Submitted, 2012 Conference on Decision and Control (CDC)
http://www.cds.caltech.edu/~murray/papers/plm12-cdc.html

Bisimulations for Reasoning about Input-to-State Stability Properties of Hybrid Systems

Pavithra Prabhakar
California Institute of Technology
& IMDEA Software Institute
pavithra@caltech.edu

Jun Liu
California Institute of Technology
liu@caltech.edu

Richard M. Murray
California Institute of Technology
murray@caltech.edu

***Abstract*— In this paper, we investigate pre-orders for reasoning about input-to-state stability properties of hybrid systems. We define the notions of *uniformly continuous input simulations* and *bisimulations*, which extend the notions in previous work to include inputs. We show that uniformly continuous input bisimulations preserve incremental input-to-state stability of hybrid systems, and thus provide a basis for constructing abstractions for verification. We show that Lyapunov function based input-to-state stability analysis can be cast in our framework as constructing a simpler one-dimensional system, using a uniformly continuous input simulation, which is input-to-state stable, and thus, inferring the input-to-state stability of the original system.**

I. Introduction

Input-to-state stability is an important component when studying robustness of systems with respect to input. In this paper, we investigate pre-orders, namely, a reflexive, transitive ordering, on systems for reasoning about stability with respect to inputs in the context of verification of input-to-state stability properties of hybrid systems.

Hybrid systems are systems exhibiting mixed discrete-continuous behaviors, and typically arise due to the involvement of digital computers to control physical systems. With the ubiquitous use of embedded processors in safety-critical applications such as aeronautics, automotives, industrial process control, robotics and others, ensuring reliable performance of these systems is essential. Hence, scalable verification of hybrid systems has gained prominence in recent years. However, the problem has remained stubbornly challenging owing to the mixed discrete continuous behaviors.

Scalable verification relies on being able to construct abstractions or "simplications" of a system efficiently, which can then be verified easily. The notions of *simulation* and *bisimulation*, introduced in the context of concurrent processes [16], to study equivalences between processes, have been the basis for designing abstraction and minimization techniques for analysis of a variety of discrete-time properties [15]. Properties expressible in temporal logics, such as, Linear Temporal Logic, Computation Tree Logic and μ-calculus are known to be invariant under bisimulation, in that, if two systems are known to be bisimilar, then either both of them satisfy the property or none of them satisfy the property. Hence, one can reduce the analysis of a system to that of a simpler system which is bisimilar. Similarly, the weaker notion of simulation preserves the property in one direction, that is, if a system $\mathcal{A}$ is simulated by a system $\mathcal{B}$ and $\mathcal{B}$ satisfies the property, then $\mathcal{A}$ satisfies the property. Properties in a safe fragment of the above logics are preserved by simulation in the above sense.

Even in the hybrid setting, bisimulations have been used to design algorithms for analysis of various classes of systems. Some of these classes include *Timed automata* [1], *O-minimal hybrid automata* [14], [4] and *STORMED hybrid systems* [20]. More recently, approximate notions of simulation and bisimulation have been proposed [7], [6] and used in the analysis of reachability and safety properties [8], [18].

However, when one turns to the analysis of stability properties, it has been shown that bisimulations do not suffice. In particular, it was shown in [5] that Lyapunov stability with respect to a set of equilibrium points is not preserved by bisimulations. Hence, additional continuity constraints were imposed on the bisimulation relation to achieve invariance under Lyapunov stability. In [17], it was shown that for stability with respect to a set of *trajectories*, the continuity constraints imposed by [5] do not suffice to preserve stability. Hence, the notion of uniformly continuous bisimulation was introduced, and Lyapunov and asymptotic stability of trajectories were shown to be invariant under this notion. In this paper, we extend these results to the case with inputs.

Current techniques for proving stability of systems is based on estabilishing the existence of Lyapunov functions. Automation of these techniques for stability analysis essentially depends on automating the search for the Lyapunov functions. While the exact charaterization of the existence and form of Lyapunov functions are know in the pure continuous case, for linear and certain classes of non-linear systems, the same is not true even for the linear case in the hybrid setting [13], [3]. This work is aimed towards developing an abstraction refinement framework for analysis of stability properties. And, the task is challenging even for the case of linear hybrid systems. Establishing pre-orders which preserve stability properties is a first step towards constructing simpler systems for the verification of stability properties.

We define a notion of incremental input-to-state stability for hybrid systems along the analogous notion for trajectories as introduced in [2]. A slight deviation is our notion of distance between executions for which we use the notion of graphical distance introduced in [9]. However, our results are not sensitive to the particular definition of distance. We introduce the notion of *uniformly continuous input simulations* and *bisimulations* which extend the classical notion of bisimulations for systems with input, and the uniform continuity constraints introduced in [17]. We show that incremental input-to-state stability is invariant under uniformly continuous input bisimulation. We also obtain that input-to-state stability is invariant under uniformly continuous input bisimulations as a special case of our results.

We examine whether the notion introduced is a reasonable pre-order for reasoning about input-to-state stability properties. In particular, we ask whether we can hope to construct simpler systems which are related to the original system by uniformly continuous input simulations/bisimulations, and show that the simplification is input-to-state stable. In order to justify the claim, we show that Lyapunov function based analysis of input-to-state stability can be cast as constructing simpler one-dimensional systems which are input-to-state stable, where the Lyapunov function serves as a uniformly continuous input simulation between the original system and the one-dimensional system.

II. Preliminaries

a) 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 $\sup_{a \in A} |f(a)|$.

b) 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 $(x, y) \in R$.

c) 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 domain of a sequence 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 $\lim_{N \to \infty} \sum_{i=0}^{N} \sigma(i) = r$.

d) 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,

1) (Identity of indiscernibles) $d(m_1, m_2) = 0$ if and only if $m_1 = m_2$.
2) (Symmetry) $d(m_1, m_2) = d(m_2, m_1)$.
3) (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.

Let us fix an extended metric space (M, d) for the rest of this section. 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 $X \subseteq \bigcup_{i \in J} U_i$.

e) 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 [11], 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 $R(x) = \{y \,|\, (x, y) \in R\}$. Further, $F^{-1} : B \rightsquigarrow A$ will denote the set valued function which maps $b \in B$ to the set $\{a \in A \,|\, b \in F(a)\}$.

f) 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 "δ, ϵ - definition" of continuity for single valued functions [11]. The function $F : A \rightsquigarrow B$ is said to be *upper semi-continuous* at $a \in Dom(F)$ 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 $a \in Dom(F)$ 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

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

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

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.

Next, we state some properties about upper semi-continuous and uniformly continuous functions.

Proposition 1: Let $F : A \rightsquigarrow B$ be a set-valued upper semi-continuous function. Then:

- F^{-1} is also an upper semi-continuous function.
- If A is compact, then F is also uniformly continuous.

g) 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 $\alpha(0) = 0$. It is said to belong to *class* $\mathcal{K}_\infty$ if $a = \infty$ 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 *class* $\mathcal{KL}$ function is a class $\mathcal{K}$ function with respect to the first argument and class L with respect to the second argument.

III. Hybrid Systems with Input

In this section, we present a general formalism for representing hybrid systems with inputs, called *hybrid input transition system*. 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 [10]. However, our abstract model captures the behaviors arising from a hybrid automaton model.

A. Trajectories

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

Given a relation $R \subseteq A_1 \times A_2$ and trajectories $\mathbf{a}_1 \in$ *Traj*(A_1) and $\mathbf{a}_2 \in$ *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 $Dom(\mathbf{a}_1) = Dom(\mathbf{a}_2)$ and for every $t \in Dom(\mathbf{a}_1)$, $R(\mathbf{a}_1(t), \mathbf{a}_2(t))$. We use $R(\mathbf{a}_1)$ to denote the set $\{\mathbf{a}_2 \,|\, R(\mathbf{a}_1, \mathbf{a}_2)\}$.

A 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 *Traj*$(U) \times$ *Traj*(S) such that $Dom(\mathbf{u}) = Dom(\mathbf{s})$. We call $\mathbf{u}$ an *input trajectory* and $\mathbf{s}$ a *state trajectory*. We will use *ISTraj*(U, S) to denote the set of all input-state trajectories over (U, S). We extend *Size* to input-state trajectories in the natural way, namely, *Size*$(\mathbf{u}, \mathbf{s}) =$ *Size*$(\mathbf{u}) =$ *Size*$(\mathbf{s})$.

We use *First*$((\mathbf{u}, \mathbf{s}))$ to denote the initial state, that is, $\mathbf{s}(0)$, and if *Size*$(\mathbf{s})$ is defined, then we use *Last*$((\mathbf{u}, \mathbf{s}))$ to denote $\mathbf{s}($*Size*$(\mathbf{s}))$. Given a state trajectory $\mathbf{s}$, we use *States*$(\mathbf{s})$ to denote the set of states occuring in $\mathbf{s}$, namely, $\{\mathbf{s}(t) \,|\, t \in Dom(\mathbf{s})\}$. Also, for a input-state trajectory we use *States*$((\mathbf{u}, \mathbf{s}))$ to denote *States*$(\mathbf{s})$. Similarly, for an input trajectory $\mathbf{u}$, we use *Inputs*$(\mathbf{u})$ to denote the set of inputs occuring in $\mathbf{u}$, namely, $\{\mathbf{u}(t) \,|\, t \in Dom(\mathbf{s})\}$.

B. 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 use 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 *Trans*(U, S).

We define *Size* on a transition $(u, (s_1, s_2))$, on a element $u \in U$ and on a pair of states (s_1, s_2) to be 0. As before, given $\tau = (u, (s_1, s_2))$, we use *First*(τ) and *Last*(τ) 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.

C. 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*(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 $\sigma : SeqDom \to \Sigma \cup \Delta$ such that for each $0 \le i < |\sigma| - 1$, *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 executions in $\mathcal{T}$ whose state signals can result from application of the input signal σ^u. Formally, $\mathcal{T}|_{\sigma^u} = \{\sigma^s \,|\, (\sigma^u, \sigma^s) \in \mathcal{T}\}$.

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*(σ) 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*(Δ) and *Inputs*$(\mathcal{H})$ denote *Inputs*$(\Sigma) \cup$ *Inputs*(Δ).

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

- $t \in [T_j, T_j + Size(\sigma(j))]]$.
- If $\sigma(j)$ is a trajectory, then $i = K_j$ and $x = \sigma(j)(t - T_j)$.
- If $\sigma(j)$ is not a trajectory, then
  - if σ is a state signal and $\sigma(j) = (x_1, x_2)$, then either $i = K_j$ and $x = x_1$, or $i = K_j + 1$ and $x = x_2$.
  - if σ is an input signal and $\sigma(i) = u$, then $i = K_j$ and $x = u$.

D. 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 $(\mathcal{H}, d_1, d_2)$ where $\mathcal{H} = (S, U, \Sigma, \Delta)$ is a hybrid input transition system, and (S, d_1) and (U, d_2) are extended metric spaces. The metric d_1 on the state space can be lifted to state signals executions and d_2 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 $(\mathcal{H}, d_1, d_2)$ 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 $d_1(\sigma_1^s, \sigma_2^s)$, is defined as $d_1(gr(\sigma_1^s), gr(\sigma_2^s))$, and the *distance between input signals* σ_1^u, σ_2^u, denoted $d_2(\sigma_1^u, \sigma_2^u)$, is defined as $d_2(gr(\sigma_1^u), gr(\sigma_2^u))$.

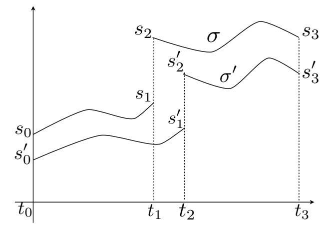

Fig. 1. Graphical Distance between Executions.

Distance between execution 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 [9], 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.

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 $d(gr(\sigma_1)|_T, gr(\sigma_2)|_T)$.

IV. 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. 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 couple of input trajectories u_1, u_2, the following is true:

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

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 every $\epsilon > 0$, there exists a $T \geq 0$, such that for every ζ_1, ζ_2 and 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 every $\epsilon > 0$, there exists a $\delta > 0$ such that for all input signals $\mathbf{u}_1, \mathbf{u}_2$ and 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.

In fact, it is straightforward to check that the conditions $C1-C3$ implies incremental input-to-state stability as given in the above definition.

Proposition 2: A set of input-state trajectories $\mathcal{T}$ is *δISS* iff it satisfies Conditions $(C1) - (C3)$.

Next, we formalize the definition of incremental input-to-state stability for hybrid input transition system using the above observation. A slight deviation is our definition of distance between trajectories, for which we use the graphical distance introduced in [9] for hybrid trajectories. However, the results in the paper are not sensitive to the particular definition of distance, in that, the results hold even when one considers the distance between two executions to be the supremum of the pointwise distance between states and inputs. We define $\mathit{Valid}(\mathcal{T}) = \{(\sigma^u, \zeta) \mid \exists \sigma^s, \mathit{First}(\sigma^s) = \zeta, (\sigma^u, \sigma^s) \in \mathcal{T}\}$. And $\mathit{InSig}(\mathcal{T}) = \{\sigma^u \mid \exists \sigma^s, (\sigma^u, \sigma^s) \in \mathcal{T}\}$.

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:

$$\forall (\sigma^u, \sigma^s) \in \mathit{Exec}(\mathcal{H}), d_1(\mathit{First}(\sigma^s), \mathit{First}(\mathcal{T}|_{\sigma^u})) < \delta$$
$$\Rightarrow \exists (\sigma^u, \hat{\sigma}^s) \in \mathcal{T}, d_1(\sigma^s, \hat{\sigma}^s) < \epsilon$$

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

$$\forall (\sigma^u, \sigma^s) \in \mathit{Exec}(\mathcal{H}), d_1(\mathit{First}(\sigma^s), \mathit{First}(\mathcal{T}|_{\sigma^u})) < \delta \Rightarrow$$
$$\exists (\sigma^u, \hat{\sigma}^s) \in \mathcal{T}, \forall \epsilon > 0, \forall t \geq T(\epsilon), d_1(\sigma^s|_t, \hat{\sigma}^s|_t) < \epsilon.$$

$(D3)$ 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:

$$\forall \hat{\sigma}^u, [d_2(\sigma^u, \hat{\sigma}^u) < \delta \Rightarrow \forall (\hat{\sigma}^u, \hat{\sigma}^s) \in \mathit{Exec}(\mathcal{H}),$$
$$[\mathit{First}(\hat{\sigma}^s) = \zeta \Rightarrow \exists (\sigma^u, \sigma^s) \in \mathcal{T},$$
$$\mathit{First}(\sigma^s) = \zeta, d_1(\sigma^s, \hat{\sigma}^s) < \epsilon]]$$

V. Input (Bi)-Simulations

In this section, we define the notion of pre-order under which, we will show in the next section, *δISS* is invariant.

First, we define the notion of input (bi)-simulation, which 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 [12].

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 a *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 $\mathcal{H}_2 \preceq_{(R_1^{-1}, R_2^{-1})} \mathcal{H}_1$.

We can show that input bisimulation does not preserve incremental input-to-state stability of systems. Essentially, one can use the counter-example from [17] with input space $U = \{0\}$. Hence, we strengthen the pre-order with uniform continuity conditions.

A. Uniformly Continuous Input (Bi)-Simulation

We will assume $\mathcal{H}_1$ and $\mathcal{H}_2$ are metric input 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 $\mathcal{H}_1 \preceq^C_{(R_1, R_2)} \mathcal{H}_2$.

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

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', then define $(R_1, R_2)(\mathcal{H})$ to be the hybrid input transition system $(S', U', \Sigma', \Delta')$, where:

- $\Sigma' = \{(u', (s'_1, s'_2)) \mid \exists (u, (s_1, s_2)) \in \Sigma, u' \in R_2(u), s'_1 \in R_2(s_1), s'_2 \in R_1(s_2)\}$.
- $\Delta' = \{(\mathbf{u}', \mathbf{s}') \mid \exists (\mathbf{u}, \mathbf{s}) \in \Delta, \mathbf{u}' \in R_2(\mathbf{u}), \mathbf{s}' \in R_1(\mathbf{s})\}$.

Proposition 3: Let $\mathcal{H} = (S, U, \Sigma, \Delta)$ be a hybrid input transition system. Let (R_1, R_2) be a pair with $R_1 \subseteq S \times S'$ and $R_2 \subseteq U \times U'$. Then we have the following:

1) (R_1, R_2) is an input bisimulation between $\mathcal{H}$ and $(R_1, R_2)(\mathcal{H})$.
2) If the relations R_1, R_2, R_1^{-1} and R_2^{-1} are uniformly continuous, then (R_1, R_2) is a uniformly continuous input bisimulation between $\mathcal{H}$ and $(R_1, R_2)(\mathcal{H})$.

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

Theorem 1: Let $(\mathcal{H}, d_1, d_2)$,$(\mathcal{H}', d'_1, d'_2)$ and $(\mathcal{H}'', d''_1, d''_2)$, where $\mathcal{H} = (S, U, \Sigma, \Delta)$, $\mathcal{H}' = (S', U', \Sigma', \Delta')$ and $\mathcal{H}' = (S'', U'', \Sigma'', \Delta'')$, be three metric hybrid transition systems.

- Then, $\mathcal{H} \preceq_{(Id_1, Id_2)} \mathcal{H}$, where $Id_1 = \{(s, s) \mid s \in S\}$ and $Id_1 = \{(u, u) \mid u \in U\}$.
- And also, if $\mathcal{H} \preceq^C_{(R_1, R_2)} \mathcal{H}'$ and $\mathcal{H}' \preceq^C_{(R'_1, R'_2)} \mathcal{H}''$, then $\mathcal{H}' \preceq^C_{(R'_1 \circ R_1, R'_2 \circ R_2)} \mathcal{H}''$, $A \circ B = \{(x, z) \mid \exists (x, y) \in A, (y, z) \in B\}$

VI. 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 $(\sigma_1^u, \zeta_1) \in Valid(\mathcal{T}_1)$, there exists $(\sigma_2^u, \zeta_2) \in Valid(\mathcal{T}_2)$ 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 $\zeta_1 \in R_2^{-1}(First(\sigma_2^s))$ such that $(\sigma_1^u, \zeta_1) \in Valid(\mathcal{T}_1)$, there exists σ_1^s with $First(\sigma_1^s) = \zeta_1$, $R_1(\sigma_1^s, \sigma_2^s)$ and $(\sigma_1^u, \sigma_1^s) \in \mathcal{T}_1$.

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

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

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

$(A6)$ There exists $\delta > 0$ such that for every $x \in B_\delta(First(\mathcal{T}_1))$, 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 2: Let $(\mathcal{H}_1, d_1, d_2)$ and $(\mathcal{H}_2, d'_1, d'_2)$, 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 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 $\mathcal{T}_1$.

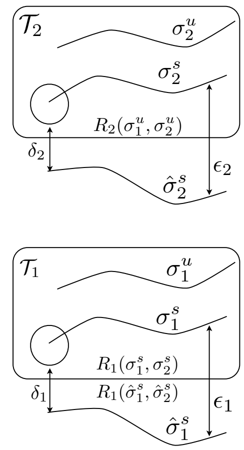

Fig. 2. Illustration for Proof of Condition $(D1)$

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.

Let us fix an input signal σ_1^u. Let $(\sigma_1^u, \sigma_1^s) \in Exec(\mathcal{H}_1)$ such that $d_1(First(\sigma_1^s), First(\mathcal{T}_1|_{\sigma_1^u}) < \delta_1$ (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 $d_1(\sigma_1^s, \hat{\sigma}_1^s) < \epsilon_1$.

Note that Condition $(A1)$ also implies that there exists $\sigma_2^u \in InSig(\mathcal{T}_2)$ 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 $(First(\sigma_1^s), \zeta_2) \in R_1$. Therefore, from input simulation relation, there exists σ_2^s such that $(\sigma_2^u, \sigma_2^s) \in Exec(\mathcal{H}_2)$ 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 $d_1(First(\sigma_1^s), First(\mathcal{T}_1)\sigma_1^u) < \delta_1$, $d_1(R_1(First(\sigma_1^s)), R_1(First(\mathcal{T}_1)|_{\sigma_1^u})) < \delta_1$. From Condition $(A5)$, $d_1(R_1(First(\sigma_1^s)), First(\mathcal{T}_2|_{R_2(\sigma_1^u)})) < \delta_1$, or equivalently $d_1(R_1(First(\sigma_1^s)), First(\mathcal{T}_2|_{\sigma_2^u})) < \delta_1$. In particular, $d_1(First(\sigma_2^s), First(\mathcal{T}_2|_{\sigma_2^u})) < \delta_1$. 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 $d_1(\sigma_2^s, \hat{\sigma}_2^s) < \epsilon_2$. 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, $d_1(\sigma_1^s, \hat{\sigma}_1^s) < \epsilon_1$ since $R_1^{-1}(s)$ is a singleton for every $s \in States(\mathcal{T}_2)$ (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.

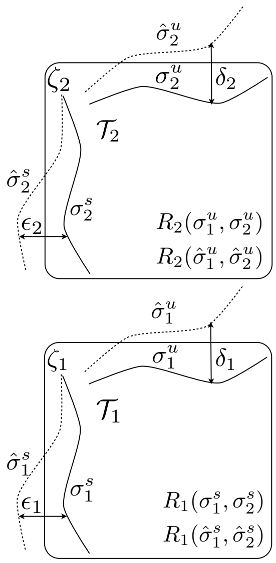

Fig. 3. Illustration for Proof of Condition $(D3)$

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

Let us fix an input signal σ_1^u and state ζ_1 such that $(\sigma_1^u, \zeta_1) \in$ *Valid*$(\mathcal{T}_1)$ (see Figure 3). Let $\hat{\sigma}_1^u$ be such that $d_2(\sigma_1^u, \hat{\sigma}_1^u) < \delta_1$ and let $(\hat{\sigma}_1^u, \hat{\sigma}_1^s) \in$ *Exec*$(\mathcal{H}_1)$ with *First*$(\hat{\sigma}_1^s) = \zeta_1$. We need to show that there exists σ_1^s such that *First*$(\sigma_1^s) = \zeta_1$, $(\sigma_1^u, \sigma_1^s) \in \mathcal{T}_1$ and $d_1(\hat{\sigma}_1^s, \sigma_1^s) < \epsilon_1$.

From Condition $(A1)$ of semi-consistency, we have that there exists $(\sigma_2^u, \zeta_2) \in$ *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 $(\hat{\sigma}_2^u, \hat{\sigma}_2^s) \in$ *Exec*$(\mathcal{H}_2)$ with *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, $d_2(\sigma_1^u, \hat{\sigma}_1^u) < \delta_1$ and $R_2(u)$ is a singleton for every $u \in$ *Inputs*$(\mathcal{T}_1)$ (from Condition $(A3)$) implies that $d_1(\sigma_2^u, \hat{\sigma}_2^u) < \delta_2$. From the definition of δ*ISS* 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 $d_1(\sigma_2^s, \hat{\sigma}_2^s) < \epsilon_2$.

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

Theorem 3: Let $(\mathcal{H}_1, d_1, d_2)$ and $(\mathcal{H}_2, d_1', d_2')$, 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 simulation 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 $\mathcal{T}_1$.

A. Modelling Input-to-State Stability of Continuous Dynamical Systems

We define input-to-state stability of dynamical systems and formulate it in our framework: Consider a continuous dynamical system

$$\dot{x} = f(x, u), \tag{1}$$
$$x \in X \subseteq \mathbb{R}^n, u \in U \subseteq \mathbb{R}^m, x_0 \in X_0 \subseteq X,$$

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 (1) 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 (1) 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_1 and d_2 be the standard Euclidean norms on $\mathbb{R}^n$ and $\mathbb{R}^m$, respectively.

The notion of input-to-state stability captures the notion of "bounded input-bounded state".

Definition. The System (1) is said to be *input-to-state stable* (*ISS*) if there exists a $\mathcal{KL}$ function β, a class $\mathcal{K}$ function γ such that

$$\|x(t)\| \leq \beta(\|x_0\|, t) + \gamma(\|u\|_\infty), \tag{2}$$

for all $t \geq 0, x_0 \in X_0$ and $u \in D_u$.

Let $\mathcal{T}_{0,0}$ be the set of trajectories with 0 input and 0 initial state, that is, $\mathcal{T}_{0,0} = \{(\mathbf{0}, \mathbf{0})\}$. It is easy to see that input-to-state stability of System (1) is equivalent to δ*ISS* of $\mathcal{H}_{f,X_0,X,U}$ with respect to $\mathcal{T}_{0,0}$.

Proposition 4: System (1) is input-to-state stable if and only if the system $\mathcal{H}_{f,X_0,X,U}$ is δ*ISS* with respect to $\mathcal{T}_{0,0}$.

Hence, we can use Theorem 2 and Theorem 3 to reason about input-to-state stability of systems.

VII. Applications of Theorem 2

First, we illustrate through an example of a linear system with inputs, how we can prove input-to-stability using our results.

A. A simple example

Consider a linear system with input, that is,

$$\dot{x} = f(x, u) = Ax + Bu, A \in \mathbb{R}^{n \times n}, B \in \mathbb{R}^{n \times m}, \tag{3}$$
$$x \in X \subseteq \mathbb{R}^n, u \in U \subseteq \mathbb{R}^m, x_0 \in X_0 \subseteq X,$$

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 $R_1(x) = x^T P x$ and a function $R_2 : \mathbb{R}^m \to \mathbb{R}^+$ given by $R_2(u) = |u|$.

Then, $\dot{R_1}(x) = \dot{x}^T P x + x^T P \dot{x} = x^T(A^T P + PA)x + u^T B^T P x + x^T P B u \leq -\lambda R_1(x) + \mu \|u\|_\infty$, where λ and μ are positive constants depending on P, Q and B.

Consider the one-dimensional system:

$$\dot{y} \leq -\lambda y + \mu \|v\|_\infty, y \geq 0. \tag{4}$$

Note that the solutions to the system satisfy $y(t) \leq e^{-\lambda t} y(0) + \mu/\lambda \|v\|_\infty$. This system is trivially input-to-state stable since it is in the form required by Inequality 2.

We will show that (R_1, R_2) is a uniformly continuous input simulation from System (3) to System (4). 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*$(\mathcal{H}_{f,X_0,X,U})$. Hence, from Theorem 2 System (3) is input-to-state stable.

B. Lyapunov Functions for Input-to-State Stability

Next we show that Lyapunov function based input-to-state stability can be cast as constructing simpler one dimensional systems, using uniformly continuous input simulations, which are input-to-state stable.

Let us consider System (1) and 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 (1) 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 4: [19] (*ISS* Theorem) Let $V : X \to \mathbb{R}^+$ be an *ISS* Lyapunov function for the System (1). Then System (1) is input-to-state stable.

Following theorem formulates Lyapunov analysis in our framework:

Theorem 5: Let V be an *ISS* Lyapunov function for System (1), and let $N : \mathbb{R}^n \to \mathbb{R}^+$ be the function $u \mapsto |u|$. Then:

- $(V, N)(\mathcal{H}_{f,X_0,X,U})$ is input bisimilar to $\mathcal{H}_{f,X_0,X,U}$.
- V, V^{-1}, N and N^{-1} are uniformly continuous over *States*$(\mathcal{H})$ and *Inputs*$(\mathcal{H})$.
- (V, N) is consistent with $\mathcal{T}_0$ and $(V, N)(\mathcal{T}_0)$.
- $(V, N)(\mathcal{H}_{f,X_0,X,U})$ is δ*ISS* with respect to $(V, N)(\mathcal{T}_0)$.

Hence $\mathcal{H}_{f,X_0,X,U}$ is δ*ISS* with respect to $\mathcal{T}_0$.

Proof: (Sketch.) Follows from Proposition 3 and Theorem 4. ∎

VIII. Conclusions

In this paper, we investigated pre-orders for reasoning about input-to-state stability properties. We introduced the notion of uniformly continuous input simultaions and bisimulations as pre-orders which preserve 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 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. Our broad goal is to develop an abstraction refinement technique for analysis of stability properties.

References

[1] R. Alur and D. Dill. A theory of timed automata. *Theoretical Computer Science*, 126:183–235, 1994.

[2] David Angeli. A lyapunov approach to incremental stability properties. *IEEE Transactions on Automatic Control*, 47:410–421, 2000.

[3] M. S. Branicky. Stability of hybrid systems: state of the art. In *Conference on Decision and Control*, pages 120–125, 1997.

[4] Thomas Brihaye and Christian Michaux. On the expressiveness and decidability of o-minimal hybrid systems. *Journal of Complexity*, 21(4):447–478, 2005.

[5] Pieter J. L. Cuijpers. On bicontinuous bisimulation and the preservation of stability. In *Proceedings of the International Conference on Hybrid Systems: Computation and Control*, pages 676–679, 2007.

[6] Antoine Girard, A. Agung Julius, and George J. Pappas. Approximate simulation relations for hybrid systems. *Discrete Event Dynamic Systems*, 18(2):163–179, 2008.

[7] Antoine Girard and George J. Pappas. Approximate bisimulation relations for constrained linear systems. *Automatica*, 43(8):1307–1317, 2007.

[8] Antoine Girard, Giordano Pola, and Paulo Tabuada. Approximately bisimilar symbolic models for incrementally stable switched systems. In *Proceedings of the International Conference on Hybrid Systems: Computation and Control*, pages 201–214, 2008.

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

[10] Thomas A. Henzinger. The Theory of Hybrid Automata. In *Logic In Computer Science*, pages 278–292, 1996.

[11] Hlne Frankowska Jean-Pierre Aubin. *Set-valued Analysis*. Boston : Birkhauser, 1990.

[12] Dilsun Kirli Kaynar, Nancy A. Lynch, Roberto Segala, and Frits W. Vaandrager. Timed I/O Automata: A Mathematical Framework for Modeling and Analyzing Real-Time Systems. In *Real-Time Systems Symposium*, pages 166–177. IEEE Computer Society, 2003.

[13] H. K. Khalil. *Nonlinear Systems*. Prentice-Hall, Upper Saddle River, NJ, 1996.

[14] G. Lafferriere, G.J. Pappas, and S. Sastry. O-minimal Hybrid Systems. *Mathematics of Control, Signals, and Systems*, 13(1):1–21, 2000.

[15] David Lee and Mihalis Yannakakis. Online Minimization of Transition Systems (Extended Abstract). In *Symposium on Theory Of Computing*, pages 264–274. ACM, 1992.

[16] Robin Milner. *Communication and Concurrency*. Prentice-Hall, Inc, 1989.

[17] P. Prabhakar, G. Dullerud, and M. Viswanathan. Pre-orders for reasoning about stability. In *Hybrid Systems: Computation and Control (to appear)*, 2012.

[18] P. Prabhakar, V. Vladimerou, M. Viswanathan, and G.E. Dullerud. Verifying tolerant systems using polynomial approximations. In *Proceedings of the IEEE Real Time Systems Symposium*, 2009.

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

[20] V. Vladimerou, P. Prabhakar, M. Viswanathan, and G. E. Dullerud. Stormed hybrid systems. In *ICALP Proceedings*, Reykjavík, 2008.