A Zero-crossing Detection Algorithm for Robust Simulation of Hybrid Systems Jumping on Surfaces

David A. Copp[a,*], **Ricardo G. Sanfelice**[b,**]

[a]*Department of Mechanical Engineering, University of California, Santa Barbara, CA 93106-9560, USA*
[b]*Department of Computer Engineering, University of California, Santa Cruz, CA 95064, USA*

Abstract

Hybrid systems are inherently fragile with respect to perturbations when their state experiences jumps on surfaces. Zero-crossing detection algorithms are capable of robustly detecting the crossing of such surfaces, but, up to now, the effects of adding such algorithms to the system being simulated are unknown. In this paper, we propose a mathematical model for hybrid systems that incorporates zero-crossing detection as well as a hybrid simulator for it. First, we discuss adverse effects that measurement noise and discretization can have on hybrid systems jumping on surfaces and prove that, under mild regularity conditions, zero-crossing detection algorithms can robustify the original system. Then, we show that integration schemes with zero-crossing detection actually compute a robustified version of the fragile nominal model. In this way, we rigorously characterize their effect on solutions to the simulated system. Finally, we show that both the model and simulator are not only robust, but also that the hybrid simulator preserves asymptotic stability properties, semiglobally and practically (on the step size), of the original system. Several examples throughout the paper illustrate these ideas and results.

Keywords: Hybrid Systems, Zero-crossing Detection, Simulation

1. Introduction

This work considers dynamical systems with a state that experiences instantaneous resets (jumps) when it hits a *switching surface* $\mathcal{S}$. A switching surface is typically defined as the zero-level set of a continuously differentiable function, defining in this way a codimension one submanifold of $\mathbb{R}^n$; see, e.g., [1, 2, 3]. The state is denoted by x and takes values from a region of operation $\mathcal{X} \subset \mathbb{R}^n$. When x is away from the surface $\mathcal{S}$, the continuous dynamics (flows) of the system are given by a differential equation. This can be written more precisely as

$$\dot{x} = f(x) \qquad \text{when } x \in \mathcal{X} \setminus \mathcal{S}. \tag{1}$$

When x hits the surface $\mathcal{S}$ while in the region of operation $\mathcal{X}$, the state is reset via a difference equation which defines the jumps of the system. More precisely, the new value of x after the jump, denoted x^+, is determined by

$$x^+ = g(x) \qquad \text{when } x \in \mathcal{S} \cap \mathcal{X}. \tag{2}$$

In this way, the trajectories are allowed to flow when $x \in \mathcal{X} \setminus \mathcal{S}$ and are allowed to jump when $x \in \mathcal{S} \cap \mathcal{X}$. This model captures the dynamics of control systems in which a controller makes decisions when certain variables hit a surface. For instance, in reset control systems (see, e.g., [4], [5], [6], [7]), the output of the controller is reset to zero whenever its input and output satisfy an algebraic condition. Reset controllers have

[*] Principal corresponding author
[**] Corresponding author
Email addresses: dacopp@engr.ucsb.edu (David A. Copp), ricardo@ucsc.edu (Ricardo G. Sanfelice)
URL: https://hybrid.soe.ucsc.edu (Ricardo G. Sanfelice)

been found useful in applications as they improve the performance of linear systems [4]. In state-dependent impulsive control systems, (see, e.g., [1], [8]), jumps occur when the state of the system belongs to a surface in the state space of the system. Impulsive controllers are widely used in robotics to switch among several feedback laws when the state of the system reaches a surface [9, 10].

Several difficulties arise when dealing with systems jumping on surfaces. One such difficulty is with "grazing" of the flowing solution at the boundary of the switching surface $\mathcal{S}$ without crossing over it. This can lead to non-unique solutions and is known as the grazing phenomenon. For a discussion of methods to handle this phenomenon, see [11]. Further difficulties may be caused by perturbations due to measurement noise or numerical errors due to discretization. For example, suppose that the value of the state x of the system is perturbed when nearby $\mathcal{S}$ (e.g., due to measurement noise). Letting e denote this perturbation on x, the perturbed state can be written as $x + e$. Suppose that, for a given solution $x(t)$ to system (1)-(2) (using an appropriate notion of solution), e is zero when $x(t) \neq \mathcal{S}$ but equal to a nonzero constant ε when $x(t) = \mathcal{S}$. Then, when the perturbation e is added, for any nonzero ε, the same solution $x(t)$ will not satisfy the condition $x(t) + e(t) \in \mathcal{S}$, and therefore, will not jump at the instant that it would without noise. This suggests that arbitrarily small perturbations to (1)-(2) can generate trajectories that are nowhere close to the trajectories of the nominal system; see [12] for related discussions. Similarly, issues can arise from numerical errors introduced by discretization, such as numerical integration errors. In the presence of small numerical integration errors, which can be made arbitrarily small by adjusting the step size used in the simulator, trajectories obtained through simulation may never hit the surface, and therefore, never jump.

One way to resolve the issue of the numerical solution never hitting the switching surface, which is widely used in simulation packages, is to include *zero-crossing detection* (ZCD) algorithms to detect crossing of the surface $\mathcal{S}$. For instance, in MATLAB/Simulink, a block called *Hit Crossing* detects when the input of the block u reaches a specified offset parameter value u^* (other blocks and local ZCD options are also available in MATLAB/Simulink). The output of the block equals 1 when the value of the input has hit or crossed the offset parameter, or 0 otherwise. In fact, MATLAB/Simulink help files note the following [13]:

> "if the input signal is exactly the value of the offset value after the hit crossing is detected, the block continues to output a value of 1. If the input signals at two adjacent points bracket the offset value (but neither value is exactly equal to the offset), the block outputs a value of 1 at the second time step."

More precisely, denoting the step size by s, this block determines if the sign of $u - u^*$ at the integration time t has changed with respect to its value at the previous integration time $t - s$, i.e., it determines whether

$$(u(t) - u^*)(u(t-s) - u^*) \leq 0. \tag{3}$$

In order to determine whether this condition holds, a memory state must be added to keep track of the sign (relative to u^*) of the previous input to the block. While zero-crossing detection might be a remedy to detect the crossings of $\mathcal{S}$, a simulator utilizing ZCD actually modifies the original system by incorporating an extra mechanism for the detection of zero crossings. On the other hand, the relationship between the simulations obtained with ZCD and the true solutions of a system is not well understood.

The purpose of this paper is to introduce a mathematical framework for theoretical study of such algorithms and their effect in simulation of hybrid systems. To that end, the effect of perturbations to hybrid systems jumping on surfaces is highlighted. In particular, we point out that measurement noise and discretization due to numerical integration can lead to simulations that do not hit the surface and never jump. Following the ideas discussed above in zero-crossing detection algorithms used in software packages, we propose a mathematical model of a hybrid system incorporating a zero-crossing algorithm. We prove that, under mild regularity conditions, such a resulting hybrid system not only includes all of the nominal solutions (no perturbations) to the original system but is also robust to measurement noise. We argue that, rather than computing the solutions to the discretization of the fragile nominal model (1)-(2), integration schemes with zero-crossing detection actually compute the solutions of a robustified version of the fragile nominal model. Finally, we propose a hybrid simulator for the hybrid system with incorporated zero-crossing detection. As a difference to [14, 15, 16], we focus on detection of zero-crossing rather than accurate location, which is another important issue in simulating systems like (1)-(2). Even though we do not discuss finding accurate locations of zero-crossings, we show that our proposed hybrid simulator enjoys the following properties, which are illustrated in examples throughout the paper:

1) Trajectories obtained with the proposed simulator with zero-crossing detection approximate the solutions to the original hybrid system with arbitrary precision. See Theorem 5.8.
2) The proposed simulator with zero-crossing detection preserves the asymptotic stability properties of the original hybrid system. Furthermore, the proposed simulator with zero-crossing detection has an asymptotically stable compact set that converges to the asymptotically stable compact set of the original hybrid system. See Theorems 5.10 and 5.11.

An important feature of the proposed hybrid simulator is that it confers the above properties to the simulation of the original hybrid system by minimally affecting the original system. In fact, the only addition to the original system consists of a scalar variable with appropriately defined dynamics for robust triggering of jumps; cf. the thickening of the switching surfaces proposed in [12] and the domain relaxations proposed in [17]. Furthermore, the proposed hybrid simulator is designed to guarantee stability and robustness properties when implemented numerically through the use of well-known solvers, which makes it suitable for the simulation of complex systems, such as hybrid and cyber-physical systems.

The remainder of this paper is organized as follows. Section 2 introduces notation used throughout the paper. Section 3 presents a mathematical model of hybrid systems with dynamics (1)-(2), a hybrid simulator for it, as well as issues with perturbations. A model of a hybrid system with added zero-crossing detection and a hybrid simulator for it are given in Section 4. Our main results appear in Section 5. A preliminary version of the results in this paper appeared without proof in [18].

2. Notation

The following notation will be used throughout the paper.

- $\mathbb{R}^n$ denotes the n-dimensional Euclidean space.
- $\mathbb{R}$ denotes the real numbers.
- $\mathbb{R}_{\geq 0}$ denotes the nonnegative real numbers, i.e., $\mathbb{R}_{\geq 0} = [0, \infty)$.
- $\mathbb{N}$ denotes the natural numbers including 0, i.e., $\mathbb{N} = \{0, 1, ...\}$.
- int S denotes the interior of the set S.
- sgn denotes the sign function defined as $\operatorname{sgn}(\xi) = 1$ if $\xi > 0$, $\operatorname{sgn}(\xi) = -1$ if $\xi < 0$, and $\operatorname{sgn}(\xi) = 0$ if $\xi = 0$.
- $\mathbb{B}$ denotes the closed unit ball centered at the origin in Euclidean space.
- $x + \varepsilon\mathbb{B}$ denotes a closed ball of radius ε centered at x.
- Given a set $\mathcal{S}$, $\overline{\mathcal{S}}$ denotes its closure.
- $T_{\mathcal{S}}(x)$ denotes the tangent cone of the surface of $\mathcal{S}$ at x (i.e., the collection of vectors that are tangent to the surface).
- Given a set $\mathcal{S}$, $\operatorname{co}\mathcal{S}$ denotes the convex hull of $\mathcal{S}$, and $\overline{\operatorname{co}}\mathcal{S}$ the closure of the convex hull.
- Given sets $\mathcal{A}_s$ and $\mathcal{A}$, $d_H(\mathcal{A}_s, \mathcal{A})$ denotes the Hausdorff distance between sets $\mathcal{A}_s$ and $\mathcal{A}$.
- Given a set $K \subset \mathbb{R}^n$ and a point $x \in \mathbb{R}^n$, $|x|_K := \inf_{y \in K} |x - y|$.

3. Hybrid Systems Jumping on Surfaces

Throughout this paper, we model systems and their simulators within the hybrid systems framework of [19] and [20]. In this way, we write the system (1)-(2) as

$$\mathcal{H}: \quad \left\{ \begin{array}{rcll} \dot{x} & = & f(x) & x \in \mathcal{X} \setminus \mathcal{S} =: C \\ x^+ & = & g(x) & x \in \mathcal{S} \cap \mathcal{X} =: D. \end{array} \right. \tag{4}$$

Following [19], a solution to a hybrid system is a function defined on a hybrid time domain satisfying certain conditions. A set $E \subset \mathbb{R}_{\geq 0} \times \mathbb{N}$ is a *compact hybrid time domain* if

$$E = \bigcup_{j=0}^{J-1} ([t_j, t_{j+1}] \times \{j\})$$

for some finite sequence of times $0 = t_0 \leq t_1 \leq t_2 ... \leq t_J$. The set E is a *hybrid time domain* if for all $(T, J) \in E$, $E \cap ([0, T] \times 0, 1, ...J)$ is a compact hybrid domain. By *hybrid arc* or *hybrid trajectory* we understand a pair consisting of a hybrid time domain $\operatorname{dom} x$ and a function $x : \operatorname{dom} x \to \mathbb{R}^n$ such that, for each j, $t \mapsto x(t, j)$ is locally absolutely continuous for $(t, j) \in \operatorname{dom} x$. A hybrid arc $\phi : \operatorname{dom} \phi \to \mathbb{R}^n$ is a *solution* to a hybrid system $\mathcal{H}$ with data (C, f, D, g) if

(S0) $\phi(0, 0) \in C \cup D$;

(S1) For each $j \in \mathbb{N}$ and each $I_j := \{t \ : \ (t, j) \in \operatorname{dom} \phi \ \}$ with nonempty interior $\operatorname{int} I_j$,

$$\begin{array}{ll} \phi(t, j) \in C & \text{for all } t \in \operatorname{int} I_j \\ \dot{\phi}(t, j) = f(\phi(t, j)) & \text{for almost all } t \in \operatorname{int} I_j; \end{array}$$

(S2) For each $(t, j) \in \operatorname{dom} \phi$ such that $(t, j + 1) \in \operatorname{dom} \phi$,

$$\phi(t, j) \in D, \qquad \phi(t, j + 1) = g(\phi(t, j)).$$

Now we introduce a few examples that will be revisited throughout the paper.

Example 3.1. *(bouncing ball) Consider a bouncing ball as a point mass vertically bouncing on a horizontal surface. The ball is dropped from some initial height and experiences acceleration due to gravity. At each impact with the surface, the velocity can be approximated as instantaneously reversing direction and possibly decreasing in magnitude due to energy dissipation. If we denote the state of the system as* $x = [x_1 \ x_2]^\top$ *where the height above the surface is* x_1 *and the vertical velocity is* x_2*, then we can write the bouncing ball system in our hybrid systems framework as*

$$\begin{array}{l} \mathcal{X} \subset \mathbb{R}^2, \\ f(x) = \left[\begin{array}{c} x_2 \\ -M(x_1) - N(x_1) x_2 \end{array} \right], \quad g(x) \ = \ \left[\begin{array}{c} 0 \\ -\lambda x_2 \end{array} \right], \\ \mathcal{S} = \{ x \in \mathcal{X} \ : \ x_1 = 0, x_2 \leq -\delta \ \}, \end{array} \tag{5}$$

where M *and* N *are given as*

$$M(x_1) = \left\{ \begin{array}{ll} \frac{\gamma}{\varepsilon} x_1 & \textit{for} \ \ x_1 < \varepsilon \\ \gamma & \textit{for} \ \ x_1 \geq \varepsilon \end{array} \right. , \quad N(x_1) = \left\{ \begin{array}{ll} -x_1 + \varepsilon & \textit{for} \ \ x_1 < \varepsilon \\ 0 & \textit{for} \ \ x_1 \geq \varepsilon \end{array} \right. ,$$

$\gamma > 0$ *is the gravitational constant,* $\lambda \in [0, 1)$ *is the restitution coefficient, and* δ *and* ε *are positive constants. This model comes from the non-Zeno bouncing ball model described in [21, Example 4.2]. In this model, the functions* M *and* N *are used to capture compression of the ball at impacts with the surface. When* $x_1 \geq \varepsilon > 0$*, this model is equivalent to a Zeno bouncing ball model.* △

Example 3.2. *(unicycle avoiding obstacle) Consider a mobile robot of the unicycle type being steered towards a target while avoiding a circular obstacle [2]. Let $\xi = [\xi_1\ \xi_2\ \xi_3]^\top$, where $[\xi_1\ \xi_2]^\top$ denotes the robot's position and ξ_3 the robot's orientation. Let q be the controller state, $q \in \{1, 2\}$, where $q = 1$ means travelling towards the target, and $q = 2$ means travelling away from the obstacle. The value of q is chosen depending on the robot's radial distance from the obstacle. Two circular surfaces, $\mathcal{S}_q$, with radii $a_q, a_2 > a_1$, are defined around the obstacle for this purpose. When the robot enters one of these surfaces, the continuous dynamics of the system do not change, but if a change in direction is desired, the controller state must be updated from $q = 1$ to $q = 2$, or vice versa, according to the update law $q^+ = 3 - q$. The current orientation of the robot depends on the value of a function defining a local mode-based controller.*

Denoting the state of the system by $x = [\xi_1\ \xi_2\ \xi_3\ q]^\top$, the closed-loop system can be written as

$$\begin{aligned}
&\mathcal{X} \subset \mathbb{R}^3 \times \{1, 2\},\\
&f(x) = \begin{bmatrix} \begin{bmatrix} v\cos(\xi_3) \\ v\sin(\xi_3) \\ \kappa(q, \xi) \end{bmatrix} \\ 0 \end{bmatrix}, \quad g(x) \;=\; \begin{bmatrix} \xi \\ 3 - q \end{bmatrix},\\
&\mathcal{S} = \cup_{q\in\{1,2\}}(\mathcal{S}_q \times \{q\}),\\
&\mathcal{S}_q = \left\{x \in \mathcal{X}\ :\ (\xi_1^\circ - \xi_1)^2 + (\xi_2^\circ - \xi_2)^2 = a_q^2\ \right\},
\end{aligned}\tag{6}$$

where v is the tangential velocity of the robot, the function $\kappa(q, \xi)$ defines the mode-based controller which the robot should use in order to steer the robot to the target ($q = 1$) or away from the obstacle ($q = 2$), and $(\xi_1^\circ, \xi_2^\circ)$ denotes the obstacle's location. △

Example 3.3. *(unicycle on a track) Consider a mobile robot of the unicycle type being steered to travel along a given track with state ξ as in Example 3.2. Let q be the controller state, $q \in \{1, 2\}$, where $q = 1$ means traveling to the left, and $q = 2$ means traveling to the right.*[1] *The value of q is chosen depending on whether the robot crosses the left or right boundaries of the track. The boundaries are given by surfaces $\mathcal{S}_q$.*

Denoting the state of the system by $x = [\xi_1\ \xi_2\ \xi_3\ q]^\top$, the closed-loop system is given by

$$\begin{aligned}
&\mathcal{X} \subset \mathbb{R}^3 \times \{1, 2\},\\
&f(x) = \begin{bmatrix} \begin{bmatrix} v\cos(\xi_3) \\ v\sin(\xi_3) \\ \kappa(q, \xi) \end{bmatrix} \\ 0 \end{bmatrix}, \quad g(x) = \begin{bmatrix} \xi \\ 3 - q \end{bmatrix},\\
&\mathcal{S} = \cup_{q\in\{1,2\}}(\mathcal{S}_q \times \{q\}),\\
&\mathcal{S}_1 = \{x \in \mathcal{X}\ :\ \xi_1 = 1,\ q = 1\ \},\\
&\mathcal{S}_2 = \{x \in \mathcal{X}\ :\ \xi_1 = -1,\ q = 2\ \},
\end{aligned}\tag{7}$$

where v is the tangential velocity of the robot, and the function $\kappa(q, \xi)$ defines the mode-based controller which the robot should use in order to stay within the track. △

3.1. Nonrobustness to measurement noise

As pointed out in Section 1, if the measurements of the state x are noisy, solutions to $\mathcal{H}$ may fail to jump due to never belonging to $\mathcal{S}$. In fact, for every nominal solution to $\mathcal{H}$, it is possible to construct an

[1] A control mode to keep the vehicle within the track could also be incorporated.

arbitrarily small measurement noise signal e so that $x + e \in \mathcal{S}$ never holds. A hybrid system $\mathcal{H}$ with data (C, f, D, g) and measurement noise $e : \operatorname{dom} e \to \mathbb{R}^n$ is denoted $\mathcal{H}^e$ and is given by

$$\mathcal{H}^e : \begin{cases} \dot{x} & = & f(x+e) & x+e \in C \\ x^+ & = & g(x+e) & x+e \in D. \end{cases}$$

A *solution* to $\mathcal{H}^e$, that is, a solution to $\mathcal{H}$ for a given measurement noise e, consists of a hybrid arc $\phi^e : \operatorname{dom} \phi^e \to \mathbb{R}^n$ satisfying

(S0^e) $\phi^e(0,0) + e(0,0) \in C \cup D$;

(S1^e) For each $j \in \mathbb{N}$ and each $I_j = \{t \ : \ (t,j) \in \operatorname{dom} \phi^e \ \}$ with nonempty interior $\operatorname{int} I_j$,

$$\begin{array}{ll} \phi^e(t,j) + e(t,j) \in C & \text{for all } t \in \operatorname{int} I_j \\ \dot{\phi}^e(t,j) = f(\phi^e(t,j) + e(t,j)) & \text{for almost all } t \in \operatorname{int} I_j; \end{array}$$

(S2^e) For each $(t,j) \in \operatorname{dom} \phi^e$ such that $(t, j+1) \in \operatorname{dom} \phi^e$,

$$\phi^e(t,j) + e(t,j) \in D, \quad \phi^e(t,j+1) = g(\phi^e(t,j) + e(t,j)).$$

The following proposition formalizes the fact that arbitrarily small measurement noise may prevent solutions to $\mathcal{H}$ from jumping on $\mathcal{S}$. We say that a function $\ell : \mathbb{R}^n \to \mathbb{R}^n$ is locally bounded on an open set O if for each compact set $K \subset O$ there exists a compact set $K' \subset \mathbb{R}^n$ such that $\ell(K) \subset K'$. Also, we say that a set K is a codimension one submanifold of $\mathbb{R}^n$ if $K \subset \mathbb{R}^n$ and $n - \dim(K) = 1$.

Proposition 3.4. *(no jumps due to measurement noise) Suppose $\mathcal{H} = (C, f, D, g)$ as in* (4) *is such that*

1. *$f : \mathbb{R}^n \to \mathbb{R}^n$ and $g : \mathbb{R}^n \to \mathbb{R}^n$ are locally bounded on an open set containing $\mathcal{X}$;*
2. *$\mathcal{S} \cap \mathcal{X}$ is a codimension one submanifold of $\mathbb{R}^n$.*

Then, for each $\varepsilon > 0$, $T > 0$, and $x_0 \in \mathcal{X}$, every solution ϕ^e to $\mathcal{H}$ with measurement noise e and $\phi^e(0,0) = x_0$ satisfies $\operatorname{dom} \phi^e \subset [0,T] \times \{0\}$*, for some measurable function $e : \operatorname{dom} e \to \varepsilon\mathbb{B}$.*

Proof. Following [12, Definition 2.12], the Krasovskii regularization of $\mathcal{H}$ is given by $\widehat{\mathcal{H}} = (\widehat{C}, \widehat{f}, \widehat{D}, \widehat{g})$, where $\widehat{C} = \overline{C}, \widehat{D} = \overline{D}$ while $\widehat{f} : \widehat{C} \rightrightarrows \mathbb{R}^n, \widehat{g} : \widehat{D} \rightrightarrows \mathbb{R}^n$ are defined by

$$\forall x \in \widehat{C} \quad \widehat{f}(x) := \bigcap_{\delta > 0} \overline{co} f((x + \delta\mathbb{B}) \cap C), \qquad \forall x \in \widehat{D} \quad \widehat{g}(x) := \bigcap_{\delta > 0} \overline{g((x + \delta\mathbb{B}) \cap D)}.$$

Using item 2 of the assumptions, the flow set $C = \mathcal{X} \setminus \mathcal{S}$ is such that $\overline{C} = \overline{\mathcal{X}}$. Then, $\widehat{C} = \overline{\mathcal{X}}$. It follows that for every $x_0 \in \mathcal{X}$ there exists a maximal solution $\widehat{\phi}$ to $\widehat{\mathcal{H}}$ that only flows, i.e., $\operatorname{dom} \widehat{\phi} \subset \mathbb{R}_{\geq 0} \times \{0\}$. Using item 1 of the assumptions, [12, Theorem 3.1] implies that every Krasovskii solution to $\mathcal{H}$ is a Hermes solution to $\mathcal{H}$, in particular, the solution $\widehat{\phi}$ is a Hermes solution to $\mathcal{H}$. Then, for every compact hybrid time domain $E := [0,T] \times \{0\} \subset \operatorname{dom} \widehat{\phi}$, there exists a sequence of solutions $\widehat{\phi}_i$ to $\mathcal{H}$ with measurement noise $e_i : \operatorname{dom} e_i \to \varepsilon_i \mathbb{B}$, $\varepsilon_i \searrow 0$, with domains $\operatorname{dom} \widehat{\phi}_i = \operatorname{dom} e_i$ graphically converging to the truncation of $\widehat{\phi}$ to E. Picking i such that that $\varepsilon_i < \varepsilon$, the claim follows with $\phi^e = \widehat{\phi}_i$ and $e = e_i$. ■

3.2. Properties of numerical simulations of $\mathcal{H}$

The simulation of $\mathcal{H}$ can be interpreted as the numerical computation of the solutions to the discretized version of $\mathcal{H}$, defining a simulator. A *hybrid simulator for $\mathcal{H}$* is given by the family of systems $\mathcal{H}^s$ parameterized by step size s satisfying $s \in (0, s^*]$, with the maximum step size $s^* > 0$. The data of the hybrid simulator $\mathcal{H}^s$ is denoted by (C^s, f^s, D^s, g^s). For simplicity, we will assume that the region of operation $\mathcal{X}$ is not discretized. Following [20], a hybrid simulator $\mathcal{H}^s$ for the hybrid system $\mathcal{H}$ is written as

$$\mathcal{H}^s \left\{ \begin{array}{rcll} x_s^+ & = & f^s(x_s) & x_s \in \mathcal{X} \setminus \mathcal{S}^s =: C^s \\ x_s^+ & = & g^s(x_s) & x_s \in \mathcal{S}^s \cap \mathcal{X} =: D^s. \end{array} \right.$$

Comparing $\mathcal{H}$ with $\mathcal{H}^s$, the dynamics for the flows of $\mathcal{H}$ have been replaced by the integration scheme $x_s^+ = f^s(x_s)$, where f^s is constructed from f. The jump map of $\mathcal{H}$ has been replaced by the discretized map g^s, and the flow and jump sets C and D have been replaced by the discretized sets C^s and D^s, where $\mathcal{S}^s$ is the discretization of $\mathcal{S}$.

The dynamics of the hybrid simulator $\mathcal{H}^s$ are purely discrete, so the solutions to $\mathcal{H}^s$ are given on discrete versions of hybrid time domains. Following the definition of hybrid time domain, a subset $E \subset \mathbb{N} \times \mathbb{N}$ is a compact discrete time domain if

$$E = \bigcup_{j=0}^{J-1} \bigcup_{k=K_j}^{K_{j+1}} (k, j)$$

for some finite sequence $0 = K_0 \leq K_1 \leq K_2 ... \leq K_J$, $K_j \in \mathbb{N}$ for every $j \leq J$, $j \in \mathbb{N}$. It is a discrete time domain if $\forall (K, J) \in E$, $E \cap (\{0, 1, \ldots K\} \times \{0, 1, \ldots J\})$ is a compact discrete time domain. Solutions to $\mathcal{H}^s$ are given by *discrete arcs* $\phi^s : \operatorname{dom} \phi^s \to \mathbb{R}^n$, where $\operatorname{dom} \phi^s$ is a discrete time domain. Then, a discrete arc $\phi^s : \operatorname{dom} \phi^s \to \mathbb{R}^n$ is a *solution to the hybrid system* $\mathcal{H}$ with a hybrid simulator $\mathcal{H}^s$, $s > 0$, if

(S1$_s$) for all $k, j \in \mathbb{N}$ such that $(k, j), (k+1, j) \in \operatorname{dom} \phi^s$,

$$\phi^s(k, j) \in C^s, \quad \phi^s(k+1, j) = f^s(\phi^s(k, j));$$

(S2$_s$) for all $k, j \in \mathbb{N}$ such that $(k, j), (k, j+1) \in \operatorname{dom} \phi^s$,

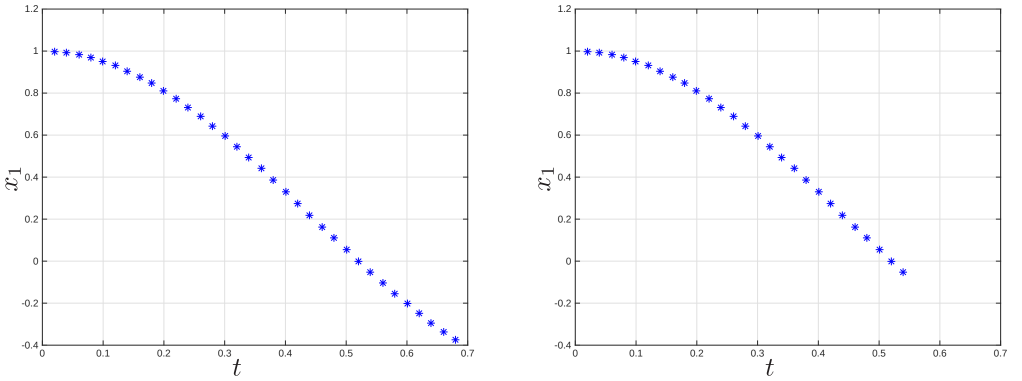
$$\phi^s(k, j) \in D^s, \quad \phi^s(k, j+1) = g^s(\phi^s(k, j)).$$

Similar to the lack of robustness to measurement noise highlighted in Proposition 3.4, numerical simulations of hybrid systems are susceptible to problems due to numerical approximations. For example, when implementing the hybrid system $\mathcal{H}$ in a simulator, the discretization in time produced by the ODE solver may prevent jumps from being triggered since the condition $x_s \in \mathcal{S}^s$ may not hold. To illustrate this, consider the flow map f discretized with an Euler integration scheme, i.e. $f^s(x) = x + s$. It follows that for every $s^* > 0$ and each $x_0 \in \mathcal{X}$, every solution ϕ^s to $\mathcal{H}^s$ with some step size $s \in (0, s^*]$ and $\phi^s(0, 0) = x_0$ has less jumps than the original solution. In fact, fix the initial condition $x_0 \in \mathcal{X}$, and suppose that for each $s \in (0, s^*]$ and every solution ϕ^s to $\mathcal{H}_s$, we have $\phi^s(k^*, 0) \in \mathcal{S}^s$ for some $k^* \in \mathbb{N}$ (depending on s and ϕ^s). Then, by the definition of solution to $\mathcal{H}^s$, we have $\phi^s(k, 0) = f^s(\phi^s(k-1, 0))$ for each $k = 1, 2, \ldots, k^*$ with $\phi^s(0, 0) = x_0$. Equivalently, we can write that $\phi^s(k^*, 0) = f^s \circ f^s \circ \ldots \circ f^s(x_0) =: (f^s)^{k^*}(x_0)$, where $(f^s)^{k^*}$ denotes k^* compositions of f^s. By continuity in s of the resulting map, we have that $\phi^s(k^*, 0)$ cannot be in $\mathcal{S}^s$ for each s.

Now, we illustrate this issue in the three examples of hybrid systems jumping on surfaces introduced earlier. Throughout this paper, all numerical examples are simulated using the Hybrid Equations (HyEQ) Toolbox as described in [22].

Example 3.5. *(bouncing ball (revisited)) Consider a hybrid simulator $\mathcal{H}^s$ for the system in Example 3.1 with $g^s = g$, $\mathcal{S}^s = \mathcal{S}$, and f^s given by the Euler integration scheme, i.e., $f^s(x) = x + sf(x) = x + s\left[x_2, \ -M(x_1) - N(x_1)x_2\right]^\top$. The type of solutions to $\mathcal{H}^s$ depend on the definition of $\mathcal{X}$. For the choice $\mathcal{X} = \left\{x \in \mathbb{R}^2 \ : \ x_1 \geq 0 \ \right\} \cup \left\{x \in \mathbb{R}^2 \ : \ x_2 \leq -\delta \ \right\}$, $\mathcal{H}^s$ has solutions that only flow and step over $\mathcal{S}$ (for some step size s), which corresponds to the ball passing through the surface on which it is supposed to bounce.*

For the choice $\mathcal{X} = \left\{x \in \mathbb{R}^2 \ : \ x_1 \geq 0 \ \right\}$, $\mathcal{H}^s$ has solutions that may flow and step over the surface $\mathcal{S}$ (because x_1 may never be identically equal to 0), leaving $\mathcal{X}$ after a finite number of steps. In other words, the simulation stops prematurely because solutions are not in $\mathcal{X}$ or $\mathcal{S}$.



(a) Height of a bouncing ball with solutions that pass through $\mathcal{S}$. The region of operation is $\mathcal{X} = \{ x \in \mathbb{R}^2 \ : \ x_1 \geq 0 \} \cup \{ x \in \mathbb{R}^2 \ : \ x_2 \leq -\delta \}$.

(b) Height of a bouncing ball with solutions that stop prematurely. The region of operation is $\mathcal{X} = \{ x \in \mathbb{R}^2 \ : \ x_1 \geq 0 \}$.

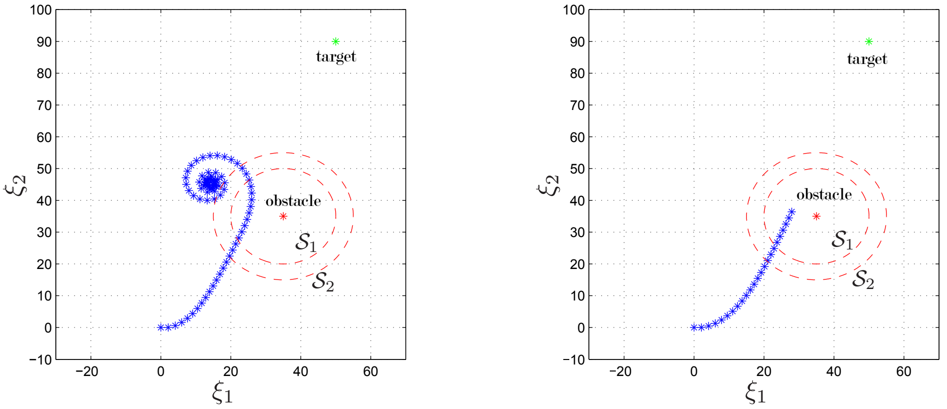
Figure 1: Solutions to the bouncing ball system in Example 3.5.

Figures 1(a) and 1(b) show simulation results with this simulator for $\gamma = 9.81$, $\lambda = 0.8$, $\delta = 0.1$, $\varepsilon = 1$, *and step size* $s = 0.02$. *The ball is dropped from an initial height of* 1, *and the step size is such that the solution continues flowing and passes through* $\mathcal{S}$. *In the simulation shown in Figure 1(b), the integration stops prematurely due to the simulated solution leaving* $\mathcal{X}$. *In each case, the results obtained from the simulations do not represent the behavior of the physical system because, due to discretization, the system does not jump, i.e., solutions to* $\mathcal{H}^s$ *never satisfy the condition* $x \in \mathcal{S}$. △

Example 3.6. *(unicycle avoiding obstacle (revisited)) Consider a hybrid simulator* $\mathcal{H}^s$ *for the system in Example 3.2 with* $g^s = g$, $\mathcal{S}^s = \cup_{q \in \{1,2\}} (\mathcal{S}_q \times \{q\})$, *and* f^s *given by the Euler integration scheme, i.e.,* $f^s(x) = x + sf(x) = x + s \left[v\cos(\xi_3), \ v\sin(\xi_3), \ \kappa(q,\xi), \ 0 \right]^\top$. *The robot changes steering modes when it crosses the surface* $\mathcal{S}_q$ *in order to maneuver around the obstacle and towards the target. As pointed out in Section 1, arbitrarily small measurement noise can prevent the robot from switching modes when reaching a boundary. This could lead the robot to collide with the obstacle or move away from the target.*

Figure 2 shows a solution to $\mathcal{H}^s$ *starting at* $(\xi_1, \xi_2) = (0, 0)$ *moving towards the green target at* $(\xi_1^*, \xi_2^*) = (50, 90)$ *while operating in mode* $q = 1$. *The obstacle is at* $(\xi_1^\circ, \xi_2^\circ) = (35, 35)$ *with radius smaller than* a_1. *The radii of the surfaces around the obstacle are* $a_1 = 15$ *and* $a_2 = 20$. *The control law is chosen as (see [2])* $\kappa(1,\xi) = 0.9(\arctan \frac{90-\xi_2}{50-\xi_1} - \xi_3)$ *and* $\kappa(2,\xi) = 0.2(\xi_3 - \arctan \frac{35-\xi_2}{35-\xi_1})$. *Figure 2(a) shows a solution to* $\mathcal{H}^s$ *where the robot crosses the surface* $\mathcal{S}_1$, *changes mode to* $q = 2$ *to move away from the obstacle, and then, due to discretization effects, steps over* $\mathcal{S}_2$ *and fails to change mode back to* $q = 1$. *On the other hand, Figure 2(b) shows the situation where a jump to mode* $q = 2$ *is not triggered. Both of these cases can occur due to numerical error in the simulation.* △

Example 3.7. *(unicycle on a track (revisited)) Consider a hybrid simulator* $\mathcal{H}^s$ *for the system in Example 3.3 with* $g^s = g$, $\mathcal{S}^s = \cup_{q \in \{1,2\}} (\mathcal{S}_q \times \{q\})$, *and* f^s *given by the Euler integration scheme, i.e.,* $f^s(x) = x + sf(x) = x + s \left[v\cos(\xi_3), \ v\sin(\xi_3), \ \kappa(q,\xi), \ 0 \right]^\top$. *The robot changes steering modes when it crosses the surface* $\mathcal{S}_q$ *in order to steer the robot back inside the track. Figure 3 shows a solution to* $\mathcal{H}^s$ *when the tangential velocity of the vehicle is* $v = 1$, *the control law is* $\kappa(1,\xi) = \frac{\pi}{4} - \xi_3$ *and* $\kappa(2,\xi) = \frac{3\pi}{4} - \xi_3$, *the initial position in the plane is* $(\xi_1, \xi_2) = (0, 0)$, *and the initial orientation angle is* $\xi_3 = \frac{\pi}{4}$ *radians. With an arbitrarily small step size, discretization effects cause the solution to step over the surface* $\mathcal{S}$ *and miss the mode change.* △



(a) Solution of robot missing second mode change and never reaching target.

(b) Solution of robot missing first mode change and potentially colliding with obstacle.

Figure 2: Solutions to the unicycle system in Example 3.6 that do not capture mode changes. The target is located at $(\xi_1^*, \xi_2^*) = (50, 90)$ and the obstacle at $(\xi_1^\circ, \xi_2^\circ) = (35, 35)$.

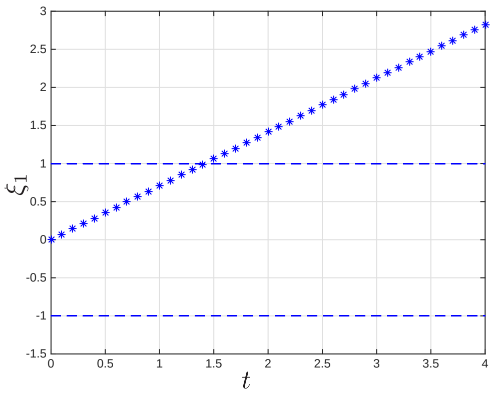


Figure 3: Solution for Example 3.7 that steps over $\mathcal{S}$.

4. A hybrid model for hybrid systems jumping on surfaces with zero-crossing detection

4.1. A hybrid system model of $\mathcal{H}$ with zero-crossing detection

For the simulation of nonlinear systems, software packages use special algorithms to capture when solutions hit a surface. As pointed out in Section 1, a memory state variable can be added to evaluate condition (3) and determine when the state should jump.

For example, when using the *Hit Crossing* block in MATLAB/Simulink, shown in Figure 4, to detect the jumps of system (4) with scalar state x and $\mathcal{S} := \{x \in \mathbb{R} \ : \ x = x^* \ \}$, the input u would be equal to the state of the system x, and the output of the block would be equal to a value $\{0, 1\}$, where 0 corresponds to flowing and 1 corresponds to a hit on or crossing of $x^* \in \mathbb{R}$. Denoting the memory state by z and initializing it to a value with the same sign as $x(0) - x^*$, which determines which "side" of $\mathcal{S}$ the state starts from, the operation of the *Hit Crossing* block might be described as follows:

$$\text{when } (x(t) - x^*)z(t) > 0 \quad \Rightarrow \quad \text{keep flowing} \tag{8}$$

$$\text{when } (x(t) - x^*)z(t) \le 0 \quad \Rightarrow \quad \text{jump.} \tag{9}$$

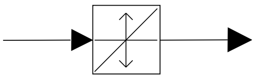


Figure 4: Hit Crossing block in MATLAB/Simulink.

In general, this mechanism can be captured by a function that changes sign according to the location of x with respect to $\mathcal{S}$. We call such a function a *zero-crossing function* (cf. [23, Chapter 3]).

Definition 4.1 (zero-crossing function). *A* zero-crossing function *on a set $\mathcal{X} \subset \mathbb{R}^n$ for a switching surface $\mathcal{S}$ is given by a function $h : \mathcal{X} \to \mathbb{R}$ that implicitly characterizes $\mathcal{S}$ and splits $\mathcal{X}$ into two sets $\mathcal{X}_1, \mathcal{X}_2 \subset \mathbb{R}^n$ as follows:*

$$\begin{aligned} \mathcal{S} \cap \mathcal{X} &= \{x \in \mathcal{X} : h(x) = 0\}, \\ \mathcal{X}_1 &= \{x \in \mathcal{X} : h(x) < 0\}, \\ \mathcal{X}_2 &= \{x \in \mathcal{X} : h(x) > 0\}. \end{aligned}$$

For Example 3.1, a zero-crossing function h on $\mathcal{X} = \left\{x \in \mathbb{R}^2 \ : \ x_1 \geq 0 \ \right\} \cup \left\{x \in \mathbb{R}^2 \ : \ x_2 \leq -\delta \ \right\}$, for $\mathcal{S}$ as in (5), is given by

$$h(x) := c_1(x)x_1^2 + c_2(x_2)x_2^2 \qquad \forall x \in \mathcal{X} \tag{10}$$

with $c_1(x)x_1^2$ continuously differentiable and such that $c_1(x) = 1$ if $x_1 \geq 0$ and $c_1(x) = -1$ if $x_1 < 0$ and $x_2 \leq -\delta$, while c_2 is such that $c_2(x_2) = 1$ if $x_2 > -\delta$ and $c_2(x_2) = 0$ otherwise. The sets $\mathcal{X}_1$ and $\mathcal{X}_2$ become

$$\begin{aligned} \mathcal{X}_1 &= \{x \in \mathcal{X} \ : \ x_1 < 0, x_2 \leq -\delta \ \}, \\ \mathcal{X}_2 &= \{x \in \mathcal{X} \ : \ x_1 > 0 \ \} \cup \{x \in \mathcal{X} \ : \ x_1 = 0, x_2 > -\delta \ \}. \end{aligned}$$

For Example 3.2, a zero-crossing function h on $\mathcal{X} = \mathbb{R}^3 \times \{1, 2\}$ for $\mathcal{S}$ as in (6) is given by

$$h(x) = h_q(\xi) := (\xi_1^\circ - \xi_1)^2 + (\xi_2^\circ - \xi_2)^2 - a_q^2 \qquad \forall x \in \mathcal{X}, \tag{11}$$

and the sets $\mathcal{X}_1$ and $\mathcal{X}_2$ become

$$\begin{aligned} \mathcal{X}_1 &= \left(\left\{\xi \in \mathbb{R}^3 \ : \ h_1(\xi) < 0 \ \right\} \times \{1\}\right) \cup \left(\left\{\xi \in \mathbb{R}^3 \ : \ h_2(\xi) < 0 \ \right\} \times \{2\}\right), \\ \mathcal{X}_2 &= \left(\left\{\xi \in \mathbb{R}^3 \ : \ h_1(\xi) > 0 \ \right\} \times \{1\}\right) \cup \left(\left\{\xi \in \mathbb{R}^3 \ : \ h_2(\xi) > 0 \ \right\} \times \{2\}\right). \end{aligned}$$

For Example 3.3, a zero-crossing function h on $\mathcal{X} = \mathbb{R}^3 \times \{1, 2\}$ for $\mathcal{S}$ as in (7) is given by

$$h(x) = h(\xi) := \begin{cases} 1 - |\xi_1| & |\xi_1| < 1 \text{ or } |\xi_1| > 1 \\ 0 & |\xi_1| = 1 \end{cases} \qquad \forall x \in \mathcal{X}, \tag{12}$$

and the sets $\mathcal{X}_1$ and $\mathcal{X}_2$ become

$$\begin{aligned} \mathcal{X}_1 &= \left\{\xi \in \mathbb{R}^3 \ : \ |\xi_1| > 1 \ \right\} \\ \mathcal{X}_2 &= \left\{\xi \in \mathbb{R}^3 \ : \ |\xi_1| < 1 \ \right\}. \end{aligned}$$

A version of the hybrid system $\mathcal{H}$ with zero-crossing detection capabilities is denoted $\mathcal{H}_{\text{ZCD}} = (C_{\text{ZCD}}, f_{\text{ZCD}}, D_{\text{ZCD}}, g_{\text{ZCD}})$ and is given by

$$\begin{aligned} \begin{bmatrix} \dot{x} \\ \dot{z} \end{bmatrix} &= \begin{bmatrix} f(x) \\ 0 \end{bmatrix} =: f_{\text{ZCD}}(x) \qquad (x, z) \in C_{\text{ZCD}} \\ \begin{bmatrix} x^+ \\ z^+ \end{bmatrix} &= \begin{bmatrix} g(x) \\ h(g(x)) \end{bmatrix} =: g_{\text{ZCD}}(x) \quad (x, z) \in D_{\text{ZCD}}, \end{aligned} \tag{13}$$

where $z \in \mathbb{R}$ is a memory state, h is a *zero-crossing function* on $\mathcal{X}$ for $\mathcal{S}$, and

$$C_{\text{ZCD}} := \{(x,z) \in \mathcal{X} \times \mathbb{R} \; : \; h(x)z \geq 0 \,\},$$
$$D_{\text{ZCD}} := \{(x,z) \in \mathcal{X} \times \mathbb{R} \; : \; h(x)z \leq 0 \,\}.$$

The memory state z is added to keep track of whether the state x is in the side of $\mathcal{S}$ with h negative ($x \in \mathcal{X}_1$) or in the side of $\mathcal{S}$ with h positive ($x \in \mathcal{X}_2$). At jumps, if $g(x) \in \mathcal{X}_1$, then $h(g(x)) < 0$. Similarly if $g(x) \in \mathcal{X}_2$, then $h(g(x)) > 0$. The value of z is always reset to $h(g(x))$ so that after each jump x is in the flow set. In this way, solutions flow when $h(x)$ and z have the same sign (i.e., $h(x)z \geq 0$) and jump when $h(x)$ evaluated along the solution attempts to take a different sign from that of z ($h(x)z \leq 0$).

Note that $h(x)$ captures the term $x(t) - x^*$ in (8) and (9) (see also (3)). Furthermore, C_{ZCD} corresponds to the condition (8) while D_{ZCD} corresponds to (9). The equality included in C_{ZCD} does not affect solutions when the flow map f is not tangent to $\mathcal{S}$.

4.2. A numerical simulation model of $\mathcal{H}_{ZCD}$

Given a hybrid system $\mathcal{H}$ as in (4) and its augmentation with zero-crossing detection $\mathcal{H}_{\text{ZCD}} = (C_{\text{ZCD}}, f_{\text{ZCD}}, D_{\text{ZCD}}, g_{\text{ZCD}})$ as in (13), a *hybrid simulator* for $\mathcal{H}_{\text{ZCD}}$ is given by the family of systems $\mathcal{H}^s_{\text{ZCD}}$ parameterized by step size s satisfying $s \in (0, s^*]$, $s^* > 0$. The data of the hybrid simulator $\mathcal{H}^s_{\text{ZCD}}$ is given by $(C^s_{\text{ZCD}}, f^s_{\text{ZCD}}, D^s_{\text{ZCD}}, g^s_{\text{ZCD}})$. The hybrid simulator $\mathcal{H}^s_{\text{ZCD}}$ is given by

$$\begin{bmatrix} x_s^+ \\ z_s^+ \end{bmatrix} = \begin{bmatrix} f^s(x_s) \\ 0 \end{bmatrix} =: f^s_{\text{ZCD}}(x_s, z_s) \qquad (x_s, z_s) \in C^s_{\text{ZCD}},$$
$$\begin{bmatrix} x_s^+ \\ z_s^+ \end{bmatrix} = \begin{bmatrix} g^s(x_s) \\ h^s(g^s(x_s)) \end{bmatrix} =: g^s_{\text{ZCD}}(x_s, z_s) \qquad (x_s, z_s) \in D^s_{\text{ZCD}},$$
$$C^s_{\text{ZCD}} := \{(x_s, z_s) \in \mathcal{X} \times \mathbb{R} \; : \; h^s(x_s)z_s \geq 0 \,\},$$
$$D^s_{\text{ZCD}} := \{(x_s, z_s) \in \mathcal{X} \times \mathbb{R} \; : \; h^s(x_s)z_s \leq 0 \,\}.$$

The dynamics of the x component for the flows of $\mathcal{H}_{\text{ZCD}}$ have been replaced by the integration scheme $x_s^+ = f^s(x_s)$, where f^s is constructed from f. The jump map of $\mathcal{H}_{\text{ZCD}}$ has been replaced by g^s_{ZCD}, and the flow and jump sets C_{ZCD} and D_{ZCD} have been replaced by the sets C^s_{ZCD} and D^s_{ZCD}, respectively. The function h^s is the discretization of the switching function h, and the memory state variable z has been replaced by z_s. The operation of z_s is the discretized equivalent to that of z in $\mathcal{H}_{\text{ZCD}}$. Because the dynamics of the hybrid simulator $\mathcal{H}^s_{\text{ZCD}}$ are purely discrete, the solutions to $\mathcal{H}^s_{\text{ZCD}}$ are given on discrete versions of hybrid time domains as previously described.

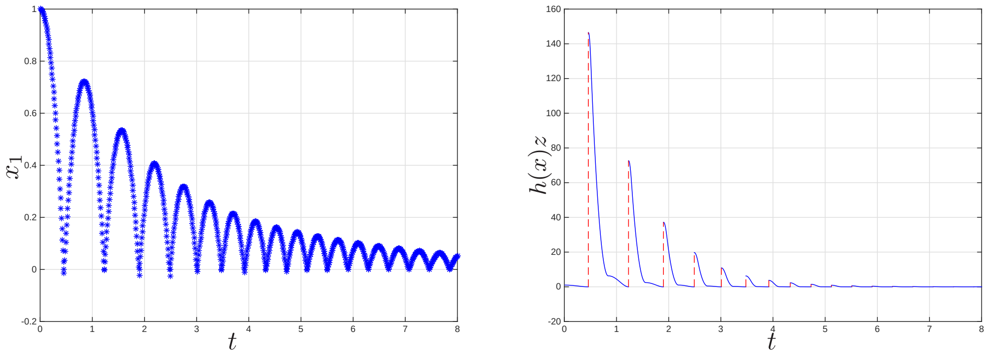
Now we revisit the examples and add zero-crossing detection to the models.

Example 4.2. *(bouncing ball (revisited)) Consider again the bouncing ball model in Example 3.1 with* $\mathcal{X} = \{x \in \mathbb{R}^2 \; : \; x_1 \geq 0 \,\} \cup \{x \in \mathbb{R}^2 \; : \; x_2 \leq -\delta \,\}$*. When adding zero-crossing detection to the model in* (5)*, we obtain the hybrid system*

$$f_{ZCD}(x,z) = \begin{bmatrix} x_2 \\ -M(x_1) - N(x_1)x_2 \\ 0 \end{bmatrix}, \quad g_{ZCD}(x,z) \; = \; \begin{bmatrix} 0 \\ -\lambda x_2 \\ h(g(x)) \end{bmatrix}$$
$$\mathcal{S} = \{x \in \mathcal{X} \; : \; x_1 = 0, x_2 \leq -\delta \,\},$$

where g is the jump map of the hybrid system in Example 3.1, and the functions M and N are also given in Example 3.1. Now consider a hybrid simulator with zero-crossing detection $\mathcal{H}^s_{ZCD}$ with $g^s_{ZCD} = g_{ZCD}$, $h^s = h$, $\mathcal{S}^s = \mathcal{S}$, and f^s_{ZCD} given by the Euler integration scheme, i.e., $f^s_{ZCD}(x,z) = [x^\top z]^\top + sf(x,z) = [x^\top z]^\top + s\left[x_2, \; -M(x_1) - N(x_1)x_2, \; 0\right]^\top$.

Figure 5(a) shows the solution to $\mathcal{H}^s_{ZCD}$ when $\gamma = 9.81$, $\lambda = 0.8$, $\delta = 0.3$, $\varepsilon = 0.15$, the initial state is $[x^\top \; z]^\top = [1, \; 0, \; 1]^\top$, and the step size is $s = 0.01$. In contrast to the simulator in Example 3.5, notice that this time the correct results are produced. Figure 5(b) shows a plot of $h(x)z$, with h as in (10)*, as a function of t. Notice that the value of $h(x)z$ switches sign at the same instant that the solution to $\mathcal{H}^s_{ZCD}$ jumps because that is the instant where the memory state z changes sign.* △



(a) Height of the bouncing ball. (b) $h(x)z$

Figure 5: Solution for Example 4.2 using $\mathcal{H}^s_{\text{ZCD}}$ framework.

Example 4.3. *(unicycle avoiding obstacle (revisited)) Consider again the unicycle model in Example 3.2. Adding zero-crossing detection to the model in* (6)*, we obtain the hybrid system*

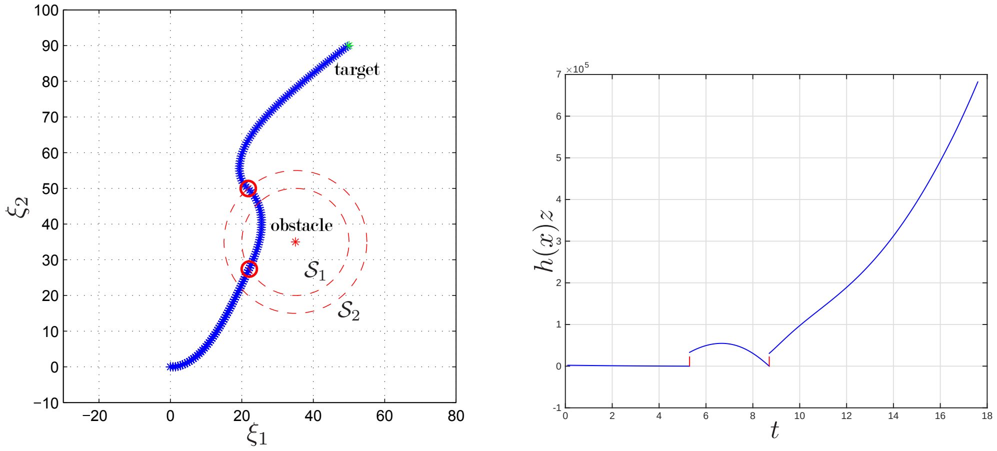
$$
\begin{aligned}
&\mathcal{X} \subset \mathbb{R}^3 \times \{1,2\}, \\
&f_{ZCD}(x,z) = \begin{bmatrix} \begin{bmatrix} v\cos(\xi_3) \\ v\sin(\xi_3) \\ \kappa(q,\xi) \end{bmatrix} \\ 0 \\ 0 \end{bmatrix}, \quad g_{ZCD}(x,z) \;=\; \begin{bmatrix} \xi \\ 3-q \\ h(g(x)) \end{bmatrix} \\
&\mathcal{S} = \cup_{q\in\{1,2\}}(\mathcal{S}_q \times \{q\}) \\
&\mathcal{S}_q = \left\{\xi \in \mathcal{X} \;:\; (\xi_1^\circ - \xi_1)^2 + (\xi_2^\circ - \xi_2)^2 = a_q^2 \right\},
\end{aligned}
$$

where g is the jump map of the hybrid system in Example 3.2. Now consider a hybrid simulator with zero-crossing detection $\mathcal{H}^s_{ZCD}$ with $g^s_{ZCD} = g_{ZCD}$, $h^s = h$, $\mathcal{S}^s = \cup_{q\in 1,2}(\mathcal{S}_q \times q)$, and f^s_{ZCD} given by the Euler integration, i.e., $f^s_{ZCD}(x,z) = x + s f_{ZCD}(x,z) = x + s\left[v\cos(\xi_3),\ v\sin(\xi_3),\ \kappa(q,\xi),\ 0,\ 0\right]^\top$.

Figure 6(a) shows a solution to $\mathcal{H}^s_{ZCD}$ with the control modes given in Example 3.6 when the initial state is $[\xi_1\ \xi_2\ \xi_3\ q\ z]^\top = [0,\ 0,\ 0,\ 1,\ 1]^\top$, and the step size is $s = 0.1$. Figure 6(b) shows a plot of $h(x)z$, with h as in (11)*, as a function of t. In contrast to the simulation in Example 3.6, the mode changes are made successfully, and the unicycle reaches the target.* △

Example 4.4. *(unicycle on a track (revisited)) Consider again the unicycle model in Example 3.3. When adding zero-crossing detection to the model in* (7)*, we obtain the hybrid system $\mathcal{H}_{ZCD}$ given by*

$$
\begin{aligned}
&\mathcal{X} \subset \mathbb{R}^3 \times \{1,2\}, \\
&f_{ZCD}(x,z) = \begin{bmatrix} \begin{bmatrix} v\cos(\xi_3) \\ v\sin(\xi_3) \\ \kappa(q,\xi) \end{bmatrix} \\ 0 \\ 0 \end{bmatrix}, \quad g_{ZCD}(x,z) \;=\; \begin{bmatrix} \xi \\ 3-q \\ h(g(x)) \end{bmatrix} \\
&\mathcal{S} = \cup_{q\in\{1,2\}}(\mathcal{S}_q \times \{q\}) \\
&\mathcal{S}_1 = \{x \in \mathcal{X} \;:\; \xi_1 = 1, q = 1 \} \\
&\mathcal{S}_2 = \{x \in \mathcal{X} \;:\; \xi_1 = -1, q = 2 \},
\end{aligned}
$$

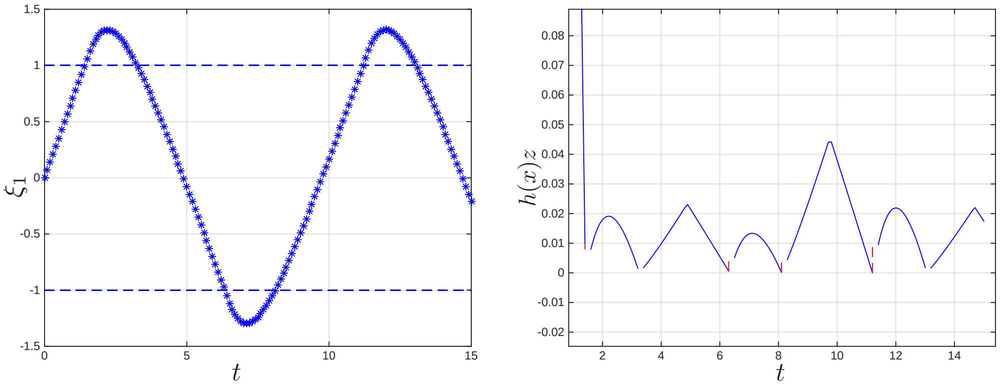


(a) Solution to unicycle avoiding obstacle example. Solution starts from $(0, 0)$. The location where jumps (mode changes) occur are labeled with red circles.

(b) $h(x)z$

Figure 6: Solution for Example 4.3 using $\mathcal{H}^s_{\mathrm{ZCD}}$ framework.

where g is the jump map of the hybrid system in Example 3.3. Now consider a hybrid simulator for this system, denoted $\mathcal{H}^s_{ZCD}$, with $g^s_{ZCD} = g_{ZCD}$, $h^s = h$, $\mathcal{S}^s = \cup_{q\in\{1,2\}}(\mathcal{S}_q \times \{q\})$, and f^s_{ZCD} given by Euler integration, i.e., $f^s_{ZCD}(x, z) = x + sf_{ZCD}(x, z) = x + s\left[v\cos(\xi_3),\ v\sin(\xi_3),\ -\xi_3 + \kappa(q, \xi),\ 0,\ 0\right]^\top$. Figure 7(a) shows a solution to $\mathcal{H}^s_{ZCD}$ with the control law $\kappa(1, \xi) = \frac{\pi}{4} - \xi_3$ and $\kappa(2, \xi) = \frac{3\pi}{4} - \xi_3$ when the initial state is $[\xi_1\ \xi_2\ \xi_3\ q\ z]^\top = [0,\ 0,\ \frac{\pi}{4},\ 1,\ 1]^\top$, and the step size is $s = 0.1$. In contrast to the simulation in Example 3.7, the mode changes are made successfully. In fact, when the unicycle encounters the boundary, the solution jumps, and, with some overshoot, the unicycle is steered back inside the track. Figure 7(b) shows a plot of $h(x)z$, with $h(x)$ as in (12), as a function of t. △



(a) Solution of unicycle on track example. Solution starts from $(\xi_1, \xi_2, \xi_3) = (0, 0, \pi/4)$.

(b) $h(x)z$

Figure 7: Solution for Example 4.4 using $\mathcal{H}^s_{\mathrm{ZCD}}$ framework.

5. Main Results

In this section, we investigate properties of $\mathcal{H}_{\text{ZCD}}$, state results addressing the previous concerns of measurement noise and robustness, and give properties of the simulation framework for hybrid systems incorporating zero-crossing detection.

5.1. Nominal Properties of $\mathcal{H}_{\text{ZCD}}$

Given $\mathcal{H}$ as in (4), we are interested in conditions on the data of a hybrid system $\mathcal{H}$ under which $\mathcal{H}_{\text{ZCD}}$ has basic regularity properties leading to robustness to perturbations. For this reason, the following mild conditions are imposed on the data of $\mathcal{H}$.

Assumption 5.1. *Given a hybrid system $\mathcal{H} = (C, f, D, g)$ as in* (4) *with associated sets $\mathcal{X}$ and $\mathcal{S}$, the following conditions hold:*

1. *$\mathcal{X}$ is closed (relative to $\mathbb{R}^n$);*
2. *$f : \mathbb{R}^n \to \mathbb{R}^n$ is continuous on $\mathcal{X}$;*
3. *$g : \mathbb{R}^n \to \mathbb{R}^n$ is continuous on $\mathcal{X}$;*
4. *There exists a continuous zero-crossing function h on $\mathcal{X}$ for $\mathcal{S}$.*

The following lemma shows that, under these assumptions, $\mathcal{H}_{\text{ZCD}}$ has regular data by construction.

Lemma 5.2. *(regularity of data of $\mathcal{H}_{\text{ZCD}}$) Suppose that a hybrid system $\mathcal{H} = (C, f, D, g)$ as in* (4) *with associated sets $\mathcal{X}$ and $\mathcal{S}$ satisfies Assumption 5.1. Then, the data of $\mathcal{H}_{\text{ZCD}}$ is such that C_{ZCD} and D_{ZCD} are closed, and f_{ZCD} and g_{ZCD} are continuous.*

Proof. For every sequence $(u_i, v_i) \in C_{\text{ZCD}}$ with $\lim_{i\to\infty}(u_i, v_i) = (u^*, v^*)$, we have, by definition of C_{ZCD}, $h(u_i)v_i \geq 0$. Using the continuity of h, $\lim_{i\to\infty} h(u_i) = h(u^*)$. Then, $0 \leq \lim_{i\to\infty} h(u_i) \lim_{i\to\infty} v_i = h(u^*)v^*$. This implies that $(u^*, v^*) \in C_{\text{ZCD}}$ and, hence, that C_{ZCD} is closed. It follows similarly that D_{ZCD} is closed. Continuity of f_{ZCD} and of g_{ZCD} follows directly from their definition and the continuity of f, h, and g. ■

When the flows of the hybrid system are transverse to the switching surface and the jump map does not map points back to the switching surface, the construction of $\mathcal{H}_{\text{ZCD}}$ is such that it captures all of the solutions to $\mathcal{H}$ (and vice versa). The following proposition states this transversality condition.

Proposition 5.3. *(properties of solutions to $\mathcal{H}_{\text{ZCD}}$) Given a hybrid system $\mathcal{H} = (C, f, D, g)$ as in* (4) *with associated sets $\mathcal{X}$ and $\mathcal{S}$, assume the following:*

1. *$f : \mathbb{R}^n \to \mathbb{R}^n$ is continuous on $\mathcal{X}$;*
2. *Every maximal solution from D to*

$$\dot{x} = f(x) \qquad x \in C \tag{14}$$

is trivial, i.e., each maximal solution ϕ to (14) *with $\phi(0,0) \in D$ is such that* $\operatorname{dom}\phi = \{(0,0)\}$.

3. *The jump map g satisfies $g(x) \notin D$ for all $x \in D$.*

Then, for every solution ϕ to $\mathcal{H}$, there exists a solution ψ to $\mathcal{H}_{\text{ZCD}}$ such that $\phi \equiv \psi_x$, where ψ_x denotes the x component of ψ. Furthermore, for every solution ψ to $\mathcal{H}_{\text{ZCD}}$ there exists a solution ϕ to $\mathcal{H}$ such that[2] *$\psi_x \equiv \phi$.*

[2] There exist solutions ψ' to $\mathcal{H}_{\text{ZCD}}$ that start from D and initially jump due to the value of z, that is, $(0,1)$ is an element of $\operatorname{dom}\psi'$. For such solutions, the equivalence $\psi_x \equiv \phi$ holds for the solution to $\mathcal{H}_{\text{ZCD}}$ defined as $\psi(t,j) = \psi'(t,j+1)$ for each $(t,j) \in \operatorname{dom}\psi'$.

PROOF. Let ϕ be a solution to $\mathcal{H}$ starting from $\overline{\mathcal{X}} \setminus \mathcal{S}$, which exists due to item 1, and let $\{t_{j+1}\}_{j=0}^{j_{\max}}$, $j_{\max} = \sup_j\{\operatorname{dom}\phi\} - 1$, be the sequence determining the jump instants (t_{j+1}, j) of ϕ. First, fix $j = 0$. By definition of solutions to $\mathcal{H}$, we have $\phi(t, 0) \in \mathcal{X} \setminus \mathcal{S}$ for all $t \in (0, t_1)$. Without loss of generality, using the definition of h, we can assume that, for all $t \in (0, t_1)$, we have $\phi(t, 0) \in \mathcal{X}_2$, i.e., $h(\phi(t, 0)) > 0$. By definition of the jump set of $\mathcal{H}$, we have $h(\phi(t_1, 0)) = 0$. By item 2, flows are not possible for $t > t_1$. Then, the function $\psi = [\psi_x^\top \; \psi_z]^\top$ defined for all $t \in [0, t_1]$ as $\psi_x(t, 0) = \phi(t, 0)$, $\psi_z(t, 0) = h(\phi(0, 0))$ is a maximal solution to

$$\left.\begin{array}{rcl} \dot{x} & = & f(x) \\ \dot{z} & = & 0 \end{array}\right\} \qquad (x, z) \in C_{\text{ZCD}}.$$

Now, let $j = 1$ and note that by the condition on g and $\mathcal{S}$ in item 3 of the assumptions we have $\phi(t_1, 1) = g(\phi(t_1, 0)) \notin D$. If $\phi(t_1, 1) \notin \mathcal{X}$ or $\phi(t_1, 1)$ is a point in the boundary of $\mathcal{X}$ from where flowing into $\mathcal{X} \setminus \mathcal{S}$ is not possible, then the augmentation of ψ by $\psi_x(t_1, 1) = \phi(t_1, 1)$, $\psi_z(t_1, 1) = h(\phi(t_1, 1))$ finishes the proof. Otherwise, there exists $t_2 > t_1$ such that $\phi(t, 1) \in \overline{\mathcal{X}} \setminus \mathcal{S}$ for all $t \in (t_1, t_2)$. The argument for $j = 0$ can be applied again (and successively) to augment ψ and construct a solution ψ. The other direction follows using a similar construction. ■

The properties in Proposition 5.3 guarantee that solutions to $\mathcal{H}_{\text{ZCD}}$ do not get repeatedly mapped to nor "slide" along (tangent to) the switching surface $\mathcal{S}$. We now check these properties for Examples 3.1, 3.2, and 3.3. Considering item 1, it is easy to see that f in each of the Examples 3.1, 3.2, and 3.3 is continuous on $\mathcal{X}$ by inspection. Satisfaction of item 2 for Examples 3.1, 3.2, and 3.3 can be confirmed by noting that f on D points outward of C (see also Figures 5, 6, and 7).

To check item 3, we first compute $g(D)$ and then show that $g(D) \cap D = \emptyset$ for each example. For Example 3.1, $g(D) = \left\{x \in \mathbb{R}^2 \;:\; x_1 = 0, x_2 \geq \lambda\delta \right\}$. Therefore, $g(D) \cap D = \emptyset$ because $\lambda\delta > -\delta$, so item 3 is satisfied for Example 3.1. For Example 3.2,

$$\begin{aligned} g(D) = \left\{(\xi, q) \in \mathbb{R}^3 \times \{1, 2\} \;:\; (\xi_1^o - \xi_1)^2 + (\xi_2^o - \xi_2)^2 = a_1,\; q = 2 \right\} \cup \\ \left\{(\xi, q) \in \mathbb{R}^3 \times \{1, 2\} \;:\; (\xi_1^o - \xi_1)^2 + (\xi_2^o - \xi_2)^2 = a_2,\; q = 1 \right\}. \end{aligned}$$

The value of q always switches thereby ensuring that $g(D) \cap D = \emptyset$, so item 3 is satisfied for Example 3.2. For Example 3.3,

$$g(D) = \left\{(\xi, q) \in \mathbb{R}^3 \times \{1, 2\} \;:\; \xi_1 = 1,\; q = 2 \right\} \cup \left\{(\xi, q) \in \mathbb{R}^3 \times \{1, 2\} \;:\; \xi_1 = -1,\; q = 1 \right\}.$$

Again, $g(D) \cap D = \emptyset$ because of the same reasoning as that for Example 3.2. Therefore, items 1, 2, and 3 in Proposition 5.3 are satisfied in Examples 3.1, 3.2, and 3.3.

5.2. *Robustness to measurement noise of $\mathcal{H}_{\text{ZCD}}$*

Next we show that $\mathcal{H}_{\text{ZCD}}$ is robust to measurement noise. This is proven by embedding the hybrid system $\mathcal{H}_{\text{ZCD}}$ with measurement noise e, denoted $\mathcal{H}_{\text{ZCD}}^e$, into an inflated version of $\mathcal{H}_{\text{ZCD}}$. More precisely, given $\varepsilon_1 > 0$, $\mathcal{H}_{\text{ZCD}}$ is inflated via an outer perturbation giving the perturbed hybrid system

$$\mathcal{H}_{\text{ZCD}}^{\varepsilon_1} : \quad \left\{ \begin{array}{rcll} \dot{x} & \in & \overline{\text{co}} f_{\text{ZCD}}(x + \varepsilon_1\mathbb{B}) & \quad x \in C_{\text{ZCD}}^{\varepsilon_1} \\ x^+ & \in & g_{\text{ZCD}}(x + \varepsilon_1\mathbb{B}) & \quad x \in D_{\text{ZCD}}^{\varepsilon_1}, \end{array} \right. \tag{15}$$

where $\overline{\text{co}}$ denotes the closed convex hull operation, and

$$\begin{array}{rcl} C_{\text{ZCD}}^{\varepsilon_1} & := & \{x \in \mathbb{R}^n \;:\; (x + \varepsilon_1\mathbb{B}) \cap C_{\text{ZCD}} \neq \emptyset \} \\ D_{\text{ZCD}}^{\varepsilon_1} & := & \{x \in \mathbb{R}^n \;:\; (x + \varepsilon_1\mathbb{B}) \cap D_{\text{ZCD}} \neq \emptyset \}. \end{array} \tag{16}$$

This perturbed hybrid system is such that it captures all of the solutions to $\mathcal{H}_{\text{ZCD}}^e$ with measurement noise $e : \operatorname{dom} e \to \varepsilon_1\mathbb{B}$. Under the conditions in Assumption 5.1, it follows that every solution to $\mathcal{H}_{\text{ZCD}}^{\varepsilon_1}$ is close (in an appropriate sense and on compact hybrid time domains) to an unperturbed solution to $\mathcal{H}_{\text{ZCD}}$. Then, the equivalence result in Proposition 5.3 permits relating these solutions to those of $\mathcal{H}$.

Before presenting the robustness result, we introduce a notion of closeness of hybrid arcs from [19]. The same property can be defined for two discrete arcs as well as for a hybrid arc and a discrete arc; see [20] for more details.

Definition 5.4. *((T, J, μ)-closeness) Given $T, J \geq 0$ and $\mu > 0$, two hybrid arcs $x_1 : \operatorname{dom} x_1 \to \mathbb{R}^n$ and $x_2 : \operatorname{dom} x_2 \to \mathbb{R}^n$ are (T, J, μ)-close if*

(a) for all $(t, j) \in \operatorname{dom} x_1$ with $t \leq T, j \leq J$ there exists s such that $(s, j) \in \operatorname{dom} x_2$, $|t - s| < \mu$, and

$$|x_1(t, j) - x_2(s, j)| < \mu,$$

(b) for all $(t, j) \in \operatorname{dom} x_2$ with $t \leq T, j \leq J$ there exists s such that $(s, j) \in \operatorname{dom} x_1$, $|t - s| < \mu$, and

$$|x_2(t, j) - x_1(s, j)| < \mu.$$

Theorem 5.5. *(robustness of $\mathcal{H}_{ZCD}$ to measurement noise) Suppose that a hybrid system $\mathcal{H} = (C, f, D, g)$ as in (4) with associated sets $\mathcal{X}$ and $\mathcal{S}$ satisfy Assumption 5.1 and items 2 and 3 of Proposition 5.3. Let $K \subset \mathbb{R}^n$ be a compact set such that every maximal solution ϕ to $\mathcal{H}$ with $\phi(0, 0) \in K$ is either bounded or complete. Then, for every $\mu > 0$ and $(T, J) \in \mathbb{R}_{\geq 0} \times \mathbb{N}$ there exists $\varepsilon^* > 0$ such that, for every measurable signal $e : \operatorname{dom} e \to \varepsilon\mathbb{B}$, $0 < \varepsilon \leq \varepsilon^*$, every solution ψ^e to $\mathcal{H}^e_{ZCD}$ with*[3] *$\psi^e_x(0, 0) \in K + \varepsilon\mathbb{B}$, $\psi^e_z(0, 0) = h(\psi^e_x(0, 0))$, is such that there exists a solution ϕ to $\mathcal{H}$ with $\phi(0, 0) \in K$ such that ψ^e_x and ϕ are (T, J, μ) close.*

Proof. Using Assumption 5.1, by Lemma 5.2, $\mathcal{H}_{\text{ZCD}}$ is such that C_{ZCD} and D_{ZCD} are closed, and f_{ZCD} and g_{ZCD} are continuous. Let $e : \operatorname{dom} e \to \varepsilon_1\mathbb{B}$, with $\varepsilon_1 > 0$ to be specified later, and consider the inflation of $\mathcal{H}_{\text{ZCD}}$ given by (15)-(16). By construction, each solution to $\mathcal{H}^e_{\text{ZCD}}$ is a solution to $\mathcal{H}^{\varepsilon_1}_{\text{ZCD}}$. Using [20, Theorem 3.4], we have that, for every $\mu_1 > 0$ and given (T, J), there exists ε_1^* such that, for each $\varepsilon_1 \in (0, \varepsilon_1^*]$, for each solution ψ^{ε_1} to $\mathcal{H}^{\varepsilon_1}_{\text{ZCD}}$ with $\psi^{\varepsilon_1}_x(0, 0) \in K + \varepsilon_1\mathbb{B}$ (in particular, those that are solutions to $\mathcal{H}^e_{\text{ZCD}}$) there exists a solution ψ to $\mathcal{H}_{\text{ZCD}}$ with $\psi_x(0, 0) \in K$ that is (T, J, μ_1)-close to ψ^{ε_1} (and, in particular, since each solution to $\mathcal{H}^e_{\text{ZCD}}$ is a solution to $\mathcal{H}^{\varepsilon_1}_{\text{ZCD}}$, to the associated solution ψ^e). Using Assumption 5.1 and items 2 and 3 of Proposition 5.3, the claim follows with $\varepsilon^* = \varepsilon_1^*$ and $\mu = \mu_1$ using the second result in Proposition 5.3, which implies that there exists a solution ϕ to $\mathcal{H}$ with the property $\psi^e_x \equiv \phi$. ■

5.3. Properties of $\mathcal{H}^s_{ZCD}$

As pointed out in Section 3.2, a hybrid simulator for $\mathcal{H}$ that simply discretizes its dynamics may not be capable of reproducing the jumps of the solutions to $\mathcal{H}$; see, e.g., [15, 16]. As a consequence, solutions to $\mathcal{H}$ cannot be reproduced by $\mathcal{H}^s$ with arbitrary precision. In this section, we present conditions on the data of $\mathcal{H}^s$ that guarantee that when zero-crossing detection is incorporated, which results in the hybrid simulator $\mathcal{H}^s_{\text{ZCD}}$, solutions to $\mathcal{H}$ can be reproduced with arbitrary precision. To this end, the following conditions on the data of $\mathcal{H}^s$ are imposed.

Assumption 5.6. *The data f^s and g^s of the hybrid simulator $\mathcal{H}^s = (C^s, f^s, D^s, g^s)$ for the hybrid system $\mathcal{H} = (C, f, D, g)$ and the associated zero-crossing function h^s satisfy the following:*

(B0) f^s is such that, for each compact set $K \subset \mathbb{R}^n$, there exists $\rho \in \mathcal{K}_\infty$ and $s^* > 0$ such that for each $x \in C^s \cap K$ and each $s \in (0, s^*]$,

$$f^s(x) \in x + s \operatorname{co} f(x + \rho(s)\mathbb{B}) + s\rho(s)\mathbb{B};$$

[3] ψ^e_x denotes the x component of ψ and ψ^e_z the z component.

(B1) g^s is such that for any positive sequence $\{s_i\}_{i=1}^{\infty}$, $s_i \searrow 0$,

$$\lim_{s_i \searrow 0} g^{s_i}(x) = g(x) \qquad \forall x \in \mathbb{R}^n;$$

(B2) h^s is such that for any positive sequence $\{s_i\}_{i=1}^{\infty}$, $s_i \searrow 0$,

$$\lim_{s_i \searrow 0} h^{s_i}(x) = h(x) \qquad \forall x \in \mathbb{R}^n.$$

Integration schemes such as Euler, as in Examples 3.5, 3.6, and 3.7, and Runge-Kutta satisfy condition (B0). Conditions (B1) and (B2) hold when the perturbed functions are continuous in the step size; see also [20, Examples 4.8 and 4.9].

When the data of the simulator $\mathcal{H}^s$ and zero-crossing function associated with $\mathcal{H}$ satisfy Assumption 5.6, the data of $\mathcal{H}^s_{\text{ZCD}}$ have regularity properties guaranteeing closeness between the solutions to $\mathcal{H}$ and its simulations obtained via $\mathcal{H}^s_{\text{ZCD}}$. The next lemma formalizes this fact.

Lemma 5.7. *(regularity of data of $\mathcal{H}^s_{ZCD}$) Assume $\mathcal{H}^s = (C^s, f^s, D^s, g^s)$ and h^s are such that Assumption 5.6 hold. Then, $\mathcal{H}^s_{ZCD} = (C^s_{ZCD}, f^s_{ZCD}, D^s_{ZCD}, g^s_{ZCD})$ is such that f^s_{ZCD} and g^s_{ZCD} satisfy (B0) and (B1), respectively, in Assumption 5.6, and C^s and D^s are such that*

(B3) for any positive sequence $\{s_i\}_{i=1}^{\infty}$ such that $s_i \searrow 0$, $\limsup_{i\to\infty} C^{s_i} \subset C$, $\limsup_{i\to\infty} D^{s_i} \subset D$, where $\limsup_{i\to\infty} C^{s_i}$, $\limsup_{i\to\infty} D^{s_i}$ are the outer limits of the sequence of sets C^{s_i}, D^{s_i}, respectively.

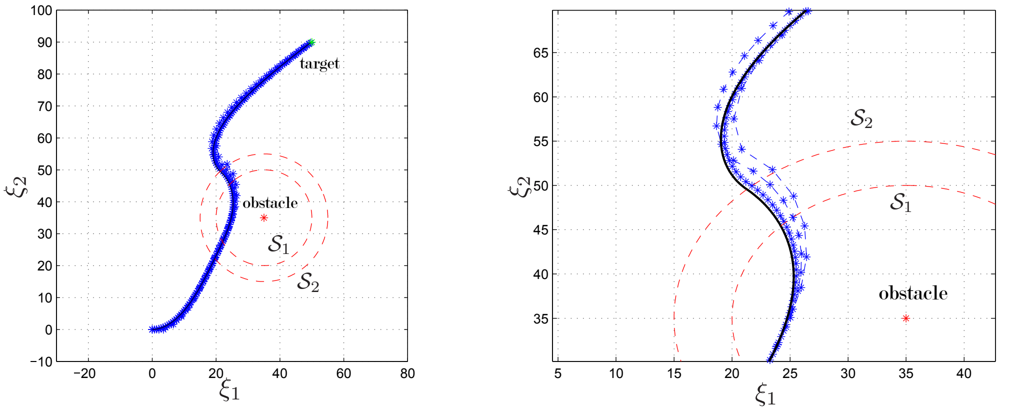
Proof. Since f^s and g^s satisfy (B0) and (B1), by construction, the functions f^s_{ZCD} and g^s_{ZCD} also satisfy (B0) and (B1). Let $s_i \searrow 0$ be a positive sequence and take $(x', z') \in C^{s_i} = \{(x, z) \ : \ h^{s_i}(x)z \geq 0 \ \}$. Since h^s satisfies (B2), we have that there exists $\delta_i > 0$, $\delta_i \searrow 0$, and $N > 0$ such that $h^{s_i}(x') \in h(x') + \delta_i\mathbb{B}$ for all $i > N$. It follows that $C^{s_i} \subset C \cup \delta_i\mathbb{B}$ for all $i > N$. Then, using [24, Proposition 5.12], C^{s_i} satisfies (B3). Similar arguments show that D^{s_i} satisfies (B3). ■

As pointed out in Section 3.2, solutions to $\mathcal{H}$ cannot be approximated by $\mathcal{H}^s$ with arbitrary precision. However, on finite simulation horizons (T, J), the solutions to hybrid simulators $\mathcal{H}^s_{\text{ZCD}}$, with data satisfying (B0)-(B3), can approximate the solutions to $\mathcal{H}$ with arbitrary precision, and this key relationship between the solutions to $\mathcal{H}$ and its simulations via $\mathcal{H}^s_{\text{ZCD}}$ is stated in the following result.

Theorem 5.8. *(closeness between solutions to $\mathcal{H}$ and $\mathcal{H}^s_{ZCD}$) Suppose that a hybrid system $\mathcal{H} = (C, f, D, g)$ as in (4) with associated sets $\mathcal{X}$ and $\mathcal{S}$ satisfies Assumption 5.1 and items 2 and 3 of Proposition 5.3. Furthermore, suppose $\mathcal{H}^s_{ZCD}$ satisfies Assumption 5.6. Then, for every compact set $K \subset \mathbb{R}^n$, $\mu > 0$, and simulation horizon $(T, J) \in \mathbb{R}_{\geq 0} \times \mathbb{N}$ there exists $s^* > 0$ with the following property: for any $s \in (0, s^*]$ and any solution $\psi^s_x(0,0) \in K$ there exists a solution ϕ to $\mathcal{H}$ with $\phi(0,0) \in K$ such that ψ^s_x and ϕ are (T, J, μ)-close.*

Proof. Using Assumption 5.1, by Lemma 5.2, $\mathcal{H}_{\text{ZCD}}$ is such that C_{ZCD} and D_{ZCD} are closed, and f_{ZCD} and g_{ZCD} are continuous. Using Assumption 5.6, by Lemma 5.7, $\mathcal{H}^s_{\text{ZCD}}$ is such that f^s_{ZCD} and g^s_{ZCD} satisfy (B0) and (B1) in Assumption 5.6, and C^s and D^s satisfy (B3). Using [20, Theorem 5.2], it follows that for every K, $\mu > 0$, and (T, J) as in the statement, there exists $s^* > 0$ with the following property: for every $s \in (0, s^*]$ and every solution ψ^s to $\mathcal{H}^s_{\text{ZCD}}$ with $\psi^s_x(0,0) \in K$ there exists a solution ψ to $\mathcal{H}_{\text{ZCD}}$ with $\psi_x(0,0) \in K$ such that ψ^s and ψ are (T, J, μ)-close. The proof of the claim is complete after an application of Proposition 5.3, which permits obtaining a solution ϕ to $\mathcal{H}$ that is equivalent to ψ_x. ■

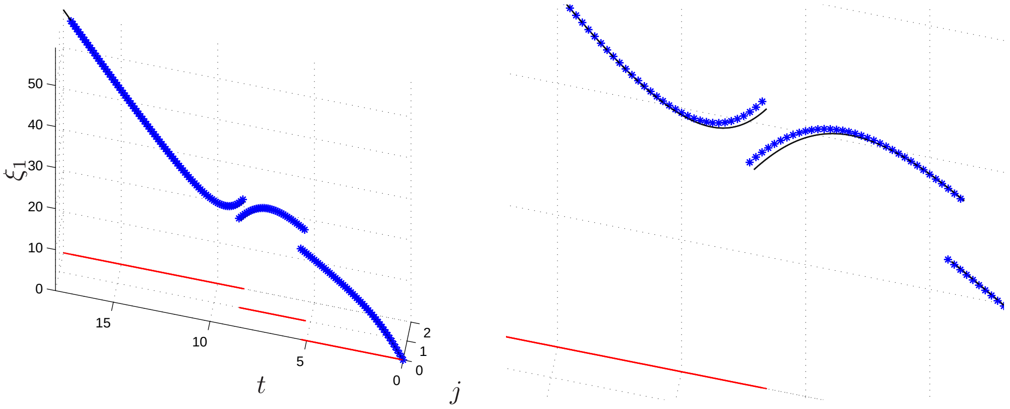
Example 5.9. *(unicycle avoiding obstacle (revisited)) Consider the unicycle model with zero-crossing detection in Example 4.3. The results of Theorem 5.8 are illustrated by plotting the solutions of $\mathcal{H}^s_{ZCD}$ from Example 4.3 for different step sizes. Figure 8 shows the exact solution (in black) and simulated solutions (in blue) in the same plot. Figure 8(b) shows a zoomed in version of Figure 8(a). Similarly, Figure 9 shows the hybrid arc solution and closeness of the simulated solution to the exact solution in a finite simulation horizon (T, J). Figure 9(b) shows a zoomed in version of Figure 9(a). Notice that the simulated solutions converge to the exact solution as the step size is decreased.* △



(a) Exact solution (in black), and simulated solutions with decreasing step size (in blue *).

(b) Zoomed in version showing closeness. Exact solution shown as solid black line.

Figure 8: Closeness between exact solution and simulated solutions of Example 3.2. Solutions start from $(0, 0)$.



(a) Discrete arc (blue *), exact hybrid arc solution (black), and exact hybrid time domain (red) are shown.

(b) Zoomed in version showing closeness.

Figure 9: Closeness between exact hybrid arc solution and simulated solutions of Example 3.2.

5.4. Application to hybrid systems $\mathcal{H}$ with asymptotically stable compact sets

In this section, we consider the setting when a compact set is asymptotically stable for the hybrid system $\mathcal{H}$ in (4). This is the case when there exists a compact set $\mathcal{A} \subset \mathbb{R}^n$ with the following properties:

- *stable* if for each $\varepsilon > 0$ there exists $\delta > 0$ such that each solution ϕ to $\mathcal{H}$ with $|\phi(0,0)|_{\mathcal{A}} \leq \delta$ satisfies $|\phi(t,j)|_{\mathcal{A}} \leq \varepsilon$ for all $(t,j) \in \operatorname{dom}\phi$;
- *attractive* if there exists $\mu > 0$ such that every solution ϕ to $\mathcal{H}$ with $|\phi(0,0)|_{\mathcal{A}} \leq \mu$ is bounded and if it is complete satisfies $\lim_{(t,j)\in\operatorname{dom}\phi, t+j\to\infty} |\phi(t,j)|_{\mathcal{A}} = 0$;
- *asymptotically stable* if stable and attractive.

When the attractivity property holds for every solution starting from $\overline{C} \cup D$, a stable compact set $\mathcal{A}$ is said to be globally asymptotically stable.

Whether or not a hybrid simulator preserves the asymptotic stability properties of $\mathcal{H}$ depends on the effect of perturbations. Therefore, in light of the lack of robustness of $\mathcal{H}$ to measurement noise pointed out in Proposition 3.4, it is not expected for $\mathcal{H}^s$ to preserve asymptotic stability because numerical integration can be interpreted similarly as a perturbation. In spite of this, the simulator with zero-crossing detection given by $\mathcal{H}^s_{\text{ZCD}}$, when designed with regular data, does, in fact, preserve stability as stated in the following result.

Theorem 5.10. *(semiglobal practical stability) Suppose that a hybrid system $\mathcal{H} = (C, f, D, g)$ as in* (4) *with associated sets $\mathcal{X}$ and $\mathcal{S}$ satisfies Assumption 5.1 and items 2 and 3 of Proposition 5.3. Furthermore, suppose that $\mathcal{A}$ is a globally asymptotically stable compact set for $\mathcal{H}$ and that $\mathcal{H}^s_{ZCD}$ satisfies Assumption 5.6. Then, $\mathcal{A}$ is semiglobally practically asymptotically stable for $\mathcal{H}^s_{ZCD}$, i.e., there exists $\beta \in \mathcal{KL}$ such that, for every compact set $K \subset \mathbb{R}^n$, $\varepsilon > 0$, and simulation horizon $(T, J) \in \mathbb{R}_{\geq 0} \times \mathbb{N}$ there exists $s^* > 0$ such that, for each $s \in (0, s^*]$, every solution ϕ^s to $\mathcal{H}^s_{ZCD}$ with $\phi^s(0,0) \in K$ satisfies for all $(k,j) \in \operatorname{dom}\phi^s$*

$$|\phi^s(k,j)|_{\mathcal{A}} \leq \beta(|\phi^s(0,0)|_{\mathcal{A}}, ks + j) + \varepsilon.$$

Proof. By the equivalence of solutions in Proposition 5.3, $\mathcal{A}$ is globally asymptotically stable for $\mathcal{H}_{\text{ZCD}}$. Using the assumptions, Lemma 5.2 and Lemma 5.7 imply that $\mathcal{H}_{\text{ZCD}}$ and $\mathcal{H}^s_{\text{ZCD}}$ have regular data. It follows that the conditions in the semiglobal practical asymptotic stability result for general simulators in [20, Theorem 5.3] hold for these systems, which establishes the claim. ■

The semiglobal practical asymptotic stability property established in the result above holds for sufficiently small step size. The bound s^* on the step size s decreases with ε, which defines the level of closeness to $\mathcal{A}$ that solutions to $\mathcal{H}^s$ should arrive at.

Our final result follows directly from [20, Theorem 5.4], Proposition 5.3, Lemma 5.2, and Lemma 5.7. It establishes that the semiglobally asymptotically stable set for $\mathcal{H}^s_{\text{ZCD}}$, denoted $\mathcal{A}_s$, converges to $\mathcal{A}$ as the step size s becomes smaller. In other words, the set $\mathcal{A}_s$ depends continuously on the step size.

Theorem 5.11. *(continuity of asymptotically stable sets) Suppose the assumptions of Theorem 5.10 hold. Then, there exists $s^* > 0$ such that for each $s \in (0, s^*]$, the hybrid simulator $\mathcal{H}^s_{ZCD}$ has a semiglobally asymptotically stable compact set $\mathcal{A}_s$ satisfying $d_H(\mathcal{A}_s, \mathcal{A}) \to 0$ as $s \searrow 0$.*

Example 5.12. *(bouncing ball (revisited)) Recall Example 3.5 where the solutions to a bouncing ball example pass through the surface on which it is supposed to bounce. We know that the origin, i.e., the compact set $\mathcal{A} = \{(0,0)\}$, is globally asymptotically stable for $\mathcal{H}_{ZCD}$, but when solutions pass through $\mathcal{S}$, this stability property no longer holds. This means that in Example 3.5, the origin is not globally asymptotically stable for $\mathcal{H}^s$ because every solution does not converge to the origin. This can be seen in Figure 1(a). However, if we instead simulate the bouncing ball example with the $\mathcal{H}^s_{ZCD}$ framework, the stability property holds. This property is highlighted in Figure 10 which shows the tail of a simulated solution to $\mathcal{H}^s_{ZCD}$ as given in Example 4.2 with $\gamma = 9.81$, $\lambda = 0.8$, $\delta = 0.3$, $\varepsilon = 0.15$, the initial state $[x^\top\ z]^\top = [1,\ 0,\ 1]^\top$, and step size $s = 0.001$. As a result of the non-Zeno bouncing ball model, the tail of this simulated solution shows the height of the ball oscillating between positive and negative values. In this way, the solution is asymptotically converging to the compact set $\mathcal{A} = \{(0,0)\}$, which follows from Theorem 5.10.* △

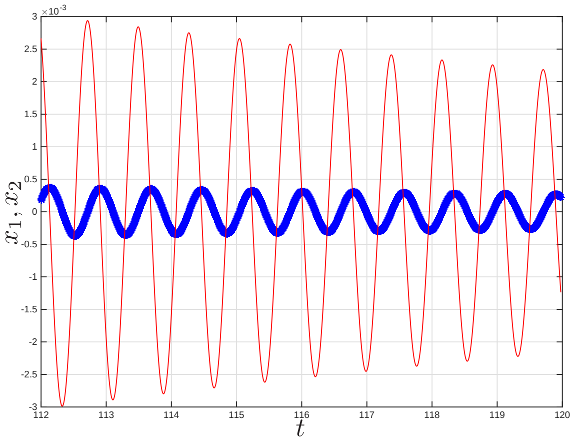


Figure 10: Tail of a solution to the bouncing ball in Example 4.2. Height (x_1) is denoted in blue, and velocity (x_2) is denoted in red. Following Theorem 5.10, the solution is asymptotically converging to the compact set $\mathcal{A} = \{(0, 0)\}$.

6. Conclusion

A mathematical framework for hybrid systems incorporating zero-crossing detection algorithms was introduced. Unlike previous work in the literature, the proposed framework allows for analytical study of the effects of zero-crossing detection algorithms in the simulation of hybrid systems. Adverse effects of perturbations in hybrid systems jumping on surfaces was highlighted, and a hybrid model and simulator incorporating zero-crossing detection were proposed. We established that the resulting system with zero-crossing detection is robust to measurement noise and to discretization effects in numerical simulation. Our results suggest that integration schemes with zero-crossing detection algorithms actually compute the solutions of a robustified version of the fragile nominal model. These results are assured when the simulator is implemented numerically using standard available solvers, so this approach is suitable for simulation of complex systems, e.g. hybrid and cyber-physical systems. These results were illustrated in several numerical examples throughout the paper.

7. Acknowledgements

This research has been partially supported by the National Science Foundation under CAREER Grant no. ECS-1150306 and by the Air Force Office of Scientific Research under Grant no. FA9550-12-1-0366.

References

[1] D. D. Bainov and P. S. Simeonov. *Systems with Impulse Effect: Stability, Theory, and Applications*. Ellis Horwood Limited, 1989.

[2] M. Boccadoro, Y. Wardi, M. Egerstedt, and E. Verriest. Optimal control of switching surfaces in hybrid dynamical systems. *Discrete Event Dynamic Systems-Theory and Applications*, 15(4):433–448, 2005.

[3] B. Morris and J. W. Grizzle. Hybrid invariant manifolds in systems with impulse effects with application to periodic locomotion in bipedal robots. *IEEE Trans. Aut. Control*, 54(8):1751–1764, 2009.

[4] J. C. Clegg. A nonlinear integrator for servomechanisms. *Transactions A.I.E.E.*, 77(Part II):41–42, 1958.

[5] K. R. Krishnan and I. M. Horowitz. Synthesis of a non-linear feedback system with significant plant-ignorance for prescribed system tolerances. *International Journal of Control*, 19:689–706, 1974.

[6] O. Beker, C. V. Hollot, Y. Chait, and H. Han. Fundamental properties of reset control systems. *Automatica*, 40(6):905–915, 2004.

[7] L. Zaccarian, D. Nesic, and A. R. Teel. First order reset elements and the clegg integrator revisited. *Proc. 24th American Control Conference*, pages 563–568, 2005.

[8] V. Chellaboina, S. P. Bhat, and W. H. Haddad. An invariance principle for nonlinear hybrid and impulsive dynamical systems. *Nonlinear Analysis, Theory, Methods and Applications*, 53:527–550, 2003.

[9] R. C. Arkin. *Behavior Based Robotics*. The MIT Press, 1998.

[10] M. Egerstedt. Behavior based robotics using hybrid automata. *Hybrid Systems: Computation and Control Lecture Notes in Computer Science*, 1790:103–116, 2000.

[11] V. Donde and I. A. Hiskens. Shooting methods for locating grazing phenomena in hybrid systems. *International Journal of Bifurcation and Chaos*, 16(03):671–692, 2006.

[12] R. G. Sanfelice, R. Goebel, and A. R. Teel. Generalized solutions to hybrid dynamical systems. *ESAIM: Control, Optimisation and Calculus of Variations*, 14(4):699–724, 2008.

[13] The Mathworks, Inc. Hit Crossing. http://www.mathworks.com/help/simulink/slref/hitcrossing.html, 1984-2015.

[14] L. Shampine, I. Gladwell, and R. Brankin. Reliable solution of special event location problems for ODEs. *ACM Transactions on Mathematical Software*, 17(1):11–25, 1991.

[15] J. M. Esposito, V. Kumar, and G. J. Pappas. Accurate event detection for simulating hybrid systems. In *Hybrid Systems: Computation and Control: 4th International Workshop*, pages 204–217, 2001.

[16] F. Zhang, M. Yeddanapudi, and P. J. Mosterman. Zero-crossing location and detection algorithms for hybrid system simulation. In *17th IFAC World Congress*, pages 7967–7972, 2008.

[17] S. Burden, H. Gonzalez, R. Vasudevan, R. Bajcsy, and S. S. Sastry. Numerical integration of hybrid dynamical systems via domain relaxation. In *50th IEEE Conference on Decision and Control*, pages 3958–3965, December 2011.

[18] D. A. Copp and R. G. Sanfelice. On the effect and robustness of zero-crossing detection algorithms in simulation of hybrid systems jumping on surfaces. In *Proc. 31st American Control Conference*, pages 2449–2454, 2012.

[19] R. Goebel, R. G. Sanfelice, and A. R. Teel. Hybrid dynamical systems. *IEEE Control Systems Magazine*, pages 28–93, 2009.

[20] R. G. Sanfelice and A. R. Teel. Dynamical properties of hybrid systems simulators. *Automatica*, 46(2):239–248, 2010.

[21] R. G. Sanfelice and A. R. Teel. A nested matrosov theorem for hybrid systems. In *Proc. 27th American Control Conference*, pages 2915–2920, 2008.

[22] R. G. Sanfelice, D. A. Copp, and P. Ñañez. A toolbox for simulation of hybrid systems in Matlab/Simulink: Hybrid equations (HyEQ) Toolbox. In *16th International Conference on Hybrid Systems: Computation and Control*, pages 101–106, 2013.

[23] R. I. Leine and H. Nijmeijer. *Dynamics and Bifurcations of Non-Smooth Mechanical Systems*. Springer, 2004.

[24] R. T. Rockafellar and R. J-B Wets. *Variational Analysis*. Springer, 1998.