to $f_j\colon D_j \times U \to \mathbb{R}^n$, guaranteed to exist by Lemma 5.6 of [10]. Abusing notation, throughout the paper, when the symbol f_j is used, it is understood that we are referring to the extended version of the function. Similarly, for each $e \in \Gamma$ we extend $R_e\colon \mathbb{R}^n \to \mathbb{R}^n$, where for each $x \in \mathbb{R}^n$ we still define $R_e(x) = A_e x + b_e$. We impose the following assumptions to simplify the discussion sliding vector fields throughout the paper.

Assumption 1. *Let $e \in \Gamma$. Then A_e is invertible, and there exists an edge $e' \in \Gamma$ such that $R_e(G_e) = G_{e'}$, $R_{e'}(G_{e'}) = G_e$, and, $\forall x \in G_e$, $x = R_{e'}(R_e(x))$.*

We say that the edge e is *reversible* if it satisfies Assumption 1, since a transition along e can be 'reversed' by a transition along e', which we refer to as the *partner* of e. We will often use e' to refer to the partner of e with out explicitly stating their relationship. The one-to-one correspondence between G_e and $G_{e'}$ defined by R_e and $R_{e'}$ will simplify our initial discussion of Filippov solutions along the guard sets of our hybrid systems. In the optional appendix, we outline

[1] This ensures continuous state trajectories are unique and well defined on our continuous domains, since continuous functions are Lipschitz over compact sets.

[2] We incur no loss of generality by considering time invariant vector fields. Indeed, one may add time as a continuous state z, with dynamics $\dot{z} = 1$ and initial condition $z(t_0) = t_0$.

[3] We choose the convention that $\hat{g}_e$ 'points out' of D_j along G_e – i.e. $g_e(x) \leq 0,\ \forall x \in D_j$.

how to overcome this assumption in theory, and in Section 8 we produce numerical examples where the edges are not reversible. However, in both cases we only consider Zeno trajectories involving at most two edges of a hybrid system.

Assumption 2. *For each pair of edges $e \neq \bar{e}$, $G_e \cap G_{\bar{e}} = \emptyset$.*

While much work has been done to extend the solution concept of Filippov to cases where multiple continuous domains interface [4], many open problems regarding the existence and uniqueness of solutions in such cases remain, and we wish to avoid such questions here in favor of presenting the main conceptual and technical components of our relaxation framework. We are currently investigating ways to extend our results to hybrid systems with non-linearities in their guard sets and reset maps, and overlapping guards. We now endow our hybrid systems with the topology from [3], which uses the concept of a *quotient space* [[9], Ch. 3].

Given a topological space $\mathcal{S}$ and a function $f\colon A \to B$, where $A, B \subset \mathcal{S}$, we define the following equivalence relation: $A \sim B = \left\{(a,b) \in \mathcal{S} \times \mathcal{S} \colon a \in f^{-1}(b)\right\}$, and denote the quotient of $\mathcal{S}$ under $A \sim B$ by $\frac{\mathcal{S}}{\Lambda_f}$. Quotienting a space is often informally referred to as applying "topological glue" – that is, for each $a \in A$ the sets a and $f(a)$ are "glued" together, becoming a single set in $\frac{\mathcal{S}}{\Lambda_f}$. We embed our hybrid systems in the quotient space defined by their collection of reset maps. For hybrid system $\mathcal{H}$ define $\widehat{R}\colon \coprod_{e\in\Gamma} G_e \to \coprod_{j\in\mathcal{J}} \partial D_{j'}$ by $\widehat{R}(x) = R_e(x)$ for each $x \in G_e$. Then the hybrid quotient space of $\mathcal{H}$ is $\mathcal{M} = \frac{\coprod_{j\in J} D_j}{\Lambda_{\widehat{R}}}$.

The construction of $\mathcal{M}$ for a bimodal hybrid system $\mathcal{H}$ is depicted in Figure 2, wherein the trajectory x undergoes a discrete transition. Note, that for partners (1,2) and (2,1), the sets $G_{(1,2)}$ and $G_{(2,1)}$, while disjoint in $D_{(1,2)} \coprod D_{(2,1)}$, compose a single *hybrid surface* in $\mathcal{M}$. Thus, the trajectories of our hybrid systems, to be defined in Section 7, are in fact continuous on this space [3]. Speaking informally, the construction of $\mathcal{M}$ reduces $\mathcal{H}$ into a switched system where the single hybrid surface $G_{(1,2)}/G_{(2,1)}$ separates D_1 and D_2. In section 5, we do in fact demonstrate how to represent a discrete transition on $\mathcal{M}$ using the execution of a switched system. This will empower us to use the solution concept of Filippov to construct Zeno (sliding) trajectories along the hybrid surface $G_{(1,2)}/G_{(2,1)}$.

To understand the main difficulty in accomplishing this task, note that, when we construct $\mathcal{M}$ and describe a transition along the edge $(1,2)$, we apply an implicit change of coordinates wherein *i*) we align the vector $\hat{g}_{(1,2)}$ with the vector $-\hat{g}_{(2,1)}$, so as a trajectory leaves D_1 it flows to the interior of D_2, and *ii*) the matrix $A_{(1,2)}$ defines a correspondence between the surfaces $G_{(1,2)}$ and $G_{(2,1)}$; that is, $A_{(1,2)}$ transforms vectors in the $(n-1)$ dimensional subspace parallel to $G_{(1,2)}$

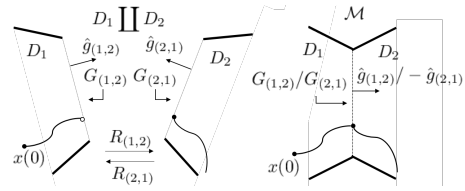

Figure 2: A hybrid trajectory x transitions from D_1 to D_2, represented on both $D_1 \coprod D_2$ (left) and $\mathcal{M}$ (right).

to lie in the $(n-1)$-dimensional subspace parallel to $G_{(2,1)}$. In Section 5 we make this transformation explicit.

Formally, in order metricize the hybrid quotient space we employ the induced length metrics from [2]. Let $d\colon \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ be a metric, then define $\tilde{d}_{\mathcal{M}}\colon \mathcal{M}\times\mathcal{M} \to \mathbb{R}$ for each $x, y \in \mathcal{M}$ by

$$\tilde{d}_{\mathcal{M}}(x,y) = \inf_{k\in\mathbb{N}} \left\{ \sum_{i=1}^{k} d(p_i, q_i) \colon x = p_1,\ y = q_k,\ q_i \sim p_{i+1} \right\}. \tag{2}$$

Intuitively, $\tilde{d}_{\mathcal{M}}(x,y)$ is simply the length of the shortest curve between x and y on $\mathcal{M}$, which may traverse multiple edges to connect the two points.

4 RELAXED HYBRID SYSTEMS

We now produce our definition for *relaxed hybrid systems* inspired by [3], attaching an ε-thick strip to each of the guard sets. In Section 6, we will define the continuously differentiable vector fields that we impart over these strips, which we will use to approximate Zeno trajectories. The relaxation of the bimodal hybrid system $\mathcal{H}$ from Figure 2, $\mathcal{H}^\varepsilon$, is depicted on the left in Figure 3.

Concretely, for each $e \in \Gamma$ we define the *relaxed strip* $S_e^\varepsilon = \{p + \hat{g}_e q \in \mathbb{R}^n \colon p \in G_e \text{ and } q \in [0,\varepsilon]\}$ and then for each $j \in \mathcal{J}$ define the *relaxed domain* $D_j^\varepsilon = D_j \cup_{e\in\mathcal{N}_j} S_e^\varepsilon$. Next, for each $e = (j,j') \in \Gamma$ we then define the *relaxed guard set* $G_e^\varepsilon = \left\{x \in S_e^\varepsilon \colon g_e^\varepsilon(x) = \hat{g}_e^T x - (c_e + \varepsilon) = 0\right\}$and define the *relaxed reset map* $R_e^\varepsilon\colon G_e^\varepsilon \to \partial D_{j'}$ by $R_e^\varepsilon(x) = R_e(P_e(x))$, where $P_e\colon \mathbb{R}^n \to \mathbb{R}^n$ is defined by $P_e(x) = x - \hat{g}_e g_e(x)$. Intuitively, P_e projects points onto the plane containing G_e, so that $R_e^\varepsilon(G_e^\varepsilon) = R_e(G_e)$. We now provide our definition of a *relaxed hybrid system*, which we take from [3].

Definition 4.1. Let $\mathcal{H}$ be a hybrid system. We then define the ε-relaxation of $\mathcal{H}$ to be the seven-tuple $\mathcal{H}^\varepsilon = (\mathcal{J}, \Gamma, \mathcal{D}^\varepsilon, U, \mathcal{F}^\varepsilon, \mathcal{G}^\varepsilon, \mathcal{R}^\varepsilon)$ where:

(1) $\mathcal{D}^\varepsilon = \left\{D_j^\varepsilon\right\}_{j\in\mathcal{J}}$ is the set of relaxed domains ;
(2) $\mathcal{F}^\varepsilon = \left\{f_j^\varepsilon\right\}_{j\in\mathcal{J}}$ is the set of relaxed vector fields, where $f_j^\varepsilon\colon D_j^\varepsilon \times U \to \mathbb{R}^n$;
(3) $\mathcal{G}^\varepsilon = \left\{G_e^\varepsilon\right\}_{e=\in\Gamma}$ is the set of relaxed guard set; and,
(4) $\mathcal{R}^\varepsilon = \left\{R_e^\varepsilon\right\}_{e=\in\Gamma}$ is the set of relaxed reset maps.

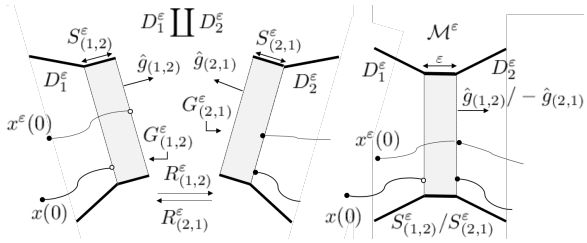

Figure 3: A hybrid trajectory x and relaxed hybrid trajectory x^ε transition from mode 1 to mode 2, represented on both $D_1^\varepsilon \coprod D_2^\varepsilon$ (left) and $\mathcal{M}^\varepsilon$ (right).

We embed our relaxed hybrid systems in the disjoint union $\coprod_{j\in\mathcal{J}} D_j^\varepsilon$ and we adopt the *relaxed hybrid quotient space* introduced in [3]. Let $\mathcal{H}^\varepsilon$ be a relaxed hybrid system and let $\widehat{R}^\varepsilon\colon \coprod_{e\in\Gamma} G_e^\varepsilon \to \coprod_{j\in\mathcal{J}} \partial D_j$ be characterized by $\widehat{R}_e^\varepsilon(x) = R_e^\varepsilon(x)$ for each $x \in G_e^\varepsilon$. We then define the relaxed hybrid quotient space of $\mathcal{H}^\varepsilon$ to be $\mathcal{M}^\varepsilon = \frac{\coprod_{j\in J} D_j^\varepsilon}{\Lambda_{\widehat{R}^\varepsilon}}$.

The construction of $\mathcal{M}^\varepsilon$ for our example bimodal hybrid system is shown on the right in Figure 3. Note that the strips $S^\varepsilon_{(1,2)}$ and $S^\varepsilon_{(2,1)}$ form a single *hybrid strip* in $\mathcal{M}^\varepsilon$ [4], and a similar change of coordinates occurs when traversing e in $\mathcal{M}^\varepsilon$, as was described for the same transition in $\mathcal{M}$. The trajectories of our relaxed hybrid systems will again be continuous on $\mathcal{M}^\varepsilon$; such a trajectory x^ε is depicted in Figure 3, where we also reproduce the trajectory x from Figure 2. By representing x on $\mathcal{M}^\varepsilon$, as in [3], we will be able to compare the distance between the trajectories of a hybrid system and its ε-relaxation using the $\widetilde{d}_{\mathcal{M}^\varepsilon}$ metric, which is defined analogously to how $\widetilde{d}_{\mathcal{M}}$ was constructed in (2). In particular, given two trajectories $x, x^\varepsilon\colon [0,T] \to \mathcal{M}^\varepsilon$, we will use the metric $\rho^\varepsilon(x, x^\varepsilon) = \sup\left(\widetilde{d}_{\mathcal{M}^\varepsilon}(x(t), x^\varepsilon(t)) | t \in [0,T]\right)$ from [3], to bound the distance between different trajectories.

5 REPRESENTING DISCRETE JUMPS WITH SWITCHED SYSTEMS

We now demonstrate how to describe a discrete transition of a hybrid system using the execution of a switched system, which will allow us to use Filippov solutions to describe the composition of continuous and discrete dynamics along guard sets. We begin by making the change of coordinates that occurs during a discrete transition explicit for a given edge $e = (j, j')$ with partner e'.

Definitively, if $\{v_e^i\}_{i=1}^{n-1}$ is a basis for the subspace parallel to G_e, then $\left\{v_e^1, \ldots, v_e^{(n-1)}, \hat{g}_e\right\}$ is a basis for $\mathbb{R}^n$, and when edge e is traversed this basis is transformed element wise

[4]This is not technically true, since for each pair of partner edges we only "glued" G_e^ε to $G_{e'}$ and $G_{e'}^\varepsilon$ to G_e. However, it is notationally cumbersome to "glue" the entire width of S_e^ε to $S_{e'}^\varepsilon$, and so we choose to abuse notation here.

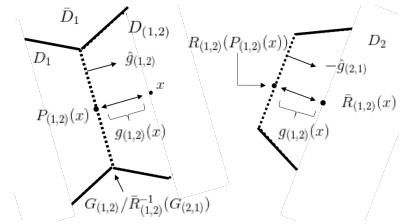

Figure 4: On the left is depicted $\bar{D}_j = D_j \cup D_e$, as well as a point $x \in D_e$. The various components of $\bar{R}_e(x)$ are depicted on $\bar{D}_j$ and $D_{j'}$ (on the right).

to the basis $\left\{A_e v_e^1, \ldots, A_e v_e^{(n-1)}, -\hat{g}_{e'}\right\}$, where $\left\{A_e v_e^i\right\}_{i=1}^{(n-1)}$ is a basis for the subspace parallel to $G_{e'}$ (indeed this set is linearly independent since we assumed A_e to be full rank), and $-\hat{g}_{e'}$ is orthogonal to this subspace.

In order to perform this change of basis automatically during simulation of a discrete transition, we will appropriately translate, rotate and resize $D_{j'}$, appending it to D_j, so we may directly simulate how a trajectory evolves into the interior of $D_{j'}$ after traversing edge e. We denote this transformed version of $D_{j'}$ by D_e, which is depicted on the left side of Figure 4. To accomplish this task define the map $\bar{R}_e\colon \mathbb{R}^n \to \mathbb{R}^n$ by $\bar{R}_e(x) = R_e(P_e(x)) - \hat{g}_{e'} g_e(x)$, and then define $D_e = \left\{x \in \mathbb{R}^n\colon \bar{R}_e(x) \in D_{j'}\right\}$. The various components of $\bar{R}_e$ are also depicted in Figure 4.

To understand the action of $\bar{R}_e$, recall that $P_e(x)$ projects points onto the plane containing G_e, so $R_e(P_e(G_e)) = G_{e'}$. Thus, the first term in $\bar{R}_e$ maintains the one-to-one correspondence between G_e and $G_{e'}$ that defines the edge; indeed note that domains D_j and D_e are separated by the surface $G_e/\bar{R}_e^{-1}(G_{e'})$. The term $-\hat{g}_{e'} g_e(x)$, on the other hand, aligns the transverse coordinates $\hat{g}_e$ and $-\hat{g}_{e'}$, in the sense that $g_{e'}(\bar{R}_e(x)) = -g_e(x)$, so that as a trajectory x leaves D_j and enters the interior of D_e, $\bar{R}_e(x)$ leaves $G_{e'}$ and enters the interior of $D_{j'}$. Another way to understand $\bar{R}_e$ is to consider the reformulation

$$\begin{aligned}
\bar{R}_e(x) &= R_e(P_e(x)) - \hat{g}_{e'} g_e(x) && (3)\\
&= R_e(x - \hat{g}_e g_e(x)) - \hat{g}_{e'} g_e(x) && (4)\\
&= A_e(I \cdot x - \hat{g}_e(\hat{g}_e^T x - c_e)) + b_e - \hat{g}_{e'}(\hat{g}_e^T x - c_e) && (5)\\
&= \bar{A}_e x + \bar{b}_e, && (6)
\end{aligned}$$

where $\bar{A}_e = A_e(I - \hat{g}_e \hat{g}_e^T) - \hat{g}_{e'} \hat{g}_e^T$ and $\bar{b}_e = A_e \hat{g}_e c_e + b_e + \hat{g}_{e'} c_e$. The matrix $\bar{A}_e$ applies the change of basis that occurs when e is traversed, and is thus invertible. In particular, the matrix $(I - \hat{g}_e \hat{g}_e^T)$ is the natural projection onto the subspace orthogonal to $\hat{g}_e$, so the term $A_e(I - \hat{g}_e \hat{g}_e^T)$ applies the change of basis along G_e and $G_{e'}$, while the dyad $-\hat{g}_{e'} \hat{g}_e^T$ rotates vectors in the direction $\hat{g}_e$ to align with $-\hat{g}_{e'}$. These definitions enable the following result.

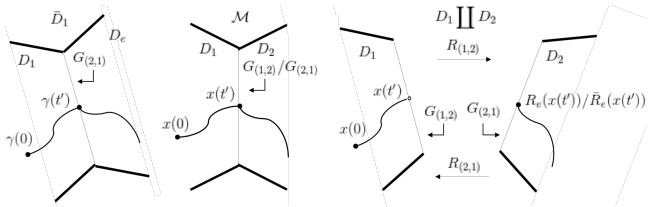

Figure 5: The curve γ is simulated on $\bar{D}_1$, and then interpreted to construct the corresponding transition on $\mathcal{M}$ (center) and $D_1 \coprod D_2$ (right).

Lemma 5.1. *Let $e = (j, j')$. Let $f_e \colon D_e \times U \to \mathbb{R}^n$ be defined by $f_e(x, u) = \bar{A}_e^{-1} f_{j'}(\bar{R}_e(x), u)$. Then $\forall x \in D_e$ if we take $\frac{\mathrm{d}}{\mathrm{d}t} x = f_e(x, u)$ we have that $\frac{\mathrm{d}}{\mathrm{d}t} \bar{R}_e(x) = f_{j'}(\bar{R}_e(x), u)$.*

To prove the claim we compute $\frac{\mathrm{d}}{\mathrm{d}t} \bar{R}_e(x) = \bar{A}_e f_e(x, u) = \bar{A}_e \bar{A}_e^{-1} f_{j'}(\bar{R}_e(x), u) = f_{j'}(\bar{R}_e(x), u)$. Intuitively, f_e evaluates the vector field $f_{j'}$ at the point $\bar{R}_e$, then reverses the change of coordinates that occurs when traversing edge e (by passing the vector field through $\bar{A}_e^{-1}$), effectively transplanting the vector field $f_{j'}$ onto the domain D_e.

More generally, for a domain j with possibly more than one guard set we define $\bar{D}_j = D_j \cup_{e \in \mathcal{N}_j} D_e$ [5], then define the switched system $\bar{f}_j \colon \bar{D}_j \times U \to \mathbb{R}^n$ by

$$\bar{f}_j(x, u) = \begin{cases} f_j(x, u) & \text{if } x \in D_j \\ f_e(x, u) & \text{if } x \in D_e, \forall e \in \mathcal{N}_j, \end{cases} \tag{7}$$

where f_e is defined as in Lemma 5.1, for each $e \in \mathcal{N}_j$. Using this switched system, we can accurately describe transitions out of mode j. For example, suppose the hybrid system is instantiated with initial condition $x(0) \in D_j$, and evolves under the vector field f_j until time t' where $x(t') \in G_e$, for some $e = (j, j') \in \mathcal{N}_j$. The system is then reset to the point $R_e(x(t')) = \bar{R}_e(x(t'))$, and x then evolves under the influence of $f_{j'}$. Alternatively, we can simulate the auxiliary curve defined by $\frac{\mathrm{d}}{\mathrm{d}t} \gamma = \bar{f}_j(\gamma, u)$ with initial condition $\gamma(0) = x(0)$, allowing γ to flow into the interior of D_e. Note that since they share the same differential equation and initial condition, we will have $x(t) = \gamma(t), \forall t \in [0, t')$. At t', we have $\bar{R}_e(\gamma(t')) = x(t') \in D_{j'}$, and $\forall t \geq t'$ by Lemma 5.1 we have $\frac{\mathrm{d}}{\mathrm{d}t} \bar{R}_e(\gamma) = f_{j'}(\bar{R}_e(\gamma), u)$. Thus, $\forall t \geq t'$, we have $\bar{R}_e(\gamma(t)) = x(t)$, since the two curves share the same initial condition and differential equation. Thus, we can construct the trajectory x by simulating γ, and interpreting $x(t) = \gamma(t) \in D_j$ for $t \in [0, t')$, and interpreting $x(t) = \bar{R}_e(\gamma(t)) \in D_{j'}$ for $t \geq 0$. These curves are depicted in Figure 5.

Ultimately, this process empowers us to describe discrete transitions of hybrid systems using the solution concept of Filippov. For edge $e = (j, j')$, carefully inspecting f_e, one can see that $\hat{g}_e^T f_e(x, u) = -\hat{g}_{e'}^T f_{j'}(\bar{R}_e(x), u)$, so sliding solutions for $\bar{f}_j$ arise along G_e when f_j points into G_e and $f_{j'}$ points into $G_{e'}$, at corresponding points along the hybrid surface $G_e/G_{e'}$. At this point we wish to remark that, while many authors (e.g. [8]) have discussed the possibility of using Filippov solutions to describe sliding Zeno executions for hybrid systems with jumps, to the best of our knowledge, we provide the first explicit means of doing so. Yet, some hybrid transitions which display Zeno phenomena may reduce to a switched system for which the solution concept of Filippov is undefined, since both vector fields are parallel to their respective guard sets. [6] Our relaxations, however, will resolve this issue.

6 RELAXED VECTOR FIELDS

While we have gained the ability to describe hybrid transitions using the solution concept of Filippov, such trajectories are difficult to approximate numerically, as they require accurately detecting when the guard sets are crossed, and when sliding solution arise and terminate. In order to add slack to our numerical calculations, we extend the method of Teixeira (see e.g. [11]) to relax our collection of switched systems.

For each edge $e = (j, j')$ we define analogs to $\bar{R}_e$ and D_e for the relaxation $\mathcal{H}^\varepsilon$. In particular, we define $\bar{R}_e^\varepsilon \colon \mathbb{R}^n \to \mathbb{R}^n$ by $\bar{R}_e^\varepsilon(x) = R_e(P_e(x)) + \hat{g}_{e'} g_e^\varepsilon(x)$ and then we define $D_e^\varepsilon = \left\{x \in \mathbb{R}^n \colon \bar{R}_e^\varepsilon(x) \in D_{j'}\right\}$, which is depicted on the left in Figure 6 for our example bimodal hybrid system. Note that $\bar{R}_e^\varepsilon(S_e^\varepsilon) = S_{e'}^\varepsilon$. We may also refactor $\bar{R}_e^\varepsilon(x) = \bar{R}_e(x) + \hat{g}_{e'}\varepsilon = \bar{A}_e x + \bar{b}_e + \hat{g}_{e'}\varepsilon$.

We will use the following class of functions from [11] to smoothly transition between the dynamics of f_j and and the (projected) dynamics of $f_{j'}$ when crossing S_e^ε. We say that $\varphi \colon \mathbb{R} \to [0, 1]$ is a *transition function* if *i)* $\varphi(a) = 0$ for $a \leq 0$ and $\varphi(a) = 1$ for $a \geq 1$, *ii)* $\varphi'(a) > 0$ for $a \in (0, 1)$, *iii)* φ' is Lipschitz continuous, and *iv)* $\forall a, \varphi(1 - a) = \varphi(a)$ (i.e. φ is symmetric around 0.5). For the rest of the paper we assume a single transition function has been chosen. [7] For the edge e we then define $\varphi_e^\varepsilon(x) = \varphi(\frac{g_e(x)}{\varepsilon})$, and now define our relaxation of f_e.

Lemma 6.1. *Let $e = (j, j')$. Define $f_e \colon D_j^\varepsilon \cup D_e^\varepsilon \times U \to \mathbb{R}^n$ by $f_e^\varepsilon(x, u) = (1 - \varphi_e^\varepsilon(x)) f_j(x, u) + \varphi_e^\varepsilon(x) \bar{A}_e^{-1} f_{j'}(\bar{R}_e^\varepsilon(x), u)$. Then*

[5] Note that even when two guards $G_e, G_{e'} \subset \partial D_j$ do not intersect, it may be the case that $D_e \cap D_{e'}$ has a non-empty interior. We ignore this technicality, since in practice we will sample neither f_e nor $f_{e'}$ from this region.

[6] For example, the two numerical examples we consider in Section 8 fall into this category.

[7] For example, in our code we employ

$$\varphi(a) = \begin{cases} 0 & \text{if } a \leq 0 \\ \frac{1}{2} + \frac{1}{2} sine(\pi a - \frac{\pi}{2}) & \text{if } 0 < a < 1 \\ 1 & \text{if } 1 \leq a \end{cases} \tag{8}$$

$\forall x \in D_e^\varepsilon$ *if we take* $\frac{\mathrm{d}}{\mathrm{d}t}x = f_e^\varepsilon(x,u)$ *we have that* $\frac{\mathrm{d}}{\mathrm{d}t}\bar{R}_e^\varepsilon(x) = f_j(\bar{R}_e^\varepsilon(x), u)$. [8]

It was shown in [12] that for each $\varepsilon > 0$ the vector field f_e^ε is continuously differentiable. Note that when $g_e(x) \leq 0$ (and $x \in D_j$), $\varphi_e^\varepsilon(x) = 0$ and $f_e^\varepsilon(x,u)$ returns $f_j(x,u)$. Similarly, when $g_e(x) \geq \varepsilon$ (and $x \in D_e^\varepsilon$), f_e^ε returns $\bar{A}_e^{-1} f_{j'}^\varepsilon(\bar{R}_e^\varepsilon(x), u)$. When $0 \leq g_e(x) \leq \varepsilon$, f_j^ε produces a convex combination of these vector fields. In the case that f_j points into G_e and $f_{j'}$ points into $G_{e'}$, the trajectories of f_e^ε will remain confined to S_e^ε; thus, Zeno executions are approximated by well defined trajectories on our relaxed strips.

We can use $\bar{R}_e^\varepsilon$ and f_e^ε to keep track of how a relaxed trajectory evolves in $D_{j'}$ after a relaxed transition along edge $e = (j, j')$ occurs, employing the same procedure that was developed using f_e and D_e in the previous section. In particular, we simulate the auxiliary curve $\frac{\mathrm{d}}{\mathrm{d}t}\gamma^\varepsilon = f_e^\varepsilon(\gamma^\varepsilon, u)$, allowing this curve to flow through S_e^ε and into D_e^ε. We can then use the map $\bar{R}_e^\varepsilon$ to keep track of how such a trajectory would have propagated into $D_{j'}$ after crossing G_e^ε and being reset to this domain. This process is depicted in Figure 6. Finally, we define $\bar{D}_j^\varepsilon = D_j^\varepsilon \cup_{e \in \mathcal{N}_j} D_e^\varepsilon$ and then define $f_j^\varepsilon : \bar{D}_j^\varepsilon \times U \to \mathbb{R}^n$ by

$$f_j^\varepsilon = \begin{cases} f_j(x,u) & \text{if } x \in D_j \\ f_e^\varepsilon(x,u) & \text{if } x \in D_e^\varepsilon \bigcup S_e^\varepsilon,\ \forall e \in \mathcal{N}_j, \end{cases} \tag{9}$$

which can also be shown to be continuously differentiable, and thus has a Lipschitz continuous gradients, since continuous functions are Lipschitz on compact domains – a property that will be useful later. For a given relaxed hybrid system $\mathcal{H}^\varepsilon$, we endow the relaxed domain D_j^ε with the vector field $f_j^\varepsilon|_{D_j^\varepsilon \times U}$, for the purposes of Definition 4; however, our theoretical analysis and discrete approximations will rely on simulating f_j^ε past the relaxed strips $\{S_e^\varepsilon\}_{e \in \mathcal{N}_j}$ and into the projected domains $\{D_e^\varepsilon\}_{e \in \Gamma}$.

Lemma 6.2. *Let e and e' be partner edges. Then $\forall x \in S_e^\varepsilon$, if we take $\frac{\mathrm{d}}{\mathrm{d}t}x = f_e^\varepsilon(x,u)$, then $\frac{\mathrm{d}}{\mathrm{d}t}\bar{R}_e^\varepsilon(x) = f_{e'}^\varepsilon(\bar{R}_e^\varepsilon(x), u)$.*

In other words, the vector fields f_e^ε and $f_{e'}^\varepsilon$ produce *equivalent flows* over $\mathcal{M}^\varepsilon$, and thus we can represent a relaxed transition along $e = (j, j')$ on either S_e^ε or $S_{e'}^\varepsilon$. Moreover, this implies that, if a relaxed trajectory repeatedly flows back and forth across $S_e^\varepsilon / S_{e'}^\varepsilon$, we can simulate this behavior on either $\bar{D}_j^\varepsilon$ or $\bar{D}_{j'}^\varepsilon$, and don't need to switch between the two vector fields each time a transition occurs – this fact will greatly simplify out analysis later. The proof of Lemma 6.2 is largely algebraic, and uses the fact that φ is symmetric about 0.5.

We conclude this section by studying how the trajectories of f_j^ε converge to those of $\bar{f}_j$ as we take $\varepsilon \to 0$. For the following two theorems, assume we have fixed an input

[8] Again, we compute $\frac{\mathrm{d}}{\mathrm{d}t}\bar{R}_e^\varepsilon(x) = \bar{A}_e f_e^\varepsilon(x,u) = \bar{A}_e(\bar{A}_e)^{-1} f_{j'}(\bar{R}_e^\varepsilon(x), u) = f_{j'}(\bar{R}_e^\varepsilon(x), u)$.

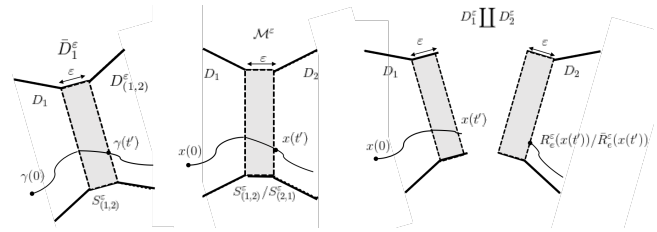

Figure 6: The curve γ^ε is simulated on $\bar{D}_1^\varepsilon$, and then interpreted to construct the corresponding transition on $\mathcal{M}^\varepsilon$ (center) and $D_1^\varepsilon \coprod D_2^\varepsilon$ (right).

$u \in PC([0,T], U)$, and then let $x^\varepsilon : [0,T] \to \bar{D}_j^\varepsilon$ be the corresponding solution generated by f_j^ε with initial condition $x_0^\varepsilon \in D_j$, and let $\bar{x} : [0,T] \to \bar{D}_j$ be the trajectory generated by $\bar{f}_j$ with initial condition $x_0 \in D_j$ and the same input. We leave the proofs to the appendix. A version of the following result for autonomous vector fields may be found in [7].

Theorem 6.3. *Assume that for each $e = (j, j') \in \mathcal{N}_j$ and each $(x,u) \in G_e \times U$ either $\hat{g}_e^T f_j(x,u) > 0$ or $\hat{g}_e^T f_e(x,u) < 0$. Finally assume $\left\|x_0^\varepsilon - x_0\right\| \leq k\varepsilon$, for some $k > 0$.* [9] *Then $\exists \varepsilon_0 > 0$ and $C > 0$ such that for each $\varepsilon_0 > \varepsilon > 0$ we may bound $\|\bar{x} - x^\varepsilon\|_\infty \leq C\varepsilon$.*

The hypothesis of Theorem 6.3 guarantee that Filippov solutions are unique and well defined for $\bar{f}_j$ along the guard sets of D_j, since the escaping region is empty. Thus, our relaxed vector fields converge to Filippov solutions, when applicable. We next examine how our relaxations behave when Filippov solutions are ill-defined.

Theorem 6.4. *Suppose there exists a Lipschitz continuous function $l : \mathbb{R} \to \mathbb{R}^n$ such that $l(\varepsilon) = x_0^\varepsilon$ and $l(0) = x_0$.*[10] *Then there exists a uniformly continuous $x^0 : [0,T] \to \bar{D}_j$ such that $x^0(0) = x_0$ and $\left\|x^0 - x^\varepsilon\right\|_\infty \to 0$ as as $\varepsilon \to 0$.*

Theorem 6.4 implies that our relaxations converge uniformly to a unique, well defined limit, even when the solution concept of Filippov breaks down.

7 EXECUTIONS

Having demonstrated our relaxation approach to describe single discrete transitions of a hybrid system, we modify the algorithmic construction presented in [3] to define the trajectories of our relaxations through multiple transitions.

Definition 7.1. An execution for a relaxed hybrid dynamical system $\mathcal{H}^\varepsilon$, given data $x_0 \in D_j$ and $u \in PC([0,T], U)$,

[9] During each discrete transition our relaxations will incur an error of order ε. By adding error to our initial conditions here, we will be able to call this result inductively to prove convergence when trajectories undergo multiple transitions.

[10] Again we add slack to our initial condition so this result may be called inductively.

denoted $x^\varepsilon\colon [0,T] \to \mathcal{M}^\varepsilon$ is constructed via the following algorithm.

(1) Set $x^\varepsilon(0) = x_0$ and $t = 0$, and let $j \in \mathcal{J}$.
(2) Simulate the differential equation $\dot{\gamma^\varepsilon}(s) = f_j^\varepsilon(\gamma^\varepsilon(s), u(s))$ forward in time with initial condition $\gamma^\varepsilon(t) = x^\varepsilon(t)$ until time $t' = \min\left\{T, \inf\left\{s\colon \gamma^\varepsilon(s^+) \notin D_j^\varepsilon\right\}\right\}$.
(3) If $t' = T$ or $\nexists e \in \mathcal{N}_j$ such that $\gamma^\varepsilon(t') \in G_e^\varepsilon$, let $x^\varepsilon(s) = \gamma^\varepsilon(s)$, $\forall s \in [t, t']$. Then terminate the execution.
(4) Else let $e = (j, j')$ be such that $\gamma^\varepsilon(t') \in G_e^\varepsilon$. For each $s \in [t, t')$ set $x^\varepsilon(s) = \gamma^\varepsilon(s)$. Set $x^\varepsilon(t') = R_e^\varepsilon(\gamma^\varepsilon(t'))$, set $t = t'$ and set $j = j'$. Go to step 2.

First, we note that the only time the execution terminates is in line 3, when either the simulation horizon T has been reached or when the trajectory leaves a relaxed domain at a point that does not belong to a relaxed guard set. Second, we note that the trajectories generated in Definition 7.1 agree with typical definitions for the execution of a hybrid system; that is, a differential equation is simulated until a guard is reached, then the state is reset and resumes simulation. However, these trajectories are continuous over $\mathcal{M}^\varepsilon$ [3], and are even Lipschitz continuous with respect to their arguments.

Proposition 7.2. *Construct $x_1^\varepsilon, x_2^\varepsilon\colon [0,T] \to \mathcal{M}^\varepsilon$ as in Definition 7.1 using the arguments $(x_0^1, u^1), (x_0^2, u^2) \in \mathcal{M}^\varepsilon \times PC([0,T], U)$, respectively. Then there exists $L > 0$ such that $\rho^\varepsilon(x_1, x_2) \le L\left(\widetilde{d}_{\mathcal{M}^\varepsilon}(x_0^1, x_0^2) + \left\|u^1 - u^2\right\|_2\right)$.*

Note that this fundamental systems theoretic property is missing from previous relaxation approaches as [3] and [8]. Due to space constraints, we do not formally compute variations over our relaxations. However, the result follows from two observations. Firstly, as demonstrated by Lemmas 6.1 and 6.2, each portion of a relaxed execution constructed via 7.1 has a one-to-one, affine (and therefore Lipschitz) correspondence to the trajectories generated by a vector field that has Lipschitz continuous gradients. Secondly, by Theorem 5.6.7 of [15], the flows generated by each of these vector fields are Lipschitz continuous with respect to their arguments. In Section 8 we demonstrate how to compute variations through a relaxed transition in the numerical setting. Although we must construct their trajectories in an algorithmic manner, our class of relaxed hybrid systems may largely be viewed simply as classical dynamical systems – that is, systems whose trajectories are continuous and have variations which are Lipschitz continuous. Furthermore, the convergence results of Theorems 6.3 and 6.4 hold when multiple transitions occur. Note that the following construction is similar to the definition of an execution of a hybrid system from [3], but unlike this work we are able to describe sliding solutions along our guard sets.

Theorem 7.3. *Assume that for each $e = (j, j') \in \Gamma$ and each $(x, u) \in G_e \times U$ either $\hat{g}_e^T f_j(x, u) > 0$ or $\hat{g}_e^T f_e(x, u) < 0$. For each $x_0 \in D_j$ and $u \in PC([0,T], U)$ let x^ε be constructed via Definition 7.1. Then $\exists \varepsilon_0 > 0$ and $C > 0$ such that $\forall \varepsilon_0 > \varepsilon > 0$, $\rho^\varepsilon(x, x^\varepsilon) \le C\varepsilon$, where $x\colon [0,T] \to \mathcal{M}$ is generated by the following algorithm.*

(1) Set $x(0) = x_0$ and $t = 0$, and let $j \in \mathcal{J}$.
(2) Simulate the differential equation $\dot{\gamma}(s) = \bar{f}_j(\gamma(s), u(s))$ forward in time (using the solution concept of Filippov) with initial condition $\gamma(t) = x(t)$ until time $t' = \min\left\{T, \inf\left\{s\colon \gamma(s^+) \notin D_j\right\}\right\}$.
(3) If $t' = T$ or $\nexists e \in \mathcal{N}_j$ such that $\gamma(t') \in G_e$, let $x(s) = \gamma(s)$, $\forall s \in [t, t']$. Then terminate the execution.
(4) Else let $e = (j, j')$ be such that $\gamma(t') \in G_e$. For each $s \in [t, t')$ set $x(s) = \gamma(s)$. Set $x(t') = R_e^\varepsilon(\gamma(t'))$, set $t = t'$ and set $j = j'$. Go to step 2.

We again leave the proof to Appendix A.

Theorem 7.4. *Fix $x \in D_j$, $u \in PC([0,T], U)$ and let x^ε be constructed by the algorithm in Definition 7.1. Then there exists a uniformly continuous $x\colon [0,T] \to \mathcal{M}$ such that $\rho^\varepsilon(x, x^\varepsilon) \to 0$ as $\varepsilon \to 0$, where $x(0) = x_0 \in D_j$.*

We omit the proof in the interest of brevity since the proof is analogous to that of Theorem 7.3, except Theorem 6.4 is called inductively in place of 6.3. We employ this limit to define the execution of our relaxed hybrid systems.

Definition 7.5. Let $\mathcal{H}$ be a hybrid system. Then given data $x_0 \in \mathcal{M}$ and $u \in PC([0,T], U)$ we define the corresponding trajectory of $\mathcal{H}$ to be $x\colon [0,T] \to \mathcal{M}$, where $x = \lim_{\varepsilon \to 0} x^\varepsilon$, and for each $\varepsilon > 0$ we construct x^ε using the algorithm in definition 7.1.

Taken together, Theorems 7.3 and 7.4 imply that executions of our hybrid systems, as in Definition 7.5, are unique and we defined, even when traditional solution concepts for hybrid systems would have produced Zeno executions. Note that his property is fundamental, yet missing from current methods such as [8], [3], and [1]. While further work is needed to more carefully characterize this limit in cases where Filippov solutions are ill-defined, in Section 8 we provide numerical evidence that in such cases our relaxations converge to solutions which make physical sense.

We now introduce the provably convergent numerical integration scheme that we use to approximate the trajectories of our relaxed hybrid systems. Again, our discretization scheme is largely similar to the one proposed in [3]. We begin with the following definition of a numerical integrator.

Definition 7.6. [3] Given a relaxed hybrid system $\mathcal{H}^\varepsilon$, we say $\mathcal{A}\colon \mathbb{R}^n \times U \times \mathcal{J} \times \mathbb{R} \to \mathbb{R}^n$ is a numerical integrator of order ω, if for each $j \in \mathcal{J}$ and $h = T/N$ (where $N \in \mathbb{N}$), and each $x_0 \in D_j$ and $u \in PC([0,T], U)$ we have

$$\sup\left(\left\|x(kh) - z^{\varepsilon,h}(kh)\right\|\colon k \in \{0, 1, \ldots, N\}\right) = O(h^\omega),$$

where $x(0) = x_0$ and $\frac{\mathrm{d}}{\mathrm{d}t}x = f_j^\varepsilon(x,u)$, and $z^{\varepsilon,h}(0)$ and $z^{\varepsilon,h}((k+1)h) = \mathcal{A}(z(kh), u(kh), j, h)$.

As was noted in [3], this definition of a numerical integrator is compatible with a large class of discretization schemes, including Euler and the Runge-Kutta family.

Definition 7.7. Given a relaxed hybrid system $\mathcal{H}^\varepsilon$, initial condition $x_0 \in D_j$, input $u \in PC([0,T],U)$, step size $h = \frac{T}{N}$ (where $N \in \mathbb{N}$), we construct the discrete approximation $z^{\varepsilon,h} : [0,t] \to \mathcal{M}^\varepsilon$ according to the following algorithm.

(1) Let $z^{\varepsilon,h}(0) = x_0$, $t = 0$, $k = 0$ and $j \in \mathcal{J}$.
(2) If $k = N$, terminate the execution. Otherwise, let $\gamma^{k+1} = \mathcal{A}\left(z^{\varepsilon,k}(kh), u(kh), j, h\right)$.
(3) For each $t \in [kh, (k+1)h)$ set $z^{\varepsilon,h}(t) = \frac{(k+1)h-t}{h}\gamma^{k+1} + \frac{t-kh}{h}z^{\varepsilon,h}(kh)$.
(4) If $\gamma^{k+1} \notin \bar{D}_j^\varepsilon$, then let $\bar{t} = \inf\left\{t : z^{\varepsilon,h}(t) \in \bar{D}_j^\varepsilon\right\}$ and return $z^{\varepsilon,h}|_{[0,\bar{t}]}$. Terminate the execution.
(5) If $\exists e = (j,j') \in \mathcal{N}_j$ such that $\gamma^{k+1} \in D_e^\varepsilon$, set $z^{\varepsilon,h}((k+1)h) = \bar{R}_e(\gamma^{k+1})$, set k = k+1, and set $j = j'$. Go to step 2.
(6) Otherwise, set $z^{\varepsilon,h}((k+1)h)$ and k = k+1. Go to step 2.

Our definition of a numerical approximation for relaxed hybrid systems differs from [3] in one crucial way. The discretization scheme proposed in [3] requires that a time step be placed in a relaxed strip when simulating a discrete transition. This requires many sample steps be taken until one is placed correctly. On the other hand, our numerical approximation can step *over* the relaxed strip S_e^ε when approximating a discrete transition along edge e, since we can use the Lipschitz vector field f_e^ε and map $\bar{R}_e^\varepsilon$ to simultaneously model how the trajectory evolves on either side of the transition, and thus there is no need to modulate the step size during numerical approximation of a discrete transition, as is depicted in Figure 7.

However, the proof of convergence for our algorithm follows directly from an argument similar to that of Theorem 27 of [3]. In particular, since, for each $\varepsilon > 0$ the vector fields we integrate over are Lipschitz continuous, we incur a numerical error of order $O(h^\omega)$ on each mode, and using an argument similar to that of Theorem 27 of [3] [11], one can show for each $x_0 \in D_j$, $u \in PC([0,T],U)$, and step size h small enough that $\rho^\varepsilon\left(x^\varepsilon, z^{\varepsilon,h}\right) \leq Ch^\omega$ for some $C > 0$, were x^ε and $z^{\varepsilon,h}$ are constructed via Definitions 7.1 and 7.7, respectively. Using these conditions, if we construct the hybrid executions x using Definition 7.5, applying the triangle inequality on ρ^ε, one can then show that $\lim_{\varepsilon\to 0}\lim_{h\to 0}\rho^\varepsilon\left(x, z^{\varepsilon,h}\right) = 0$, as was demonstrated in Corollary 28 of [3]. Moreover, the rate of convergence in h is of order ω. When the hypothesis of Theorem 7.3 are satisfied, the rate of convergence is linear in ε, but unknown when Filippov solutions are ill-defined on the guard sets. Before proceeding to our examples, we note that the relaxation scheme we developed in this paper allowed us to construct a provably convergent numerical algorithm capable of simulating all of the trajectories of our hybrid systems, even those that continue past Zeno, an improvement over existing methods such as [3] and [5].

[11] Alternatively, one can make an argument similar to the proof of Theorem 7.3.

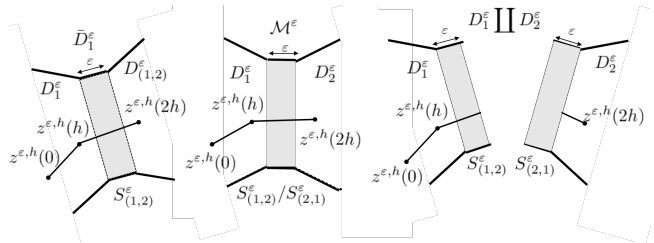

Figure 7: A numerical approximation of a discrete transition is constructed on $\bar{D}_1^\varepsilon$ (left) and the interpreted on $\mathcal{M}^\varepsilon$ (center) and $D_1^\varepsilon \coprod D_2^\varepsilon$ (right).

8 NUMERICAL EXAMPLES

We present several numerical examples, demonstrating the utility of the techniques developed in this paper. We first use an example which is often used to model limbs in the dynamic walking literature. This example is inspired by [2] and [14].

Example 1: (Double Pendulum) Consider the double pendulum with a mechanical stop which is depicted in Figure 8. The system has two angular degrees of freedom $q = (\theta_1, \theta_2)$ whose dynamics are Lagrangian. When the second link impacts the mechanical stop (i.e when $\theta_2 = 0$ and $\dot{\theta}_2 \leq 0$), the angular velocities of are reset according to $(\dot{\theta}_1, \dot{\theta}_2) \to (\dot{\theta}_1 + k(1+c)\dot{\theta}_2, -c\dot{\theta}_2)$, where $c \in [0,1]$ is the *coefficient of restitution*, and $k > 0$. The interested reader may find the explicit representations of these dynamics in [14], where it was demonstrated this system may be faithfully modeled by a unimodal hybrid system with a single edge. When $\theta_2 = \dot{\theta}_2 = 0$, the arm may be locked in place until the imaginary force $\lambda(q,\dot{q})$ becomes nonpositive, at which point the second arm begins to swing freely again. It was shown in [14] that this hidden locked mode corresponds to a Zeno execution. However, using our relaxation procedure, we can model the dynamics of this hidden mode using well defined solutions on the relaxed strip for this hybrid system. In Figure 8, we simulate trajectories for this system for both $c = 0.5$ and $c = 0$, with physical parameters $m_1 = m_2 = L_1 = L_2 = g = 1$, Euler step size $h = 10^{-6}$, $\varepsilon = 10^{-6}$ and initial condition $\left(\theta_1, \dot{\theta}_1, \theta_2, \dot{\theta}_2\right) = (25°, 0, 35°, 0)$ (using the extensions to our framework outlined in the optional appendix). In both simulations, time steps that lie in

the relaxed strip are bold and colored black. Note, in both cases the double pendulum settles into a (decaying) periodic orbit, wherein the second arm is periodically locked into place (and the simulation remains confined to the strip) until the imaginary force dissipates and the second arm swings freely.

For our second experiment involving the double pendulum, we demonstrate how our relaxation framework may be used to conduct sensitivity analysis around a nominal trajectory, as the trajectory progresses through a hybrid transition. In particular, we fix $c = 0$ and $\varepsilon = 10^{-3}$, and once again fix $m_1 = m_2 = L_1 = L_2 = g = 1$, and choose the initial condition of $\left(\theta_1, \dot{\theta}_1, \theta_2, \dot{\theta}_2\right) = (20°, 0, 2°, 0) = x_0$ for our nominal trajectory, which is depicted in Figure 8 d). We choose again choose an Euler step-size of $h = 10^{-6}$ for each of the simulations of this experiment. Time instances that lie in the relaxed strip are again blackened. Note, this nominal trajectory only undergoes one transition, thus we can simulate the entirety of this trajectory on a single extended domain, without needing to ever reset the trajectory. We denote this domain D and its vector field by f. Moreover, since this vector field has gradients that are Lipschitz continuous we can numerically approximate variations over this vector field, using the techniques of, e.g., Chapter 5.6.3 of [15]. For a given $\delta > 0$ we let x^δ be the trajectory corresponding to the perturbed initial condition $x_0^\delta = x_0 + \delta x_0$, where $\delta x_0 = \delta(0, 0, 1°, 0)$. Next, applying Theorem 5.6.13 of [15], and linearizing about the nominal trajectory x^0, for each δ we approximate x^δ with $\hat{x}^\delta = x^0 + Dx^\delta$ where Dx^δ is the solution to the linearized difference equation

$$Dx^\delta((k+1)h) = \frac{\mathrm{d}}{\mathrm{d}x} f(x^0(kh)) Dx^\delta(kh) \tag{10}$$

with initial condition $Dx^\delta(0) = \delta x_0$. For various values of δ we simulate both x^δ and $\hat{x}^\delta$, and in Figure 8 e) we appropriately interpret x^δ and $\hat{x}^\delta$ on $\mathcal{M}^\varepsilon$, and plot the difference $\rho^\varepsilon\left(x^\delta, \hat{x}^\delta\right)$. As this figure clearly demonstrates, using this technique we are able to accurately compute variations through a relaxed transition, even for a trajectory that is simulating past Zeno, as we take δ to be sufficiently small.

Example 2: (Bouncing Ball) For our second example, we simulate the famous bouncing ball. This system consists of a ball repeatedly bouncing on the ground, losing a fraction of its energy during each impact. The ball bounces vertically and has two continuous states, its height x_1 and its vertical velocity x_2. These two states evolve according to $\frac{\mathrm{d}}{\mathrm{d}t} x_1 = x_2$ and $\frac{\mathrm{d}}{\mathrm{d}t} x_2 = -g$, where g is the gravitational constant. When an impact occurs, the velocity is reset according to $x_2 \to -cx_2$, where $c \in [0, 1]$ is again the coefficient of restitution. It can be shown [8] that the ball undergoes an infinite number of bounces by the finite time $t_\infty = \frac{x_2(0)}{g} + \frac{(1/c+1)\sqrt{x_2(0)^2 + 2g x_1(0)}}{g(1/c-1)}$, at which time it comes to rest (i.e. $x_1(t) = x_2(t) = 0, \forall t \geq t_\infty$). Thus, a faithful hybrid representation of the system is necessarily Zeno. We simulate this example to benchmark the performance of our relaxations, since we know analytically when and where Zeno occurs. We simulate the bouncing ball with initial condition $(x_1(0), x_2(0)) = (1, 0)$, $g = 1$ and $c = 0.5$ for various Euler steps sizes h, and for each simulation fix $\varepsilon = 0.01 * h$. For each simulation we let $\rho = \sup\{\|(x_1(t), x_2(t))\|_\infty : t \in [t_\infty, T]\}$, and use this metric to measure the convergence of our relaxed trajectories to the Zeno point. We plot the results in Figure 8 f). Note that we do not provide theoretical guarantees of the rate of convergence for this example, since the vector field is parallel to the transition surface at the origin (the Zeno accumulation point). However, the plot in Figure 8 f) nevertheless demonstrates that we have (near) linear convergence as we take h and ε to zero, under the ρ metric. We are currently working to provide formal guarantees for the rate of convergence in such cases.

9 CONCLUSION

In this paper we developed a novel class of relaxations, which we used to construct unique, well defined solutions for hybrid systems, even past the point of Zeno. The trajectories of our hybrid systems were shown to be Lipschitz continuous with respect to initial conditions and inputs, and naturally gave rise to a broad class of provably convergent discretization schemes. We provided several numerical examples, wherein we were able to accurately simulate and performed sensitivity analysis on Zeno executions. While further work is needed to extend our current framework, including the addition of non-linear guards and reset maps as well as overlapping guards, it is our conviction that the techniques developed here will provide an avenue to extract further important systems theoretic properties from hybrid systems. Moreover, we are currently working to extend our sensitivity analysis techniques to trajectories undergoing multiple transitions, with the intention of using these techniques to assess the stability of periodic orbits in hybrid systems. Such an approach has many practical applications, including finding stable periodic gates for dynamic walking robots [17].

REFERENCES

[1] Ames, A. D., Zheng, H., Gregg, R. D., and Sastry, S. Is there life after zeno? taking executions past the breaking (zeno) point. In *2006 American Control Conference* (June 2006), pp. 6 pp.–.

[2] Burden, S., Gonzalez, H., Vasudevan, R., Bajcsy, R., and Sastry, S. S. Numerical integration of hybrid dynamical systems via domain relaxation. In *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on* (2011), IEEE, pp. 3958–3965.

[3] Burden, S. A., Gonzalez, H., Vasudevan, R., Bajcsy, R., and Sastry, S. S. Metrization and simulation of controlled hybrid systems. *IEEE*

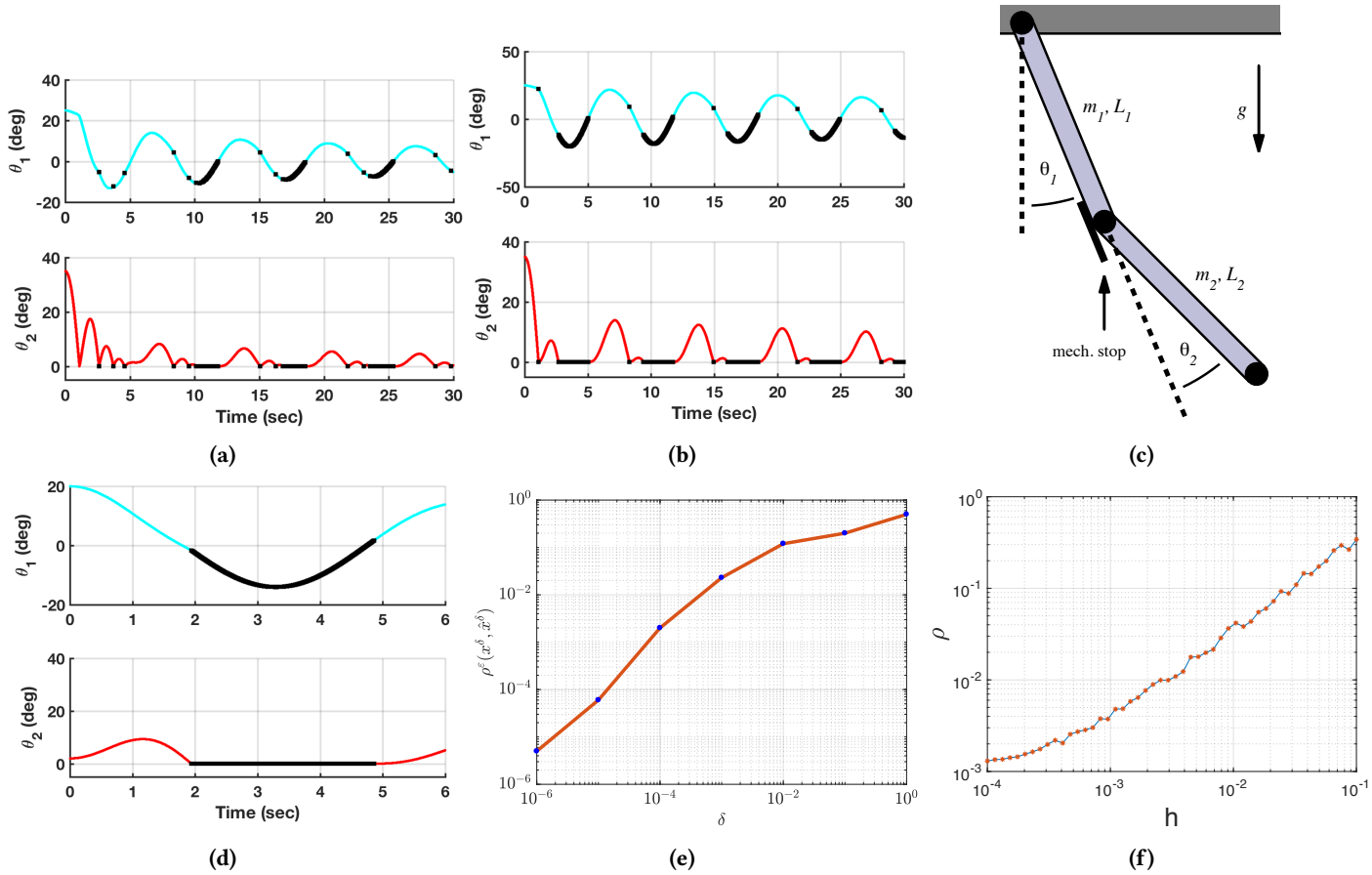

Figure 8: Double Pendulum with a mechanical stop: a) simulation with $c = 0.5$ b) simulation with $c = 0$ c) schematic of the system d) reference trajectory for sensitivity analysis e) numerical error of sensitivity analysis. Bouncing Ball: f) simulation error under ρ metric.

Transactions on Automatic Control 60, 9 (2015), 2307–2320.

[4] Dieci, L., and Lopez, L. Sliding motion on discontinuity surfaces of high co-dimension. a construction for selecting a filippov vector field. *Numerische Mathematik 117*, 4 (2011), 779–811.

[5] Esposito, J. M., Kumar, V., and Pappas, G. J. *Accurate Event Detection for Simulating Hybrid Systems*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001, pp. 204–217.

[6] Filippov, A. F. *Differential equations with discontinuous righthand sides: control systems*, vol. 18. Springer Science & Business Media, 2013.

[7] Fiore, D., Hogan, S. J., and di Bernardo, M. Contraction analysis of switched systems via regularization. *Automatica 73* (2016), 279–288.

[8] Johansson, K. H., Egerstedt, M., Lygeros, J., and Sastry, S. On the regularization of zeno hybrid automata. *Systems & control letters 38*, 3 (1999), 141–150.

[9] Kelley, J. L. *General topology*. Courier Dover Publications, 2017.

[10] Lee, J. M. Smooth manifolds. In *Introduction to Smooth Manifolds*. Springer, 2003, pp. 1–29.

[11] Llibre, J., da Silva, P. R., Teixeira, M. A., et al. Regularization of discontinuous vector fields on r3 via singular perturbation. *Journal of Dynamics and Differential Equations 19*, 2 (2007), 309–331.

[12] Llibre, J., da Silva, P. R., Teixeira, M. A., et al. Sliding vector fields via slow–fast systems. *Bulletin of the Belgian Mathematical Society-Simon Stevin 15*, 5 (2008), 851–869.

[13] Lygeros, J., Johansson, K. H., Simic, S. N., Zhang, J., and Sastry, S. S. Dynamical properties of hybrid automata. *IEEE Transactions on automatic control 48*, 1 (2003), 2–17.

[14] Or, Y., and Ames, A. D. Formal and practical completion of lagrangian hybrid systems. In *American Control Conference, 2009. ACC'09.* (2009), IEEE, pp. 3624–3631.

[15] Polak, E. *Optimization: algorithms and consistent approximations*, vol. 124. Springer Science & Business Media, 2012.

[16] Simić, S. N., Johansson, K. H., Sastry, S., and Lygeros, J. Towards a geometric theory of hybrid systems. In *International Workshop on Hybrid Systems: Computation and Control* (2000), Springer, pp. 421–436.

[17] Westervelt, E. R., Grizzle, J. W., and Koditschek, D. E. Hybrid zero dynamics of planar biped walkers. *IEEE transactions on automatic control 48*, 1 (2003), 42–56.

[18] Zhang, J., Johansson, K. H., Lygeros, J., and Sastry, S. Zeno hybrid systems. *International Journal of Robust and Nonlinear Control 11*, 5 (2001), 435–451.

A PROOFS

For each of the following proofs, we provide the main arguments, omitting some details in the interest of brevity.

A.1 Proof of Theorem 6.3

We supply the proof for the case where $|\mathcal{N}_j| = 1$; the generalization to the case where D_j has multiple, disjoint guard sets is straightforward. We first demonstrate that the claim holds for all input functions $\hat{u} \in PCD([0,T],U)$, where PCD denotes the class of piecewise continuously differentiable functions. We transform the system $\frac{\mathrm{d}}{\mathrm{d}t}\bar{x} = \bar{f}_j(\bar{x},\hat{u})$ into the autonomous system defined by $(\dot{\tilde{x}},\dot{z}) = \tilde{f}_j(\tilde{x},z) = (\bar{f}_j(\tilde{x},\hat{u}(z)),1)$ which we endow with initial condition $(x_0,0)$. Note that $z(t) = t, \forall t \in [0,T]$, and thus $\tilde{x}(t) = \bar{x}(t)$, $\forall t \in [0,T]$. Let $\tilde{f}_j^\varepsilon$ be the ε-relaxation of $\tilde{f}_j$, and let $(\tilde{x}^\varepsilon, z)$ be the resulting trajectory, with initial data $(x^\varepsilon, 0)$. Next, note that $\hat{u}$ must be non differentiable on a finite number of points $0 = t_1 < t_2 < \ldots < t_p = T$, $p \in \mathbb{N}$. Thus, on each interval $(t_i, t_{i+1}), \forall i = 1,2,\ldots,p-1$, $\tilde{f}_j$ is continuously differentiable in z. Thus, restricting both trajectories to the time interval $[t_1,t_2]$, we have $\left\|(\tilde{x},z)|_{[t_1,t_2]} - (\tilde{x}^\varepsilon,z)|_{[t_1,t_2]}\right\|_\infty = O(\varepsilon)$ for each $\varepsilon < \varepsilon_0$ for some $\varepsilon_0 > 0$, by an argument similar to Lemma 2 of [7]. Thus, by a straight forward inductive argument we obtain $\|(\tilde{x},z) - (\tilde{x}^\varepsilon,z)\|_\infty = O(\varepsilon)$, and thus $\|\bar{x} - x^\varepsilon\|_\infty \le C\varepsilon$, for some $C > 0$. The result for our desired $u \in PC([0,T],U)$ follows from noting that $PCD([0,T],U)$ is dense in $PC([0,T],U)$ under the L^2 norm, and thus we may choose $\hat{u}$ to be arbitrarily close to the desired input u.

A.2 Proof of theorem 6.4

First, note that f_j^ε is continuously differentiable in ε for each $\varepsilon > 0$, since it is constructed using a finite number of compositions and multiplications of functions which are each continuously differentiable in ε. Thus, $\frac{\partial f_j^\varepsilon}{\partial \varepsilon}$ must be Lipschitz continuous for each $\varepsilon \in [\underline{\varepsilon}, \bar{\varepsilon}]$, where $\bar{\varepsilon} > \underline{\varepsilon} > 0$, since continuous functions are Lipschitz on compact domains. Thus, by Lemma 5.6.7 of [15], $x^\varepsilon(t)$ is a Lipschitz continuous function of ε, $\forall t \in [0,T]$ and $\varepsilon \in (\underline{\varepsilon}, \bar{\varepsilon})$. Thus, as $\varepsilon \downarrow \underline{\varepsilon}$, x^ε must converge uniformly to some uniformly continuous function $x^\varepsilon \colon [0,T] \to \bar{D}_j^{\underline{\varepsilon}}$. The desired result follows by noting that $\underline{\varepsilon}$ may be chosen to be arbitrarily small.

A.3 Proof of Theorem 7.3

In this case when no transitions occur, the result follows from the uniqueness of trajectories on our continuous domains. Suppose now that x undergoes one transition along edge $e = (j,j')$. We may represent the trajectories of both x and x^ε through this transition using the domain $\bar{D}_j^\varepsilon$. In particular, let γ be the solution to $\frac{\mathrm{d}}{\mathrm{d}t}\gamma = \bar{f}_j(\gamma,u)$ with initial condition $\gamma(0) = x_0$, and let γ^ε be defined by $\frac{\mathrm{d}}{\mathrm{d}t}\gamma^\varepsilon = f_j^\varepsilon(\gamma^\varepsilon,u)$ with initial condition $\gamma^\varepsilon(0) = x_0$. By Theorem 6.3, $\|\gamma - \gamma^\varepsilon\|_\infty \le C\varepsilon$, where $\varepsilon_0, C > 0$, and $\varepsilon < \varepsilon_0$. For all t such that $\gamma(t), \gamma^\varepsilon(t) \in D_j$, we immediately have that $\widetilde{d}_{\mathcal{M}^\varepsilon}(x(t),x^\varepsilon(t)) \le C\varepsilon$. For all t such that $\gamma(t) \in D_e$ and $\gamma^\varepsilon(t) \in D_e^\varepsilon$ (i.e. when x and x^ε have both transitioned to mode j'), bound $\widetilde{d}_{\mathcal{M}^\varepsilon}(x(t),x^\varepsilon(t)) = \|x(t) - x^\varepsilon(t)\| = \left\|\bar{R}_e(\gamma(t)) - \bar{R}_e^\varepsilon(\gamma^\varepsilon(t))\right\| = \left\|\bar{A}_e(\gamma(t) - \gamma^\varepsilon(t)) - \varepsilon\hat{g}_{e'}\right\| \le \bar{C}\varepsilon$, for some $\bar{C} > 0$. If $\gamma(t) \in D_e$ but $\gamma^\varepsilon(t) \in D_j^\varepsilon$ (so that $x(t) \in D_{j'}$ but $x^\varepsilon(t) \in D_j^\varepsilon$), then by an application of the triangle inequality on $\widetilde{d}_{\mathcal{M}^\varepsilon}$, we may bound $\widetilde{d}_{\mathcal{M}^\varepsilon}(x(t),x^\varepsilon(t)) \le \|\gamma(t) - \gamma^\varepsilon(t)\| + \left\|\bar{R}_e(\gamma(t)) - \bar{R}_e^\varepsilon(\gamma^\varepsilon(t))\right\| \le \tilde{C}\varepsilon$, for some $\tilde{C} > 0$. The case where $\gamma(t) \in D_j$ but $\gamma^\varepsilon(t) \in D_e^\varepsilon$ follows similarly. Finally it is important to note that, while x was transitioning, x^ε may have transitioned back and forth along e and its partner e' multiple times, yet, as a consequence of Lemma 6.2, γ^ε nevertheless fully captures the behavior of x^ε near $S_e^\varepsilon / S_{e'}^\varepsilon$. For trajectories where x undergoes multiple transitions, Theorem 6.3 may be called inductively to complete the proof.

B NON-REVERSIBLE EDGES

In this section we demonstrate how to extend our framework to encompass non-reversible edges. In the interest of brevity, we show how this may be done for a unimodal hybrid systems with one continuous domain D (and vector field f), and a single non-reversible edge e. However, the generalization to more complicated hybrid systems with non-reversible edges is straightforward. In order to avoid needlessly introducing a large amount of slightly modified notation, we only outline these additional techniques. In particular, we demonstrate how to construct relaxed executions for these hybrid systems, and then discuss when and how the convergence theorems from the main document apply here, but do not explicitly construct the corresponding switched systems, as their structure will become apparent from the relaxations.

We proceed by noting that $R_e(G_e) \subset \partial D$ by Definition 3.1. Since D is a convex polytope, there exists a unit vector $\hat{h}_e \in \mathbb{R}^n$ and scalar d_e such that

$$R_e(G_e) \subset \left\{x \in \partial D \colon h_e(x) = \hat{h}_e^T x - d_e = 0\right\}, \tag{11}$$

where by convention we choose $\hat{h}_e$ such that it points out of D along $R_e(G_e)$ – that is, $h(x) \le 0$, $\forall x \in D$. Note, we do not assume that $R_e(G_e) \cap G_e = \emptyset$. We now define the map $\bar{R}_e^\varepsilon \colon \mathbb{R}^n \to \mathbb{R}^n$ by

$$\bar{R}_e^\varepsilon(x) = R_e(P_e(x)) - \hat{h}_e g_e^\varepsilon(x), \tag{12}$$

simply replacing $\hat{g}_{e'}$ with $\hat{h}_e$ when defining $\bar{R}_e$. In this case, we may now simplify

$$\bar{R}_e^\varepsilon(x) = \bar{A}_e x + \bar{b}_e^\varepsilon, \tag{13}$$

where we now have that $\bar{A}_e = A_e(I - \hat{g}_e\hat{g}_e^T) - \hat{h}_e\hat{g}_e^T$ and $\bar{b}_e^\varepsilon = A_e\hat{g}_e c_e + b_e + \hat{h}_e(c_e + \varepsilon)$. If it is the case that $\bar{A}_e$ is

full rank, then we may construct relaxed transitions along e using the same procedure as in the main document. That is, we use the vector fields f_e^ε, and map $\bar{R}_e^\varepsilon$ to construct relaxed transitions.

However, when $\bar{A}_e$ is not full rank these objects are ill-defined. Specifically, when $\bar{A}_e$ is not full rank we cannot use $\bar{A}_e$ to project the component of f lying in the subspace $range(\bar{A}_e)^\perp$ back through e. [12] In order to overcome this deficiency, let $\{v_e^i\}_{i=1}^{p_e}$ be a basis for $range(\bar{A}_e)^\perp$, where $p_e = n - rank(\bar{A}_e)$ [13]. In order to capture the flow of f along $span\left\{\{v_e^i\}_{i=1}^{p_e}\right\}$, we add the auxiliary state $z \in \mathbb{R}^{p_e}$ to our continuous state space when in mode j, and now define $\tilde{R}_e^\varepsilon \colon \mathbb{R}^n \times \mathbb{R}^{p_e} \to \mathbb{R}^n$ by

$$\tilde{R}_e(x,z) = \bar{R}_e^\varepsilon(x) + [\, v_e^1 | \ldots | v_e^{p_e} \,] z, \tag{15}$$

which we may reformulate into

$$\tilde{R}_e(x,z) = \left[\bar{A}_e | v_e^1 | \ldots | v_e^{p_e}\right] \left[\begin{smallmatrix} x \\ z \end{smallmatrix}\right] + \bar{b}_e^\varepsilon, \tag{16}$$

and then define $\tilde{A}_e = \left[\bar{A}_e | v_e^1 | \ldots | v_e^{p_e}\right] \in \mathbb{R}^{n \times (n+p_e)}$, which is surjective by construction, since $range(\tilde{A}_e) = \mathbb{R}^n$. [14] We will now employ the right inverse of, $\tilde{A}_e$, namely $\tilde{A}_e^\dagger \in \mathbb{R}^{(p_e+n)\times n}$, to project the dynamics of f back through edge e, and capture this flow during a relaxed transition using the augmented state $(x^\varepsilon, z) \in \mathbb{R}^{n+p_e}$. For the rest of the section, let $\vec{0}$ denote the p_e-dimensional zero vector. When we begin a relaxed execution, we will instantiate $z = \vec{0}$, and we will reset $z \to \vec{0}$ whenever a relaxed transition occurs, for reasons that will become clear momentarily. We can now define the analogue to D_e^ε,

$$\hat{D}_e^\varepsilon = \left\{(x,z) \in \mathbb{R}^n \times \mathbb{R}^{p_e} \colon \tilde{R}_e^\varepsilon(x,z) \in D\right\}. \tag{18}$$

Next, we define

$$M = \sup\left\{\|z\|_\infty \colon \exists x \in \mathbb{R}^n \text{ such that } \tilde{R}_e^\varepsilon(x,z) \in D\right\}, \tag{19}$$

and then define the analogue to D^ε,

$$\hat{D}^\varepsilon = D^\varepsilon \times [-M, M]^{p_e}, \tag{20}$$

and finally the analogue to S_e^ε

$$\hat{S}_e^\varepsilon = S_e^\varepsilon \times [-M, M]^{p_e}. \tag{21}$$

That is, we confine the auxiliary state to z to $\{-M, M\}^{p_e}$, so that our augmented continuous domain remains compact, but we allow z to be large enough such that we can capture the full scope of $\hat{D}_e^\varepsilon$ using this extra variable. Next, we define the augmented guard set

$$\hat{G}_e^\varepsilon = \left\{(x,z) \in \tilde{S}_e^\varepsilon \colon x \in G_e^\varepsilon\right\}, \tag{22}$$

which will triggers a discrete transition when crossed and the state is reset according to $\hat{R}_e^\varepsilon \colon \mathbb{R}^{n+p_e} \to \mathbb{R}^{n+p_e}$, [15] where

$$\hat{R}_e^\varepsilon(x,z) = \left[\begin{smallmatrix} \tilde{R}_e^\varepsilon(x,z) \\ \vec{0} \end{smallmatrix}\right]. \tag{23}$$

Note, a discrete transition occurs when $x \in G_e^\varepsilon$, and does not depend on the value of z. Moreover, after the transition occurs, z is reset $\vec{0}$ so that it is ready to simulate the next transition along e.

Finally we are ready to define the relaxed vector field $\hat{f}^\varepsilon \colon \hat{D}^\varepsilon \cup \tilde{D}_e^\varepsilon \times U \to \mathbb{R}^{n+p_e}$ by

$$\hat{f}^\varepsilon((x,z),u) = (1 - \varphi_e^\varepsilon(x)) \left[\begin{smallmatrix} f(x,u) \\ \vec{0} \end{smallmatrix}\right] + \varphi_e^\varepsilon(x) \tilde{A}_e^\dagger f(\tilde{R}_e^\varepsilon(x,z), u), \tag{24}$$

which may be shown to be continuously differentiable. Note that, under this vector field, when $x \in D$, $\varphi_e^\varepsilon(x) = 0$ and $\frac{\mathrm{d}}{\mathrm{d}t}(x,z) = (f(x,u), \vec{0})$, thus the auxiliary z state does not affect the evolution of the original state x when $(x,z) \notin \tilde{S}_e^\varepsilon$. That is, the auxiliary state z remains dormant until the real state reaches S_e^ε, and then z begins to flow, capturing the component of f that lies in $range(\bar{A}_e)^\perp$, as x traverses S_e^ε. Finally note that whenever $(x,z) \in \tilde{D}_e^\varepsilon$ (and $g_e(x) \geq \varepsilon$), the vector field $\hat{f}^\varepsilon$ returns $\tilde{A}_e^\dagger f(\tilde{R}_e^\varepsilon(x,z), u)$, which leads to the following result.

Lemma B.1. *Let $\hat{f}^\varepsilon$ be defined as in (24). Then $\forall (x,z) \in \tilde{D}_e^\varepsilon$, if we take $\frac{\mathrm{d}}{\mathrm{d}t}(x,z) = \hat{f}^\varepsilon((x,z),u)$ then we have that $\frac{\mathrm{d}}{\mathrm{d}t}\hat{R}_e^\varepsilon(x,z) = (f(\tilde{R}_e^\varepsilon(x,z),u), \vec{0})$.*

To prove the claim we compute

$$\frac{\mathrm{d}}{\mathrm{d}t}\hat{R}_e^\varepsilon(x,z) = \left[\begin{smallmatrix} \tilde{A}_e \hat{f}^\varepsilon((x,z),u) \\ \vec{0} \end{smallmatrix}\right] \tag{25}$$

$$= \left[\begin{smallmatrix} \tilde{A}_e \tilde{A}_e^\dagger f(\tilde{R}_e^\varepsilon(x,z),u) \\ \vec{0} \end{smallmatrix}\right] \tag{26}$$

$$= \left[\begin{smallmatrix} f(\tilde{R}_e^\varepsilon(x,z),u) \\ \vec{0} \end{smallmatrix}\right]. \tag{27}$$

[12] For example, the double pendulum with a mechanical stop from Section 8 falls into this category when $c = 0$; in particular, for this case we cannot project the differential equation for $\dot{\theta}_2$ back through the edge $\bar{e}$ of this hybrid system as in this case (if we arrange the state $x = (\theta_1, \dot{\theta}_1, \theta_2, \dot{\theta}_2)$) then $\hat{g}_e = \hat{h}_e = (0, 0, -1, 0)^T$ and the matrix $\bar{A}_{\bar{e}}$ works out to be

$$\bar{A}_{\bar{e}} = \left[\begin{smallmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & k \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \end{smallmatrix}\right]. \tag{14}$$

[13] In the case of the double pendulum where $c = 0$, we have that $p_{\bar{e}} = 1$ and we choose $v_{\bar{e}}^1 = (0, 0, 0, 1)^T$.

[14] For the double pendulum when $c = 0$ we arrive at

$$\tilde{A}_{\bar{e}} = \left[\begin{smallmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & k(1+c) & 0 \\ 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{smallmatrix}\right] \tag{17}$$

, which is full rank and can thus be used to project the differential equation for $\dot{\theta}_2$ back through $\bar{e}$

[15] Note we have overridden the original definition of $\hat{R}_e^\varepsilon$ from the main document.

Consequently, we can use the vector field $\hat{f}^\varepsilon$ and the map $\hat{R}_e^\varepsilon$ to keep track of how (x^ε, z) evolves during a relaxed transition. Note, that even though we are not projecting the dynamics for z back through the edge e, since it is always reset to a value of zero and has trivial dynamics when in $\hat{D}^\varepsilon$, this is not needed to keep track of how z will evolve immediately after the transition.

Concretely, if the real state is instantiated at $x_0 \in D$, in order to describe a relaxed transition along e, we simulate the auxiliary curve $\hat{\gamma}^\varepsilon \colon [0,T] \to \hat{D}^\varepsilon \cup \tilde{D}_e^\varepsilon$ defined by $\frac{\mathrm{d}}{\mathrm{d}t}\hat{\gamma}^\varepsilon = \hat{f}^\varepsilon(\hat{\gamma}^\varepsilon, u)$ with initial condition $(x_0, \vec{0})$, allowing the curve to flow into $\tilde{D}_e^\varepsilon$. We then interpret $(x^\varepsilon(t), z(t)) = \hat{\gamma}^\varepsilon(t)$, $\forall t$ such that $\hat{\gamma}^\varepsilon(t) \in \hat{D}^\varepsilon$, and interpret $(x^\varepsilon(t), z(t)) = \hat{R}_e^\varepsilon(\hat{\gamma}^\varepsilon(t)) = (\tilde{R}_e^\varepsilon(x,z), \vec{0})$, $\forall t$ such that $\hat{\gamma}^\varepsilon(t) \in \tilde{D}_e^\varepsilon$.

It is straightforward to show that analogues to Theorems 6.3 and 6.4 hold when studying the convergence of the trajectories of $\hat{f}^\varepsilon$ as we take $\varepsilon \to 0$; that is, sliding solutions arise when applicable and the trajectories of $\hat{f}^\varepsilon$ always converge to a unique well defined limit as we take $\varepsilon \to 0$. In some cases, such as the double pendulum with mechanical stop, it makes physical sense for the trajectory to "get stuck" in the relaxed strip for some time. However, in general we leave it to the practitioner to interpret this behavior.

In order to discuss analogues to Theorems 7.3 and 7.4, we must first settle on the topology for the class of hybrid systems we consider here. In particular, we now define our relaxed hybrid quotient space by

$$\mathcal{M}^\varepsilon = \frac{D}{\Lambda_{\hat{R}_e^\varepsilon}}. \tag{28}$$

In order to construct trajectories over $\mathcal{M}^\varepsilon$ with multiple transitions, we further modify the construction from [3].

Definition B.2. An execution for a relaxed unimodal hybrid dynamical system with a single non-reversible edge, given data $x_0 \in D$ and $u \in PC([0,T], U)$, denoted $(x^\varepsilon, z) \colon [0,T] \to \mathcal{M}^\varepsilon$ is constructed via the following algorithm.

(1) Set $x(0) = x_0$, $z(0) = 0$ and $t = 0$.
(2) Simulate the differential equation $\dot{\hat{\gamma}}^\varepsilon(s) = \hat{f}^\varepsilon(\hat{\gamma}^\varepsilon(s), u(s))$ forward in time with initial condition $\hat{\gamma}^\varepsilon(t) = (x^\varepsilon(t), z(t))$ until time $t' = \min\{T, \inf\{s \colon \hat{\gamma}^\varepsilon(s^+) \notin \hat{D}^\varepsilon\}\}$.
(3) If $t' = T$ or $\hat{\gamma}^\varepsilon(t') \notin \hat{G}_e^\varepsilon$, let $(x^\varepsilon(s), z(s)) = \hat{\gamma}^\varepsilon(s)$, $\forall s \in [t, t']$. Then terminate the execution.
(4) Otherwise we have that $\hat{\gamma}^\varepsilon(s) \in \hat{G}_e^\varepsilon$. For each $s \in [t, t')$ set $(x^\varepsilon(s), z(s)) = \hat{\gamma}^\varepsilon(s)$. Set $(x^\varepsilon(t'), z(t')) = \hat{R}_e^\varepsilon(\hat{\gamma}^\varepsilon(t'))$, set $t = t'$. Go to step 2.

Using this construction, it is straightforward to show that analogues to the results of Proposition 1, and Theorems 7.3 and 7.4 hold for the class of hybrid systems with non-reversible edges we have considered so far in this appendix. We omit the details in the interest of brevity.