Towards an Approximate Conformance Relation for Hybrid I/O Automata

Morteza Mohaqeqi
Department of Information Technology
Uppsala University, Sweden
morteza.mohaqeqi@it.uu.se

Mohammad Reza Mousavi
Centre for Research on Embedded Systems,
School of IT, Halmstad University, Sweden
m.r.mousavi@hh.se

Several notions of conformance have been proposed for checking the behavior of cyber-physical systems against their hybrid systems models. In this paper, we explore the initial idea of a notion of approximate conformance that allows for comparison of both observable discrete actions and (sampled) continuous trajectories. As such, this notion will consolidate two earlier notions, namely the notion of Hybrid Input-Output Conformance (HIOCO) by M. van Osch and the notion of Hybrid Conformance by H. Abbas and G.E. Fainekos. We prove that our proposed notion of conformance satisfies a semi-transitivity property, which makes it suitable for a step-wise proof of conformance or refinement.

1 Introduction

Conformance testing is a practical and rigorous technique for verifying system correctness against a specification. In the context of cyber-physical systems, several notions of conformance have been proposed for testing such systems against their hybrid systems models, of which hybrid input-output conformance (*hioco*) [17] and (τ,ε)-hybrid conformance (*hconf*) [2, 3] are two notable examples. As we noted earlier in [16, 13], these two notions are substantially different in that hconf allows for approximate comparison of system dynamics, while hioco provides direct support for discrete actions and non-deterministic and partial specifications. Hence, as indicated in [16, 13], combining the advantages of these approaches will be beneficial and will lead to a flexible and realistic definition of conformance with the above-mentioned characteristics.

In this paper, we define an approximate notion of conformance that allows for approximate comparison of both (sampled) continuous signals and observable discrete actions. To this end, we take the notion of hconf [2, 3], and interpret discrete actions as special signals with discrete values (0 and ∞). As a first sanity check, we set out to prove that our notion of conformance is a pre-order (i.e., it is reflexive and transitive), making it suitable for step-wise conformance testing and refinement. Reflexivity immediately follows from the definition; transitivity, however, requires a twist in accumulating the conformance bounds (error margins). We formulate the notion of semi-transitivity and prove that our proposed notion is indeed semi-transitive.

We illustrate the definitions using a simple example of a thermostat. The thermostat has two input discrete actions ON and OFF and one output (continuous) variable x denoting the current temperature. The dynamics of the temperature are different depending on the state of the thermostat, which is in turn determined by the discrete input actions. We provide a formalization of an ideal thermostat in hybrid (I/O) automata and analyze the conformance of its implementation with respect to the specification.

The rest of this paper is organized as follows. In Section 2, we present an overview of the related work and position our approach with respect to the state of the art in the literature. In Section 3, we

Kargahi and Trivedi: Verification and Validation of
Cyber-Physical Systems 2016 (V2CPS-16)
EPTCS 232, 2016, pp. 53–64, doi:10.4204/EPTCS.232.8

introduce the preliminary definitions regarding hybrid I/O automata and their traces. In Section 4, we review the notion of hioco. In Section 5, we first review the notion of hconf and show how it can be extended to treat conformance for systems with discrete actions. Additionally, we state and prove it semi-transitivity property. In Section 6, we conclude the paper and present some directions for future research.

2 Related Work

There is a substantial literature on approximate notions of behavioral equivalence and/or pre-order for hybrid systems. We have already mentioned two examples of such notions, namely hybrid input-output conformance (*hioco*) [17] and (τ,ε)-hybrid conformance (*hconf*) [2, 3]. Additionally, there are a number of notions of approximate equivalence / pre-order based on Metrics such as [10, 11, 18]. In [16, 13], we formally relate some of the basic notions of conformance for hybrid systems and their underlying semantic models.

We expect the extension of our notion to the non-deterministic case to be straightforward; in [17] and [3] already non-deterministic models are considered. The extension to richer semantic domains, e.g., with probabilistics or stochastics, are far less trivial. We refer to [12, 19] for related ideas in this direction.

3 Preliminaries

This section introduces the basic definitions and notations used throughout the rest of the paper, including a formal definition of hybrid I/O automata and their associated traces.

3.1 Notation and Basic Definitions

We denote the set of real numbers by $\mathbb{R}$ and the set of non-negative integers by $\mathbb{N}$. Given a tuple $\alpha = (a_1, a_2, \ldots, a_n)$, the ith element of the tuple is denoted by $\gamma_i(\alpha)$, that is, $\gamma_i(\alpha) = a_i$.

For a function f, we denote its domain by $dom(f)$. For a set S, the restriction of f to S, denoted by $f\lceil S$, is a function with the domain $dom(f) \cap S$, where $(f\lceil S)(c) = f(c), \forall c \in S$.

In order to describe the system dynamics, we define and use the following notions of valuation and trajectory.

Definition 1 (Valuation) *Consider a set of real-valued variables V. A valuation of V is a function of type $V \mapsto \mathbb{R}$, which assigns a real number to each variable $v \in V$. The set of all valuations of V is denoted by $Val(V)$.*

Definition 2 (Trajectory) *Let $D \subset \mathbb{R}$ be an interval. A trajectory σ is a function $\sigma : D \to Val(V)$ that maps each element in interval D to a valuation. The set of all trajectories associated to V is denoted by $trajs(V)$.*

Consider a set of variables V and the corresponding set of valuations $Val(V)$. The *restriction* of a valuation $val \in Val(V)$ to a set $V' \subseteq V$, denoted by $val \downarrow V'$, is a valuation $val' \in Val(V')$ where for all $v \in V'$, $val'(v) = val(v)$. Also, the restriction of a trajectory $\sigma : D \to Val(V)$ to a set $V' \subseteq V$, denoted by $\sigma \downarrow V'$, is a trajectory $D \to Val(V')$ where $(\sigma \downarrow V')(t) = \sigma(t) \downarrow V'$ for all $t \in D$.

For a trajectory σ, the first valuation of σ (i.e., $\sigma(min(dom(\sigma)))$), if it exists, is denoted by $\sigma.fval$. Moreover, if $dom(\sigma)$ is right-closed, then $\sigma.lval$ is defined as the last valuation of σ.

For an interval $D \subseteq \mathbb{R}$ and an arbitrary $t \geq 0$, we define $D+t \triangleq \{t'+t | t' \in D\}$ as the time shift of D. Moreover, for a trajectory σ with domain D, $\sigma+t$ is defined as a function with domain $D+t$ such that $\forall t' \in D : (\sigma+t)(t'+t) = \sigma(t')$.

For a trajectory σ, define $\tau \trianglerighteq t \triangleq (\tau\lceil[t,\infty)) - t$. Additionally, trajectory σ is a prefix of trajectory σ', denoted as $\sigma \leq \sigma'$, if and only if $\sigma = \sigma'\lceil dom(\sigma)$. Also, the concatenation of two trajectories σ and σ' is defined by $\sigma^\frown\sigma' \triangleq \sigma \cup \sigma'$.

3.2 Hybrid Automata

Hybrid I/O automata (HIOA) [14] are defined as the extension of hybrid automata [5] with an additional classification of external actions and variables as inputs and outputs. Hence, we first review the definition of hybrid automata and then introduce its extension to input and output actions and variables.

Definition 3 (Hybrid Automata) *A hybrid automaton (HA) $\mathscr{A} = (W, X, Q, \Theta, E, H, D, \mathcal{T})$ consists of*

- *A set W of external variables and a set X of internal variables, disjoint from each other. We write $V \equiv W \cup X$.*
- *A set $Q \subseteq Val(X)$ of states.*
- *A nonempty set $\Theta \subseteq Q$ of start states.*
- *A set E of external actions and a set H of internal actions, disjoint from each other. We write $A \equiv E \cup H$.*
- *A set $D \subseteq Q \times A \times Q$ of discrete transitions. We use $\mathbf{x} \xrightarrow{a}_{\mathscr{A}} \mathbf{x}'$ as shorthand for $(\mathbf{x}, a, \mathbf{x}') \in D$. We sometimes drop the subscript and write $\mathbf{x} \xrightarrow{a} \mathbf{x}'$, when we think $\mathscr{A}$ should be clear from the context. We say that a is enabled in $\mathbf{x}$ if there exists an $\mathbf{x}'$ such that $\mathbf{x} \xrightarrow{a} \mathbf{x}'$.*
- *A set $\mathcal{T}$ of trajectories for V such that $\tau(t)\lceil X \in Q$ for every $\tau \in \mathcal{T}$ and $t \in dom(\tau)$. Given a trajectory $\tau \in \mathcal{T}$ we denote $\tau.fval\lceil X$ by $\tau.fstate$ and, if τ is closed, we denote $\tau.lval\lceil X$ by $\tau.lstate$. We require that the following axioms hold:*

  ***T1** (Prefix closure)*

  For every $\tau \in \mathcal{T}$ and every $\tau' \leq \tau$, $\tau' \in \mathcal{T}$.

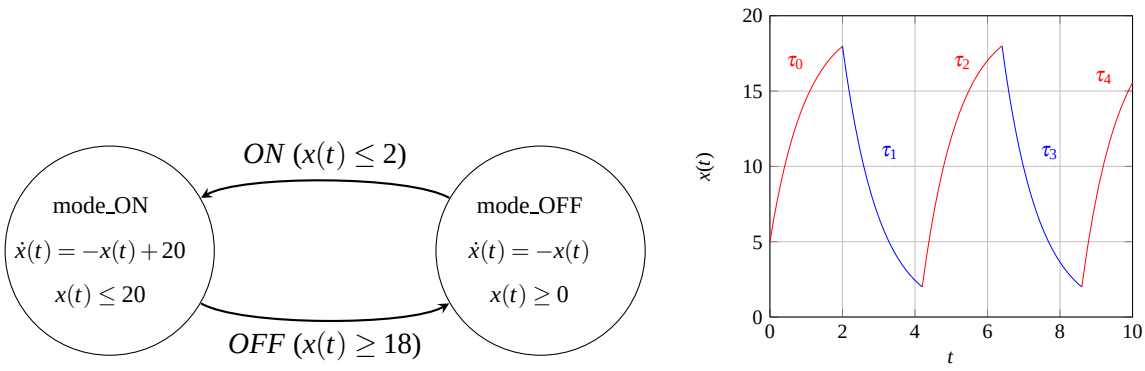
  ***T2** (Suffix closure) For every $\tau \in \mathcal{T}$ and every $t \in dom(\tau)$, $\tau \trianglerighteq t \in \mathcal{T}$*

  ***T3** (Concatenation closure) Let $\tau_0, \tau_1, \tau_2, \ldots$ be a sequence of trajectories in $\mathcal{T}$ such that, for each nonfinal index i, τ_i is closed and $\tau_i.lstate = \tau_{i+1}.fstate$. Then $\tau_0^\frown\tau_1^\frown\tau_2 \ldots \in \mathcal{T}$.*

Example 4 *Figure 1a, due to [4], shows the hybrid automaton of the thermostat described in Section 1 with the set of variables of $W = \{y\}$, $X = \{l, x\}$, where l is a two-valued variable (mode_ON and mode_OFF) determining the location of the automaton, x is a continuous real-valued variable, and y is always equal to x, i.e., $y(t) = x(t), \forall t \geq 0$. Further, the set of actions is specified as $E = \{ON\, OFF\}$, and $H = \{\}$. Also, we have*

$$Q = \{(l, x) \mid l \in \{mode\_ON, mode\_OFF\},\ 0 \leq x \leq 20\}$$

and $\Theta = \{(mode\_ON, 5)\}$. Trajectories generated by this HIOA are specified according to the differential equations given for each location. Figure 1b depicts a set of trajectories for variable x (or equivalently external variable y).



(a) Hybrid automaton of the thermostat

(b) A sample of the continuous dynamics of the system

Figure 1: Thermostat example

3.3 Hybrid I/O Automata

As mentioned before, an HIOA is a hybrid automaton in which the set of variables and actions are classified as input and output, as specified below.

Definition 5 (Hybrid I/O Automata [14]) *A hybrid I/O automaton (HIOA) is a tuple* $(\mathscr{H}, W_I, W_O, E_I, E_O)$ *where*

- $\mathscr{H} = (W, X, Q, \Theta, E, H, D, \mathcal{T})$ *is a hybrid automaton.*
- W_I *and* W_O *partition* W *into input and output variables, respectively. Variables in* $Z \equiv X \cup W_O$ *are called locally controlled.*
- E_I *and* E_O *partition* E *into input and output actions, respectively. Actions in* $L \equiv H \cup E_O$ *are called locally controlled.*
- *The following additional axioms are satisfied:*
  - ***E1*** *(Input action enabling)*
    For every $\mathbf{x} \in Q$ *and every* $a \in E_I$*, there exists* $\mathbf{x}' \in Q$ *such that* $\mathbf{x} \xrightarrow{a} \mathbf{x}'$*.*
  - ***E2*** *(Input trajectory enabling)*
    For every $\mathbf{x} \in Q$ *and every* $v \in trajs(W_I)$*, there exists* $\tau \in \mathcal{T}$ *such that* $\tau.fstate = \mathbf{x}$*,* $\tau \downarrow W_I \leq v$ *and either*
    1. $\tau \downarrow W_I = v$*, or*
    2. τ *is closed and some* $l \in L$ *is enabled in* $\tau.lstate$*.*

Example 6 *Consider the hybrid automaton described in Example 4. Classifying the set of variables into input and output variables* $W_I = \{\}$*,* $W_O = \{y\}$ *and the set of actions into input and output actions as* $E_I = \{ON\}$*,* $E_O = \{OFF\}$ *reveals the corresponding HIOA.*

For an HIOA, a state is said to be *agile* if it affords some trajectory, i.e., there exists trajectory $\sigma \in \mathcal{T}$ for which $s \xrightarrow{\sigma}$. To show that a state is agile, we add a spacial action ξ to the set of output actions E_O (the usage of this action is explained in the next section).

3.4 Hybrid sequences, executions, traces and solution pairs

To describe the behavior of a hybrid (I/O) automaton, we use the notions of execution, trace and solution pair. To this end, we first define a notation of hybrid sequence.

Definition 7 (Hybrid Sequence [14]) *Take a set of variables V and a set of actions A. Then, an (V,A)-sequence is defined as a finite or infinite sequence $\alpha = \tau_0, a_1, \tau_1, a_2, \ldots$, in which*

- *each τ_i denotes a trajectory in trajs(V);*
- *each α_i denotes an action in A;*
- *if α is finite then it ends with a trajectory;*
- *if τ_i is not the last trajectory in the sequence then dom(τ_i) is closed.*

Any (V,A)-sequence is called a hybrid sequence.

Next, we define the notions of execution fragment and execution for a hybrid I/O automaton.

Definition 8 (Execution Fragment and Execution [14]) *A hybrid sequence $\alpha = \tau_0, a_1, \tau_1, a_2, \ldots$ is an execution fragment for an HIOA A if (i) each τ_i is a trajectory in $\mathcal{T}$, and (ii) if τ_i is not the last trajectory then $\tau_i.lstate \xrightarrow{a_{i+1}} \tau_{i+1}.fstate$. For an execution fragment α, we define $\alpha.fstate$ to be $\tau_0.fstate$.*

An execution fragment $\alpha = \tau_0, a_1, \tau_1, a_2, \ldots$ is called an execution *of a hybrid automata $\mathscr{A}$ if $\tau_0.fstate$ is a start state, i.e., $\tau_0.fstate \in \Theta$.*

We write $execs_{\mathscr{A}}$ to denote the set of all executions of $\mathscr{A}$.

To capture the *external* behavior of a hybrid (I/O) automaton, we use the notion of *trace*. Intuitively, a trace of an execution specifies the sequence of its external actions and the evolution of the external variables.

Definition 9 (Trace [14]) *Consider an execution α. The trace of α, denoted by trace(α), is the (E,W)-restriction of α (It is recalled that E and W denote the set of external actions and external variables, respectively).*

Example 10 *Figure 1b depicts a set of trajectories for variable x of the HIOA specified in Example 6. According to this sample behavior, $\alpha = \tau_0, OFF, \tau_1, ON, \tau_2$ specifies a trace of the HIOA.*

Definition 11 (Solution Pair [3]) *Assume that u and y are two traces for a HIOA $\mathcal{H}$; (u, y) is a solution pair to $\mathcal{H}$ if*

- $\mathrm{dom}(u) = \mathrm{dom}(y)$, *and*
- *there exists a trace α to $\mathcal{H}$ such that* $\mathrm{dom}(\alpha) = \mathrm{dom}(u)$, $u = \alpha \downarrow W_I$, *and* $y = \alpha \downarrow W_O$.

4 I/O Conformance for Hybrid I/O Automata

In this section, we review the notion of hioco [17], which is an exact notion of conformance, and adopt it for the HIOA formalism. Subsequently, in the next section, we will adapt the ideas explored in this section to come up with an approximate notion of conformance. To this end, we first define the output and the trajectories associated with a state of an HIOA.

Definition 12 *Assume that $\mathscr{A} = (\mathscr{H}, W_I, W_O, E_I, E_O)$ is a hybrid IO automaton with the sets Q and Θ of states and starting states, respectively. For a state $s \in \Theta$ we define*

$$\mathbf{out}(s) = \begin{cases} \{o \in E_O \mid \{\xi\} \cup s \xrightarrow{o}\} & \text{, if } s \text{ is agile} \\ \{o \in E_O \mid s \xrightarrow{o}\} & \text{, otherwise} \end{cases} \tag{1}$$

Furthermore, for a set $C \subseteq Q$ we define:

$$\mathbf{out}(C) = \bigcup_{s \in C} \mathbf{out}(s).$$

Definition 13 (after operator) *For an HIOA $\mathscr{A}$ and a trace t,* **after** *operator is defined as*

$$\mathscr{A} \textbf{ after } t = \{s \mid s_0 \xrightarrow{t} s\} \tag{2}$$

Definition 14 (Trajectories of a state) *For an HIOA $\mathscr{A}$ and a state s,* $\mathbf{traj}(s)$ *is defined as*

$$\mathbf{traj}(s) = \{\sigma \in \Im \mid s \xrightarrow{\sigma}\} \tag{3}$$

In order to define conformance, we need to focus on those trajectories of the system under test for which a trajectory with the same behavior exists in the specification. This is achieved by the following infilter operator.

Definition 15 (infilter) *Consider two sets of trajectories Σ_I and Σ_S on a set of variables V. Assume $V_I \subseteq V$ is the set of input variables; then*

$$\mathbf{infilter}(\Sigma_I, \Sigma_S) = \{\sigma \in \Sigma_I \mid \exists_{\sigma' \in \Sigma_S} : \sigma \downarrow V_I = \sigma' \downarrow V_I\} \tag{4}$$

Finally, the notion of HIOCO is defined HIOA below.

Definition 16 (extension of hioco) *An (input-enabled) HIOA I is said to hybrid input-output conform to another HIOA S, denoted by I* **hioco** *S, if and only if for all traces $t \in traces(S)$:*

$$\mathbf{out}(I \textbf{ after } t) \subseteq \mathbf{out}(S \textbf{ after } t), \text{and}$$
$$\mathbf{infilter}\,(\mathbf{traj}(I \textbf{ after } t), \mathbf{traj}(S \textbf{ after } t)) \subseteq \mathbf{traj}(S \textbf{ after } t)$$

Example 17 *Assume that we refer to the HIOA specified in Example 6 by S. As specified in Example 10, $\alpha = \tau_0, OFF, \tau_1, ON, \tau_2$ is a trace of S. Then, we have* $\mathbf{out}(S \textbf{ after } \alpha) = \{OFF\}$. *Now, assume an imaginary HIOA I which generates a trace $\alpha' = \tau_0, OFF, \tau_1, ON, \tau_2, ON$. It is seen that $ON \in$* $\mathbf{out}(I \textbf{ after } \alpha)$. *As a consequence, I does not conform to S since* $\mathbf{out}(I \textbf{ after } \alpha) \not\subseteq \mathbf{out}(S \textbf{ after } \alpha)$.

5 Approximate Conformance for Hybrid I/O Automata

As it can be noted in Definition 16, HIOCO considers an implementation to be conforming when it exactly follows (features a non-empty subset of the behavior of) the specification. However, due to (unavoidable) inaccuracies during the implementation and testing of a CPS, this is not a feasible and practical approach. We wish to allow for *limited* deviations of the implementation from the specification. In this regard, Abbas et al. [2, 3] define a notion of closeness which is measures the degree of similarity between the observed behavior of two systems. However, the approach ignores the input and output actions and mainly focuses on the continuous behavior of the system. In the remainder of this section, we first review the hybrid conformance relation proposed by Abbas et al. [2, 3] and then extend it, following the recipe of HIOCO, with discrete input and output actions.

5.1 Conformance Relation based on the Notion of Closeness

In the hybrid conformance approach [2, 3], the behavior of a system is specified using a notion of trace in which the types of observable discrete actions are not explicitly specified. Instead, the number of discrete jumps are considered relevant in the definition of hybrid conformance. This kind of trace, which we call an *action-insensitive trace* (or an *a-trace*) hereafter, is defined based on the notion of hybrid time domain, defined below.

Definition 18 (Hybrid Time Domain [3]) *A hybrid time domain E is a subset of $\mathbb{R}_+ \times \mathbb{N}$ defined as*

$$E = \bigcup_{j=0}^{J-1} [t_j, t_{j+1}] \times \{j\} \tag{5}$$

where $0 = t_0 \leq t_1 \leq t_2 \leq ... \leq t_J$. We denote the set of all hybrid time domains by $\mathbb{T}$.

The hybrid time domain is used to specify the evolution of a CPS regarding both continuous evolution (using time intervals $[t_j, t_{j+1}]$) and discrete jumps (using integer numbers j). Next, we define the notion of action-insensitive trace which is subsequently used to define the approximate notion of conformance.[1]

Definition 19 (Action-Insensitive Trace [3]) *Take a hybrid time domain E and a set of variables V. An action-insensitive trace (a-trace) over E is a function $\phi : E \to Val(V)$, where for every j, $t \mapsto \phi(t, j)$ is absolutely continuous in t over the interval $I_j = \{t | (t, j) \in E\}$. The set of all a-traces defined over the variable set V is denoted by a-trace(V).*

As it can be noted, the notion of a-trace is an abstraction of a trace, as defined in Definition 9. In fact, an a-trace can be obtained from a given trace by removing its actions and keeping only the number of jumps.

Definition 20 ((τ,ε)-closeness [3]) *Consider a test duration $T \in \mathbb{R}_+$, a maximum number of jumps $J \in \mathbb{N}$, and $\tau, \varepsilon > 0$; then two a-traces y_1 and y_2 are said to be (τ,ε)-close, denoted by $y_1 \approx_{(\tau,\varepsilon)} y_2$, if*

1. *for all $(t,i) \in \mathrm{dom}(y_1)$ with $t \leq T, i \leq J$, there exists $(s,j) \in \mathrm{dom}(y_2)$ such that $|t-s| \leq \tau$ and $\|y_1(t,i) - y_2(s,j)\| \leq \varepsilon$, and*
2. *for all $(t,i) \in \mathrm{dom}(y_2)$ with $t \leq T, i \leq J$, there exists $(s,j) \in \mathrm{dom}(y_1)$ such that $|t-s| \leq \tau$ and $\|y_2(t,i) - y_1(s,j)\| \leq \varepsilon$.*

Definition 21 (Conformance Relation [3]) *Consider two hybrid I/O automata $\mathcal{H}_1$ and $\mathcal{H}_2$. Given a test duration $T \in \mathbb{R}_+$, a maximum number of jumps $J \in \mathbb{N}$, and $\tau, \varepsilon > 0$, $\mathcal{H}_2$* conforms to *$\mathcal{H}_1$, denoted by $\mathcal{H}_2 \approx_{(\tau,\varepsilon)} \mathcal{H}_1$, if and only if for all solution pairs (u, y_1) of $\mathcal{H}_1$, there exists a solution pair (u, y_2) of $\mathcal{H}_2$ such that the corresponding output a-traces y_1 and y_2 are (τ,ε)-close.*

Figure 2 shows two a-traces y_1 and y_2 where $y_1 \approx_{(0.8,1)} y_2$. As mentioned, the notion of hconf ($\approx_{(\tau,\varepsilon)}$) is not sensitive to the existence/absence of actions. As a result, if the HIOA generating y_1 produces output action *OFF* at $t = 2$ and the other HIOA produces no action, the two a-traces are still regarded as conforming according to hconf.

[1] In [3], the term *Hybrid Arc* is used to refer to this concept.

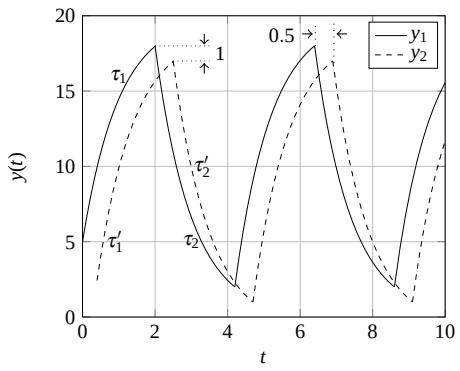


Figure 2: Two signals which are $\approx_{\tau,\varepsilon}$ for $\tau = 0.8$ and $\varepsilon = 1$ but not for $\tau = 0.8$ and $\varepsilon = 0.4$.

5.2 Extension

The goal of this section is to extend the notion of closeness and the corresponding conformance relation to the case of hybrid I/O automata. The main difference of our extended conformance with that of Abbas et al. [2] is that we include the input and output actions for the comparison of two behaviors (i.e., the comparison of specification and implementation).

According to Definition 20, the notion of closeness is specified on the basis of comparing a-traces. However, as mentioned in the previous section, an a-trace is an abstraction of a trace of a HIOA, in which, the actions are not explicitly specified. As a result, a natural way for extending the notion of closeness to HIOA is to augment the closeness definition (Definition 20) with additional conditions that also compare the corresponding actions of the traces.

We extend the definition of a-trace such that it can reflect the occurrence of actions as well as the change of continuous variables. For this purpose, we consider each action as a special continuous variables. This variable is assumed to get only two values: 0 and ∞, where a value of 0 denotes the absence of the corresponding action while the value of ∞ denotes an instantaneous occurrence of the action.

To this end, we first assume that the set of real numbers is augmented with a special number ∞ which is regarded to be larger than any number in $\mathbb{R}$, and hence larger than any arbitrary ε. Further, for notational convenience, we define $\infty - \infty = 0$. Let $\mathbb{R}_\infty$ denote $\mathbb{R}$ extended with ∞.

We extend the notion of trace and solution pair (Definitions 9 and 11) such that for any action in a given HIOA, there is a fresh variable. Such variables get only two values, namely 0 and ∞. This naturally extends the notion of a-trace to our notion of trace which has the same type (apart from the fact that $Val(V)$ ranges over $\mathbb{R}_\infty$); the only difference is that the set of variables also includes the freshly introduced action variables.

In order to extend the notion of closeness (and thus, hconf conformance relation), we need to define the meaning of a norm ($\|\|$) for the signals containing a mixture of normal real-valued variables and these extended type of variables. Assume y_1 and y_2 are two vector of variables of the same length, i.e.,

$$y_1 = (y_1[1], \ldots, y_1[n])$$
$$y_2 = (y_2[1], \ldots, y_2[n])$$

We define the distance of y_1 and y_2, denoted by $\|y_2 - y_1\|_e$, as

$$\|y_2 - y_1\|_e = \begin{cases} \infty, & \exists i \in \{1,\ldots,n\} s.t. (y_1[i] = \infty \textit{ and } y_2[i] \neq \infty) \textit{ or } (y_1[i] \neq \infty \textit{ and } y_2[i] = \infty) \\ \|y_2 - y_1\|, & \text{otherwise} \end{cases} \quad (6)$$

where $\|y_2 - y_1\|$ denotes the ordinary Euclidean distance. Then, we can modify the definition of (τ,ε)-closeness based on this notion of distance.

Definition 22 (Modified (τ,ε)-closeness) *Consider a test duration $T \in \mathbb{R}_+$, a maximum number of jumps $J \in \mathbb{N}$, and $\tau, \varepsilon > 0$; then two a-traces y_1 and y_2 are said to be (τ,ε)-close, denoted by $y_1 \approx^e_{(\tau,\varepsilon)} y_2$, if*

1. *for all $(t,i) \in \mathrm{dom}(y_1)$ with $t \leq T, i \leq J$, there exists $(s,j) \in \mathrm{dom}(y_2)$ such that $|t-s| \leq \tau$ and $\|y_1(t,i) - y_2(s,j)\|_e \leq \varepsilon$, and*
2. *for all $(t,i) \in \mathrm{dom}(y_2)$ with $t \leq T, i \leq J$, there exists $(s,j) \in \mathrm{dom}(y_1)$ such that $|t-s| \leq \tau$ and $\|y_2(t,i) - y_1(s,j)\|_e \leq \varepsilon$.*

Definition 23 (Conformance Relation) *Consider two hybrid I/O automata $\mathcal{H}_1$ and $\mathcal{H}_2$. Given a test duration $T \in \mathbb{R}_+$, a maximum number of jumps $J \in \mathbb{N}$, and $\tau, \varepsilon > 0$, $\mathcal{H}_2$* conforms to *$\mathcal{H}_1$, denoted by $\mathcal{H}_2 \approx^e_{(\tau,\varepsilon)} \mathcal{H}_1$, if and only if for all solution pairs (u,y_1) of $\mathcal{H}_1$, there exists a solution pair (u,y_2) of $\mathcal{H}_2$ such that the corresponding output a-traces y_1 and y_2 are (τ,ε)-close, considering the modified notion of closeness in Definition 22.*

Example 24 *Consider two traces $\alpha = \tau_1, OFF, \tau_2$ and $\alpha' = \tau_1', \tau_2'$ where τ_1, τ_2, τ_1', and τ_2' are trajectories shown in Fig. 2. Let ϕ_1 and ϕ_2 be the corresponding a-traces obtained according to the abovementioned method for encoding the actions as numeric variables. Then, according to the modified notion of closeness, we will have $\phi_1 \not\approx^e_{(\tau,\varepsilon)} \phi_2$.*

5.3 Step-wise Approximate Refinement

In order to apply our notion of conformance in a step-wise refinement trajectory, one needs to compose approximate conformance relations. This can be achieved through the following notion of semi-transitivity.

Definition 25 (Semi-Transitivity) *Let $\mathcal{H}_1$, $\mathcal{H}_2$, and $\mathcal{H}_3$ be three arbitrary hybrid I/O automata such that $\mathcal{H}_2 \approx_{(\tau_1,\varepsilon_1)} \mathcal{H}_1$ and $\mathcal{H}_3 \approx_{(\tau_2,\varepsilon_2)} \mathcal{H}_2$ for some $\tau_1, \tau_2, \varepsilon_1, \varepsilon_2 > 0$. Then, the conformance relation is* semi-transitive *if $\mathcal{H}_3 \approx_{(\tau_1+\tau_2,\varepsilon_1+\varepsilon_2)} \mathcal{H}_1$.*

Theorem 26 *The extended conformance relation is semi-transitive.*

Proof For the proof, we first formally specify the assumptions and the thesis of the theorem; the latter is further refined and proven in a number of steps:

1. Assumption: $\mathcal{H}_1 \approx^e_{(\tau_1,\varepsilon_1)} \mathcal{H}_2$ and $\mathcal{H}_2 \approx^e_{(\tau_2,\varepsilon_2)} \mathcal{H}_3$.
   Claim: $\mathcal{H}_1 \approx^e_{(\tau_1+\tau_2,\varepsilon_1+\varepsilon_2)} \mathcal{H}_3$.
2. Assumption Rewriting (according to Definition 22):

$$\forall (u,y_1) \in \mathcal{H}_1 : \exists (u,y_2) \in \mathcal{H}_2 \text{ such that } y_1 \approx^e_{(\tau_1,\varepsilon_1)} y_2, \tag{7}$$
$$\forall (u,y_2) \in \mathcal{H}_2 : \exists (u,y_3) \in \mathcal{H}_3 \text{ such that } y_2 \approx^e_{(\tau_2,\varepsilon_2)} y_3. \tag{8}$$

   Claim: $\forall (u,y_1) \in \mathcal{H}_1 : \exists (u,y_3) \in \mathcal{H}_3$ such that $y_1 \approx^e_{(\tau_1+\tau_2,\varepsilon_1+\varepsilon_2)} y_3$.
3. Assumption Rewriting:

$$\begin{aligned}&\forall (u,y_1) \in \mathcal{H}_1 : \exists (u,y_2) \in \mathcal{H}_2 \text{ such that}\\ &\quad \forall (t,i) \in \mathrm{dom}(y_1) : \exists (s,j) \in \mathrm{dom}(y_2) \text{ such that } |t-s| \leq \tau_1 \text{ and } \|y_1(t,i) - y_2(s,j)\|_e \leq \varepsilon_1,\end{aligned} \tag{9}$$

and also

$$\begin{aligned}&\forall(u,y_2)\in\mathcal{H}_2:\exists(u,y_3)\in\mathcal{H}_3 \text{ such that}\\&\forall(s,j)\in\mathrm{dom}(y_2):\exists(v,k)\in\mathrm{dom}(y_3) \text{ such that } |v-s|\leq\tau_2 \text{ and } \|y_2(s,j)-y_3(v,k)\|_e\leq\varepsilon_2,\end{aligned}\quad(10)$$

conditioned on that $t\leq T$ and $i\leq J$.
Claim:

$$\begin{aligned}&\forall(u,y_1)\in\mathcal{H}_1:\exists(u,y_3)\in\mathcal{H}_3 \text{ such that}\\&\forall(t,i)\in\mathrm{dom}(y_1):\exists(v,k)\in\mathrm{dom}(y_3) \text{ such that } |v-t|\leq\tau_1+\tau_2 \text{ and } \|y_1(t,i)-y_3(v,k)\|_e\leq\varepsilon_1+\varepsilon_2.\end{aligned}\quad(11)$$

Now, we proceed with the proof. Fix an arbitrary $(u,y_1)\in\mathcal{H}_1$; it follows from (9) that $\exists y_2$ such that $(u,y_2)\in\mathcal{H}_2$ for which $\forall(t,i)\in\mathrm{dom}(y_1)$ there exists $(s,j)\in\mathrm{dom}(y_2)$ such that

$$|t-s|\leq\tau_1 \quad(12)$$

and

$$\|y_1(t,i)-y_2(s,j)\|_e\leq\varepsilon_1 \quad(13)$$

Similarly, from (10), it follows that considering y_2, $\exists y_3$ such that $(u,y_3)\in\mathcal{H}_3$ for which $\forall(s,j)\in\mathrm{dom}(y_2)$ there exists $(v,k)\in\mathrm{dom}(y_3)$ such that

$$|s-v|\leq\tau_2 \quad(14)$$

and

$$\|y_2(s,j)-y_3(v,k)\|_e\leq\varepsilon_2. \quad(15)$$

Now, fix an arbitrary $\forall(t,i)\in\mathrm{dom}(y_1)$ and assume (s,j) and (v,k) are the corresponding points as specified above. Since ε_1 has a bounded value (i.e., $\varepsilon_1<\infty$), we can conclude from (13) that the value of those variables in $y_1(t,i)$ and $y_2(s',j')$ which are associated with the actions is exactly the same. This is because that otherwise $\|y_1(t,i)-y_2(s',j')\|_e=\infty>\varepsilon_1$ (according to (6)) which violates (13). The same results hold for $y_2(s',j')$ with respect to $y_3(v,k)$. As a consequence, the norm operator in (13) and (15) is reduced to the normal Euclidean norm which yields the following relation (according to the triangular property of the Euclidean norm)

$$\|y_1(t,i)-y_3(v,k)\|_e\leq\varepsilon_1+\varepsilon_2. \quad(16)$$

Besides, from (12) and (14) it is concluded that

$$|t-v|\leq\tau_1+\tau_2 \quad(17)$$

Consequently, the claim declared in (11) is achieved. □

6 Conclusions and Discussion

In this paper, we proposed a notion of conformance parameterized by conformance bounds in time and value. In addition to approximate comparison of (sampled) continuous signals, our notion allows for approximate matching of discrete signals by allowing them to happen within specified time margins. In

this sense, it consolidates two earlier notions of hybrid conformance by Abbas and Fainekos [2] and by van Osch [17].

In the remainder of this section, we present the directions of our ongoing research. As an immediate next step, we would like to prove the conservative extension property of the extended conformance relation compared to the other conformance relations, namely hioco and hconf. This will generalize our earlier result in [16]. Moreover, we envisage other important issues such as sound sampling rates [15], test-case generation algorithms, coverage [8, 9], links to temporal and modal logics [7], and compositionality [1, 6].

References

[1] Houssam Abbas & Georgios E. Fainekos (2015): *Towards composition of conformant systems*. *CoRR* abs/1511.05273. Available at `http://arxiv.org/abs/1511.05273`.

[2] Houssam Abbas, Hans Mittelmann & Georgios E. Fainekos (2014): *Formal property verification in a conformance testing framework*. In: *Formal Methods and Models for Codesign (MEMOCODE), 2014 Twelfth ACM/IEEE International Conference on*, pp. 155–164, doi:`10.1109/MEMCOD.2014.6961854`.

[3] Houssam Y. Abbas (2015): *Test-Based Falsification and Conformance Testing for Cyber-Physical Systems*. Ph.D. thesis, Arizona State University. Available at `http://hdl.handle.net/2286/R.A.150686`.

[4] Arend Aerts, Mohammad Reza Mousavi & Michel A. Reniers (2016): *Model-based testing of cyber-physical systems*. In: *Cyber-Physical Systems: Foundations, Principles and Applications*, Elsevier. To appear.

[5] Rajeev Alur, Costas Courcoubetis, Thomas A. Henzinger & Pei Hsin Ho (1993): *Hybrid automata: An algorithmic approach to the specification and verification of hybrid systems*. In Robert L. Grossman, Anil Nerode, Anders P. Ravn & Hans Rischel, editors: *Hybrid Systems*, Springer Berlin Heidelberg, pp. 209–229, doi:`10.1007/3-540-57318-6_30`.

[6] Nikola Benes, Przemyslaw Daca, Thomas A. Henzinger, Jan Kretínský & Dejan Nickovic (2015): *Complete Composition Operators for IOCO-Testing Theory*. In: *Proceedings of the 18th International ACM SIGSOFT Symposium on Component-Based Software Engineering (CBSE 2015)*, ACM, pp. 101–110.

[7] Jyotirmoy V. Deshmukh, Rupak Majumdar & Vinayak S. Prabhu (2015): *Quantifying Conformance Using the Skorokhod Metric*. In Daniel Kroening & S. Corina Păsăreanu, editors: *Proceedings of the 27th International Conference on Computer Aided Verification (CAV 2015), Part II*, Springer, pp. 234–250, doi:`10.1007/978-3-319-21668-3_14`.

[8] Tommaso Dreossi, Thao Dang, Alexandre Donzé, James Kapinski, Xiaoqing Jin & Jyotirmoy V. Deshmukh (2015): *Efficient Guiding Strategies for Testing of Temporal Properties of Hybrid Systems*. In Klaus Havelund, Gerard J. Holzmann & Rajeev Joshi, editors: *Proceedings of the 7th International Symposium of NASA Formal Methods (NFM 2015)*, *Lecture Notes in Computer Science* 9058, Springer, pp. 127–142, doi:`10.1007/978-3-319-17524-9_10`.

[9] Georgios E. Fainekos (2015): *Automotive control design bug-finding with the STaLiRo tool*. In: *Proceedings of the American Control Conference (ACC 2015)*, IEEE, p. 4096, doi:`10.1109/ACC.2015.7171969`.

[10] Antoine Girard & George J. Pappas (2011): *Approximate Bisimulation: A Bridge Between Computer Science and Control Theory*. *European Journal of Control* 17(5-6), pp. 568–578, doi:`10.3166/ejc.17.568-578`.

[11] A. Agung Julius, Alessandro D'Innocenzo, Maria Domenica Di Benedetto & George J. Pappas (2009): *Approximate equivalence and synchronization of metric transition systems*. *Systems & Control Letters* 58(2), pp. 94–101, doi:`10.1016/j.sysconle.2008.09.001`.

[12] A. Agung Julius & George J. Pappas (2009): *Approximations of Stochastic Hybrid Systems*. *IEEE Trans. Automat. Contr.* 54(6), pp. 1193–1203, doi:`10.1109/TAC.2009.2019791`.

[13] Narges Khakpour & Mohammad Reza Mousavi (2015): *Notions of Conformance Testing for Cyber-Physical Systems: Overview and Roadmap (Invited Paper)*. In: *Proceedings of the 26th International Conference on Concurrency Theory (CONCUR 2015)*, LIPIcsLeibniz International Proceedings in Informatics.

[14] Nancy Lynch, Roberto Segala & Frits Vaandrager (2003): *Hybrid I/O automata*. *Information and Computation* 185(1), pp. 105–157, doi:10.1016/S0890-5401(03)00067-1.

[15] Morteza Mohaqeqi & Mohammad Reza Mousavi (2016): *Sound Test-Suites for Cyber-Physical Systems*. In: *Proceedings of the 10th International Symposium on Theoretical Aspects of Software Engineering (TASE 2016)*, IEEE Computer Society, doi:10.1109/TASE.2016.33.

[16] Morteza Mohaqeqi, Mohammad Reza Mousavi & Walid Taha (2014): *Conformance Testing of Cyber-Physical Systems: A Comparative Study*. In: *Proceedings of the 14th International Workshop on Automated Verification of Critical Systems (AVOCS 2014)*, *Electronic Communications of the EASST* 70.

[17] Michiel van Osch (2006): *Hybrid input-output conformance and test generation*. In: *FATES/RV*, Springer Berlin Heidelberg, pp. 70–84.

[18] Paulo Tabuada (2007): *Approximate Simulation Relations and Finite Abstractions of Quantized Control Systems*. In: *Proceedings of the 10th International Workshop on Hybrid Systems: Computation and Control (HSCC 2007)*, *Lecture Notes in Computer Science* 4416, Springer, pp. 529–542, doi:10.1007/978-3-540-71493-4_41.

[19] Majid Zamani, Alessandro Abate & Antoine Girard (2015): *Symbolic models for stochastic switched systems: A discretization and a discretization-free approach*. *Automatica* 55, pp. 183–196, doi:10.1016/j.automatica.2015.03.004.