Foundations for Approximation based Analysis of Stability Properties of Hybrid Systems

Pavithra Prabhakar
IMDEA Software Institute
pavithra.prabhakar@imdea.org

***Abstract*— We discuss pre-orders for reasoning about stability properties of hybrid systems, including Lyapunov stability, asymptotic stability, input-to-state stability and incremental input-to-state stability. We strengthen the classical notion of equivalence between processes, namely, bisimulation with continuity and uniformity conditions and show that the new notions preserve the stability properties for hybrid systems. We demonstrate the usefulness of our results by casting various classical methods for proving stability as constructing simpler systems which are related to the original systems by uniformly continuous notions of pre-orders and proving the simpler systems to be stable.**

I. Introduction

One of the main challenges in automated verification of cyber-physical systems is the scalability of the verification techniques. In order to accomodate verification of large systems, abstractions have been considered as an important tool in reducing the state-space of the systems being analysed [6], [1]. Broadly speaking, in order to verify that a system satisfies a property, an abstract system, which is "simpler", is constructed and checked for satisfaction of the desired property. The abstraction mechanism is such that it "preserves" the property of interest, that is, by verifying the simpler model, one can make some conclusions about the satisfaction of the property by the original model.

The crux of developing such abstraction based techniques is understanding the relationship between the original and the abstract system which preserve the desired properties. In this paper, we study the notions of equivalence and pre-orders between systems which preserve stability properties of cyber-physical systems.

In the context of process algebra, bisimulations have been proposed as the classical notion of process equivalence [21] and various classes of properties, including, safety, Linear Temporal Logic (LTL), μ-calculus, are known to be invariant under this notion, that is, if two processes are bisimilar, then either both satisfy the property or both do not. A weaker notion, namely, that of simulation, preserves properties in one direction, and defines a pre-order on the class of systems. For example, if a system A is simulated by a system B, and B is safe, then it can be concluded that A is safe. However, in general, the converse does not hold.

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* [2], *O-minimal hybrid automata* [20], [4] and *STORMED hybrid systems* [25]. More recently, approximate notions of simulation and bisimulation have been proposed [10], [9] and used in the analysis of reachability and safety properties [11], [22].

It turns out that simulations and bisimulations do not preserve stability properties. Stability is a property of the system which captures the notion that small changes to initial state and/or inputs to the system do not change the resulting behavior of the system drastically. We show that different notions of stability are not invariant under bisimulation, that is, there are systems which are bisimilar, but one of them is stable, and the other is not (according to the same notion of stability). Further, even if we strengthen the notions by adding continuity, stability is still not preserved.

We propose the notions of uniformly continuous (input) simulations and bisimulations, which require that the bisimulation relation (seen as a set valued function) be uniformly continuous. We show that different stability properties such as Lyapunov stability, asymptotic stability, and properties involving input, namely, input-to-state stability and incremental input-to-state stability [3] are preserved by this strengthened notion.

Finally, we ask, if we can hope to build concrete abstraction methods where the relationship between the original and the abstract system is as proposed above. We demonstrate the feasibility of constructing such abstraction methods by showing that some of the classical methods for proving stability proposed in the literature, in fact, provide such concrete methods for abstraction. More precisely, we show that, these methods for proving stability can be cast as constructing simpler systems which are related to the original systems by uniformly continuous pre-orders, and showing that the simpler systems are stable. In particular, we consider Lyapunov's second method for proving stability and its extension to input-to-state stability, and the Hartman-Grobman theorem [13], [14], [15] for proving stability of non-linear systems by linearization.

The results in this paper are based on the results in [23], [24], and is a joint work with Geir Dullerud, Jun Liu, Richard Murray and Mahesh Viswanathan.

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.

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 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 $\lim_{N\to\infty} \sum_{i=0}^{N} \sigma(i) = r$.

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, 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.

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 [17], 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 [17]. 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.

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 [16]. However, our abstract model captures the behaviors arising from a hybrid automaton model.

A. Trajectories

A *trajectory* τ over a set A is a function $\tau : I \to A$, where $I \in PreInt$. 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 $Dom(\tau) = [0, \infty)$.

g) Relating trajectories: 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).

h) Size, First, Last, States, Inputs of Input-State Trajectories: 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 *Last*$((\mathbf{u}, \mathbf{s}))$ to denote the last state, that is, $\mathbf{s}(Size(\mathbf{s}))$, if *Size*$(\mathbf{s})$ is not ∞, and is not defined otherwise. Given a state trajectory $\mathbf{s}$, we use *States*$(\mathbf{s})$ to denote the set of states occurring in $\mathbf{s}$, namely, $\{\mathbf{s}(t) \mid t \in Dom(\mathbf{s})\}$. 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 occurring in $\mathbf{u}$, namely, $\{\mathbf{u}(t) \mid t \in Dom(\mathbf{u})\}$.

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 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).

i) Size, First, Last, States, Inputs of Transitions: 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*(U, S) is a set of transitions and $\Delta \subseteq$ *ISTraj*(U, S) is a set of input-state trajectories.

Next, we define an execution of a hybrid input 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 executions in $\mathcal{T}$ whose state signals can result from application of the input signal σ^u. Formally, $\mathcal{T}|_{\sigma^u} = \{\sigma^s \mid (\sigma^u, \sigma^s) \in \mathcal{T}\}$.

j) 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*(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).

k) 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 $(\mathcal{H}, d^s, d^u)$ 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

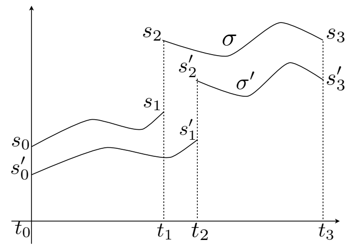

Fig. 1. Graphical Distance between Executions.

used to define various notions of stability. 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 as $d^s(gr(\sigma_1^s), gr(\sigma_2^s))$, 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 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 [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.

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

E. Modeling systems without input

Systems without input can be specified using Hybrid Input Transition Systems by considering a singleton input alphabet $U = \{u\}$. We call such systems Hybrid Transition Systems (HTS). We often represent such systems by dropping the components corresponding to U, that is, a hybrid input transition system $\mathcal{H} = (S, U, \Sigma, \Delta)$ where $U = \{u\}$, is also represented by a hybrid transition system $\mathcal{H}' = (S, \Sigma', \Delta')$, where $\Sigma' = \{(s_1, s_2) \,|\, (u, (s_1, s_2)) \in \Sigma\}$ and $\Delta' = \{\mathbf{s} \,|\, (\mathbf{u}, \mathbf{s}) \in \Delta\}$. And all concepts defined for entities with input is extended to that without input in the natural way.

IV. Stability Properties

In this section, we introduce various properties related to the stability of systems (a good introductory book is [19]). Intuitively, stability is a property that requires that a system when started close to the ideal starting state or with the ideal control input, behaves in a manner that is close to its ideal, desired behavior. We first define notions of stability for closed systems, and then we define notions of stability involving inputs.

A. Lyapunov Stability

We first define the notion of Lyapunov stability. Given a *HTS* $\mathcal{H}$ and a set of executions $\mathcal{T} \subseteq \textit{Exec}(\mathcal{H})$, we say that $\mathcal{H}$ is *Lyapunov stable* (*LS*) with respect to $\mathcal{T}$, if for every $\epsilon > 0$ in $\mathbb{R}^+$, there exists a $\delta > 0$ in $\mathbb{R}^+$ such that the following condition holds:

$$\forall \sigma \in \textit{Exec}(\mathcal{H}), d^s(\textit{First}(\sigma(0)), \textit{First}(\mathcal{T})) < \delta \implies$$

$$\exists \rho \in \mathcal{T}, d^s(\sigma, \rho) < \epsilon. \tag{1}$$

The above statement says that for every execution σ of the system $\mathcal{H}$ which starts with in a distance δ of some execution ρ' in $\mathcal{T}$, there exists an execution ρ in $\mathcal{T}$ which is with in distance ϵ from σ.

B. Asymptotic Stability

Next we define a stronger notion of stability called asymptotic stability which in addition to Lyapunov stability requires that the executions starting close also converge as time goes to infinity. A *HTS* $\mathcal{H}$ is said to be *asymptotically stable* (*AS*) with respect to a set of execution $\mathcal{T} \subseteq \textit{Exec}(\mathcal{H})$, if it is Lyapunov stable and there exists a $\delta > 0$ in $\mathbb{R}^+$ such that

$$\forall \sigma \in \textit{Exec}(\mathcal{H}), d^s(\textit{First}(\sigma(0)), \textit{First}(\mathcal{T})) < \delta \implies$$

$$\exists \rho \in \mathcal{T}, \forall \epsilon > 0, \exists T \geq 0, d^s(\sigma|_T, \rho|_T) < \epsilon. \tag{2}$$

So a system $\mathcal{H}$ is asymptotically stable with respect to a set of its executions $\mathcal{T}$ if $\mathcal{H}$ is Lyapunov stable with respect to $\mathcal{T}$ and every execution starting within a distance of δ from the starting point of some execution in $\mathcal{T}$ converges to some execution in $\mathcal{T}$.

C. Incremental Input-to-State Stability

Next, we define a notion of stability with respect to input, namely, incremental input-to-state stability of hybrid input transition systems. Our definition is motivated by the definition of incremental input-to-state stability of [3], and an equivalent characterization from [24].

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:

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

$(D2)$ 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 The notions of stability with respect to an equilibrium point are a special case of the notions of stability with respect to trajectories, as an equilibrium point has the property that the only trajectory from the equilibrium point is one that stays there.

V. Pre-orders

In this section, we define pre-orders under which the stability properties defined in Section IV are invariant. We first recall standard pre-orders from the literature.

A. Simulations and Bisimulations

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 [18].

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 $\mathcal{H}_1 \preceq_{(R_1,R_2)} \mathcal{H}_2$.

Definition. A pair of relations (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 $\mathcal{H}_2 \preceq_{(R_1^{-1},R_2^{-1})} \mathcal{H}_1$.

Notation For systems without input, we consider only the relations between states with the interpretation that the relation R_2 is an identity relation on the unique input $\{u\}$. In this case, we refer to the pre-orders as simply simulations or bisimulations.

B. Insufficiency of the pre-orders

Bisimulation is the canonical notion of congruence for processes. However, stability is not bisimulation invariant. This was first observed by Cuijpers in [7]. The stability requirement suggests that continuity requirements must be imposed on the witnessing bisimulation (or simulation) relation. Cuijpers considered the problem of Lyapunov stability of an equilibrium state x_*, which informally requires that if the system is started close to x_* then it stays close to x_* at all times. He showed that if a system T_1 with equilibrium point x_* is simulated by T_2 with equilibrium point y_* by a relation R that relates x_* and y_*, is upper semi-continuous, and R^{-1} is lower semi-continuous [17], and if T_2 is Lyapunov stable near y_* then T_1 is Lyapunov stable near x_*. Cuijpers' result, unfortunately, does not extend when one considers stronger notions of stability, like asymptotic stability, or the (Lyapunov or asymptotic) stability of trajectories [1].

1) Asymptotic Stability of Trajectories: Consider a standard dynamical system[2] D_1 that has two state variables x, y taking values in $\mathbb{R}$, with the set of initial states being $\{(0, y) \mid y \in \mathbb{R}^+\}$. The execution map of D_1 is the function $f((x, y), t)$ which prescribes the state at time t provided the state at time 0 was (x, y); specifically, here $f((0, y), t) = (t, y)$. Observe that such a system is stable with respect to the trajectory $\tau = [t \mapsto (t, 0)]_{t \in \mathbb{R}_{\geq 0}}$, as executions that start close to $(0, 0)$ remain close to τ at all times. (see Fig. 2). Let us consider another dynamical system D_2 (shown in Fig. 3) that has the same state space, and same initial states, but whose execution map is $g((0, y), t) = (t, y(1 + t))$. Observe that this system is not stable with respect to its trajectory τ, because no matter how close an initial condition

[1]Stability near an equilibrium point is the special case of stability of a trajectory, as the only trajectory from the equilibrium point stays at the equilibrium point.

[2]A standard dynamical system, in this paper, refers to a hybrid system without any discrete transitions.

Fig. 2. A Lyapunov stable HTS

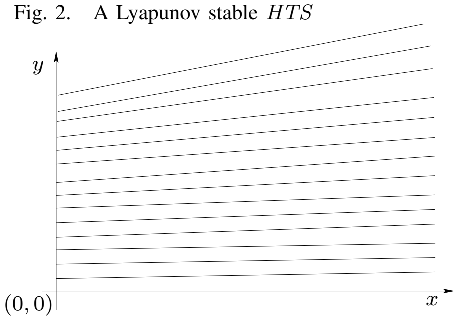

Fig. 3. An unstable HTS

$(0, y_0)$ is to the origin, the resulting execution $g((0, y_0), t)$ will diverge from τ. On the other hand, the relation $R = \{((x_1, y_1), (x_2, y_2)) \mid x_2 = x_1 \text{ and } y_2 = y_1(1 + x_1)\}$ is a bisimulation between the systems D_1 and D_2. Observe that R is bi-continuous and hence R is the kind of bisimulation considered by Cuijpers. This also shows that strengthening the pre-orders with continuity also does not preserve Lyapunov stability.

m) Asymptotic Stability of Trajectories: Next let us consider a dynamical system D_3 which is similar to D_2 except that $g((0, y), t) = ye^{-t}$. Note that D_3 is asymptotically stable. Then the relation R' between D_1 and D_3 given by $\{((x_1, y_1), (x_2, y_2)) \mid x_1 = x_2 \text{ and } y_2 = y_1 e^{-x_1}\}$ is a bi-continuous bisimulation between D_1 and D_3, however, D_3 is asymptotically stable, where as D_1 is not. So the continuity conditions in [7] on bisimulation relations, do not suffice to reason about asymptotic stability of trajectories. In fact, they do not suffice even to reason about asymptotic stability with respect to a set of points.

n) Incremental input-to-state stability: The counter-example corresponding to insufficiency of uniformly continuous bisimulations for preserving asymptotic stability can be extended to show the insufficiency of uniformly continuous input bisimulations for preserving incremental input-to-state stability. Essentially, one can use the counter-example with input space $U = \{u\}$.

C. Uniformly Continuous Input (Bi)-Simulation

Let $(\mathcal{H}_1, d_1^s, d_1^u)$ and $(\mathcal{H}_2, d_2^s, d_2^u)$ be two metric hybrid 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 $\mathcal{H}_1 \preceq^C_{(R_1,R_2)} \mathcal{H}_2$, and $\mathcal{H}_1 \preceq^C \mathcal{H}_2$ to denote that there exists (R_1, R_2) such that $\mathcal{H}_1 \preceq^C_{(R_1,R_2)} \mathcal{H}_2$. Next, we show that uniformly continuous input simulations define a pre-order on systems.

Theorem 1: 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 input 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: (Sketch.) Reflexivity follows from the fact that $\mathcal{H}_1 \preceq_{(Id_1,Id_2)} \mathcal{H}_2$, where $Id_1 = \{(s, s) \mid s \in S\}$ and $Id_1 = \{(u, u) \mid u \in U\}$. 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 $A \circ B = \{(x, z) \mid \exists (x, y) \in A, (y, z) \in B\}$ (since composition of continuous relations is continuous). ∎

Again, for systems without input, we refer to the pre-orders as uniformly continuous simulations and bisimulations.

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.

VI. Stability Preservation Theorems

A. Lyapunov and Asymptotic Stability Preservation

The main result of this section is that uniformly continuous simulations serve as the right foundation for abstractions when verifying Lyapunov and Asymptotic stability properties. That is, we will show that if $\mathcal{H}_1$ is uniformly continuously simulated by $\mathcal{H}_2$ and $\mathcal{H}_2$ is stable with respect to $\mathcal{T}_2$ then $\mathcal{H}_1$ will be stable with respect to $\mathcal{T}_1$. However, for such an observation to hold, the simulation relation between H_1 and H_2 should also relate the executions $\mathcal{T}_1$ and $\mathcal{T}_2$. So before proving the main result of this section, we first formally define how the simulation relation should relate the sets $\mathcal{T}_1$ and $\mathcal{T}_2$.

Definition. Given $HITS$s $\mathcal{H}_1$ and $\mathcal{H}_2$, and sets of executions $\mathcal{T}_1 \subseteq Exec(\mathcal{H}_1)$ and $\mathcal{T}_2 \subseteq Exec(\mathcal{H}_2)$, a binary relation $R \subseteq S_1 \times S_2$ is said to be *semi-complete* with respect to $\mathcal{T}_1$ and $\mathcal{T}_2$ if the following hold:

- $R(First(\mathcal{T}_1)) = First(\mathcal{T}_2)$.
- For every $\rho_2 \in \mathcal{T}_2$, there is an execution in $\rho_1 \in \mathcal{T}_1$ such that $\rho_2 \in R(\rho_1)$.
- For every $x \in States(\mathcal{T}_2)$, $R^{-1}(x)$ is a singleton.
- There exists $\delta > 0$ such that for all $x \in B_\delta(First(\mathcal{T}_1))$, there exists a y such that $R(x, y)$.

R is *complete* with respect to $\mathcal{T}_1$ and $\mathcal{T}_2$ if R and R^{-1} are semi-complete with respect to $\mathcal{T}_1$ and $\mathcal{T}_2$.

The next theorem states that uniformly continuous simulations preserve Lyapunov and asymptotic stability.

Theorem 2 (Stability Preservation Theorem): Let $\mathcal{H}_1$ and $\mathcal{H}_2$ be two hybrid transition systems and $\mathcal{T}_1 \subseteq$

$Exec(\mathcal{H}_1)$ and $\mathcal{T}_2 \subseteq Exec(\mathcal{H}_2)$ be two sets of execution. Let $R \subseteq S_1 \times S_2$ be a uniformly continuous simulation from $\mathcal{H}_1$ to $\mathcal{H}_2$, and let R be semi-complete with respect to $\mathcal{T}_1$ and $\mathcal{T}_2$. Then the following hold:

1) If $\mathcal{H}_2$ is Lyapunov stable with respect to $\mathcal{T}_2$ then $\mathcal{H}_1$ is Lyapunov stable with respect to $\mathcal{T}_1$.
2) If $\mathcal{H}_2$ is asymptotically stable with respect to $\mathcal{T}_2$ then $\mathcal{H}_1$ is asymptotically stable with respect to $\mathcal{T}_1$.

The above theorem implies that the stability of a system $\mathcal{H}_1$ can be concluded by analyzing a potentially simpler system $\mathcal{H}_2$ which uniformly continuously simulates $\mathcal{H}_1$.

As a corollary of Theorem 3, we obtain that Lyapunov stability and asymptotic stability are invariant under uniformly continuous bisimulations.

Definition. A *uniformly continuous bisimulation* between two *HITS*s $\mathcal{H}_1$ and $\mathcal{H}_2$ is a binary relation $R \subseteq S_1 \times S_2$ such that R is a uniformly continuous simulation from $\mathcal{H}_1$ to $\mathcal{H}_2$ and R^{-1} is a uniformly continuous simulation from $\mathcal{H}_2$ to $\mathcal{H}_1$.

Corollary 1: Let $\mathcal{H}_1$ and $\mathcal{H}_2$ be two hybrid transition systems and $\mathcal{T}_1 \subseteq Exec(\mathcal{H}_1)$ and $\mathcal{T}_2 \subseteq Exec(\mathcal{H}_2)$ be two sets of execution. Let $R \subseteq S_1 \times S_2$ be a uniformly continuous bisimulation between $\mathcal{H}_1$ and $\mathcal{H}_2$, and let R be complete with respect to $\mathcal{T}_1$ and $\mathcal{T}_2$. Then the following hold:

1) $\mathcal{H}_1$ is Lyapunov stable with respect to $\mathcal{T}_1$ if and only if $\mathcal{H}_2$ is Lyapunov stable with respect to $\mathcal{T}_2$.
2) $\mathcal{H}_1$ is asymptotically stable with respect to $\mathcal{T}_1$ if and only if $\mathcal{H}_2$ is asymptotically stable with respect to $\mathcal{T}_2$.

B. Incremental Input-to-State Stability Preservation

Next, we extend the above results to incremental input-to-state stability.

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 *input 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 *input 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 input 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 $\mathcal{T}_1 \subseteq Exec(\mathcal{H}_1)$ 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 input 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 $\mathcal{T}_1$.

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 simulation from $\mathcal{H}_1$ to $\mathcal{H}_2$, and let (R_1, R_2) be input 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 $\mathcal{T}_1$.

o) Input-to-state stability: Input-to-state stability is a special case of incremental input-to-state stability, where the unique reference trajectory is the one where the state always stays at the origin. The uniformly continuous input simulations and bisimulations preserve input-to-state stability between two systems with the origin as the reference. In other words, Theorems 3 and 4 can be extended to input-to-state stability.

Remark We have chosen to use graphical distance as the notion of distance between 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 point-wise distance between states and inputs, or the Skorokhod metric (see [8], [5] for more details).

VII. Applications

In this section, we argue that various methods used in proving Lyapunov, asymptotic and (incremental) input-to-state stability of systems can be formulated as constructing a simpler system which is related to the original system by a uniformly continuous pre-order and showing that the simpler system is stable.

p) Lyapunov's Second Method: A Lyapunov function maps the state space of a dynamical system to the one-dimensional non-negative reals, and is such that, the value of the image of the state under the function decreases along any trajectory. One can interpret this function as a mapping from the original state-space to a one-dimensional state-space in a compact neighborhood around the origin which is a uniformly continuous bisimulation from the hybrid transition system corresponding to the original dynamical system to its image under the Lyapunov function. Note that the one dimensional system is trivially Lyapunov stable since along any trajectory of this system, the value of the state decreases. Similarly, the Lyapunov function based analysis for switched systems and for analyzing input-to-state stability (see, for example, [19]) can be cast as constructing simpler systems which are uniformly continuously similar and input similar, respectively, to the corresponding one dimensional systems.

q) Hartman-Grobman Theorem: Hartman-Grobman theorem [13], [14], [15] states that there is a homeomorphism between a non-linear system with a hyperbolic equilibrium point and its linearization around that point in a small neighborhood around the equilibrium. One can again interpret linearization as constructing a simpler linear system which is uniformly continuously bisimilar to the non-linear system in a small neighborhood around the equilibrium.

Hence, Lyapunov functions and linearization can be interpreted as concrete methods for constructing simpler systems which are related to the original systems by uniformly continuous pre-orders and hence preserve stability properties.

VIII. Conclusions

In this paper, we investigated pre-orders for reasoning about various stability properties for both closed and open hybrid systems. We introduced the notion of uniformly continuous input simulations and bisimulations as pre-orders which preserve stability of systems. We showed that the notion is a reasonable pre-order to consider by establishing Lyapunov function based analysis and linearization as concrete methods for constructing simpler systems according to the pre-orders introduced here.

In the future, we intend to develop concrete techniques for constructing abstractions based on uniformly continuous pre-orders. Our broad goal is to develop an abstraction refinement technique for analysis of stability properties.

IX. Acknowledgements

The author would like to thank Geir Dullerud, Jun Liu, Richard Murray and Mahesh Viswanathan for various inputs on the paper, and California Institute of Technology Center for Mathematics of Information for a fellowship for conducting part of this work.

References

[1] R. Alur, T. Dang, and F. Ivancic. Counter-Example Guided Predicate Abstraction of Hybrid Systems. In *Tools and Algorithms for the Construction and Analysis of Systems*, pages 208–223, 2003.

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

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

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

[5] Paul Caspi and Albert Benveniste. Toward an approximation theory for computerised control. In Alberto L. Sangiovanni-Vincentelli and Joseph Sifakis, editors, *EMSOFT*, volume 2491 of *Lecture Notes in Computer Science*, pages 294–304. Springer, 2002.

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

[7] Pieter J. L. Cuijpers. On bicontinuous bisimulation and the preservation of stability. In *HSCC*, pages 676–679, 2007.

[8] Jennifer M. Davoren. Epsilon-tubes and generalized skorokhod metrics for hybrid paths spaces. In Rupak Majumdar and Paulo Tabuada, editors, *HSCC*, volume 5469 of *Lecture Notes in Computer Science*, pages 135–149. Springer, 2009.

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

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

[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] D.M. Grobman. Homeomorphism of systems of differential equations. *Doklady Akad. Nauk SSSR*, 128:880–881, 1959.

[14] D.M. Grobman. Topological classification of neighborhoods of a singularity in n-space. *Mat. Sbornik*, 56(98):77–94, 1962.

[15] P. Hartman. A lemma in the theory of structural stability of differential equations. *Proceedings of the American Mathematical Society*, 11:610–620, 1960.

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

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

[18] 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.

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

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

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

[22] 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.

[23] Pavithra Prabhakar, Geir Dullerud, and Mahesh Viswanathan. Pre-orders for reasoning about stability. In *Proceedings of the 15th ACM international conference on Hybrid Systems: Computation and Control*, HSCC '12, pages 197–206, New York, NY, USA, 2012. ACM.

[24] Pavithra Prabhakar, Jun Liu, and Richard Murray. Bisimulations for reasoning about input-to-state stability properties of hybrid systems. Preprint, 2012.

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