International Conference on Embedded Software (EMSOFT), 2013
http://www.cds.caltech.edu/~murray/papers/plm13-emsoft.html

Pre-orders for Reasoning about Stability Properties with respect to Input of Hybrid Systems

Pavithra Prabhakar
IMDEA Software Institute
Madrid, Spain
pavithra.prabhakar@imdea.org

Jun Liu
University of Sheffield
Sheffield, UK
j.liu@sheffield.ac.uk

Richard M. Murray
California Inst. of Tech.
Pasadena, CA, USA
murray@caltech.edu

ABSTRACT

Pre-orders on systems are the basis for abstraction based verification of systems. In this paper, we investigate pre-orders for reasoning about stability with respect to inputs of hybrid systems. First, we present a superposition type theorem which gives a characterization of the classical incremental input-to-state stability of continuous systems in terms of the traditional ϵ-δ definition of stability. We use this as the basis for defining a notion of incremental input-to-state stability of hybrid systems. Next, we present a pre-order on hybrid systems which preserves incremental input-to-state stability, by extending the classical definitions of bisimulation relations on systems with input, with uniform continuity constraints. We show that the uniform continuity is a necessary requirement by exhibiting counter-examples to show that weaker notions of input bisimulation with just continuity requirements do not suffice to preserve stability. Finally, we demonstrate that the definitions are useful, by exhibiting concrete abstraction functions which satisfy the definitions of pre-orders.

Categories and Subject Descriptors

D.2.4 [**Software Engineering**]: Software/Program Verification—*Formal Methods*

General Terms

Design, Verification

Keywords

Stability, (Bi)-Simulations, Abstraction, Verification

1. INTRODUCTION

The ubiquitous use of embedded processors to control safety critical systems such as aeronautics, automotive and medical devices, has pressurized the need for scalable methods for reliable development of embedded control systems. A unique feature of such systems is the interaction between the discrete components - a network of embedded processors, and the continuous components - the physical system that the processors control. Such systems with mixed discrete continuous behaviors have been popularly termed *hybrid systems*.

In this paper, we develop the foundations for scalable verification of a robustness property of hybrid systems, namely, stability with respect to inputs. Intuitively, we expect small perturbations in the system input to lead to only small changes in the behavior of the system. The small perturbations in input capture disturbances such as quantization errors in actuators and sensors, which are often ignored during control design, but are, nevertheless, present in any digital implementation of the control law. Hence, stability of systems with respect to input is an important property required of any control design.

Input-to-state stability [22, 21] is the classical notion of stability with respect to inputs for purely continuous systems. Incremental input-to-state stability [2] generalizes input-to-state stability with respect to equilibrium points to that with respect to reference trajectories. The notion of input-to-state stability has been defined for hybrid systems and has been well-investigated [4, 18, 13]. More recently, input-to-output stability notions have been explored for discrete systems [23]. However, the notion of incremental input-to-state stability has not been investigated in the literature for hybrid systems. We present a characterization of the classical definition of incremental input-to-state stability for continuous systems as a super-position theorem "separating" the stability with respect to states and with respect to inputs. The characterization gives an ϵ-δ definition similar to that for Lyapunov and asymptotic stability. We generalize this definition to the setting of hybrid systems.

Current methods for verifying stability are deductive. They rely on exhibiting a certificate of stability in the form of a Lyapunov function (see, for example, [16])- a continuously differentiable function which is positive definite and whose value decreases along any trajectory of the system. Automation of the search for a Lyapunov function essentially relies on searching for the coefficients of a template, say, a polynomial, such that the requirements of the Lyapunov function are satisfied [19]. However, there is little support for systematically iterating over templates so as to prune the search space.

The work in the paper is motivated by an alternate approach, namely, algorithmic verification [6]. It is a completely automatic approach in which an exhaustive state space exploration is performed to deduce the property. However, it suffers from the state space explosion problem, and the crucial element in achieving algorithmic verification methods is to devise efficient abstraction mechanisms [1, 24]. An abstraction reduces the state space of the system such that the satisfaction of the property by the original system can be deduced by the property about the reduced system. However, if the reduced system does not satisfy the property, one cannot infer useful information about the original system. Hence, abstraction techniques are often accompanied by an abstraction refinement loop [7, 1], where a refinement is an abstraction of the system which is more precise than the current abstraction.

The basis for developing an abstraction refinement technique lies in understanding the relations between systems (the concrete and the abstract) which preserve the properties of interest. In this paper, we investigate such relations which preserve stability properties of hybrid systems with respect to input. More precisely, we define a pre-order - a reflexive, transitive relation - on the class of hybrid systems, such that if a system is stable with respect to inputs, then all system below it in the order are stable as well.

Simulations and bisimulations [17] are the canonical notions of pre-order and equivalence on systems with respect to several discrete-time properties such as LTL, CTL and μ-calculus. Alternate weaker notions such as approximate simulations and bisimulations have been investigated to achieve state-space reduction [10, 11]. It has been shown in [8, 20] that simulations and bisimulations do not preserve Lyapunov and asymptotic stability and hence, additional constraints such as continuity and uniform continuity are imposed on the simulation and bisimulation relations to force stability preservation. In this paper, we investigate incremental input-to-state stability. We show that the classical definitions of input simulations and bisimulations do not preserve incremental input-to-state stability, even with continuity constraints imposed on the input and state spaces. Hence, uniform continuity constraints are added to the relations corresponding to both the state space and the input space. Finally, we demonstrate that the definitions are reasonable, by exhibiting concrete abstraction functions which satisfy the definition of the pre-order. To this end, we cast the well-known Lyapunov function based analysis as a concrete abstraction function which reduces a system to a simple one-dimensional system, for which incremental input-to-state stability can be easily inferred. The future work will focus on developing new abstraction-refinement mechanism.

To summarize, the main contributions of the paper are:

1. A superposition type theorem for incremental input-to-state stability of continuous dynamical systems.

2. Using the alternate characterization of incremental input-to-state stability of continuous dynamical systems given by the superposition theorem to define a notion of incremental input-to-state stability for hybrid systems.

3. A definition of pre-order on the class of hybrid systems which respects input-to-state stability and thus defines the basis for an abstraction refinement framework.

4. Examples demonstrating concrete abstraction functions which fall under the definition of the pre-order.

Organization of the paper.

In Section 2, we define the necessary definitions. In Section 3, we define hybrid input transition systems and related concepts. In Section 4, we present a superposition type characterization of the notion of incremental input-to-state stability of continuous dynamical systems, which we use in Section 5 to present a definition of incremental input-to-state stability for hybrid systems. In Section 6, we define the pre-orders on hybrid systems, namely, uniformly continuous input simulations and bisimulations, and show the inadequacy of weaker notions for incremental input-to-state stability preservation. In Section 7, we show that the new pre-order introduced in Section 6 preserves incremental input-to-state stability. Finally, in Section 8, we present concrete abstraction techniques based on Lyapunov function based incremental input-to-state stability analysis, and conclude with Section 9.

2. PRELIMINARIES

Notation.

Let $\mathbb{R}$ and $\mathbb{R}^+$ denote the set of reals and non-negative reals, respectively. Let $\mathbb{R}_\infty$ denote the set $\mathbb{R}^+ \cup \{\infty\}$, where ∞ denotes the largest element of $\mathbb{R}_\infty$, that is, $x < \infty$ for all $x \in \mathbb{R}^+$. Also, for all $x \in \mathbb{R}_\infty$, $x + \infty = \infty$. Let $\mathbb{N}$ denote the set of all natural numbers $\{0, 1, 2, \cdots\}$, and let $[n]$ denote the first n natural numbers, that is, $[n] = \{0, 1, 2, \cdots, n-1\}$. Let $PreInt$ denote the set consisting of all closed intervals of the form $[0, T]$, where $T \in \mathbb{R}^+$, and the infinite interval $[0, \infty)$. Given an $x \in \mathbb{R}^n$, we use $|x|$ to denote the Euclidean norm of x. And, given a function $f : A \to \mathbb{R}^m$, we use $\|f\|_\infty$ to denote $\sup_{a \in A} |f(a)|$.

Functions and Relations.

Given a function F, let $Dom(F)$ denote the domain of F. Given a function $F : A \to B$ and a set $A' \subseteq A$, $F(A')$ denotes the set $\{F(a) \,|\, a \in A'\}$. Given a binary relation $R \subseteq A \times B$, R^{-1} denotes the set $\{(x, y) \,|\, (y, x) \in R\}$. For a binary relation R, we will interchangeably use "$(x, y) \in R$" and "$R(x, y)$" to denote that $(x, y) \in R$.

Sequences.

A sequence σ is a function whose domain is either $[n]$ for some $n \in \mathbb{N}$ or the set of natural numbers $\mathbb{N}$. We denote the set of all domains of sequences as $SeqDom$. Length of a sequence σ, denoted $|\sigma|$, is n if $Dom(\sigma) = [n]$ or ∞ otherwise. Given a sequence $\sigma : \mathbb{N} \to \mathbb{R}$ and an element r of $\mathbb{R}_\infty$ we use $\sum_{i=0}^{\infty} \sigma(i) = r$ to denote the standard limit condition $\lim_{N \to \infty} \sum_{i=0}^{N} \sigma(i) = r$.

Extended Metric Space.

An *extended metric space* is a pair (M, d) where M is a set and $d : M \times M \to \mathbb{R}_\infty$ is a distance function such that for all m_1, m_2 and m_3, the following hold: (Identity of indiscernibles) $d(m_1, m_2) = 0$ if and only if $m_1 = m_2$, (Symmetry) $d(m_1, m_2) = d(m_2, m_1)$, and (Triangle inequality) $d(m_1, m_3) \leq d(m_1, m_2) + d(m_2, m_3)$. When the metric on M is clear we will simply refer to M as a metric space.

We define an open ball of radius ϵ around a point x to be the set of all points which are within a distance ϵ from x. Formally, an *open ball* is a set of the form $B_\epsilon(x) = \{y \in M \,|\, d(x, y) < \epsilon\}$. An *open set* is a subset of M which is a union of open balls. Given a set $X \subseteq M$, a *neighborhood* of X is an open set in M which contains X. Given a subset X of M, an ϵ-neighborhood of X is the set $B_\epsilon(X) = \bigcup_{x \in X} B_\epsilon(x)$. A subset X of M is *compact* if for every collection of open sets $\{U_\alpha\}_{\alpha \in A}$ such that $X \subseteq \bigcup_{\alpha \in A} U_\alpha$, there is a finite subset J of A such that $X \subseteq \bigcup_{i \in J} U_i$.

Set Valued Functions.

We consider set valued functions and define continuity of these functions. We choose not to treat set valued functions as single valued functions whose co-domain is a power set, since as argued in [3], it leads to strong notions of continuity, which are not satisfied by many functions. A *set valued function* $F : A \rightsquigarrow B$ is a function which maps every element of A to a set of elements in B. Given a set $A' \subseteq A$, $F(A')$ will denote the set $\bigcup_{a \in A'} F(a)$. Given a binary relation $R \subseteq A \times B$, we use R also to denote the set valued function $R : A \rightsquigarrow B$ given by $R(x) = \{y \mid (x, y) \in R\}$. Further, $F^{-1} : B \rightsquigarrow A$ will denote the set valued function which maps $b \in B$ to the set $\{a \in A \mid b \in F(a)\}$.

Continuity of Set Valued Functions.

Let $F : A \rightsquigarrow B$ be a set valued function, where A and B are extended metric spaces. We define upper semi-continuity of F which is a generalization of the "ϵ, δ - definition" of continuity for single valued functions [3]. The function $F : A \rightsquigarrow B$ is said to be *upper semi-continuous* at $a \in \mathit{Dom}(F)$ if and only if

$$\forall \epsilon > 0, \exists \delta > 0 \,\text{such that}\, F(B_\delta(a)) \subseteq B_\epsilon(F(a)).$$

If F is upper semi-continuous at every $a \in \mathit{Dom}(F)$ we simply say that F is upper semi-continuous. Next we define a "uniform" version of the above definition, where, analogous to the case of single valued functions, corresponding to an ϵ, there exists a δ which works for every point in the domain.

Definition. A function $F : A \rightsquigarrow B$ is said to be *uniformly continuous* if and only if

$$\forall \epsilon > 0, \exists \delta > 0 \,\text{such that}$$

$$\forall a \in \mathit{Dom}(A), F(B_\delta(a)) \subseteq B_\epsilon(F(a)).$$

Given an $\epsilon > 0$, we call a $\delta > 0$ satisfying the above condition, a *uniformity constant of* F corresponding to ϵ. We refer to uniform upper semi-continuity as just uniform continuity, because it turns out that the two notions of upper and lower semi-continuity coincide with the addition of uniformity condition, i.e., uniform upper semi-continuity is equivalent to uniform lower semi-continuity.

Class $\mathcal{K}$, L, $\mathcal{K}_\infty$ and $\mathcal{KL}$ functions.

A continuous function $\alpha : [0, a) \to [0, \infty)$ is said to belong to *class* $\mathcal{K}$ if it is strictly increasing and $\alpha(0) = 0$. A continuous function $\alpha : [0, \infty) \to [0, \infty)$ is said to belong to *class* $\mathcal{K}_\infty$ if α is a class $\mathcal{K}$ function and $\alpha(r) \to \infty$ as $r \to \infty$. A continuous function $\varphi : [0, \infty) \to [0, \infty)$ is said to be of *class* L if it is monotonically decreasing and $\lim_{s \to \infty} \varphi(s) = 0$. A continuous function $\beta : [0, a) \times [0, \infty) \to [0, \infty)$ is a *class* $\mathcal{KL}$ function if it is a class $\mathcal{K}$ function with respect to the first argument and class L with respect to the second argument, that is, for a fixed s, $\beta(r, s)$ is a class $\mathcal{K}$ function and for a fixed r, $\beta(r, s)$ is a class L function.

3. HYBRID SYSTEMS WITH INPUT

In this section, we present a general formalism for representing hybrid systems with inputs, called *hybrid input transition systems*. Hybrid systems are systems exhibiting mixed discrete-continuous behaviors. We represent the continuous behavior using a pair of input and state *trajectories* which capture the values of input and state over an interval of time; and represent the discrete behavior using *transitions* which capture instantaneous changes to the state due to impulse inputs. We will not concern ourselves with the exact representation of the models, see, for example, the hybrid automaton model [14]. However, our abstract model captures the behaviors arising from a hybrid automaton model.

3.1 Trajectories

A *trajectory* τ over a set A is a function $\tau : I \to A$, where $I \in \mathit{PreInt}$. We denote the set of all trajectories over A as $\mathit{Traj}(A)$. Let us define a function $\mathit{Size} : \mathit{Traj}(A) \to \mathbb{R}_\infty$ which assigns a size to the trajectories. For $\tau \in \mathit{Traj}(A)$, $\mathit{Size}(\tau) = T$ if $\mathit{Dom}(\tau) = [0, T]$ and $\mathit{Size}(\tau) = \infty$ if $\mathit{Dom}(\tau) = [0, \infty)$.

Relating trajectories.

Given a relation $R \subseteq A_1 \times A_2$ and trajectories $\mathbf{a}_1 \in \mathit{Traj}(A_1)$ and $\mathbf{a}_2 \in \mathit{Traj}(A_2)$, we say that $\mathbf{a}_1$ and $\mathbf{a}_2$ are related by R, denoted $R(\mathbf{a}_1, \mathbf{a}_2)$ if $\mathit{Dom}(\mathbf{a}_1) = \mathit{Dom}(\mathbf{a}_2)$ and for every $t \in \mathit{Dom}(\mathbf{a}_1)$, $R(\mathbf{a}_1(t), \mathbf{a}_2(t))$. We use $R(\mathbf{a}_1)$ to denote the set $\{\mathbf{a}_2 \mid R(\mathbf{a}_1, \mathbf{a}_2)\}$.

Input-State Trajectories.

An input-state trajectory specifies the state evolution on an input signal. Let us fix an *input space* U and a *state space* S. An *input-state trajectory* over a pair (U, S) is a pair of trajectories $(\mathbf{u}, \mathbf{s})$ from $\mathit{Traj}(U) \times \mathit{Traj}(S)$ such that $\mathit{Dom}(\mathbf{u}) = \mathit{Dom}(\mathbf{s})$. We call $\mathbf{u}$ an *input trajectory* and $\mathbf{s}$ a *state trajectory*. We will use $\mathit{ISTraj}(U, S)$ to denote the set of all input-state trajectories over (U, S).

Size, First, Last, States, Inputs of Input-State Trajectories.

We extend *Size* to input-state trajectories in the natural way, namely, $\mathit{Size}(\mathbf{u}, \mathbf{s}) = \mathit{Size}(\mathbf{u}) = \mathit{Size}(\mathbf{s})$. We use $\mathit{First}((\mathbf{u}, \mathbf{s}))$ to denote the initial state, that is, $\mathbf{s}(0)$, and $\mathit{Last}((\mathbf{u}, \mathbf{s}))$ to denote the last state, that is, $\mathbf{s}(\mathit{Size}(\mathbf{s}))$, if $\mathit{Size}(\mathbf{s})$ is not ∞, and is not defined otherwise. Given a state trajectory $\mathbf{s}$, we use $\mathit{States}(\mathbf{s})$ to denote the set of states occurring in $\mathbf{s}$, namely, $\{\mathbf{s}(t) \mid t \in \mathit{Dom}(\mathbf{s})\}$. Also, for a input-state trajectory we use $\mathit{States}((\mathbf{u}, \mathbf{s}))$ to denote $\mathit{States}(\mathbf{s})$. Similarly, for an input trajectory $\mathbf{u}$, we use $\mathit{Inputs}(\mathbf{u})$ to denote the set of inputs occurring in $\mathbf{u}$, namely, $\{\mathbf{u}(t) \mid t \in \mathit{Dom}(\mathbf{u})\}$.

3.2 Transitions

A transition specifies the instantaneous change in a state resulting from an impulse input. A *transition* over a pair (U, S) is an element of $U \times (S \times S)$. A transition $(u, (s_1, s_2))$ denotes the fact that if an input impulse u is applied to the system in state s_1, then the system state changes to s_2. We will represent a transition $(u, (s_1, s_2))$ as $s_1 \xrightarrow{u} s_2$. We denote the set of all transition over a pair (U, S) as $\mathit{Trans}(U, S)$.

Size, First, Last, States, Inputs of Transitions.

We define *Size* of a transition $(u, (s_1, s_2))$ to be 0. As before, given $\tau = (u, (s_1, s_2))$, we use $First(\tau)$ and $Last(\tau)$ to denote the state of the system before and after the transition, namely, $First(\tau) = s_1$ and $Last(\tau) = s_2$. Also, $First((s_1, s_2)) = s_1$ and $Last((s_1, s_2)) = s_2$. Similarly, $States((s_1, s_2)) = States((u, (s_1, s_2))) = \{s_1, s_2\}$. And, $Inputs(u) = \{u\}$, for an input u.

3.3 Hybrid Input Transition Systems

We can now define a hybrid input transition system as consisting of sets of input-state trajectories and transitions.

Definition. A *hybrid input transition system* (*HITS*) $\mathcal{H}$ is a tuple (S, U, Σ, Δ), where S is a set of states, U is a set of inputs, $\Sigma \subseteq Trans(U, S)$ is a set of transitions and $\Delta \subseteq ISTraj(U, S)$ is a set of input-state trajectories.

We will just use hybrid system or hybrid transition system to refer to the above entity. Next, we define an execution of a hybrid transition system, which is a behavior of the system. An execution is a finite or infinite sequence of trajectories and transitions which have matching end-points.

Definition. An *execution* of a hybrid input transition system $\mathcal{H}$ is a sequence $\sigma : M \to \Sigma \cup \Delta$, where $M \in SeqDom$, such that for each $0 \leq i < |\sigma| - 1$, $Last(\sigma(i)) = First(\sigma(i + 1))$. Let $Exec(\mathcal{H})$ denote the set of all executions of $\mathcal{H}$.

We can view an execution as a pair consisting of an input signal and state signal. Let $\sigma \in Exec(\mathcal{H})$. Then for each $i \in Dom(\sigma)$, $\sigma(i) = (\mathbf{u}_i, \mathbf{s}_i)$, where either $(\mathbf{u}_i, \mathbf{s}_i)$ is an input-state trajectory or a transition. Let σ^u and σ^s be sequences whose domain is the same as σ such that $\sigma^u(i) = \mathbf{u}_i$ and $\sigma^s(i) = \mathbf{s}_i$. Then we also use (σ^u, σ^s) to denote the execution σ.

Given a set of executions $\mathcal{T}$ and an input signal σ^u, we use $\mathcal{T}|_{\sigma^u}$ to denote the set of all state signals of executions in $\mathcal{T}$ which result from application of the input signal σ^u. Formally, $\mathcal{T}|_{\sigma^u} = \{\sigma^s \mid (\sigma^u, \sigma^s) \in \mathcal{T}\}$.

First, Last, States, Inputs of Executions.

We extend *first* and *last* to executions and state signals in the natural way, that is, the *first* of the first element in the sequence and the *last* of the last element if the sequence is finite. Formally, for an execution or a state signal σ, $First(\sigma) = First(\sigma(0))$ and $Last(\sigma)$ is defined only if $Dom(\sigma) = [n]$ for some $n \in \mathbb{N}$ and is equal to $Last(\sigma(n))$. Similarly, $States(\sigma) = \bigcup_{i \in Dom(\sigma)} States(\sigma(i))$. Also, for an input signal σ^u, $Inputs(\sigma^u) = \bigcup_{i \in Dom(\sigma^u)} Inputs(\sigma^u(i))$. The functions are extended to sets of trajectories, state signals and executions in a natural manner. Let $States(\mathcal{H})$ denote $States(\Sigma) \cup States(\Delta)$ and $Inputs(\mathcal{H})$ denote $Inputs(\Sigma) \cup Inputs(\Delta)$.

Graph of an execution.

In order to define distance between executions, we interpret the input and state signals as sets called the graphs which have information about the linear ordering between the states and inputs at various times. The set corresponding to a state signal σ^s consists of triples (t, i, s) such that s is a state that is reached after time t has elapsed along the execution, and i is the number of discrete transitions that have taken place before time t. Similarly, the set corresponding to an input signal σ^u consists of triples (t, i, u) such that the input u was applied at time t, and the number of impulse inputs applied before time t is i.

Definition. For an input or state signal σ^a and $j \in Dom(\sigma^a)$, let $T_j = \sum_{k=0}^{j-1} Size(\sigma^a(k))$ and $K_j = |\{k \mid k < j,\ \sigma^a(k)$ is not a trajectory$\}|$. The *graph* of the signal σ^a, denoted $gr(\sigma^a)$, is the set of all triples (t, i, x) such that there exists $j \in Dom(\sigma^a)$ satisfying the following:

- $t \in [T_j, T_j + Size(\sigma^a(j))]]$.
- If $\sigma^a(j)$ is a trajectory, then $i = K_j$ and $x = \sigma^a(j)(t - T_j)$.
- If $\sigma^a(j)$ is not a trajectory, then
  - if σ^a is a state signal and $\sigma^a(j) = (s_1, s_2)$, then either $i = K_j$ and $x = s_1$, or $i = K_j + 1$ and $x = s_2$.
  - if σ^a is an input signal and $\sigma^a(i) = u$, then $i = K_j$ and $x = u$.

3.4 Metric Hybrid Input Transition System

In order to reason about stability of a system, one needs a notion of distance between behaviors of the system. Hence, we extend the definition of the hybrid system with a metric on the states and inputs which can then be extended to distance between signals and executions.

A metric hybrid input transition system is a hybrid input transition system whose state and input spaces are equipped with a metric. A *metric hybrid input transition system* (*MHS*) is a pair $(\mathcal{H}, d^s, d^u)$ where $\mathcal{H} = (S, U, \Sigma, \Delta)$ is a hybrid input transition system, and (S, d^s) and (U, d^u) are extended metric spaces. The metric d^s on the state space can be lifted to state signals and d^u to input signals, which will then be used to define input-to-state stability notions. Before defining this extension, recall that given an extended metric space (M, d), the *Hausdorff distance* between $A, B \subseteq M$, also denoted $d(A, B)$, is given by the maximum of

$$\{\sup_{p \in A} \inf_{q \in B} d(p, q), \sup_{p \in B} \inf_{q \in A} d(p, q)\}.$$

We extend d to triples used in the definition of graphs.

Definition. For $(t_1, i_1, x_1), (t_2, i_2, x_2) \in \mathbb{R}^+ \times \mathbb{N} \times M$, let

$$d((t_1, i_1, x_1), (t_2, i_2, x_2)) = \max\{|t_1 - t_2|, |i_1 - i_2|, d(x_1, x_2)\}.$$

Now we can define the distance between state signals and input signals.

Definition. Let $(\mathcal{H}, d^s, d^u)$ be a metric hybrid input transition system with $\mathcal{H} = (S, U, \Sigma, \Delta)$. The *distance between state signals* σ_1^s, σ_2^s, denoted as $d^s(\sigma_1^s, \sigma_2^s)$, is defined to be the distance between their graphs, that is, $d^s(gr(\sigma_1^s), gr(\sigma_2^s))$, and the *distance between input signals* σ_1^u, σ_2^u, denoted $d^u(\sigma_1^u, \sigma_2^u)$, is defined as $d^u(gr(\sigma_1^u), gr(\sigma_2^u))$.

Distance between executions as defined above, called *graphical distance*, captures the notion that two executions are close if their states are close at approximately same times. The notion of graphical distance is borrowed from [12], where it has been argued that allowing a wiggle time is necessary when one considers hybrid executions. Graphical distance between two executions is illustrated in Figure 1. Note that the two executions σ and σ' are not close at all times t, for example, at a time $t \in (t_1, t_2)$, the states are very far. However, for every time t and corresponding state s of σ, there exists a time $t' \in [t - \epsilon, t + \epsilon]$ such that s is close to the state of σ' at time t'. For example, s_2 is close to s_2' and times t_1 and t_2 are close.

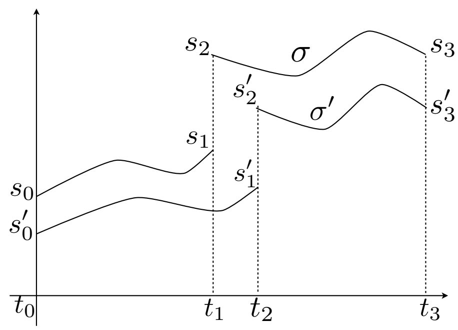

Figure 1: Graphical Distance between Executions.

In order to define convergence, we need the distance between suffixes of signals starting from some time T. Given a subset G of $\mathbb{R}^+ \times \mathbb{N} \times A$ and a $T \in \mathbb{R}^+$, let us denote by $G|_T$ the set $\{(t, i, x) \in G \,|\, t \geq T\}$. Given two signals σ_1, σ_2 and a $T \in \mathbb{R}^+$, we define $d(\sigma_1|_T, \sigma_2|_T)$ to be $d(gr(\sigma_1)|_T, gr(\sigma_2)|_T)$.

4. AN ALTERNATE CHARACTERIZATION OF INCREMENTAL INPUT-TO-STATE STABILITY OF CONTINUOUS DYNAMICAL SYSTEMS

Our definition of input-to-state stability is motivated by the following definition of incremental input-to-state stability of [2]. Let $\mathcal{T}$ be a set of input-state trajectories over $(\mathbb{R}^m, \mathbb{R}^n)$ such that for each $\zeta \in \mathbb{R}^n$ and input trajectory $\mathbf{u}$, there exists a unique element $(\mathbf{u}, \mathbf{s}) \in \mathcal{T}$ with *First*$(\mathbf{s}) = \zeta$. Given ζ and $\mathbf{u}$, let us denote the unique trajectory $\mathbf{s}$ by $\mathbf{x}(\zeta, \mathbf{u})$. Then the definition of incremental input-to-state stability from [2] is as follows:

Definition.(δISS for input-state trajectories) The set of input-state trajectories $\mathcal{T}$ is said to be *incrementally input-to-state stable* if there exists a $\mathcal{KL}$ function β and a $\mathcal{K}_\infty$ function γ such that for any $t \geq 0$, any ζ_1, ζ_2 and any pair of input trajectories u_1, u_2, the following is true:

$$|\mathbf{x}(\zeta_1, \mathbf{u}_1)(t) - \mathbf{x}(\zeta_2, \mathbf{u}_2)(t)| \leq \beta(|\zeta_1 - \zeta_2|, t) + \gamma(\|\mathbf{u}_1 - \mathbf{u}_2\|_\infty). \quad (1)$$

The above definition forces the following properties of the system $\mathcal{T}$:

($C1$) The system is Lyapunov stable "uniformly" in the input. For every $\epsilon > 0$, there exists a $\delta > 0$, such that for every input trajectory $\mathbf{u}$, and for all initial states ζ_1, ζ_2, the following holds for every $t \geq 0$.

$$|\zeta_1 - \zeta_2| < \delta \Rightarrow |\mathbf{x}(\zeta_1, \mathbf{u})(t) - \mathbf{x}(\zeta_2, \mathbf{u})(t)| < \epsilon.$$

Note that δ depends only on ϵ, in particular, it is independent of the input trajectory $\mathbf{u}$.

($C2$) The system converges "uniformly" in the input. For any $\epsilon > 0$ and initial states ζ_1, ζ_2, there exists a $T \geq 0$, such that for every input signal $\mathbf{u}$,

$$|\mathbf{x}(\zeta_1, \mathbf{u})(t) - \mathbf{x}(\zeta_2, \mathbf{u})(t)| < \epsilon, \forall t > T.$$

Note that T depends only on ϵ and is independent of $\mathbf{u}$.

($C3$) The system is input-to-state stable "uniformly" in the initial state. For any $\epsilon > 0$ and input signals $\mathbf{u}_1, \mathbf{u}_2$, there exists a $\delta > 0$ such that for every initial state ζ, the following holds for every $t \geq 0$:

$$\|\mathbf{u}_1 - \mathbf{u}_2\|_\infty < \delta \Rightarrow |\mathbf{x}(\zeta, \mathbf{u}_1)(t) - \mathbf{x}(\zeta, \mathbf{u}_2)(t)| < \epsilon.$$

Note the independence of δ with respect to ζ.

4.1 Super-position type theorem for incremental input-to-state stability

We show that the conditions $(C1) - (C3)$ characterize incremental input-to-state stability as given in Definition 4. This is summarized in the following theorem:

THEOREM 1. *A set of input-state trajectories $\mathcal{T}$ is δISS iff it satisfies Conditions $(C1) - (C3)$.*

PROOF. $\delta ISS \Rightarrow (C1) - (C3)$: It is straightforward to check that δISS implies conditions $(C1) - (C3)$. In fact, choosing $\mathbf{u}_1 = \mathbf{u}_2 = \mathbf{u}$ and δ such that $\beta(\delta, 0) < \epsilon$ in (1) implies

$$|\mathbf{x}(\zeta_1, \mathbf{u})(t) - \mathbf{x}(\zeta_2, \mathbf{u})(t)| \leq \beta(|\zeta_1 - \zeta_2|, t) < \beta(\delta, 0) < \epsilon,$$

provided that $|\zeta_1 - \zeta_2| < \delta$. This shows $(C1)$ is true. Moreover, since

$$|\mathbf{x}(\zeta_1, \mathbf{u})(t) - \mathbf{x}(\zeta_2, \mathbf{u})(t)| \leq \beta(|\zeta_1 - \zeta_2|, t) \to 0$$

as $t \to \infty$, for any given ϵ and ζ_1, ζ_2, we can choose T independent of $\mathbf{u}$ such that

$$|\mathbf{x}(\zeta_1, \mathbf{u})(t) - \mathbf{x}(\zeta_2, \mathbf{u})(t)| < \epsilon$$

for all $t > T$. This shows $(C2)$ is true. Finally, choosing $\zeta_1 = \zeta_2 = \zeta$ and δ such that $\gamma(\delta) < \epsilon$ in (1) implies

$$|\mathbf{x}(\zeta, \mathbf{u})(t) - \mathbf{x}(\zeta, \mathbf{u})(t)| \leq \gamma(|\mathbf{u}_1 - \mathbf{u}_2|_\infty) < \gamma(\delta) < \epsilon,$$

provided that $\|\mathbf{u}_1 - \mathbf{u}_2\|_\infty < \delta$. This shows $(C3)$ is true.

$(C1) - (C3) \Rightarrow \delta ISS$: The proof for the opposite implication essentially follows from the proof of Lemma 4.5 in [16]. Therefore, the detailed argument is omitted and the following is an outline of the proof. First, by (C3), we can prove there exists a $\mathcal{K}_\infty$ function γ such that

$$|\mathbf{x}(\zeta, \mathbf{u}_1)(t) - \mathbf{x}(\zeta, \mathbf{u}_2)(t)| \leq \gamma(\|\mathbf{u}_1 - \mathbf{u}_2\|_\infty), \quad t \geq 0, \quad (2)$$

holds for all initial states ζ_1, ζ_2. Second, conditions $(C1)$ and $(C2)$ imply that there exist a $\mathcal{KL}$ function β such that

$$|\mathbf{x}(\zeta_1, \mathbf{u})(t) - \mathbf{x}(\zeta_2, \mathbf{u})(t)| \leq \beta(|\zeta_1 - \zeta_2|, t), \quad t \geq 0, \quad (3)$$

holds for all input trajectory u. Now given any pair of initial states ζ_1, ζ_2 and any pair of input trajectories u_1, u_2, it follows from (2) and (3) that

$$\begin{aligned} &|\mathbf{x}(\zeta_1, \mathbf{u}_1)(t) - \mathbf{x}(\zeta_2, \mathbf{u}_2)(t)| \\ &\leq |\mathbf{x}(\zeta_1, \mathbf{u}_1)(t) - \mathbf{x}(\zeta_2, \mathbf{u}_1)(t) + \mathbf{x}(\zeta_2, \mathbf{u}_1)(t) - \mathbf{x}(\zeta_2, \mathbf{u}_2)(t)| \\ &\leq \beta(|\zeta_1 - \zeta_2|, t) + \gamma(\|\mathbf{u}_1 - \mathbf{u}_2\|_\infty), \quad t \geq 0. \end{aligned}$$

This completes the proof. $\square$

5. INCREMENTAL INPUT-TO-STATE STABILITY OF HYBRID INPUT TRANSITION SYSTEMS

In this section, we define a notion of incremental input-to-state stability of hybrid input transition systems by generalizing the characterization of incremental input-to-state stability for continuous dynamical systems given by Theorem 1. To define distances between executions, we borrow the notion of graphical distance introduced in [12] for hybrid trajectories. However, the results in the paper are not sensitive to the particular definition of distance.

We need a few definitions. We denote by $\mathit{Valid}(\mathcal{T})$, the set of all pairs of input signals and initial states such that there exists a state signal corresponding to it in $\mathcal{T}$. More precisely, $\mathit{Valid}(\mathcal{T}) = \{(\sigma^u, \zeta) \mid \exists \sigma^s, \mathit{First}(\sigma^s) = \zeta, (\sigma^u, \sigma^s) \in \mathcal{T}\}$. We denote by $\mathit{InSig}(\mathcal{T})$, the set of all input signals such that there exists a state signal corresponding to it in $\mathcal{T}$. That is, $\mathit{InSig}(\mathcal{T}) = \{\sigma^u \mid \exists \sigma^s, (\sigma^u, \sigma^s) \in \mathcal{T}\}$.

Next, we define incremental input-to-state stability (δISS) for hybrid input transition systems.

Definition.(δISS for Hybrid Systems) Given a hybrid input transition system $\mathcal{H}$ and a set of executions $\mathcal{T} \subseteq \mathit{Exec}(\mathcal{H})$, we say that $\mathcal{H}$ is *incrementally input-to-state stable* (δISS) with respect to the set of executions $\mathcal{T}$, if the following hold:

($D1$) for every $\epsilon > 0$, there exists a $\delta > 0$, such that the following holds for every input signal σ^u:

$$\forall (\sigma^u, \sigma^s) \in \mathit{Exec}(\mathcal{H}), d^s(\mathit{First}(\sigma^s), \mathit{First}(\mathcal{T}|_{\sigma^u})) < \delta$$

$$\Rightarrow \exists (\sigma^u, \hat{\sigma}^s) \in \mathcal{T}, d^s(\sigma^s, \hat{\sigma}^s) < \epsilon$$

($D2$) there exists a $\delta > 0$ and a function $T : \mathbb{R}_{>0} \to \mathbb{R}_{>0}$ such that the following holds for every input signal σ^u:

$$\forall (\sigma^u, \sigma^s) \in \mathit{Exec}(\mathcal{H}), d^s(\mathit{First}(\sigma^s), \mathit{First}(\mathcal{T}|_{\sigma^u})) < \delta \Rightarrow$$

$$\exists (\sigma^u, \hat{\sigma}^s) \in \mathcal{T}, \forall \epsilon > 0, \forall t \geq T(\epsilon), d^s(\sigma^s|_t, \hat{\sigma}^s|_t) < \epsilon.$$

($D3$) for every $\epsilon > 0$, there exists a $\delta > 0$ such that for every input signal σ^u and state ζ with $(\sigma^u, \zeta) \in \mathit{Valid}(\mathcal{T})$, the following holds:

$$\forall \hat{\sigma}^u, [d^u(\sigma^u, \hat{\sigma}^u) < \delta \Rightarrow \forall (\hat{\sigma}^u, \hat{\sigma}^s) \in \mathit{Exec}(\mathcal{H}),$$

$$[\mathit{First}(\hat{\sigma}^s) = \zeta \Rightarrow \exists (\sigma^u, \sigma^s) \in \mathcal{T},$$

$$\mathit{First}(\sigma^s) = \zeta, d^s(\sigma^s, \hat{\sigma}^s) < \epsilon]]$$

Remark There have been several proposals for defining metrics on the set of executions, including the Skorokhod metric (see [9, 5] for more details), wherein, two executions are considered close if there exists a bijective, strictly order-preserving function between the time domains of the executions, such that the distance between a time point and its image under the function is small and the values of the corresponding states are small. However, the results in the paper are not sensitive to the particular choice of the distance metric.

6. PRE-ORDERS

In this section, we define a pre-order - a reflexive, transitive relation - on the class of hybrid input transition systems, such that if a system is incrementally input-to-state stable, then all systems below it in the ordering are incrementally input-to-state stable. Defining such an ordering is the first step towards developing an abstraction refinement framework for verifying incremental input-to-state stability. We begin by defining the canonical notion of equivalence between systems with input, namely, input bisimulations. We show that this canonical notion does not suffice to preserve δISS and hence strengthen it with additional conditions. In the next section, we show that the new notion preserves δISS.

6.1 Input (Bi)-Simulations

The notion of input (bi)-simulation is an extension of the classical notion of (bi)-simulation with inputs for hybrid input transition systems. Our definition is closely related to the definition of (bi)-simulation defined in [15].

Definition. Given two hybrid input transition systems $\mathcal{H}_1 = (S_1, U_1, \Sigma_1, \Delta_1)$ and $\mathcal{H}_2 = (S_2, U_2, \Sigma_2, \Delta_2)$, a pair of binary relations (R_1, R_2), where $R_1 \subseteq S_1 \times S_2$ and $R_2 \subseteq U_1 \times U_2$, is called an *input simulation relation* from $\mathcal{H}_1$ to $\mathcal{H}_2$ if, for every $(s_1, s_2) \in R_1$, the following hold:

- For every state s_1' and input u_1 such that $(u_1, (s_1, s_1')) \in \Sigma_1$, there exist a state s_2' and an input u_2 such that $R_1(s_1', s_2')$, $R_2(u_1, u_2)$ and $(u_2, (s_2, s_2')) \in \Sigma_2$.
- For every input-state trajectory $(\mathbf{u}_1, \mathbf{s}_1) \in \Delta_1$ such that $\mathit{First}(\mathbf{s}_1) = s_1$, there exists an input-state trajectory $(\mathbf{u}_2, \mathbf{s}_2) \in \Delta_2$ such that $\mathit{First}(\mathbf{s}_2) = s_2$, $\mathbf{s}_2 \in R_1(\mathbf{s}_1)$ and $\mathbf{u}_2 \in R_2(\mathbf{u}_1)$.

We denote the fact that (R_1, R_2) is an input simulation relation from $\mathcal{H}_1$ to $\mathcal{H}_2$ by $\mathcal{H}_1 \preceq_{(R_1,R_2)} \mathcal{H}_2$. Further, (R_1, R_2) is an *input bisimulation relation* between $\mathcal{H}_1$ and $\mathcal{H}_2$ if both (R_1, R_2) and (R_1^{-1}, R_2^{-1}) are input simulation relations, that is, $\mathcal{H}_1 \preceq_{(R_1,R_2)} \mathcal{H}_2$ and $\mathcal{H}_2 \preceq_{(R_1^{-1},R_2^{-1})} \mathcal{H}_1$.

Image of $\mathcal{H}$ under (R_1, R_2).

We define the image of a hybrid input transition system on a pair of relations . Given a hybrid input transition system $\mathcal{H} = (S, U, \Sigma, \Delta)$, and a pair (R_1, R_2), where $R_1 \subseteq S \times S'$ and $R_2 \subseteq U \times U'$, for some S' and U', we define $(R_1, R_2)(\mathcal{H})$ to be the hybrid input transition system $(S', U', \Sigma', \Delta')$, where:

- $\Sigma' = \{(u', (s_1', s_2')) \mid \exists (u, (s_1, s_2)) \in \Sigma, u' \in R_2(u), s_1' \in R_1(s_1), s_2' \in R_1(s_2)\}$.
- $\Delta' = \{(\mathbf{u}', \mathbf{s}') \mid \exists (\mathbf{u}, \mathbf{s}) \in \Delta, \mathbf{u}' \in R_2(\mathbf{u}), \mathbf{s}' \in R_1(\mathbf{s})\}$.

PROPOSITION 1. *Let $\mathcal{H} = (S, U, \Sigma, \Delta)$ be a hybrid input transition system, and $R_1 \subseteq S \times S'$ and $R_2 \subseteq U \times U'$, for some S' and U' such that $R_1(s)$ and $R_2(u)$ is not empty for any $s \in S$ and $u \in U$. Then (R_1, R_2) is an input simulation from $\mathcal{H}$ to $(R_1, R_2)(\mathcal{H})$.*

6.2 Uniformly Continuous Input (Bi)-Simulation

The notion of input bisimulation does not preserve incremental input-to-state stability. Hence, we strengthen it with uniformity conditions.

Let $(\mathcal{H}_1, d_1^s, d_1^u)$ and $(\mathcal{H}_2, d_2^s, d_2^u)$ be two metric input hybrid transition systems.

Definition. A pair (R_1, R_2) is a *uniformly continuous input simulation* from $\mathcal{H}_1$ to $\mathcal{H}_2$ if (R_1, R_2) is an input simulation from $\mathcal{H}_1$ to $\mathcal{H}_2$ and R_1, R_1^{-1}, R_2 and R_2^{-1} are uniformly continuous.

We denote the fact that (R_1, R_2) is a uniformly continuous input simulation from $\mathcal{H}_1$ to $\mathcal{H}_2$ by $\mathcal{H}_1 \preceq^C_{(R_1,R_2)} \mathcal{H}_2$, and

$\mathcal{H}_1 \preceq^C \mathcal{H}_2$ to denote that there exists (R_1, R_2) such that $\mathcal{H}_1 \preceq^C_{(R_1,R_2)} \mathcal{H}_2$.

Definition. A pair (R_1, R_2) is a *uniformly continuous input bisimulation* from $\mathcal{H}_1$ to $\mathcal{H}_2$ if (R_1, R_2) is a uniformly continuous input simulation from $\mathcal{H}_1$ to $\mathcal{H}_2$, and (R_1^{-1}, R_2^{-1}) is a uniformly continuous input simulation from $\mathcal{H}_2$ to $\mathcal{H}_1$.

Next, we show that uniformly continuous input simulations define a pre-order on hybrid input transitions systems.

THEOREM 2. *Let $(\mathcal{H}_i, d_i^s, d_i^u)$, for $1 \leq i \leq 3$, where $\mathcal{H}_i = (S_i, U_i, \Sigma_i, \Delta_i)$, be three metric hybrid transition systems. Then we have the following properties about $\preceq^C$:*

- *(Reflexivity) $\mathcal{H}_1 \preceq^C \mathcal{H}_1$.*
- *(Transitivity) If $\mathcal{H}_1 \preceq^C \mathcal{H}_2$ and $\mathcal{H}_2 \preceq^C \mathcal{H}_3$, then $\mathcal{H}_1 \preceq^C \mathcal{H}_3$,*

PROOF. (Reflexivity) Reflexivity follows from the fact that $\mathcal{H}_1 \preceq_{(Id_1,Id_2)} \mathcal{H}_2$, where $Id_1 = \{(s,s) \,|\, s \in S\}$ and $Id_2 = \{(u,u) \,|\, u \in U\}$.

(Transitivity) Transitivity follows from the fact that $\mathcal{H}_1 \preceq^C_{(R_1,R_2)} \mathcal{H}_2$ and $\mathcal{H}_2 \preceq^C_{(R_1',R_2')} \mathcal{H}_3$, then $\mathcal{H}_1 \preceq^C_{(R_1' \circ R_1, R_2' \circ R_2)} \mathcal{H}_3$, where $A \circ B = \{(x,z) \,|\, \exists (x,y) \in A, (y,z) \in B\}$, since composition of input bisimulations is an input bisimulation, and composition of uniformly continuous relations is a uniformly continuous relation. $\square$

6.3 Inadequacy of Weaker Notions of Input Bisimulations

We show that weaker extensions to the definition of input bisimulation which require only continuity instead of uniform continuity on either the input space or the state space, do not suffice to preserve incremental input-to-state stability.

6.3.1 Necessity of Uniform Continuity on the Input Space

We consider two systems $\mathcal{H}_1$ and $\mathcal{H}_2$ consisting only of trajectories such that $\mathcal{H}_1$ is δISS, whereas $\mathcal{H}_2$ is not. We then show that there exists a input bisimulation, which is uniformly continuous on the state space, but is only continuous on the input space.

The state-space of both the systems is $\mathbb{R}^2$, and the input space is $\mathbb{R}$. All the input trajectories are constant signals with values in $\mathbb{R}$. The set of initial states is $\{0\} \times [-1, 1]$. We will use variable x and y to denote the two dimensions. Hence, $\Phi((x_0, y_0), t, u)$ denotes the values of the trajectory starting from (x_0, y_0) with input u at time t. The reference trajectories, $\mathcal{T}_1$ and $\mathcal{T}_2$, are those corresponding to the initial state $(0, 0)$. Since our input signals are constant signals, we abuse notation and just use the constant value instead of the signal in Φ.

The dynamics of the first system $\mathcal{H}_1$ is given as follows:

$$\Phi((x_0, y_0), t, u) = (t, e^{-t} y_0 + u\bar{t}),$$

where $\bar{t} = t$ if $t \leq 1$, and 1 otherwise.

The dynamics of the second system $\mathcal{H}_2$ is similar to $\mathcal{H}_1$, except that the affect of the input at time $t = 1$ is e^u instead of u as in the previous system.

$$\Phi((x_0, y_0), t, u) = (t, e^{-t} y_0 + e^u \bar{t}),$$

where $\bar{t} = t$ if $t \leq 1$, and 1 otherwise.

Note that both the systems satisfy Conditions $C1$ and $C2$, where as, only the first system satisfies $C3$. Hence, the system $\mathcal{H}_1$ is δISS, where as $\mathcal{H}_2$ is not.

We can define an input bisimulation between the two systems given by:

$$R_1 = \{((x, y), (x, y))\}, R_2 = \{(e^u, u)\}.$$

This is an input bisimulation, since the trajectories starting from the initial state (x_0, y_0) and input signal e^u in $\mathcal{H}_1$ is related by R_1 to the trajectory starting from the initial state (x_0, y_0) and the input signal u in $\mathcal{H}_2$. Further, R_1 and R_1^{-1} are uniformly continuous, where as, R_2^{-1} is only continuous.

6.3.2 Necessity of Uniform Continuity on the State Space

We extend the counter-example in [20], which shows the necessity of uniform continuity on the state space for asymptotic stability preservation (without inputs), for showing the necessity of uniform continuity on the state-space for δISS preservation.

The first system $\mathcal{H}_3$ is the same as $\mathcal{H}_1$ above, except that the input space is restricted to be $\{0\}$. We define $\mathcal{H}_4$ which is the same as $\mathcal{H}_1$ except that the trajectories do not converge to the reference trajectory. The dynamics of the system $\mathcal{H}_4$ is given as follows:

$$\Phi((x_0, y_0), t, u) = (t, y_0).$$

Note that both $\mathcal{H}_3$ and $\mathcal{H}_4$ satisfy Conditions $C1$ and $C3$, where as, only $\mathcal{H}_3$ satisfies $C2$. Hence, $\mathcal{H}_3$ is δISS, where as, $\mathcal{H}_4$ is not.

We can define an input bisimulation between the two systems given by:

$$R_1 = \{((x_1, y_1), (x_2, y_2)) \,|\, x_1 = x_2, y_1 = e^{-x_1} y_2\}, R_2 = \{(u, u)\}.$$

This is an input bisimulation, since the trajectories starting from the initial state (x_0, y_0) and input signal u in $\mathcal{H}_2$ is related by R_1 to the trajectory starting from the initial state (x_0, y_0) and the input signal u in $\mathcal{H}_2$. Further, R_2 and R_2^{-1} are uniformly continuous, where as, R_1 is only continuous.

7. INCREMENTAL INPUT-TO-STATE STABILITY PRESERVATION

In this section, we present the main result of the paper, namely, that incremental input-to-state stability is invariant under uniformly continuous input bisimulations.

We need a technical consistency condition between the input bisimulation relations and the reference executions.

Definition. A pair of relations (R_1, R_2), where $R_1 \subseteq S_1 \times S_2$ and $R_2 \subseteq U_1 \times U_2$, is said to be *semi-consistent* with respect to the sets of executions $\mathcal{T}_1$ and $\mathcal{T}_2$ over (S_1, U_1) and (S_2, U_2), respectively, if the following hold:

$(A1)$ For every $(\sigma_1^u, \zeta_1) \in \mathit{Valid}(\mathcal{T}_1)$, there exists $(\sigma_2^u, \zeta_2) \in \mathit{Valid}(\mathcal{T}_2)$ such that $R_2(\sigma_1^u, \sigma_2^u)$ and $R_1(\zeta_1, \zeta_2)$.

$(A2)$ For every $(\sigma_2^u, \sigma_2^s) \in \mathcal{T}_2$, for every $\sigma_1^u \in R_2^{-1}(\sigma_2^u)$ and $\zeta_1 \in R_2^{-1}(\mathit{First}(\sigma_2^s))$ such that $(\sigma_1^u, \zeta_1) \in \mathit{Valid}(\mathcal{T}_1)$, there exists σ_1^s with $\mathit{First}(\sigma_1^s) = \zeta_1$, $R_1(\sigma_1^s, \sigma_2^s)$ and $(\sigma_1^u, \sigma_1^s) \in \mathcal{T}_1$.

$(A3)$ $R_2(u)$ is a singleton for every $u \in \mathit{Inputs}(\mathcal{T}_1)$.

$(A4)$ $R_1^{-1}(s)$ is singleton for every $s \in \mathit{States}(\mathcal{T}_2)$.

($A5$) For every σ_1^u, $R_1(\mathit{First}(\mathcal{T}_1|_{\sigma_1^u})) = \mathit{First}(\mathcal{T}_2|_{R_2(\sigma_1^u)})$.

($A6$) There exists $\delta > 0$ such that for every $x \in B_\delta(\mathit{First}(\mathcal{T}_1))$, there exists a y such that $R_1(x, y)$.

(R_1, R_2) is said to be *consistent* with respect to $\mathcal{T}_1$ and $\mathcal{T}_2$ if both (R_1, R_2) and (R_1^{-1}, R_2^{-2}) are semi-consistent with respect to $\mathcal{T}_1$ and $\mathcal{T}_2$.

THEOREM 3. *Let $(\mathcal{H}_1, d_1^s, d_1^u)$ and $(\mathcal{H}_2, d_2^s, d_2^u)$, where $\mathcal{H}_1 = (S_1, U_1, \Sigma_1, \Delta_1)$ and $\mathcal{H}_2 = (S_2, U_2, \Sigma_2, \Delta_2)$, be two metric hybrid input transition systems, and let $\mathcal{T}_1 \subseteq \mathit{Exec}(\mathcal{H}_1)$ and $\mathcal{T}_2 \subseteq \mathit{Exec}(\mathcal{H}_2)$ be two sets of executions. Let (R_1, R_2) be a uniformly continuous input simulation from $\mathcal{H}_1$ to $\mathcal{H}_2$, and let (R_1, R_2) be semi-consistent with respect to $\mathcal{T}_1$ and $\mathcal{T}_2$. Then the following holds:*

If $\mathcal{H}_2$ is δISS with respect to $\mathcal{T}_2$, then $\mathcal{H}_1$ is δISS with respect to $\mathcal{T}_1$.

PROOF. Let us assume $\mathcal{H}_2$ is δISS with respect to $\mathcal{T}_2$. We need to show that $\mathcal{H}_1$ is δISS with respect to $\mathcal{T}_1$. We will show that $\mathcal{H}_1$ satisfies conditions $(D1) - (D3)$.

Proof of satisfaction of Condition ($D1$) Let us fix an $\epsilon_1 > 0$. We need to find a $\delta_1 > 0$ such that Condition ($D1$) holds in $\mathcal{H}_1$ and $\mathcal{T}_1$. Let ϵ_2 be the uniformity constant of R_1^{-1} corresponding to ϵ_1. Let δ_2 be the constant satisfying Condition ($D1$) for $\mathcal{H}_2$ corresponding to ϵ_2. Set δ_1 to be the uniformity constant of R_2 corresponding to δ_2.

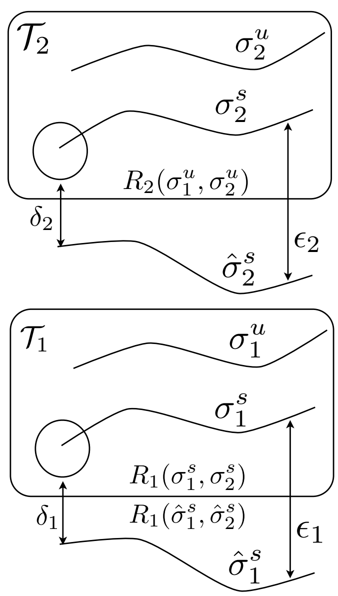

Figure 2: Illustration for Proof of Condition ($D1$)

Let us fix an input signal σ_1^u. Let $(\sigma_1^u, \sigma_1^s) \in \mathit{Exec}(\mathcal{H}_1)$ such that $d_1^s(\mathit{First}(\sigma_1^s), \mathit{First}(\mathcal{T}_1|_{\sigma_1^u}) < \delta_1$ (see Figure 2). We need to show that there exists a $\hat{\sigma}_1^s$, such that $(\sigma_1^u, \hat{\sigma}_1^s) \in \mathcal{T}_1$ and $d_1^s(\sigma_1^s, \hat{\sigma}_1^s) < \epsilon_1$.

Note that Condition ($A1$) also implies that there exists $\sigma_2^u \in \mathit{InSig}(\mathcal{T}_2)$ such that $R_2(\sigma_1^u, \sigma_2^u)$. Further, σ_2^u is unique because of Condition ($A3$) on R_2. From Condition ($A6$), there exists a ζ_2 such that $(\mathit{First}(\sigma_1^s), \zeta_2) \in R_1$. Therefore, from input simulation relation, there exists σ_2^s such that $(\sigma_2^u, \sigma_2^s) \in \mathit{Exec}(\mathcal{H}_2)$ and $R_1(\sigma_1^s, \sigma_2^s)$ (note that σ_2^u is the same as before, this follows from the uniqueness of σ_2^u). Since $d_1^s(\mathit{First}(\sigma_1^s), \mathit{First}(\mathcal{T}_1)\sigma_1^u) < \delta_1$, $d_2^s(R_1(\mathit{First}(\sigma_1^s)), R_1(\mathit{First}(\mathcal{T}_1)|_{\sigma_1^u})) < \delta_1$. From Condition ($A5$), we have $d_2^s(R_1(\mathit{First}(\sigma_1^s)), \mathit{First}(\mathcal{T}_2|_{R_2(\sigma_1^u)})) < \delta_1$, or equivalently $d_2^s(R_1(\mathit{First}(\sigma_1^s)), \mathit{First}(\ \mathcal{T}_2|_{\sigma_2^u})) < \delta_1$. In particular, $d_2^s(\mathit{First}(\sigma_2^s), \mathit{First}(\mathcal{T}_2|_{\sigma_2^u})) < \delta_1$. From the δISS of $\mathcal{H}_2$ with respect to $\mathcal{T}_2$, we have that there exists $\hat{\sigma}_2^s$ such that $(\sigma_2^u, \hat{\sigma}_2^s) \in \mathcal{T}_2$ and $d_2^s(\sigma_2^s, \hat{\sigma}_2^s) < \epsilon_2$. Then from Condition ($A2$), there exists $\hat{\sigma}_1^s$, such that $(\sigma_1^u, \hat{\sigma}_1^s) \in \mathcal{T}_1$, and $R_1(\hat{\sigma}_1^s, \hat{\sigma}_2^s)$. Now, $d_1^s(\sigma_1^s, \hat{\sigma}_1^s) < \epsilon_1$ since $R_1^{-1}(s)$ is a singleton for every $s \in \mathit{States}(\mathcal{T}_2)$ (from Condition ($A4$)).

Proof of satisfaction of Condition ($D2$) Let $\delta_2 > 0$ and $T_2 : \mathbb{R}^+ \to \mathbb{R}^+$ be such that they satisfy Condition ($D2$) for system $\mathcal{H}_2$ with respect to $\mathcal{T}_2$. Choose $\delta_1 > 0$ to be the uniformity constant of R_2 with respect to δ_2. Similarly, define $T_1 : \mathbb{R}^+ \to \mathbb{R}^+$ as follows: Given any $\epsilon_1 > 0$, set $T_1(\epsilon_1)$ to be equal to $T_2(\epsilon_2)$, where ϵ_2 is the uniformity constant of R_1^{-1} with respect to ϵ_1.

The proof essentially is the same as before, except that we need to show that $\forall \epsilon_1 > 0, \forall t \geq T_1(\epsilon_1), d_1^s(\sigma_1^s|_t, \hat{\sigma}_1^s|_t) < \epsilon_1$. Note that the above condition follows from the fact that now we have $\forall \epsilon_2 > 0, \forall t \geq T_2(\epsilon_2), d_2^s(\sigma_2^s|_t, \hat{\sigma}_2^s|_t) < \epsilon_2$. The required result follows from the definition of T_1.

Proof of satisfaction of Condition ($D3$) Let us fix an $\epsilon_1 > 0$, we need to find a $\delta_1 > 0$ such that Condition ($D3$) holds. Let ϵ_2 be the uniformity constant of R_1^{-1} corresponding to ϵ_1. Let δ_2 be the constant satisfying Condition ($D3$) for $\mathcal{H}_2$ corresponding to ϵ_2. Set δ_1 to be the uniformity constant of R_2 corresponding to δ_2.

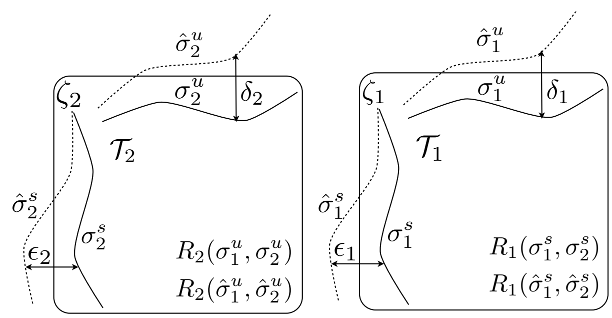

Figure 3: Illustration for Proof of Condition ($D3$)

Let us fix an input signal σ_1^u and state ζ_1 such that $(\sigma_1^u, \zeta_1) \in \mathit{Valid}(\mathcal{T}_1)$ (see Figure 3). Let $\hat{\sigma}_1^u$ be such that $d_1^u(\sigma_1^u, \hat{\sigma}_1^u) < \delta_1$ and let $(\hat{\sigma}_1^u, \hat{\sigma}_1^s) \in \mathit{Exec}(\mathcal{H}_1)$ with $\mathit{First}(\hat{\sigma}_1^s) = \zeta_1$. We need to show that there exists σ_1^s such that $\mathit{First}(\sigma_1^s) = \zeta_1$, $(\sigma_1^u, \sigma_1^s) \in \mathcal{T}_1$ and $d_1^s(\hat{\sigma}_1^s, \sigma_1^s) < \epsilon_1$.

From Condition ($A1$) of semi-consistency, we have that there exists $(\sigma_2^u, \zeta_2) \in \mathit{Valid}(\mathcal{T}_2)$ such that $R_2(\sigma_1^u, \sigma_2^u)$ and $R_1(\zeta_1, \zeta_2)$. From the fact that (R_1, R_2) is an input simulation, and $R_1(\zeta_1, \zeta_2)$, we know that there exists $(\hat{\sigma}_2^u, \hat{\sigma}_2^s) \in \mathit{Exec}(\mathcal{H}_2)$ with $\mathit{First}(\hat{\sigma}_2^s) = \zeta_2$, $R_2(\hat{\sigma}_1^u, \hat{\sigma}_2^u)$ and $R_1(\hat{\sigma}_1^s, \hat{\sigma}_2^s)$.

Now, $d_1^u(\sigma_1^u, \hat{\sigma}_1^u) < \delta_1$ and $R_2(u)$ is a singleton for every $u \in \mathit{Inputs}(\mathcal{T}_1)$ (from Condition ($A3$)) implies that $d_2^u(\sigma_2^u, \hat{\sigma}_2^u) < \delta_2$. From the definition of δISS for $\mathcal{H}_2$, we know that there exists σ_2^s such that $(\sigma_2^u, \sigma_2^s) \in \mathcal{T}_2$ and $d_2^s(\sigma_2^s, \hat{\sigma}_2^s) < \epsilon_2$.

From Condition ($A2$) of semi-consistency, there exists σ_1^s with $\mathit{First}(\sigma_1^s) = \zeta_1$, $R_1(\sigma_1^s, \sigma_2^s)$ and $(\sigma_1^u, \sigma_1^s) \in \mathcal{T}_1$. Note that $d_1^s(\sigma_1^s, \hat{\sigma}_1^s) < \epsilon_1$ since $d_2^s(\sigma_2^s, \hat{\sigma}_2^s) < \epsilon_2$, and $R_1^{-1}(s)$ is a singleton for every $s \in \mathit{States}(\mathcal{T}_2)$ (from Condition ($A4$)). □

THEOREM 4. *Let $(\mathcal{H}_1, d_1^s, d_2^u)$ and $(\mathcal{H}_2, d_2^s, d_2^u)$, where $\mathcal{H}_1 = (S_1, U_1, \Sigma_1, \Delta_1)$ and $\mathcal{H}_2 = (S_2, U_2, \Sigma_2, \Delta_2)$, be two metric* *hybrid input transition systems, and $\mathcal{T}_1 \subseteq Exec(\mathcal{H}_1)$ and $\mathcal{T}_2 \subseteq Exec(\mathcal{H}_2)$ be two sets of executions. Let (R_1, R_2) be a uniformly continuous input bisimulation from $\mathcal{H}_1$ to $\mathcal{H}_2$, and let (R_1, R_2) be consistent with respect to $\mathcal{T}_1$ and $\mathcal{T}_2$. Then the following holds:*

$\mathcal{H}_2$ is δISS with respect to $\mathcal{T}_2$ if and only if $\mathcal{H}_1$ is δISS with respect to $\mathcal{T}_1$.

Theorem 3 states that uniformly continuous input simulations serve as a notion of abstraction with respect to δISS and Theorem 4 states that uniformly continuous input bisimulations are a notion of equivalence between systems with respect to incremental input-to-state stability.

8. EXAMPLES OF CONCRETE ABSTRACTION FUNCTIONS

In this section, we argue that the notion of abstraction introduced in the paper is not too stringent by exhibiting concrete abstraction functions which satisfy the constraints imposed by the uniformly continuous input simulations/bisimulations. We show that the proofs of stability based on Lyapunov functions can be interpreted as constructing simpler hybrid input transition systems where the Lyapunov function serves as a uniformly continuous input simulation relation. We further illustrate this through a concrete example of a linear system with inputs. The main purpose of this section is to demonstrate that the definitions of uniformly continuous input simulations and bisimulations are reasonable.

8.1 Lyapunov Functions for Input-to-State Stability

In this section, we focus on systems in which the reference executions consist of the unique trajectory which always remains at the equilibrium point $\bar{0}$. In this case, the definition of incremental input-to-state stability coincides with the notion of input-to-state stability. First, we define a Lyapunov's theorem for analyzing input-to-state stability.

Consider a continuous dynamical system

$$\dot{x} = f(x, u), \tag{4}$$

$$x \in X = \mathbb{R}^n, u \in U \subseteq \mathbb{R}^m, x_0 \in X_0 \subseteq X,$$

where $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is locally Lipschitz in x and u, and X_0 and U are compact sets. We will assume that the input signal space D_u consists of functions $u : [0, \infty) \to U$ that are piecewise continuous, bounded functions of t for all $t \geq 0$.

We define the hybrid system corresponding to the System (4) to be the following: $\mathcal{H}_{f,X_0,X,U} = (X, U, \emptyset, \Delta)$, where Δ is the set of pairs $(\mathbf{u}, \mathbf{x})$, where $\mathbf{u}$ is in D_u, $\mathbf{x}(0) \in X_0$ and $\mathbf{x}$ is the solution of System (4) starting from $\mathbf{x}(0)$, that is, $\mathbf{u}, \mathbf{x}$ satisfy $\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t))$ for every $t \geq 0$. Let d^s and d^u be the standard Euclidean norms on $\mathbb{R}^n$ and $\mathbb{R}^m$, respectively.

For System (4), assume that the system $\dot{y} = f(y, 0)$ has a uniformly asymptotically stable equilibrium point at the origin.

Definition. A continuously differentiable function $V : X \to \mathbb{R}^+$ is said to be an *ISS* Lyapunov function for the System (4) if there exist class $\mathcal{K}_\infty$ functions $\alpha_1, \alpha_2, \alpha_3$ and $\mathcal{X}$ such that:

$$\alpha_1(\|x\|) \leq V(x(t)) \leq \alpha_2(\|x\|), \forall x \in X, t > 0 \tag{5}$$

$$\frac{\partial V(x)}{\partial x} f(x, u) \leq \alpha_3(\|x\|), \forall u \in D_u : \|x\| \geq \mathcal{X}(\|u\|). \tag{6}$$

THEOREM 5. *[22] (ISS Theorem) Let $V : X \to \mathbb{R}^+$ be an ISS Lyapunov function for the System (4). Then System (4) is input-to-state stable.*

Next, we show that an *ISS* Lyapunov function is essentially a uniformly continuous input simulation from System 4 to a one dimensional system obtained by the application of the Lyapunov function to System 4, such that the one-dimensional system is δISS. Hence, Theorem 3 gives an alternate proof of δISS of System 4.

Let us say that a function $F : \mathbb{R}^n \to \mathbb{R}$ has *non-zero differential* if there exists a neighborhood Y containing $\bar{0}$ such that the gradient of F at any point $y \in Y$ other than $\bar{0}$, $\nabla F(y)$, is non-zero. Following theorem formulates Lyapunov analysis in our framework:

THEOREM 6. *Let V be an ISS Lyapunov function for System (4), and let $N : \mathbb{R}^n \to \mathbb{R}^+$ be the function $u \mapsto |u|$. Let V have non-zero differential. Then:*

1. *$(V, N)(\mathcal{H}_{f,X_0,X,U})$ input simulates $\mathcal{H}_{f,X_0,X,U}$.*
2. *V, V^{-1}, N and N^{-1} are uniformly continuous over States($\mathcal{H}$) and Inputs($\mathcal{H}$) in a compact space containing an open ball around the origin.*
3. *(V, N) is consistent with $\mathcal{T}_{0,0}$ and $(V, N)(\mathcal{T}_{0,0})$.*
4. *$(V, N)(\mathcal{H}_{f,X_0,X,U})$ is δISS with respect to $(V, N)(\mathcal{T}_{0,0})$.*

Hence $\mathcal{H}_{f,X_0,X,U}$ is δISS with respect to $\mathcal{T}_{0,0}$.

PROOF. (Sketch.) The first property follows from Proposition 1. The uniform continuity of V and N follows from the fact that the state and input spaces are restricted to a compact space around the origin. The uniform continuity of V^{-1} and N^{-1} uses the non-zero differentiability. Note that to prove input-to-state stability, it suffice to prove the same in a small neighborhood around the origin in the state space and the input space. It is trivial to check the consistency of (V, N). The fourth property follows from Equations 5 and 6. The conclusion follows from the four properties using Theorem 3. □

8.2 Illustration on linear system

We illustrate Lyapunov function based analysis as an abstraction based analysis on a linear system example. Consider a linear system with inputs, that is,

$$\dot{x} = f(x, u) = Ax + Bu, A \in \mathbb{R}^{n \times n}, B \in \mathbb{R}^{n \times m}, \tag{7}$$

$$x \in X = \mathbb{R}^n, u \in U \subseteq \mathbb{R}^m, x_0 \in X_0 \subseteq X,$$

where, A is a Hurwitz matrix, and X_0 and U are compact sets.

Let P be a positive definite symmetric matrix satisfying $A^T P + PA = -Q$ for some positive definite matrix Q. Consider a function $R_1 : \mathbb{R}^n \to \mathbb{R}^+$ given by $R_1(x) = x^T Px$ and a function $R_2 : \mathbb{R}^m \to \mathbb{R}^+$ given by $R_2(u) = |u|$. Then, $\dot{R}_1(x) = \dot{x}^T Px + x^T P\dot{x} = x^T(A^T P + PA)x + u^T B^T Px + x^T PBu \leq -\lambda R_1(x) + \mu\|u\|_\infty$, where λ and μ are positive constants depending on P, Q and B.

Consider the one-dimensional system:

$$\dot{y} \leq -\lambda y + \mu \|v\|_\infty, y \geq 0. \quad (8)$$

Note that the solutions to the system satisfy $y(t) \leq e^{-\lambda t} y(0) + \mu/\lambda \|v\|_\infty$. It is easy to check from the solutions of this system, that it is trivially incrementally input-to-state stable.

We will show that (R_1, R_2) is a uniformly continuous input simulation from System (7) to System (8). Input simulation follows from the fact that if $(\mathbf{x}, \mathbf{u})$ satisfies $\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + B\mathbf{u}(t)$ for all $t \geq 0$, then by construction, $(R_1(\mathbf{x}), R_2(\mathbf{u}))$ satisfies $\dot{R_1}(\mathbf{x}) \leq -\lambda R_1(\mathbf{x}) + \mu \|R_2(\mathbf{u})\|_\infty$. Also, when R_1 and R_2 are interpreted as relations or set valued functions, then R_1, R_1^{-1}, R_2 and R_2^{-1} are continuous. Further, since X_0 and U are compact, these functions are uniformly continuous over *States*$(\mathcal{H}_{f,X_0,X,U})$ and *Inputs*$(\mathcal{H}_{f,X_0,X,U})$. Note that the set of reference executions in both the systems is $\{(\mathbf{0}, \mathbf{0})\}$, where $\mathbf{0}$ is of appropriate dimension. It is easy to see that semi-consistency is trivially satisfied. Hence, from Theorem 3 System (7) is incremental input-to-state stable.

9. CONCLUSIONS

In this paper, we investigated pre-orders for reasoning about incremental input-to-state stability properties. We introduced the notion of uniformly continuous input simulations and bisimulations as pre-orders which preserve incremental input-to-state stability of systems. We showed that the notion is a reasonable pre-order to consider by establishing Lyapunov function based analysis of incremental input-to-state stability as a special case of our analysis framework.

In the future, we intend to develop concrete techniques for constructing abstractions based on uniformly continuous input simulations and bisimulations. The notion of refinement as given by the pre-order is a new concept which does not arise in the Lyapunov function based analysis. The notion of refinements (an abstraction ordered between the concrete and the abstract system) will be the guiding principle for developing concrete refinement techniques. Our broad goal is to develop an abstraction refinement technique for analysis of stability properties.

10. ACKNOWLEDGEMENTS

The authors would like to thank the anonymous referees for their feedback in improving the earlier versions of the paper. This work was partially supported by a postdoctoral fellowship from the Center for Mathematics of Information at the California Institute of Technology.

11. REFERENCES

[1] R. Alur, T. Dang, and F. Ivančić. Predicate abstraction for reachability analysis of hybrid systems. *ACM TECS*, 5(1):152–199, February 2006.

[2] D. Angeli. A Lyapunov approach to incremental stability properties. *IEEE TAC*, 47:410–421, 2000.

[3] Jean-Pierre Aubin and Helene Frankowska. *Set-valued Analysis*. Boston : Birkhauser, 1990.

[4] C. Cai and A. R. Teel. Characterizations of input-to-state stability for hybrid systems. *Systems & Control Letters*, 58(1):47–53, 2009.

[5] P. Caspi and A. Benveniste. Toward an Approximation Theory for Computerised Control. In *EMSOFT*, pages 294–304, 2002.

[6] E. M. Clarke, Jr., O. Grumberg, and D. A. Peled. *Model checking*. MIT Press, Cambridge, MA, USA, 1999.

[7] E.M. Clarke, O. Grumberg, S. Jha, Y. Lu, and H. Veith. Counterexample-Guided Abstraction Refinement. In *CAV*, pages 154–169, 2000.

[8] P. J. L. Cuijpers. On Bicontinuous Bisimulation and the Preservation of Stability. In *HSCC*, pages 676–679, 2007.

[9] J. M. Davoren. Epsilon-Tubes and Generalized Skorokhod Metrics for Hybrid Paths Spaces. In *HSCC*, pages 135–149, 2009.

[10] A. Girard, A. A. Julius, and G. J. Pappas. Approximate Simulation Relations for Hybrid Systems. *Discrete Event Dynamic Systems*, 18(2):163–179, 2008.

[11] A. Girard, G. Pola, and P. Tabuada. Approximately bisimilar symbolic models for incrementally stable switched systems. In *HSCC*, pages 201–214, 2008.

[12] R. Goebel, R. Sanfelice, and A. Teel. Hybrid dynamical systems. *IEEE Control Systems, Control Systems Magazine*, 29:28–93, 2009.

[13] W. P. M. H. Heemels and S. Weiland. Input-to-state stability and interconnections of discontinuous dynamical systems. *Automatica*, 44(12):3079–3086, 2008.

[14] T. A. Henzinger. The Theory of Hybrid Automata. In *LICS*, pages 278–292, 1996.

[15] D. K. Kaynar, N. A. Lynch, R. Segala, and F. W. Vaandrager. Timed I/O Automata: A Mathematical Framework for Modeling and Analyzing Real-Time Systems. In *IEEE RTSS*, pages 166–177, 2003.

[16] H. K. Khalil. *Nonlinear Systems*. Prentice-Hall, Inc, 1996.

[17] R. Milner. *Communication and Concurrency*. Prentice-Hall, Inc, 1989.

[18] M. A. Müller and D. Liberzon. Input/output-to-state stability and state-norm estimators for switched nonlinear systems. *Automatica*, 48(9):2029–2039, 2012.

[19] P. A. Parrilo. Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization. Technical report, MIT, 2000.

[20] P. Prabhakar, G. E. Dullerud, and M. Viswanathan. Pre-orders for reasoning about stability. In *HSCC*, pages 197–206, 2012.

[21] E. D. Sontag. Input to state stability: Basic concepts and results. In *Nonlinear and Optimal Control Theory*, pages 163–220. Springer, 2006.

[22] E.D. Sontag and Y. Wang. On characterizations of the input-to-state stability property. *Systems & Control Letters*, 24(5):351–359, 1995.

[23] P. Tabuada, A. Balkan, S. Y. Caliskan, Y. Shoukry, and R. Majumdar. Input-output robustness for discrete systems. In *EMSOFT*, pages 217–226, 2012.

[24] A. Tiwari and G. Khanna. Series of Abstractions for Hybrid Automata. In *HSCC*, volume 2289 of *LNCS*, pages 465–478. Springer, March 2002.