Quantifying Conformance using the Skorokhod Metric *

Jyotirmoy V. Deshmukh[1], Rupak Majumdar[2], and Vinayak S. Prabhu[2,3]
[1] Toyota Technical Center; [2] MPI-SWS [3] University of Porto
jyotirmoy.deshmukh@tema.toyota.com {rupak,vinayak}@mpi-sws.org

Abstract. The conformance testing problem for dynamical systems asks, given two dynamical models (e.g., as Simulink diagrams), whether their behaviors are "close" to each other. In the semi-formal approach to conformance testing, the two systems are simulated on a large set of tests, and a metric, defined on pairs of real-valued, real-timed trajectories, is used to determine a lower bound on the distance. We show how the Skorokhod metric on continuous dynamical systems can be used as the foundation for conformance testing of complex dynamical models. The Skorokhod metric allows for both state value mismatches and timing distortions, and is thus well suited for checking conformance between idealized models of dynamical systems and their implementations. We demonstrate the robustness of the metric by proving a *transference theorem*: trajectories close under the Skorokhod metric satisfy "close" logical properties in the timed linear time logic TLTL augmented with a rich class of temporal and spatial constraint predicates. We provide an efficient window-based streaming algorithm to compute the Skorokhod metric, and use it as a basis for a conformance testing tool for Simulink. We experimentally demonstrate the effectiveness of our tool in finding discrepant behaviors on a set of control system benchmarks, including an industrial challenge problem.

1 Introduction

A fundamental question in model-based design is *conformance testing*: whether two models of a system display similar behavior. For discrete systems, this question is well-studied [28, 19, 20, 29], and there is a rich theory of process equivalences based, e.g., on bisimilarity. For continuous and hybrid systems, however, the state of the art is somewhat unsatisfactory. While there is a straightforward generalization of process equivalences to the continuous case, in practice, equivalence notions such as bisimilarity are always too strong and most systems are not bisimilar. Since equivalence is a Boolean notion, one gets no additional information about the systems other than they are "not bisimilar." Further, even if two dynamical systems are bisimilar, they may still differ in many control-theoretic properties. Thus, classical notions for equivalence and conformance have been of limited use in industrial practice.

* This research was funded in part by a Humboldt fellowship, FCT grant SFRHBPD902672012, and by a contract from Toyota Motors.

In recent years, the notion of bisimulation has therefore been generalized to *metrics* on systems, which quantify the distance between them. For example, one approach is that of ϵ-bisimulation, which requires that the states of the two systems remain "close" forever (within an ϵ-ball), rather than coincide exactly. Under suitable stability assumptions on the dynamics, one can construct ϵ-bisimulations [17, 18]. Unfortunately, proving the pre-requisites for the existence of ϵ-bisimulations for complex dynamical models, or coming up with suitable and practically tractable bisimulation functions is extremely difficult in practice. In addition, establishing ϵ-bisimulation requires full knowledge of the system dynamics making the scheme inapplicable where one system is an actual physical component with unknown dynamics. So, these notions have also been of limited industrial use so far.

Instead, a more pragmatic semi-formal approach has gained prominence in industrial practice. In this approach, the two systems are executed on the same input sequences and a metric on finite trajectories is used to evaluate the closeness of these trajectories. The key to this methodology is the selection of a *good* metric, with the following properties:

- *Transference.* Closeness in the metric must translate to preserving interesting classes of logical and functional specifications between systems, and
- *Tractability.* The metric should be efficiently computable.

In addition, there is the more informal requirement of *usability*: the metric should classify systems which the engineers consider close as being close, and conversely.

The simplest candidate metric is a *pointwise* metric that computes the maximum pointwise difference between two trajectories, sometimes generalized to apply a constant time-shift to one trajectory [15]. Unfortunately, for many practical models, two trajectories may be close only under variable time-shifts. This is the case, for example, for two dynamical models that may use different numerical integration techniques (e.g., fixed step versus adaptive step) or when some component in the implementation has some jitter. Thus, the pointwise metric spuriously reports large distances for "close" models. More nuanched hybrid distances have been proposed [1], but the transference properties of these metrics w.r.t. common temporal logics are not yet clear.

In this work we present a methodology for quantifying conformance between real-valued dynamical systems based on the *Skorokhod* metric [12]. The Skorokhod metric allows for mismatches in both the trace values and in the timeline, and quantifies temporal and spatial variation of the system dynamics under a unifying framework. The distortion of the timeline is specified by a *retiming* function r which is a continuous bijective strictly increasing function from $\mathbb{R}_+$ to $\mathbb{R}_+$. Using the retiming function, we obtain the *retimed trace* $x(\mathsf{r}(t))$ from the original trace $x(t)$. Intuitively, in the retimed trace $x(\mathsf{r}(t))$, we see exactly the same values as before, in exactly the same order, but the time duration between two values might now be different than the corresponding duration in the original trace. The amount of distortion for the retiming r is given by $\sup_{t\geq 0}|\mathsf{r}(t) - t|$. Using retiming functions, the Skorokhod distance between two traces x and y is defined to be the least value over all possible retimings r of:

$$\max\left(\sup_{t\in[0,T]}|\mathsf{r}(t)-t|,\ \sup_{t\in[0,T]}\mathcal{D}\big(x\left(\mathsf{r}(t)\right),y(t)\big)\right),$$

where $\mathcal{D}$ is a pointwise metric on values. In this formula, the first component quantifies the *timing discrepancy* of the timing distortion required to "match" two traces, and the second quantifies the *value mismatch* (in the metric space $\mathcal{O}$) of the values under the timing distortion. The Skorokhod metric was introduced as a theoretical basis for defining the semantics of hybrid systems by providing an appropriate hybrid topology [9, 8]. We now demonstrate its usefulness in the context of conformance testing.

Transference. We show that the Skorokhod metric gives a robust quantification of system conformance by relating the metric to TLTL (timed LTL) enriched with (i) predicates of the form $f(x_1,\ldots,x_n)\geq 0$, as in Signal Temporal Logic [15], for specifying constraints on trace values; and (ii) *freeze quantifiers*, as in TPTL [4], for specifying temporal constraints (freeze quantifiers can express more complex timing constraints than bounded timing constraints, e.g., of MTL). TLTL subsumes MTL and STL [15]. We prove a *transference theorem*: flows (and propositional traces) which are close under the Skorokhod metric satisfy "close" TLTL formulae for a rich class of temporal and spatial predicates, where the untimed structure of the formulae remains unchanged, only the predicates are enlarged.

Tractability. We improve on recent polynomial-time algorithms for the Skorokhod metric [25] by taking advantage of the fact that, in practice, only retimings that map the times in one trace to "close" times in the other are of interest. This enables us to obtain a streaming sliding-window based monitoring procedure which takes only $O(W)$ time per sample, where W is the window size (assuming the dimension n of the system to be a constant).

Usability. Using the Skorokhod distance checking procedure as a subroutine, we have implemented a Simulink toolbox for conformance testing. Our tool integrates with Simulink's model-based design flow for control systems, and provides a stochastic search-based approach to find inputs which maximize the Skorokhod distance between systems under these inputs.

We present three case studies from the control domain, including industrial challenge problems; our empirical evaluation shows that our tool computes sharp estimates of the conformance distance reasonably fast on each of them. Our input models were complex enough that techniques such as ϵ-bisimulation functions are inapplicable. We conclude that the Skorokhod metric can be an effective foundation for semi-formal conformance testing for complex dynamical models. Proofs of the theorems are given in the accompanying technical report [13].

Related Work. The work of [1, 2] is closely related to ours. In it, robustness properties of hybrid state sequences are derived with respect to a trace metric which also quantifies temporal and spatial variations. Our work differs in the following ways. First, we guarantee robustness properties over *flows* rather than only over (discrete) sequences. Second, the Skorokhod metric is a stronger

form of the $(T, J, (\tau, \epsilon))$-closeness degree[1,2](for systems which do not have hybrid time); and allows us to give stronger robustness transference guarantees. The Skorokhod metric requires order preservation of the timeline, which the $(T, J, (\tau, \epsilon))$-closeness function does not. Preservation of the timeline order allows us to (i) keep the untimed structure of the formulae the same (unlike in the transference theorem of [1]); (ii) show transference of a rich class of global timing constraints using freeze quantifiers (rather than only for the standard bounded time quantifiers of MTL/MITL). However, for implementations where the timeline order is not preserved, we have to settle for the less stronger guarantees provided by [1]. The work of [15] deals with spatial robustness of STL; the only temporal disturbances considered are constant time-shifts for the entire signal where the entire signal is moved to the past, or to the future by the same amount. In contrast, the Skorokhod metric incorporates variable time-shifts.

2 Conformance Testing with the Skorokhod Metric

2.1 Systems and Conformance Testing

Traces and Systems. A (finite) *trace* or a *signal* $\pi : [T_i, T_e] \mapsto \mathcal{O}$ is a mapping from a finite closed interval $[T_i, T_e]$ of $\mathbb{R}_+$, with $0 \leq T_i < T_e$, to some topological space $\mathcal{O}$. If $\mathcal{O}$ is a metric space, we refer to the associated metric on $\mathcal{O}$ as $\mathcal{D}_{\mathcal{O}}$. The time-domain of π, denoted $\mathsf{tdom}(\pi)$, is the time interval $[T_i, T_e]$ over which it is defined. The time-duration of π, denoted $\mathsf{tlen}(\pi)$, is $\sup(\mathsf{tdom}(\pi))$. The t-suffix of π for $t \in \mathsf{tdom}(\pi)$, denoted π^t, is the trace π restricted to the interval $(\mathsf{tdom}(\pi) \cap [t, \mathsf{tlen}(\pi)]$. We denote by $\pi_{\downarrow T_e'}$ the prefix trace obtained from π by restricting the domain to $[T_i, T_e'] \subseteq \mathsf{tdom}(\pi)$.

A (continuous-time) *system* $\mathfrak{A} : \left(\mathbb{R}_+^{[)} \mapsto \mathcal{O}_{\mathsf{ip}}\right) \mapsto \left(\mathbb{R}_+^{[)} \mapsto \mathcal{O}_{\mathsf{op}}\right)$, where $\mathbb{R}_+^{[)}$ is the set of finite closed intervals of $\mathbb{R}_+$, transforms input traces $\pi_{\mathsf{ip}} : [T_i, T_e] \mapsto \mathcal{O}_{\mathsf{ip}}$ into output traces $\pi_{\mathsf{op}} : [T_i, T_e] \mapsto \mathcal{O}_{\mathsf{op}}$ (over the same time domain). We require that the system is *causal*: if $\mathfrak{A}(\pi_{\mathsf{ip}}) \mapsto \pi_{\mathsf{op}}$, then for every $\min \mathsf{tdom}(\pi) \leq T_e' < \max \mathsf{tdom}(\pi)$, the system $\mathfrak{A}$ maps $\pi_{\mathsf{ip}\downarrow T_e'}$ to $\pi_{\mathsf{op}\downarrow T_e'}$. Common examples of such systems are (causal) dynamical and hybrid dynamical systems [7, 30].

Conformance Testing. Let $\mathfrak{A}_1$ and $\mathfrak{A}_2$ be systems and let $\mathcal{D}_{\mathcal{TR}}$ be a metric over output traces. For a set Π_{ip} of input traces, we define the *(quantitative) conformance* between $\mathfrak{A}_1$ and $\mathfrak{A}_2$ w.r.t. Π_{ip} as $\sup_{\pi_{\mathsf{ip}} \in \Pi_{\mathsf{ip}}} \mathcal{D}_{\mathcal{TR}}(\mathfrak{A}_1(\pi_{\mathsf{ip}}), \mathfrak{A}_2(\pi_{\mathsf{ip}}))$ The conformance between $\mathfrak{A}_1$ and $\mathfrak{A}_2$ is their conformance w.r.t. the set of all input traces.

The conformance testing problem asks, given systems $\mathfrak{A}_1, \mathfrak{A}_2$, a trace metric $\mathcal{D}_{\mathcal{TR}}$, a tolerance δ, and a set of test input traces Π_{test}, if the quantitative con-

[1] Instead of having two separate parameters τ and ϵ for time and state variation, we pre-scale time and the n state components with $n + 1$ constants, and have a single value quantifying closeness of the scaled traces.

[2] Informally, two signals x, y are $(T, J, (\tau, \epsilon))$-close if for each point $x(t)$, there is a point $y(t')$ with $|t - t'| < \tau$ such that $\mathcal{D}(x(t), y(t')) < \epsilon$; and similarly for $y(t)$.

Algorithm 1: Algorithm for Conformance Testing

Input: Systems $\mathfrak{A}_1$, $\mathfrak{A}_2$, trace metric $\mathcal{D}_{\mathcal{TR}}$, time horizon T, input parameterization F, termination criterion `terminate`?

Output: Input u that achieves maximum distance between $\mathfrak{A}_1$ and $\mathfrak{A}_2$

1 $d \leftarrow 0$, $u \leftarrow \perp$, $dmax \leftarrow 0$, $umax \leftarrow \perp$
2 **while** not(`terminate`?) **do**
3 $\quad u \leftarrow$ `pickNewInputs`(F, T, d)
4 $\quad y_1 \leftarrow$ `simulate`$(\mathfrak{A}_1, u, T)$ and $y_2 \leftarrow$ `simulate`$(\mathfrak{A}_2, u, T)$
5 $\quad d \leftarrow \mathcal{D}_{\mathcal{TR}}(y_1, y_2)$
6 $\quad$ **if** $d > dmax$ **then** $dmax \leftarrow d$, $umax \leftarrow u$
7 **end**
8 **return** *"on input umax, outputs* $\mathfrak{A}_1(umax)$ *and* $\mathfrak{A}_2(umax)$ *differ by dmax by time* T*"*

formance between $\mathfrak{A}_1$ and $\mathfrak{A}_2$ w.r.t. Π_{test} is more than δ. Clearly, conformance w.r.t. Π_{test} is a lower bound on the conformance between $\mathfrak{A}_1$ and $\mathfrak{A}_2$.

Algorithm 1 is a standard optimization-guided adaptive testing algorithm. To define the set Π_{test} of test inputs, we use a fixed finite parameterization of the input space using a finite set F of *basis functions* and fix a time horizon T. We only generate inputs obtained as a linear combination $\sum_{f \in F} p_f \cdot f$ of basis functions over the interval $[0, T]$, where the coefficients $\{p_f \mid f \in F\}$ come from a closed convex subset of $\mathbb{R}^{|F|}$.

In each step, Algorithm 1 picks an input signal u and computes the distance between the corresponding outputs $y_1 = \mathfrak{A}_1(u)$ and $y_2 = \mathfrak{A}_2(u)$. Based on heuristics that rely on the current distance, and a possibly bounded history of costs, the procedure then picks a new value for u by choosing new values for the coefficients $\{p_f \mid f \in F\}$. For instance, in a gradient-ascent based procedure, the new value of u is chosen by estimating the local gradient in each direction in the input-parameter space, and then picking the direction that has the largest (positive) gradient. In our implementation, we use the Nelder-Mead (or nonlinear simplex) algorithm to pick new inputs.

On termination (e.g., when some maximum number of iterations is reached), the algorithm returns the conformance distance between $\mathfrak{A}_1$ and $\mathfrak{A}_2$ w.r.t. the set of tests generated. One can compare the distance to some tolerance δ chosen based on engineering requirements.

Sampling and Polygonal Traces. In practice, the output behaviors of the systems are observed with a sampling process, thus y_1 and y_2 on line 4 are discrete time-sampled *sequences*. We go from these sequences to output traces by linearly interpolating between the sampled time points.

Formally, a *polygonal trace* $\pi : I_\pi \mapsto \mathcal{O}$ where $\mathcal{O}$ is a vector space with the scalar field $\mathbb{R}$ is a continuous trace such that there exists a finite sequence $\min I_\pi = t_0 < t_1 < \cdots < t_m = \max I_\pi$ of time-points such that the trace segment between t_k and t_{k+1} is affine for all $0 \leq k < m$, *i.e.*, for $t_k \leq t \leq t_{k+1}$ we have $\pi(t) = \pi(t_k) + \frac{t - t_k}{t_{k+1} - t_k} \cdot (\pi(t_{k+1}) - \pi(t_k))$.

Given a timed trace sequence tseq, let $[\![\mathsf{tseq}]\!]_{\mathsf{LI}}$ denote the polygonal trace obtained from tseq by linear interpolation. Let $\mathsf{tseq}_\pi, \mathsf{tseq}_{\pi'}$ be two corresponding samplings of the traces π, π', respectively. For a trace metric $\mathcal{D}_{\mathcal{TR}}$, we have:

$$\mathcal{D}_{\mathcal{TR}}(\pi, \pi') \leq \mathcal{D}_{\mathcal{TR}}\left([\![\mathsf{tseq}_\pi]\!]_{\mathsf{LI}}, [\![\mathsf{tseq}_{\pi'}]\!]_{\mathsf{LI}}\right) + \mathcal{D}_{\mathcal{TR}}\left([\![\mathsf{tseq}_\pi]\!]_{\mathsf{LI}}, \pi\right) + \mathcal{D}_{\mathcal{TR}}\left([\![\mathsf{tseq}_{\pi'}]\!]_{\mathsf{LI}}, \pi'\right).$$

If Δ_{samerr} is a bound on the distance between a trace and an interpolated completion of its sampling, we have that $\mathcal{D}_{\mathcal{TR}}(\pi, \pi') \leq \mathcal{D}_{\mathcal{TR}}([\![\mathsf{tseq}_\pi]\!]_{\mathsf{LI}}, [\![\mathsf{tseq}_{\pi'}]\!]_{\mathsf{LI}}) + 2 \cdot \Delta_{\mathsf{samerr}}$. Thus, a value of $2 \cdot \Delta_{\mathsf{samerr}}$ needs to be added in the testing algorithm to account for the error due to polygonal approximations.

2.2 The Skorokhod Metric

We now define the Skorokhod metric, which we use as the metric in Algorithm 1. A *retiming* $\mathsf{r} : I \mapsto I'$, for closed intervals I, I' of $\mathbb{R}_+$ is an order-preserving (i.e., monotone) continuous bijective function from I to I'; thus if $t < t'$ then $\mathsf{r}(t) < \mathsf{r}(t')$. Let $\mathsf{R}_{I \mapsto I'}$ be the class of retiming functions from I to I' and let $\mathcal{I}$ be the identity retiming. Intuitively, retiming can be thought of as follows: imagine a stretchable and compressible timeline; a retiming of the original timeline gives a new timeline where some parts have been stretched, and some compressed, without the timeline having been broken. Given a trace $\pi : I_\pi \to \mathcal{O}$, and a retiming $\mathsf{r} : I \mapsto I_\pi$; the function $\pi \circ \mathsf{r}$ is another trace from I to $\mathcal{O}$.

Definition 1 (Skorokhod Metric). *Given a retiming* $\mathsf{r} : I \mapsto I'$*, let* $||\, \mathsf{r} - \mathcal{I} \,||_{\sup}$ *be defined as* $||\, \mathsf{r} - \mathcal{I} \,||_{\sup} = \sup_{t \in I} |\, \mathsf{r}(t) - t|$*. Given two traces* $\pi : I_\pi \mapsto \mathcal{O}$ *and* $\pi' : I_{\pi'} \mapsto \mathcal{O}$*, where* $\mathcal{O}$ *is a metric space with the associated metric* $\mathcal{D}_{\mathcal{O}}$*, and a retiming* $\mathsf{r} : I_\pi \mapsto I_{\pi'}$*, let* $\|\pi - \pi' \circ \mathsf{r}\|_{\sup}$ *be defined as:*

$$\|\pi - \pi' \circ \mathsf{r}\|_{\sup} = \sup_{t \in I_\pi} \mathcal{D}_{\mathcal{O}}\big(\, \pi(t) \;,\; \pi'\left(\mathsf{r}(t)\right) \,\big).$$

The Skorokhod distance[3] *between the traces* $\pi()$ *and* $\pi'()$ *is defined to be:*

$$\mathcal{D}_{\mathcal{S}}(\pi, \pi') = \inf_{r \in \mathsf{R}_{I_\pi \mapsto I_{\pi'}}} \max(\|\mathsf{r} - \mathcal{I}\|_{\sup}\,,\, \|\pi - \pi' \circ \mathsf{r}\|_{\sup}). \quad \square \tag{1}$$

Intuitively, the Skorokhod distance incorporates two components: the first component quantifies the *timing discrepancy* of the timing distortion required to "match" two traces, and the second quantifies the *value mismatch* (in the metric space $\mathcal{O}$) of the values under the timing distortion. In the retimed trace $\pi \circ \mathsf{r}$, we see exactly the same values as in π, in exactly the same order, but the times at which the values are seen can be different.

The following theorem shows that the Skorokhod distance between polygonal traces can be computed efficiently. We remark that after retiming, the retimed version $\pi \circ \mathsf{r}$ of a polygonal trace π need not be polygonal (see e.g., [24]).

[3] The two components of the Skorokhod distance (the retiming, and the value difference components) can be weighed with different weights – this simply corresponds to a change of scale.

Theorem 1 (Computing the Distance between Polygonal Traces [25]). *Let $\pi : I_\pi \mapsto \mathbb{R}^n$ and $\pi' : I_{\pi'} \mapsto \mathbb{R}^n$ be two polygonal traces with m_π and $m_{\pi'}$ affine segments respectively. Let the Skorokhod distance between them (for the L_2 norm on $\mathbb{R}^n$) be denoted as $\mathcal{D}_\mathbb{S}(\pi, \pi')$.*

1. *Given $\delta \geq 0$, it can be checked whether $\mathcal{D}_\mathbb{S}(\pi, \pi') \leq \delta$ in time $O(m_\pi \cdot m_{\pi'} \cdot n)$.*
2. *Suppose we restrict retimings to be such that the i-th affine segment of π can only be matched to π' affine segments $i - W$ through $i + W$ for all i, where $W \geq 1$. Under this retiming restriction, we can determine, with a streaming algorithm, whether $\mathcal{D}_\mathbb{S}(\pi, \pi') \leq \delta$ in time $O((m_\pi + m_{\pi'}) \cdot n \cdot W)$.* □

Let us denote by $\mathcal{D}_\mathbb{S}^W(\pi, \pi')$ the Skorokhod difference between π, π' under the retiming restriction of the second part of Theorem 1, *i.e.*, the value obtained by restricting the retimings in Equation 1[4]. The value $\mathcal{D}_\mathbb{S}^W(\pi, \pi')$ is an upper bound on $\mathcal{D}_\mathbb{S}(\pi, \pi')$. In addition, for $W' < W$, we have $\mathcal{D}_\mathbb{S}^W(\pi, \pi') \leq \mathcal{D}_\mathbb{S}^{W'}(\pi, \pi')$.

3 Transference of Logical Properties

In this section, we demonstrate a transference result for the Skorokhod metric for a version of the timed linear time logic TLTL [4]. The logic we consider generalizes MTL and STL. We show that if the Skorokhod distance between two traces is small, they satisfy close TLTL formulae. Given a formula ϕ of TLTL satisfied by trace π_1, we can compute a "relaxation" of ϕ that will be satisfied by the "close" trace π_2. We first present the results in a propositional framework, and then extend to $\mathbb{R}^n$-valued spaces for a logic generalizing STL.

3.1 The Logic TLTL

Let $\mathcal{P}$ be a set of propositions. A *propositional trace* π over $\mathcal{P}$ is a trace where the topological space is $2^\mathcal{P}$, with the associated metric $\mathcal{D}_\mathcal{P}(\sigma, \sigma') = 0$ if $\sigma = \sigma'$, and ∞ otherwise, for $\sigma, \sigma' \in 2^\mathcal{P}$. We restrict our attention to propositional traces with finite variability: we require that there exists a finite partition of $\mathsf{tdom}(\pi)$ into disjoint subintervals $I_0, I_1, \ldots, I_m$ such that π is constant on each subinterval. The set of all timed propositional traces over $\mathcal{P}$ is denoted by $\Pi(\mathcal{P})$.

Definition 2 (TLTL($\mathcal{F}_\mathsf{T}$) Syntax). *Given a set of propositions $\mathcal{P}$, a set of (time) variables V_T, and a set $\mathcal{F}_\mathsf{T}$ of functions from $\mathbb{R}_+^l$ to $\mathbb{R}$, the formulae of* TLTL*($\mathcal{F}_\mathsf{T}$) are defined by the following grammar.*

$$\phi := p \mid \text{TRUE} \mid f_\mathsf{T}(\overline{x}) \sim 0 \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \phi_1 \vee \phi_2 \mid \phi_1 \,\mathcal{U}\, \phi_2 \mid x.\phi \quad \textit{where}$$

- *$p \in \mathcal{P}$ and $x \in V_\mathsf{T}$, and $\overline{x} = (x_1, \ldots, x_l)$ with $x_i \in V_\mathsf{T}$ for all $1 \leq i \leq l$;*
- *$f_\mathsf{T} \in \mathcal{F}_\mathsf{T}$ is a real-valued function, and $\sim$ is one of $\{\leq, <, \geq, >\}$.* □

[4] $\mathcal{D}_\mathbb{S}^W$ is not a metric over traces (the triangle inequality fails).

The quantifier "$x.$" is known as the *freeze quantifier*, and binds variable x to the current time. A variable x is defined to be *free* in ϕ as follows. The variable x is *not* free in $x.\Psi$, or in p (a proposition), or in TRUE, or in $f_{\mathsf{T}}(x_1, \ldots, x_l) \sim 0$ where $x_i \neq x$ for all i. It is also not free in ϕ if ϕ does not contain an occurrence of x. It is free in $\neg\psi$ iff x is free in ψ; and it is free in $\phi_1 \mathbin{\overset{\wedge}{\vee}} \phi_2$, or in $\phi_1 \,\mathcal{U}\, \phi_2$, iff x is free in either ϕ_1 or in ϕ_2. Finally, variable x is free in $f_{\mathsf{T}}(x_1, \ldots, x_l) \sim 0$ if some x_i is x. A formula is *closed* if it has no free variables.

Definition 3 (TLTL($\mathcal{F}_{\mathsf{T}}$) Semantics). *Let $\pi : I \mapsto 2^{\mathcal{P}}$ be a timed propositional trace, and let $\mathcal{E} : V_{\mathsf{T}} \mapsto I$ be the time environment mapping the variables in V_{T} to time values in I. The satisfaction of the trace π with respect to the* TLTL($\mathcal{F}_{\mathsf{T}}$) *formula ϕ in the time environment $\mathcal{E}$ is written as $\pi \models_{\mathcal{E}} \phi$, and is defined inductively as follows (denoting $t_0 = \min \mathsf{tdom}(\pi)$).*

$\pi \models_{\mathcal{E}} p$ *for* $p \in \mathcal{P}$ *iff* $p \in \pi(t_0)$; $\quad \pi \models_{\mathcal{E}}$ TRUE; $\quad \pi \models_{\mathcal{E}} \neg\Psi$ *iff* $\pi \not\models_{\mathcal{E}} \Psi$;
$\pi \models_{\mathcal{E}} \phi_1 \wedge \phi_2$ *iff* $\pi \models_{\mathcal{E}} \phi_1$ *and* $\pi \models_{\mathcal{E}} \phi_2$; $\quad \pi \models_{\mathcal{E}} \phi_1 \vee \phi_2$ *iff* $\pi \models_{\mathcal{E}} \phi_1$ *or* $\pi \models_{\mathcal{E}} \phi_2$;
$\pi \models_{\mathcal{E}} f_{\mathsf{T}}(x_1, \ldots, x_l) \sim 0$ *iff* $f_{\mathsf{T}}(\mathcal{E}(x_1), \ldots, \mathcal{E}(x_l)) \sim 0$ *for* $\sim \in \{\leq, <, \geq, >\}$;
$\pi \models_{\mathcal{E}} x.\psi$ *iff* $\pi \models_{\mathcal{E}[x:=t_0]} \psi$ *where* $\mathcal{E}[x:=t_0]$ *agrees with* $\mathcal{E}$ *for all* $x_i \neq x$, *and maps* x *to* t_0;
$\pi \models_{\mathcal{E}} \phi_1 \,\mathcal{U}\, \phi_2$ *iff* $\pi^t \models_{\mathcal{E}} \phi_2$ *for some* $t \in I$ *and* $\pi^{t'} \models_{\mathcal{E}} \phi_1 \vee \phi_2$ *for all* $t_0 \leq t' < t$.

A timed trace π is said to satisfy the closed formula ϕ (written as $\pi \models \phi$) if there is some environment $\mathcal{E}$ such that $\pi \models_{\mathcal{E}} \phi$. □

We define additional temporal operators in the standard way: the "eventually" operator $\Diamond\phi$ stands for TRUE$\mathcal{U}\,\phi$; and the "always" operator $\Box\phi$ stands for $\neg\Diamond\neg\phi$. TLTL($\mathcal{F}_{\mathsf{T}}$) provides a richer framework than MTL [23] for expressing timing constraints as: (i) freeze quantifiers allow specification of constraints between distant contexts, which the bounded temporal operators in MTL cannot do; and (ii) the predicates $f_{\mathsf{T}}() \sim 0$ for $f_{\mathsf{T}} \in \mathcal{F}_{\mathsf{T}}$ allow the specification of complex timing requirements not expressible in MTL. Note that even if the predicates $f_{\mathsf{T}}() \sim 0$ are restricted to be of the form $x_1 - x_2 + c \sim 0$, where x_1, x_2 are freeze variables, and c is a constant, TLTL($\mathcal{F}_{\mathsf{T}}$) is more expressive than MTL [6] (and hence more expressive than MITL on which STL is based).

*Example 1 (*TLTL($\mathcal{F}_{\mathsf{T}}$) *subsumes* MTL*).* Let $\mathcal{F}_{\mathsf{T}}$ be the set of two variable functions of the form $f(x, y) = x - y + c$ where c is a rational constant. Then TLTL($\mathcal{F}_{\mathsf{T}}$) subsumes MTL. The MTL formula $p\,\mathcal{U}_{\,[a,b]}\,q$ can be written as

$$x.\Big(p\,\mathcal{U}\,y.\big(\,(y \leq x + b) \wedge (y \geq x + a) \wedge q\big)\Big).$$

We explain the formula as follows. We assign the "current" time t_x to the variable x, and some future time t_y to the variable y. The values t_x and t_y are such that at time t_y, we have q to be true, and moreover, at all times between t_x and t_y, we have $p \vee q$ to be true. Furthermore, t_y must be such that $t_y \in [t_x + a, t_x + b]$, which is specified by the term $(y \leq x + b) \wedge (y \geq x + a)$. □

Example 2 (Temporal Constraints). Suppose we want to express that whenever the event p occurs, it must be followed by a response q, and then by r. In addition, we have the following timing requirement: if $\varepsilon_{pq}, \varepsilon_{qr}, \varepsilon_{pr}$ are the time delays between p and q, between q and r, and between p and r, respectively, then: we must have $\varepsilon_{pq}^2 + \varepsilon_{qr}^2 + \varepsilon_{pr}^2 \leq d$ for a given positive constant d. This can be written using freeze quantifiers as the TLTL formula ϕ:

$$x.\,\big(p \rightarrow \Diamond\big(y.\,\big(q \wedge \Diamond\,\big[z.\,\big(r \wedge \big((y-x)^2 + (z-y)^2 + (z-x)^2 \leq d\big)\big)\big]\big)\big)\big)\,. \quad \square$$

3.2 Transference of TLTL Properties for Propositional Traces

We now show that if a timed propositional trace π satisfies a TLTL($\mathcal{F}_\mathsf{T}$) formula ϕ, then any timed trace π' that is at most δ distance away from π satisfies a slightly relaxed version of the formula ϕ, the degree of relaxation being governed by δ; and the variance of the functions in $\mathcal{F}_\mathsf{T}$ over the time interval containing the time domains of π and π'.

We define the distance $\mathcal{D}_\mathrm{S}$ between two propositional traces as the Skorokhod distance, where we use $\mathcal{D}_\mathcal{P}$ as the distance between two sets of propositions.

Next, we define relaxations of TLTL($\mathcal{F}_\mathsf{T}$) formulae. The relaxations are defined as a syntactic transformation on formulae in negation-normal form, *i.e.*, in which negations only appear at the propositions. It can be showed that every TLTL($\mathcal{F}_\mathsf{T}$) formula can be rewritten in negation-normal form, when we additionally use the waiting for operator, $\mathcal{W}$, defined as:

$\pi \models_\mathcal{E} \phi_1 \,\mathcal{W}\, \phi_2$ iff either (1) $\pi^t \models_\mathcal{E} \phi_1$ for all $t \in I_\pi$; or (2) $\pi^t \models_\mathcal{E} \phi_2$ for some $t \in I_\pi$; and $\pi^{t'} \models_\mathcal{E} \phi_1 \vee \phi_2$ for all $\min I_\pi \leq t' < t$.

Definition 4 (δ-relaxation of TLTL($\mathcal{F}_\mathsf{T}$) formulae). *Let ϕ be a* TLTL($\mathcal{F}_\mathsf{T}$) *formula in which negations appear only on the propositional symbols. The δ relaxation of ϕ (for $\delta \geq 0$) over a closed interval J, denoted $\mathsf{rx}_J^\delta(\phi)$, is defined as:*

$$\begin{array}{l|l}
\mathsf{rx}_J^\delta(p) = p & \mathsf{rx}_J^\delta(\textsc{true}) = \textsc{true} \\
\mathsf{rx}_J^\delta(\neg p) = \neg p & \mathsf{rx}_J^\delta(\textsc{false}) = \textsc{false} \\
\mathsf{rx}_J^\delta(\phi_1 \wedge \phi_2) = \mathsf{rx}_J^\delta(\phi_1) \wedge \mathsf{rx}_J^\delta(\phi_2) & \mathsf{rx}_J^\delta(\phi_1 \vee \phi_2) = \mathsf{rx}_J^\delta(\phi_1) \vee \mathsf{rx}_J^\delta(\phi_2) \\
\mathsf{rx}_J^\delta(x.\psi) = x.\,\mathsf{rx}_J^\delta(\psi) & \\
\mathsf{rx}_J^\delta(\phi_1 \,\mathcal{U}\, \phi_2) = \mathsf{rx}_J^\delta(\phi_1) \,\mathcal{U}\, \mathsf{rx}_J^\delta(\phi_2) & \mathsf{rx}_J^\delta(\phi_1 \,\mathcal{W}\, \phi_2) = \mathsf{rx}_J^\delta(\phi_1) \,\mathcal{W}\, \mathsf{rx}_J^\delta(\phi_2)
\end{array}$$

$$\mathsf{rx}_J^\delta\left(f_\mathsf{T}(x_1,\ldots,x_l)\right) \sim 0) = \begin{cases} f_\mathsf{T}(x_1,\ldots,x_l) \;+\; K_J^{f_\mathsf{T}}(\delta) \;\sim\; 0 & \text{if } \sim \in \{>, \geq\} \\ f_\mathsf{T}(x_1,\ldots,x_l) \;-\; K_J^{f_\mathsf{T}}(\delta) \;\sim\; 0 & \text{if } \sim \in \{<, \leq\}, \end{cases}$$

where $K_J^{f_\mathsf{T}} : [0, \max \mathsf{tdom}(J) \;-\; \min \mathsf{tdom}(J)] \mapsto \mathbb{R}_+$, *and*

$$K_J^{f_\mathsf{T}}(\delta) \stackrel{\text{def}}{=} \sup_{\substack{t_1,\ldots,t_l \in J \\ t'_1,\ldots,t'_l \in J}} \left\{ \left| \begin{array}{c} f_\mathsf{T}(t_1,\ldots,t_l) \\ - \\ f_\mathsf{T}(t'_1,\ldots,t'_l) \end{array} \right| \quad \textit{s.t. } |t_i - t'_i| \leq \delta \textit{ for all } i \right\} \tag{2}$$

Thus, instead of comparing the $f_\mathsf{T}()$ values to 0, we relax by comparing instead to $\pm K_J^{f_\mathsf{T}}(\delta)$. The other cases recursively relax the subformulae. The

functions $K_J^{f_\mathsf{T}}(\delta)$ define the maximal change in the value of f_T that can occur when the input variables can vary by δ. The role of J is to restrict the domain of the freeze quantifier variables to the time interval J (from $\mathbb{R}_+$) in order to obtain the least possible relaxation on a given trace π (e.g., we do not care about the values of a function in $\mathcal{F}_\mathsf{T}$ outside of the domain $\mathsf{tdom}(\pi)$ of the trace).

Example 3 (δ-relaxation for Bounded Temporal Operators – MTL). We demonstrate how δ-relaxation operates on bounded time constraints. Consider again the MTL formula $\phi = p\mathcal{U}_{[a,b]}q$. When written as a TLTL formula and relaxed using the $\mathsf{rx}^\delta_{\mathbb{R}_+}$ function, the relaxed TLTL formula is equivalent to the MTL formula $p\mathcal{U}_{[a-2\cdot\delta\,,\,b+2\cdot\delta]}q$. □

Theorem 2 (Transference for Propositional Traces). *Let π, π' be two timed propositional traces such that $\mathcal{D}_\mathsf{S}(\pi,\pi') < \delta$ for some finite δ. Let ϕ be a closed* TLTL($\mathcal{F}_\mathsf{T}$) *formula in negation-normal form. If $\pi \models \phi$, then $\pi' \models \mathsf{rx}^\delta_{I_{\pi,\pi'}}(\phi)$ where $I_{\pi,\pi'}$ is the convex hull of $\mathsf{tdom}(\pi) \cup \mathsf{tdom}(\pi')$.* □

Theorem 2 relaxes the freeze variables over the entire signal time-range $I_{\pi,\pi'}$; it can be strengthened by relaxing over a smaller range: if $\pi \models \phi$, and $t_1, \ldots, t_k$ are time-stamp assignments to the freeze variables $x_1, \ldots, x_k$ which witness π satisfying ϕ, then x_i only needs to be relaxed over $[t_i - \delta, t_i + \delta]$ rather than the larger interval $I_{\pi,\pi'}$. These smaller relaxation intervals for the freeze variables can be incorporated in Equation 2. We omit the details for ease of presentation.

Example 4. Recall Example 2, and the formula ϕ presented in it. Suppose a trace π satisfies ϕ; and let $\mathcal{D}_\mathsf{S}(\pi,\pi') < \delta$ (using the Skorokhod metric for propositional traces). Our transference theorem ensures that (i) π' will satisfy the same untimed formula $p \rightarrow \Diamond(q \wedge \Diamond r)$; and (ii) it gives a bound on how much the timing constraints need to be relaxed in ϕ in order to ensure satisfaction by π'; it states that π' satisfies the following relaxed formula ϕ'.

$$\pi' \models x.\,\big(p \rightarrow \Diamond\big(y.\,\big(q \wedge \Diamond\,\big[z.\,\big(r \wedge \big((y-x)^2 + (z-y)^2 + (z-x)^2 \leq d^\dagger\big)\big]\big)\big)\big)$$

where $d^\dagger = d + 12\cdot\delta^2 + 4\sqrt{3}\cdot\delta\cdot\sqrt{d}$ (see [13]). □

3.3 Transference of TLTL Properties for $\mathbb{R}^n$-valued Signals

A *timed $\mathbb{R}^n$-valued trace* π is a function from a closed interval I of $\mathbb{R}_+$ to $\mathbb{R}^n$. For $\overline{\alpha} = (\alpha^0, \ldots, \alpha^n) \in \mathbb{R}^n$, we denote the k-th dimensional value α^k as $\overline{\alpha}[k]$. The π projected function onto the k-th $\mathbb{R}$ dimension is denoted by $\pi_k : I \mapsto \mathbb{R}$.

To define the semantics of TLTL formulae over timed $\mathbb{R}^n$-valued sequences, we use booleanizing predicates $\mu : \mathbb{R}^n \mapsto \mathbb{B}$, as in STL [15], to transform $\mathbb{R}^n$-valued sequences into timed propositional sequences. These predicates are part of the logical specification. In this work, we restrict our attention to traces and predicates such that each predicate varies only finitely often on the finite time traces under consideration. Since we also have freeze variables, TLTL with predicates is strictly more expressive than STL[5] (as in the propositional case [6]).

[5] STL is MITL enriched with booleanzing predicates, *i.e.*, STL is MITL($\mathcal{F}_\mathsf{S}$).

Definition 5 (TLTL($\mathcal{F}_\mathsf{T}, \mathcal{F}_\mathsf{S}$) Syntax). *Given a set of variables V_T (the freeze variables), a set of* ordered *variables V_S (the signal variables), and two sets $\mathcal{F}_\mathsf{T}, \mathcal{F}_\mathsf{S}$ of functions, the formulae of* TLTL($\mathcal{F}_\mathsf{T}, \mathcal{F}_\mathsf{S}$) *are defined by the grammar:*

$$\phi := \text{TRUE} \mid f_\mathsf{T}(\overline{x}) \sim 0 \mid f_\mathsf{S}(\overline{y}) \sim 0 \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \phi_1 \vee \phi_2 \mid \phi_1 \,\mathcal{U}\, \phi_2 \mid x.\phi \quad \textit{where}$$

- $x \in V_\mathsf{T}$, *and* $\overline{x} = (x_1, \ldots, x_l)$ *with* $x_i \in V_\mathsf{T}$ *for all* $1 \leq i \leq l$;
- $\overline{z} = (z_1, \ldots, z_d)$ *with* $z_j \in V_\mathsf{S}$ *for all* $1 \leq j \leq d$ *(with* $d \leq n$*);*
- V_T *and* V_S *are disjoint;*
- $f_\mathsf{T} \in \mathcal{F}_\mathsf{T}$ *and* $f_\mathsf{S} \in \mathcal{F}_\mathsf{S}$ *are real-valued functions, and* $\sim$ *is* $\leq, <, \geq,$ *or* $>$. $\square$

The semantics of TLTL($\mathcal{F}_\mathsf{T}, \mathcal{F}_\mathsf{S}$) is straightforward and similar to the propositional case (Definition 3). The only new ingredients are the booleanizing predicates $f_\mathsf{S}(\overline{z}) \sim 0$: we define $\pi \models_\mathcal{E} f_\mathsf{S}(z_1, \ldots, z_d) \sim 0$ iff $f_\mathsf{S}(\pi_{j_1}[t_0], \ldots, \pi_{j_d}[t_0]) \sim 0$ for any freeze variable environment $\mathcal{E}$, where $t_0 = \min \mathsf{tdom}(\pi)$, and z_i is the j_i-th variable in V_S (*i.e.*, z_i refers to the j_i-th dimension in the signal trace). We require that for a timed $\mathbb{R}^n$-valued trace π to satisfy ϕ, the arity of the functions in $\mathcal{F}_\mathsf{S}$ occurring in ϕ should not be more than n, that is, functions should not refer to dimensions greater than n for an $\mathbb{R}^n$ trace.

δ relaxation of TLTL($\mathcal{F}_\mathsf{T}, \mathcal{F}_\mathsf{S}$). Let $\mathbf{J}_{V_\mathsf{S}}$ be a mapping from V_S to closed intervals of $\mathbb{R}$, thus $\mathbf{J}_{V_\mathsf{S}}(z)$ denotes a sub-domain of $z \in V_\mathsf{S}$. The relaxation function $\mathsf{rx}^\delta_{J,\mathbf{J}_{V_\mathsf{S}}}$ which operates on TLTL($\mathcal{F}_\mathsf{T}, \mathcal{F}_\mathsf{S}$) formulae is defined analogous to the relaxation function rx^δ_J in Definition 4. We omit the similar cases, and only present the new case for the predicates formed from $\mathcal{F}_\mathsf{S}$.

$$\mathsf{rx}^\delta_{J,\mathbf{J}_{V_\mathsf{S}}}\left(f_\mathsf{S}(z_1, \ldots, z_l)) \sim 0\right) = \begin{cases} f_\mathsf{S}(z_1, \ldots, z_l) + K^{f_\mathsf{S}}_{J,\mathbf{J}_{V_\mathsf{S}}}(\delta) \sim 0 & \text{if } \sim \in \{>, \geq\}; \\ f_\mathsf{S}(z_1, \ldots, z_l) - K^{f_\mathsf{S}}_{J,\mathbf{J}_{V_\mathsf{S}}}(\delta) \sim 0 & \text{if } \sim \in \{<, \leq\} \end{cases}$$

where $K^{f_\mathsf{S}}_{J,\mathbf{J}_{V_\mathsf{S}}} : \left[0,\ \max_{z \in V_\mathsf{S}} |\max \mathbf{J}_{V_\mathsf{S}}(z) - \min \mathbf{J}_{V_\mathsf{S}}(z)|\right] \mapsto \mathbb{R}_+$ is a function s.t.

$$K^{f_\mathsf{S}}_{J,\mathbf{J}_{V_\mathsf{S}}}(\delta) = \sup_{\substack{u_i \in \mathbf{J}_{V_\mathsf{S}}(z_i);\ u'_i \in \mathbf{J}_{V_\mathsf{S}}(z'_i) \\ \text{for all } i}} \left\{ \left| \begin{array}{c} f_\mathsf{S}(u_1, \ldots, u_l) \\ - \\ f_\mathsf{S}(u'_1, \ldots, u'_l) \end{array} \right| \text{ s.t. } |u_i - u'_i| \leq \delta \text{ for all } i \right\}.$$

The functions $K^{f_\mathsf{S}}_{J,\mathbf{J}_{V_\mathsf{S}}}(\delta)$ define the maximal change in the value of f_S that can occur when the input variables can vary by δ over the intervals in $\mathbf{J}_{V_\mathsf{S}}(z)$ and J. The role of $\mathbf{J}_{V_\mathsf{S}}$ in the above definition is to restrict the domain of the signal variables in order to obtain the least possible relaxation bounds on the signal constraints; as was done in Definition 4 for the freeze variables.

Theorem 3 (Transference for $\mathbb{R}^n$-valued Traces). *Let π, π' be two $\mathbb{R}^n$-valued traces such the Skorokhod distance between them is less than δ for some finite δ. Let ϕ be a closed* TLTL($\mathcal{F}_\mathsf{T}, \mathcal{F}_\mathsf{S}$) *formula in negation-normal form. If $\pi \models \phi$, then $\pi' \models \mathsf{rx}^\delta_{I_{\pi,\pi'},\mathbf{I}_{V_\mathsf{S}}}(\phi)$, where*

- $I_{\pi,\pi'}$ *is the convex hull of* $\mathsf{tdom}(\pi) \cup \mathsf{tdom}(\pi')$*; and*

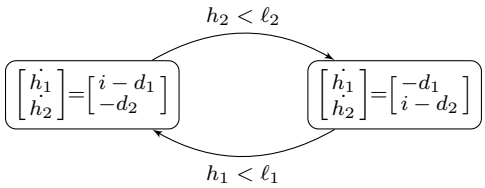

Fig. 1. System $\mathfrak{A}_1$ used for benchmarking Skorokhod Distance computation. Inflow rate i, Drain rate d_1 for tank 1 and d_2 for tank 2 are all inputs to the system.

– $\mathbf{I}_{V_\mathsf{S}}(z)$ *is the convex hull of* $\{\pi(t)[k] \mid t \in \mathsf{tdom}(\pi)\} \cup \{\pi'(t)[k] \mid t \in \mathsf{tdom}(\pi')\}$*; where* z *is the* k*-th variable in the ordered set* V_S*.* □

Theorem 3 can be strengthened similar to the strengthening mentioned for Theorem 2 by relaxing the variables over smaller intervals obtained from assignments to variables which witness $\pi \models \phi$.

Example 5 (Spatial Constraints and Transference). Recall Example 2, suppose that the events p, q, and r are defined by the following predicates over real variables α_1 and α_2. Let $p \equiv \alpha_1 + 10{\cdot}\alpha_2 \geq 3$; the predicate $q \equiv |\alpha_1| + |\alpha_2| \leq 20$; and $r \equiv |\alpha_1| + |\alpha_2| \leq 15$. Let π satisfy this formula with these predicates, and let π' be δ close to π, for a finite δ under the Skorokhod metric for $\mathbb{R}^2$. Our robustness theorem ensures that π' will satisfy the relaxed formula

$$x.\left(p^\delta \rightarrow \Diamond\left(y.\left(q^\delta \wedge \Diamond\left[z.\left(r^\delta \wedge \left((y-x)^2 + (z-y)^2 + (z-x)^2 \leq d + 12{\cdot}\delta^2\right)\right)\right]\right)\right).$$

where the relaxed predicates $p^\delta, q^\delta, r^\delta$ are defined as follows: $p^\delta \equiv \alpha_1 + 10{\cdot}\alpha_2 \geq 3 - 22{\cdot}\delta$; $q^\delta \equiv |\alpha_1| + |\alpha_2| \leq 20 + 4{\cdot}\delta$; and $r^\delta \equiv |\alpha_1| + |\alpha_2| \leq 15 + 4{\cdot}\delta$. □

4 Experimental Evaluation

We have implemented a streaming, sliding window-based monitoring routine which checks, given a fixed δ, whether the linear interpolations of two time-sampled traces are at Skorokhod distance at most δ away from each other. The least such δ value is then computed by binary search over the monitoring routine. The upper limit of the search range is set to the pointwise metric (i.e., assuming the identity retiming) between the two traces. The traces to the monitoring routine are pre-scaled, each dimension (and the time-stamp) is scaled by a different constant. The constants are chosen so that after scaling, one unit of deviation in one dimension is as undesirable as one unit of jitter in other dimensions.

We have integrated the monitoring routine in an adaptive testing procedure for Simulink blocks based on Algorithm 1. The output of Algorithm 1 is compared against tolerance levels (e.g., maximum allowed jitter) given by the engineering requirements. In the following, we evaluate the effectiveness of the Skorokhod metric in conformance testing of Simulink applications.

Skorokhod Distance Computation Benchmark. We first show that the window-based implementation is efficient using the following benchmark. Fig. 1

Table 1. Computation of $\mathcal{D}_S(\pi_1, \pi_2)$, where π_1 is a trace of system $\mathfrak{A}_1$ described in Fig. 1, and π_2 is a trace of system $\mathfrak{A}_2$, which is $\mathfrak{A}_1$ with an actuation delay. $\mathcal{D}_2$ is the pointwise L_2 distance. Both π_1 and π_2 contain equally spaced 2001 time points over a simulation horizon of 100 seconds.

Window size	Avg. $\mathcal{D}_S$	Avg. Time taken (secs)		$\frac{\mathcal{D}_2 - \mathcal{D}_S}{\mathcal{D}_2}$		
		Computation	Monitoring	Max.	Avg.	Std. dev.
20	8.58	0.81	0.13	0.11	0.03	0.03
40	8.35	1.55	0.26	0.23	0.06	0.06
60	8.09	2.31	0.39	0.34	0.1	0.09
80	7.88	3.05	0.52	0.38	0.1	0.11
100	7.72	3.77	0.64	0.38	0.1	0.11

shows a hybrid dynamical system $\mathfrak{A}_1$ consisting of two water tanks, each with an outlet from which water drains at a constant rate d_j. Both tanks share a single inlet pipe that is switched between the tanks, filling only one tank at any given time at a constant inflow rate of i. When the water-level in tank j falls below level ℓ_j, the pipe switches to fill it. The drain and inflow rates d_1, d_2 and i are assumed to be inputs to the system. Now consider a version $\mathfrak{A}_2$ that incorporates an actuation delay that is a function of the inflow rate. This means that after the level drops to ℓ_j for tank j, the inlet pipe starts filling it only after a finite time. $\mathfrak{A}_1$ and $\mathfrak{A}_2$ have the same initial water level. We perform a fixed number of simulations by systematically choosing drain and inflow rates d_1, d_2, i to generate traces (water-level vs. time) of both systems and compute their Skorokhod distance. We summarize the results in Table 1.

Recall that the Skorokhod distance computation involves a sequence of monitoring calls with different δ values picked by a binary-search procedure. Thus, the total time to compute $\mathcal{D}_S$ is the sum over the computation times for individual monitoring calls plus some bookkeeping. In Table 1, we make a distinction between the average time to monitor traces (given a δ value), and the average time to compute $\mathcal{D}_S$. There are an average of 6 monitoring calls per $\mathcal{D}_S$ computation. We ran 64 simulations by choosing different input values, and then computing $\mathcal{D}_S$ for increasing window sizes. As the window size increases, the average $\mathcal{D}_S$ decreases and the computation time increases linearly, as expected from Theorem 1. Finally, the Skorokhod distance can be significantly smaller than the simpler metric $\mathcal{D}_2$ (defined as the maximum of the pointwise L_2 norm). This discrepancy becomes more prominent with increased window size. With a window size of 100, the variation between $\mathcal{D}_S$ and $\mathcal{D}_2$ was up to 38% (mean difference of 10% with std. deviation of 11%).

Case Study I: LQR-based Controller. The first case study for conformance testing is an aircraft pitch control application taken from the openly accessible control tutorials for Matlab and Simulink [27]. The authors describe a linear dynamical system of the form: $\dot{\mathbf{x}} = (A - BK)\mathbf{x} + B\theta_{des}$. Here, $\mathbf{x}$ describes the vector of continuous state variables and θ_{des} is the desired reference provided as

Table 2. Variation in Skorokhod Distance with changing sampling time for an aircraft pitch control system with an LQR-based controller. Time taken indicates the total time spent in computing the upper bound on the Skorokhod distance across all simulations. We choose a window size chosen of 150 samples and simulate the system for 5 seconds with a variable-step solver.

Controller Sample-Time (seconds)	Skorokhod distance	Time taken (seconds) to compute $\mathcal{D}_S$	Number of simulations
0.01	0.012	232	104
0.05	0.049	96	104
0.1	0.11	70	106
0.3	0.39	45	104
0.5	1.51	40	101

an external input. One of the states in the $\mathbf{x}$ vector is the pitch angle θ, which is also the system output. The controller gain matrix K is computed using the linear quadratic regulator method [5], a standard technique from optimal control. We are interested in studying a digital implementation of the continuous-time controller obtained using the LQR method. To do so, we consider sampled-data control where the controller samples the plant output, computes, and provides the control input to the plant every Δ seconds. To model sensor delay, we add a fixed delay element to the system; thus, the overall system now represents a delay-differential equation.

Control engineers are typically interested in the step response of a system. In particular, quantities such as the overshoot/undershoot of the output signal (maximum positive/negative deviation from a reference value) and the settling time (time it takes for transient behaviors to converge to some small region around the reference value) are of interest. Given a settling time and overshoot for the first system, we would like the second system to display similar characteristics. We remark that both of these properties can be expressed in STL (and hence in TLTL($\mathcal{F}_\mathsf{T}, \mathcal{F}_\mathsf{S}$)), see [21] for details. We quantify system conformance (and thereby adherence to requirements) in terms of the Skorokhod distance, or, in other words, maximum permitted time/space-jitter value δ. For this system, we know that at nominal conditions, the settling time is approximately 2.5 seconds, and that we can tolerate an increase in settling time of about 0.5 seconds. Thus, we chose a time-scaling factor of $2 = \frac{1}{0.5}$. We observe that the range of θ is about 0.4 radians, and specify an overshoot of 20% of this range as being permissible. Thus, we pick a scaling factor of $\frac{1}{0.08}$ for the signal domain. In other words, Skorokhod distance $\delta = 1$ corresponds to either a time-jitter of 0.5 seconds, or a space-discrepancy of 0.08 radians.

We summarize the results of conformance testing for different values of sampling time Δ in Table 2. As expected, the conformance increases with increasing Δ. The time taken to compute the Skorokhod distance decreases with increasing Δ, as the number of time-points in the two traces decreases.

Table 3. Conformance testing for closed-loop A/F ratio controller at different engine speeds. We scale the signals such that 0.5 seconds of time-jitter is treated equivalent to 10% of the steady-state value (14.7) of the A/F ratio signal. The simulation traces correspond to a time horizon of 10 seconds and the window size is 300.

Engine speed (rpm)	Skorokhod distance	Computation Time (secs)	Total Time Taken (secs)	Number of simulations
1000	0.31	218	544	700
1500	0.20	240	553	700
2000	0.27	223	532	700

Case Study II: Air-Fuel Ratio Controller. In [21], the authors present three systems representing an air-fuel ratio (λ) controller for a gasoline engine, that regulate λ to a given reference value of $\lambda_{\text{ref}} = 14.7$. Of interest to us are the second and the third systems. The former has a continuous-time plant model with highly nonlinear dynamics, and a discrete-time controller model. In [22], the authors present a version of this system where the controller is also continuous. We take this to be $\mathfrak{A}_1$. The third system in [21] is a continuous-time closed-loop system where all the system differential equations have right-hand-sides that are polynomial approximations of the nonlinear dynamics in $\mathfrak{A}_1$. We call this polynomial dynamical system $\mathfrak{A}_2$. The rationale for these system versions is as follows: existing formal methods tools cannot reason about highly nonlinear dynamical systems, but tools such as Flow* [10], C2E2 [16], and CORA [3] demonstrate good capabilities for polynomial dynamical systems. Thus, the hope is to analyze the simpler systems instead. In [21], the authors comment that the system transformations are not accompanied by formal guarantees. By quantifying the difference in the system behaviors, we hope to show that if the system $\mathfrak{A}_2$ satisfies the temporal requirements φ presented in [21], then $\mathfrak{A}_1$ satisfies a relaxation of φ. We pick a scaling factor of 2 for the time domain, as a time-jitter of 0.5 seconds is the maximum deviation we wish to tolerate in the settling time, and pick $0.68 = \frac{1}{0.1 * \lambda_{\text{ref}}}$ as the scaling factor for λ (which corresponds to the worst case tolerated discrepancy in the overshoot).

Table 3 summarizes the results of conformance testing for these systems. In [14], the authors posed a challenge problem for conformance testing. They reported that the original nonlinear system and the approximate polynomial system both satisfy the STL requirements specifying overshoot/undershoot and settling time. We, however, found an input that causes the outputs of the two systems to have a high Skorokhod distance. Thus, comparing the two systems by considering equi-satisfaction of a given set of STL requirements such as overshoot/undershoot and settling time may not always be sufficient. Our experiment indicates that the Skorokhod metric may be a better measure of conformance.

Case Study III: Engine Timing Model. The Simulink demo palette from Mathworks [26] contains a system representing a four-cylinder spark ignition internal combustion engine based on a model by Crossley and Cook [11]. This

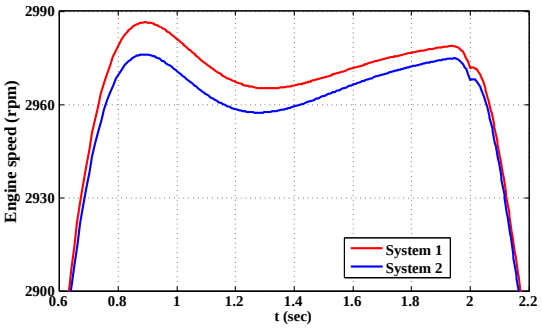

Fig. 2. Outputs showing a Skorokhod distance of 1.04.

system is then enhanced by adding a proportional plus integral (P+I) control law. The integrator is used to adjust the steady-state throttle as the desired engine speed set-point changes, and the proportional term compensates for phase lag introduced by the integrator. In an actual implementation of such a system, such a P+I controller is implemented using a discrete-time integrator. Such integrator blocks are typically associated with a particular numerical integration technique, e.g., forward-Euler, backward-Euler, trapezoidal, etc. It is expected that different numerical techniques will produce slight variation in the results. We wish to quantify the effect of using different numerical integrators in a closed-loop setting. We checked if the user-provided tolerance of $\delta = 1.0$ is satisfied by systems $\mathfrak{A}_1$ and $\mathfrak{A}_2$, where $\mathfrak{A}_1$ is the original system provided in [26] and $\mathfrak{A}_2$ is a modified system that uses the backward Euler method to compute the discrete-time integral in the controller. We scale the outputs in such a way that a value discrepancy of 1% of the the output range ($\sim$ 1000) is equivalent to a time discrepancy of 0.1 seconds. These values are chosen to bias the search towards finding signals that have a small time jitter. This is an interesting scenario for this case study where the two systems are equivalent except for the underlying numerical integration solver. We find the signal shown in Fig. 2, for which we find output traces with Skorokhod distance 1.04. The experiment uses 296 simulations and the total time taken to find the counterexample is 677 seconds.

5 Conclusion

We argue that the Skorokhod metric provides a robust basis for checking conformance between dynamical systems. We showed that it provides transference of a rich class of temporal logic properties and that it can be computed efficiently, both in theory and in practice. Our experiments indicate that conformance checking using the Skorokhod metric can be integrated into a testing flow for Simulink models and can find non-conformant behaviors effectively, allowing for independent weighing of time and value distortions.

References

1. H. Abbas and G.E. Fainekos. Formal property verification in a conformance testing framework. In *MEMOCODE*. To Appear, 2014.
2. H. Abbas, B. Hoxha, G.E. Fainekos, J.V. Deshmukh, J. Kapinski, and K. Ueda. Conformance testing as falsification for cyber-physical systems. *CoRR*, abs/1401.5200, 2014.
3. M. Althoff. Reachability analysis of nonlinear systems using conservative polynomialization and non-convex sets. In *HSCC 13*, pages 173–182, 2013.
4. R. Alur and T.A. Henzinger. A really temporal logic. *J. ACM*, 41(1):181–204, 1994.
5. Brian D. O. Anderson. *Optimal Control: Linear Quadratic Methods*. Dover Books on Engineering. Dover Publications, 2007.
6. P. Bouyer, F. Chevalier, and N. Markey. On the expressiveness of TPTL and MTL. In *FSTTCS 05*, LNCS 3821, pages 432–443. Springer, 2005.
7. M.S. Branicky. *Studies in hybrid systems: modeling, analysis, and control*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 1995.
8. M. Broucke. Regularity of solutions and homotopic equivalence for hybrid systems. In *IEEE Conference on Decision and Control*, volume 4, pages 4283–4288, Dec 1998.
9. P. Caspi and A. Benveniste. Toward an approximation theory for computerised control. In *EMSOFT*, pages 294–304. Springer, 2002.
10. X. Chen, E. Ábrahám, and S. Sankaranarayanan. Flow*: An analyzer for non-linear hybrid systems. In *CAV 13*, pages 258–263, 2013.
11. P.R. Crossley and J.A. Cook. A nonlinear engine model for drivetrain system development. In *International Conference on Control*, pages 921–925. IET, 1991.
12. J.M. Davoren. Epsilon-tubes and generalized Skorokhod metrics for hybrid paths spaces. In *HSCC*, LNCS 5469, pages 135–149. Springer, 2009.
13. J.V. Deshmukh, R. Majumdar, and V.S. Prabhu. Quantifying conformance using the Skorokhod metric (full version). *CoRR*, abs/1505.05832, 2015.
14. X. Jinand J. Deshmukh, J. Kapinski, K. Ueda, and K. Butts. Benchmarks for model transformations and conformance checking. In *ARCH 14*, 2014.
15. A. Donzé and O. Maler. Robust satisfaction of temporal logic over real-valued signals. In *FORMATS*, LNCS 6246, pages 92–106. Springer, 2010.
16. P.S. Duggirala, S. Mitra, and M. Viswanathan. Verification of annotated models from executions. In *EMSOFT 13*, page 26, 2013.
17. A. Girard, G. Pola, and P. Tabuada. Approximately bisimilar symbolic models for incrementally stable switched systems. *IEEE Trans. Automat. Contr.*, 55(1):116–126, 2010.
18. E. Haghverdi, P. Tabuada, and G.J. Pappas. Bisimulation relations for dynamical, control, and hybrid systems. *Theor. Comput. Sci.*, 342(2-3):229–261, 2005.
19. M. Hennessy and R. Milner. Algebraic laws for nondeterminism and concurrency. *J. ACM*, 32(1):137–161, 1985.
20. M.R. Henzinger, T.A. Henzinger, and P.W. Kopke. Computing simulations on finite and infinite graphs. In *FOCS: Foundations of Computer Science*, pages 453–462. IEEE Computer Society, 1995.
21. X. Jin, J.V. Deshmukh, J. Kapinski, K. Ueda, and K. Butts. Powertrain control verification benchmark. In *HSCC 14*, pages 253–262, 2014.
22. J. Kapinski, J.V. Deshmukh, S. Sankaranarayanan, and N. Arechiga. Simulation-guided lyapunov analysis for hybrid dynamical systems. In *HSCC 14*, pages 133–142. ACM, 2014.

23. R. Koymans. Specifying real-time properties with metric temporal logic. *Real-Time Syst.*, 2(4):255–299, 1990.
24. R. Majumdar and V.S. Prabhu. Computing the Skorokhod distance between polygonal traces (full paper). *CoRR*, abs/1410.6075, 2014.
25. R. Majumdar and V.S. Prabhu. Computing the Skorokhod distance between polygonal traces. In *HSCC*. ACM, 2015.
26. The Mathworks. Engine timing model with closed loop control.
27. William Messner and Dawn Tilbury. Control tutorials for matlab and simulink.
28. R. Milner. *A Calculus of Communicating Systems*. LNCS 92. Springer, 1980.
29. D. Sangiorgi and J. Rutten. *Advanced Topics in Bisimulation and Coinduction*. Cambridge University Press, 2011.
30. P. Tabuada. *Verification and Control of Hybrid Systems - A Symbolic Approach*. Springer, 2009.

Appendix

A. Transference Formalism and Proofs

Example 6 (Freeze Quantification). Suppose we want to express that whenever the event Q occurs, it is followed later by R, and then by S, such that the time difference between occurrences of Q and R is at most 5, and also the time difference between occurrences of Q and S is at most 10. This can be expressed in TLTL($\mathcal{F}_{\mathsf{T}}$) as

$$\Box\Big(x.Q \rightarrow \Diamond\big(y.\big[R \wedge (y \leq x+5) \wedge \Diamond\,(z.\,(S \wedge z \leq x+10))\,\big]\big)\Big).$$

Thus, freeze quantification, by giving a mechanism to bind times to variables, allows us to relate, with several constraints, far apart events. □

Example 7 (Freeze Quantification Functions). Suppose we want to express that whenever the event Q occurs, it must be followed by a response R within time λ^{t_Q} for some $\lambda > 1$ where t_Q is the time at which Q occurred; thus, the later Q occurs the more delay we can tolerate in the response time. The requirement can be expressed as $x.\,\big(Q \rightarrow \Diamond\big(y.\,(R \wedge 0 \leq y \leq \lambda^x)\,\big)\big)$. □

Example 8 (δ-relaxation for Bounded Temporal Operators – MTL). We demonstrate how δ-relaxation operates on bounded time constraints through an example. Consider the MTL formula $\phi = Q\,\mathcal{U}_{\,[a,b]}\,R$. The δ-relaxation of this formula over the interval $\mathbb{R}_+$ is $Q\,\mathcal{U}_{\,[a-2\cdot\delta\,,\,b+2\cdot\delta]}\,R$. This can be seen as follows. The formula ϕ can be written in TLTL syntax as:

$$x.Q\,\mathcal{U}\,y.\,((y \leq x+b) \wedge (y \geq x+a) \wedge R)\,.$$

The δ-relaxation of this formula according to Definition 4 is:

$$\begin{aligned}
\mathsf{rx}^{\delta}_{\mathbb{R}_+}\,(x.Q\,\mathcal{U}\,y.\,((y \leq x+b) \wedge (y \geq x+a) \wedge R)) \;&= \\
&= \mathsf{rx}^{\delta}_{\mathbb{R}_+}\,(x.Q\,\mathcal{U}\,y.\,((y-x-b \leq 0) \wedge (y-x-a \geq 0) \wedge R)) \\
&= x.Q\,\mathcal{U}\,y.\,\begin{pmatrix}(y-x-b-2\cdot\delta \leq 0)\;\wedge \\ (y-x-a+2\cdot\delta \geq 0) \wedge R\end{pmatrix} \\
&\qquad \text{since the Lipschitz constant of } y-x-c \text{ is } 2 \\
&\qquad \text{(for any constant } c\text{) for the } L_\infty \text{ norm.} \\
&= x.Q\,\mathcal{U}\,y.\,((y \leq x+b+2\cdot\delta) \wedge (y \geq x+a-2\cdot\delta) \wedge R) \\
&= Q\,\mathcal{U}_{\,[a-2\cdot\delta,b+2\cdot\delta]}\,R.
\end{aligned}$$

Thus, the time constraint interval boundaries are relaxed by $2\cdot\delta$. The factor of 2 arises because there are two contributing factors: the starting time of Q can be "pulled back" by δ, and the time of R can be postponed by δ; thus, the time duration in between Q and R can increase (and similarly can decrease) by $2\cdot\delta$. □

Removing Negation using the $\mathcal{W}$ Operator. The following identities hold relating the $\mathcal{W}$ operator to the $\mathcal{U}$ operator

1. $\phi_1 \,\mathcal{U}\, \phi_2 \equiv \neg\,(\neg(\phi_2)\,\mathcal{W}\,(\neg\phi_1 \wedge \neg\phi_2)\,)$; and
2. $\phi_1 \,\mathcal{W}\, \phi_2 \equiv \neg\,(\neg(\phi_2)\,\mathcal{U}\,(\neg\phi_1 \wedge \neg\phi_2)\,)$.

Informally, the first identity states that $\neg(\phi_1 \,\mathcal{U}\, \phi_2)$ holds iff either (i) ϕ_2 never holds; or (ii) there is a point where ϕ_1 is false, and at that point and all points before it, ϕ_2 has remained false. The second identity is similar. The first identity above allows us to "push" the negations down using the $\mathcal{W}$ operator. The mechanism for the three interesting cases is below.

$$\begin{aligned}
\neg\,(f_{\mathsf{T}}(x_1,\ldots,x_l) \sim 0) \;\equiv\; & f_{\mathsf{T}}(x_1,\ldots,x_l)\ \mathsf{neg}(\sim)\,0, \\
& \text{where, for } \sim \in \{\leq, <, \geq, >\} \text{ we have} \\
& \mathsf{neg}(\leq) \text{ to be } >; \qquad \mathsf{neg}(<) \text{ to be } \geq; \\
& \mathsf{neg}(\geq) \text{ to be } <; \qquad \mathsf{neg}(>) \text{ to be } \leq \\
\neg(x.\psi) \;\equiv\; & x.\neg(\psi) \\
\neg\,(\phi_1 \,\mathcal{U}\, \phi_2) \;\equiv\; & \neg(\phi_2)\,\mathcal{W}\,(\neg\phi_1 \wedge \neg\phi_2)
\end{aligned}$$

Proposition 1. *The function* rx *is a relaxation on* $\mathrm{TLTL}(\mathcal{F}_{\mathsf{T}})$ *formulae,* i.e. *if a timed propositional trace* $\pi \models \phi$ *for a* $\mathrm{TLTL}(\mathcal{F}_{\mathsf{T}})$ *formula* ϕ*, then* $\pi \models \mathsf{rx}^{\delta}_{J}(\phi)$ *for all* $\delta > 0$ *and non-empty intervals* J*.*

Proof. Observe that, over the predicates $f_{\mathsf{T}}(x_1,\ldots,x_l) \sim 0$, the function rx is indeed a relaxation, *i.e*, if $f_{\mathsf{T}}(t_1,\ldots,t_l) \sim 0$ for values $t_1,\ldots,t_l$, then $\mathsf{rx}^{\delta}_{J}\,(f_{\mathsf{T}}(t_1,\ldots,t_l)) \sim 0$) also holds. The result follows by a straightforward induction argument. □

Proof of Theorem 2. Let $\mathsf{untime}(\phi)$ be the formula where all freeze variable constraints are replaced by TRUE (*e.g.* $\mathsf{untime}(x.(Q \wedge x < 5))$ is $x.(Q \wedge \text{TRUE})$). Since $\mathcal{D}(\pi,\pi') < \delta$, we have that there exists a retiming $r : \mathsf{tdom}(\pi) \mapsto \mathsf{tdom}(\pi')$ such that

$$\pi(t) = \pi'(\mathsf{r}(t)). \tag{3}$$

Thus, both π and π' have the same untimed propositional sequence. This implies that both π and π' satisfy $\mathsf{untime}(\phi)$, which can be shown by an induction argument. The interesting cases are for the $\mathcal{U}$ and $\mathcal{W}$ operators. We sketch the argument for the $\mathcal{U}$ case (the argument for $\mathcal{W}$ is similar). The time environment $\mathcal{E}'$ for π' assigns the time $\mathsf{r}(t_x)$ to the freeze variable x where the witnessing freeze variable environment $\mathcal{E}$ for $\pi \models \phi$ assigns t_x to x. Let a Let $\pi \models_{\mathcal{E}} \phi_1 \,\mathcal{U}\, \phi_2$, and let t be the time value which demonstrates this satisfaction (as in Definition 3), with the corresponding freeze variable environment $\mathcal{E}$. To show $\pi' \models_{\mathcal{E}'} \phi_1 \,\mathcal{U}\, \phi_2$, we pick the time $\mathsf{r}(t)$, with the environment $\mathcal{E}'$ for π' which assigns the time $\mathsf{r}(t_x)$ to the freeze variable x where $\mathcal{E}(x) = t_x$. It can be checked that, due to Equation 3, we have (i) $\mathsf{r}(t) \geq \mathcal{E}'(x)$, for a freeze variable x in $\phi_1 \,\mathcal{U}\, \phi_2$ (which was previously bound); (i) ${\pi'}^{\mathsf{r}(t)} \models_{\mathcal{E}'} \phi_2$; and (ii) for all $t'_0 \leq t^{\dagger} < \mathsf{r}(t)$, we have ${\pi'}^{t^{\dagger}} \models_{\mathcal{E}'} \phi_1 \vee \phi_2$. Thus, $\mathsf{r}(t)$, and $\mathcal{E}'$ demonstrate that $\pi' \models_{\mathcal{E}'} \phi_1 \,\mathcal{U}\, \phi_2$.

We now check what is the relaxation needed on the original freeze variable constraints so that π' satisfies the relaxed constraints. Without loss of generality,

assume that each freeze variable x is only quantified once in ϕ, *i.e.* once it is bound to a value by "$x.$", it is not "re-bound" with another application of "$x.$".

Let κ_π denote an assignment of time values (from I) to the freeze variables such that all the freeze variable constraints in ϕ are satisfied, *i.e.* κ_π is an time environment witness to the satisfaction of ϕ by π. Consider a free variable assignment $\kappa_{\pi'}$ corresponding to κ_π, where $\kappa_{\pi'}(x) = \mathsf{r}(\kappa_\pi(x))$. This is a legal variable assignment compatible with some $\mathcal{U}$, $\mathcal{W}$ time witnesses which demonstrate that π' satisfies $\mathsf{untime}(\phi)$, as shown previously. Observe that by the existence of a retiming function, for all freeze variables x occurring in ϕ, we have that $|\kappa_{\pi'}(x) - \kappa_\pi(x)| < \delta$.

Since the time *values* of variables are different in κ_π and $\kappa_{\pi'}$, the original freeze constraints (*e.g.* $x < 5$) in ϕ might not be satisfied with the assignment $\kappa_{\pi'}$. Consider a freeze variable constraint $f_{\mathsf{T}}(x_1, \ldots, x_l) \sim 0$ in ϕ. We know that $f_{\mathsf{T}}(\kappa_\pi(x_1), \ldots, \kappa_\pi(x_l)) \sim 0$ is true. As $|\kappa_{\pi'}(x) - \kappa_\pi(x)| \leq \delta$ for all freeze variables x occurring in ϕ, by the definition of relaxation, we have that

1. $f_{\mathsf{T}}(\kappa_\pi(x_1), \ldots, \kappa_\pi(x_l)) + K_{\mathsf{T}}(\delta) \sim 0$ if $\sim \in \{>, \geq\}$; and
2. $f_{\mathsf{T}}(\kappa_\pi(x_1), \ldots, \kappa_\pi(x_l)) - K_{\mathsf{T}}(\delta) \sim 0$ if $\sim \in \{<, \leq\}$.

This implies that $\kappa_{\pi'}$ is also a witness to the satisfaction of $\mathsf{rx}^{\delta}_{I_{\pi,\pi'}}(\phi)$ by π'. Thus, $\pi' \models \mathsf{rx}^{\delta}_{I_{\pi,\pi'}}(\phi)$. □

Example 4 details. Since π satisfies ϕ, we must have time-stamps t_x, t_y, t_z bound to x, y, z respectively so that with these assignments, the formula ϕ is satisfied. Since π' is δ close to π, for every $\epsilon > 0$, there is a retiming from π to π' such that the times t_x, t_y, t_z in π are mapped to t'_x, t'_y, t'_z in π' such that (a) $|t_x - t'_x| \leq \delta + \epsilon$; and (b) $|t_y - t'_y| \leq \delta + \epsilon$; and (c) $|t_z - t'_z| \leq \delta + \epsilon$. Let $\delta' = \delta + \epsilon$.

The sum $(t'_x - t'_y)^2 + (t'_y - t'_z)^2 + (t'_z - t'_x)^2$ is

$$
\begin{aligned}
&= \big((t'_x - t_x) + (t_x - t_y) + (t_y - t'_y)\big)^2 \;+\; \big((t'_y - t_y) + (t_y - t_z) + (t_z - t'_z)\big)^2 \;+ \\
&\qquad\qquad ((t'_z - t_z) + (t_z - t_x) + (t_x - t'_x))^2 \\
&= 2\left((t'_x - t_x)^2 + (t'_y - t_y)^2 + (t'_z - t_z)^2\right) \;+\; (t_x - t_y)^2 + (t_y - t_z)^2 + (t_z - t_x)^2 + \\
&\qquad 2\left((t'_x - t_x)(t_x - t_y) + (t_y - t'_y)(t_x - t_y) + (t'_x - t_x)(t_y - t'_y)\right) + \\
&\qquad 2\left((t'_y - t_y)(t_y - t_z) + (t_z - t'_z)(t_y - t_z) + (t'_y - t_y)(t_z - t'_z)\right) + \\
&\qquad 2\left((t'_z - t_z)(t_z - t_x) + (t_x - t'_x)(t_z - t_x) + (t'_z - t_z)(t_x - t'_x)\right) \\
&\leq 6\delta'^2 + d + 2\delta'|t_x - t_y| + 2\delta'^2 + 2\delta'|t_y - t_z| + 2\delta'^2 + 2\delta'|t_z - t_x| + 2\delta'^2 \\
&= d + 12\delta'^2 + 4\delta'\left(|t_x - t_y| + |t_y - t_z| + |t_z - t_x|\right) \\
&\leq d + 12 \cdot \delta'^2 + 4\sqrt{3} \cdot \delta' \cdot \sqrt{d}
\end{aligned}
$$

In the last step above, we use the inequality: $|a| + |b| + |c| \leq \sqrt{3} \cdot \sqrt{a^2 + b^2 + c^2}$ This inequality is obtained by applying the Cauchy-Schwarz inequality to the tuples $(|a|, |b|, |c|)$ and $(1, 1, 1)$. Thus, by Theorem 2, for every $\epsilon > 0$, we have

$$\pi' \models x.\left(Q \rightarrow \Diamond\big(y.\left(R \wedge \Diamond\left[z.\left(S \wedge \left((y-x)^2 + (z-y)^2 + (z-x)^2 \leq d^{\dagger}\right)\right)\right]\right)\big)\right)$$

where $d^\dagger = d + 12 \cdot \delta'^2 + 4\sqrt{3} \cdot \delta' \cdot \sqrt{d}$. □

Definition 6 (δ-relaxation of TLTL$(\mathcal{F}_\mathsf{T}, \mathcal{F}_\mathsf{S})$ formulae). *Let ϕ be a* TLTL$(\mathcal{F}_\mathsf{T}, \mathcal{F}_\mathsf{S})$ *formula in which negations appear only on the prepositional symbols . The δ relaxation of ϕ (for $\delta \geq 0$), denoted $\mathsf{rx}^\delta_{I_{\mathcal{F}_\mathsf{T}}, \mathbf{I}_{V_\mathsf{S}}}(\phi)$ is defined as follows, where $I_{\mathcal{F}_\mathsf{T}}$, a closed subset of $reals_+$, is the domain of the variables in V_T; and $\mathbf{I}_{V_\mathsf{S}}$ is a mapping from V_S to closed intervals of $\mathbb{R}$ such that $\mathbf{I}_{V_\mathsf{S}}(z)$ denotes the domain of z.*

$$\mathsf{rx}^\delta_{I_{\mathcal{F}_\mathsf{T}}, \mathbf{I}_{V_\mathsf{S}}}(\text{TRUE}) = \text{TRUE}; \qquad \mathsf{rx}^\delta_{I_{\mathcal{F}_\mathsf{T}}, \mathbf{I}_{V_\mathsf{S}}}(\text{FALSE}) = \text{FALSE};$$
$$\mathsf{rx}^\delta_{I_{\mathcal{F}_\mathsf{T}}, \mathbf{I}_{V_\mathsf{S}}}(\phi_1 \wedge \phi_2) = \mathsf{rx}^\delta_{I_{\mathcal{F}_\mathsf{T}}, \mathbf{I}_{V_\mathsf{S}}}(\phi_1) \wedge \mathsf{rx}_\delta(\phi_2);$$
$$\mathsf{rx}^\delta_{I_{\mathcal{F}_\mathsf{T}}, \mathbf{I}_{V_\mathsf{S}}}(\phi_1 \vee \phi_2) = \mathsf{rx}^\delta_{I_{\mathcal{F}_\mathsf{T}}, \mathbf{I}_{V_\mathsf{S}}}(\phi_1) \vee \mathsf{rx}^\delta_{I_{\mathcal{F}_\mathsf{T}}, \mathbf{I}_{V_\mathsf{S}}}(\phi_2);$$
$$\mathsf{rx}^\delta_{I_{\mathcal{F}_\mathsf{T}}, \mathbf{I}_{V_\mathsf{S}}}(x.\psi) = x.\, \mathsf{rx}^\delta_{I_{\mathcal{F}_\mathsf{T}}, \mathbf{I}_{V_\mathsf{S}}}(\psi);$$
$$\mathsf{rx}^\delta_{I_{\mathcal{F}_\mathsf{T}}, \mathbf{I}_{V_\mathsf{S}}}(\phi_1 \,\mathcal{U}\, \phi_2) = \mathsf{rx}^\delta_{I_{\mathcal{F}_\mathsf{T}}, \mathbf{I}_{V_\mathsf{S}}}(\phi_1) \,\mathcal{U}\, \mathsf{rx}^\delta_{I_{\mathcal{F}_\mathsf{T}}, \mathbf{I}_{V_\mathsf{S}}}(\phi_2);$$
$$\mathsf{rx}^\delta_{I_{\mathcal{F}_\mathsf{T}}, \mathbf{I}_{V_\mathsf{S}}}(\phi_1 \,\mathcal{W}\, \phi_2) = \mathsf{rx}^\delta_{I_{\mathcal{F}_\mathsf{T}}, \mathbf{I}_{V_\mathsf{S}}}(\phi_1) \,\mathcal{W}\, \mathsf{rx}^\delta_{I_{\mathcal{F}_\mathsf{T}}, \mathbf{I}_{V_\mathsf{S}}}(\phi_2)$$

$$\mathsf{rx}^\delta_{I_{\mathcal{F}_\mathsf{T}}, \mathbf{I}_{V_\mathsf{S}}}(f_U(z_1, \ldots, z_l)) \sim 0) = \begin{cases} f_U(z_1, \ldots, z_l) + K_{f_U}(\delta) \sim 0 & \text{if } \sim \in \{>, \geq\}; \\ f_U(z_1, \ldots, z_l) - K_{f_U}(\delta) \sim 0 & \text{if } \sim \in \{<, \leq\}; \end{cases}$$

where $U \in \{\mathsf{T}, \mathsf{S}\}$ with K_{f_U} being as in Definition 4; and $K_{f_\mathsf{S}} : \left[0, \max_{z \in V_\mathsf{S}} |\max \mathbf{I}_{V_\mathsf{S}}(z) - \min \mathbf{I}_{V_\mathsf{S}}(z)|\right] \mapsto \mathbb{R}_+$ is a function such that:

$$K_{f_\mathsf{S}}(\delta) = \sup_{\substack{z_i \in \mathbf{I}_{V_\mathsf{S}}(z_i);\, z'_i \in \mathbf{I}_{V_\mathsf{S}}(z'_i) \\ \text{for all } i}} \left\{ \left| \begin{array}{c} f_\mathsf{S}(z_1, \ldots, z_l) \\ - \\ f_\mathsf{S}(z'_1, \ldots, z'_l) \end{array} \right| \;\; s.t.\ |z_i - z'_i| \leq \delta \text{ for all } i \right\}$$

□

The functions $K_{f_\mathsf{S}}(\delta)$ define the maximal change in the value of f_S that can occur when the input variables can vary by δ. The role of $\mathbf{I}_{V_\mathsf{S}}$ in the above definition is to restrict the domain of the signal variables in order to obtain the least possible bounds relaxation bounds on the signal constraints; as was done in Definition 4 for the freeze variables.

Proposition 2. *The function $\mathsf{rx}^\delta_{I_{\mathcal{F}_\mathsf{T}}, \mathbf{I}_{V_\mathsf{S}}}$ is a relaxation on* TLTL$(\mathcal{F}_\mathsf{T}, \mathcal{F}_\mathsf{S})$ *formulae,* i.e. *if a timed $\mathbb{R}^n$-valued trace $\pi \models \phi$ for a* TLTL$(\mathcal{F}_\mathsf{T}, \mathcal{F}_\mathsf{S})$ *formula ϕ, then $\pi \models \mathsf{rx}^\delta_{I_{\mathcal{F}_\mathsf{T}}, \mathbf{I}_{V_\mathsf{S}}}(\phi)$.*

Proof. The proof is similar to the proof of Proposition 1. □

Proof of Theorem 3. The proof use the result for the propositional case, Theorem 2. We construct the propositions p_{f_S} defined to be $\mathsf{rx}^\delta_{I_{\mathcal{F}_\mathsf{T}}, \mathbf{I}_{V_\mathsf{S}}}(f_\mathsf{S}(\overline{y})) \sim 0)$

for the constraints over V_{S} in the formula ϕ; and define the TLTL($\mathcal{F}_{\mathsf{T}}$) formula $\phi_{\mathcal{P}}$ as that obtained from ϕ by syntactically replacing each constraint $f_{\mathsf{S}}(\overline{y}) \sim 0$ in ϕ by $p_{f_{\mathsf{S}}}$. Let $\mathcal{P}_{\mathsf{S}}$ denote all such predicates for ϕ. We obtain the timed $\mathcal{P}_{\mathsf{S}}$ propositional traces $\pi_{P_{\mathsf{S}}}, \pi'_{P_{\mathsf{S}}}$ from π, π' by mapping to propositions. By the definition of the Skorokhod distance, the distance between $\pi_{\mathcal{P}_{\mathsf{S}}}$ and $\pi'_{\mathcal{P}_{\mathsf{S}}}$ is less than δ. By Theorem 2, $\pi'_{\mathcal{P}_{\mathsf{S}}} \models \phi_{\mathcal{P}}$. This implies $\pi' \models \mathsf{rx}^{\delta}_{I_{\mathcal{F}_{\mathsf{T}}}, \mathbf{I}_{V_{\mathsf{S}}}}(\phi)$. □

B. Details on Case Studies

LQR-based pitch controller. The aircraft pitch controller system has 3 state variables, and the state vector $\mathbf{x} = [\alpha\ q\ \theta]$, where α is the angle of attack, q is the pitch rate, and θ is the pitch angle. The system has a single input δ (the elevator deflection angle). In deriving the control law, the designers use the state feedback law to substitute $\delta = \theta_{des} - K\mathbf{x}$, where θ_{des} is the desired pitch angle. The resulting dynamical equations of the system are of the form $\dot{\mathbf{x}} = (A - BK)\mathbf{x} + B\theta_{des}$, and the output of the system is the state variable θ. Note that the K matrix is the gain matrix resulting from the LQR control design technique. The values of the A, B and K matrices are as given below:

$$A = \begin{bmatrix} -0.313 & 56.7 & 0 \\ -0.0139 & -0.426 & 0 \\ 0 & 56.7 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 0.232 \\ 0.0203 \\ 0 \end{bmatrix}$$

$$K = [-0.6435 \quad 169.6950 \quad 7.0711]$$

Air-Fuel Ratio Controller. The Air-Fuel (A/F) ratio control systems that we consider are simplified versions of industrial-scale models. Both versions have 2 exogenous inputs, and 4 continuous states. The inputs are engine speed (measured in rpm) and the throttle angle (in degrees). The throttle angle is a user input, and it is common to assume a series of pulses or steps as throttle angle inputs. The engine speed is considered an input to avoid modeling parts of the powertrain dynamics. In our experiments, we typically hold the engine speed constant. This is to mimic a common engine testing scenario involving a dynamometer, which is a device to provide external torque to the engine to maintain it at a constant speed. Of the 4 continuous states, we assume that 2 of these states are from the plant model (that encapsulates physical processes within the engine), while 2 states belong to the controller. The plant states p and λ denote intake manifold pressure and the A/F ratio respectively. The controller states p_e denotes the estimated manifold pressure (with the use of an observer) used in the feed-forward control, and the state i denotes the integrator state in the P+I feedback control. We check conformance with respect to the system output λ. For the dynamical system equations, please refer to [21, 22].