Abstraction, Discretization, and Robustness in Temporal Logic Control of Dynamical Systems*

Jun Liu
University of Sheffield
Mappin Street
Sheffield, S1 3JD, United Kingdom
j.liu@sheffield.ac.uk

Necmiye Ozay
University of Michigan
1301 Beal Avenue
Ann Arbor, MI 48109-2122, USA
necmiye@umich.edu

ABSTRACT

Abstraction-based, hierarchical approaches to control synthesis from temporal logic specifications for dynamical systems have gained increased popularity over the last decade. Yet various issues commonly encountered and extensively dealt with in control systems have not been adequately discussed in the context of temporal logic control of dynamical systems, such as inter-sample behaviors of a sampled-data system, effects of imperfect state measurements and unmodeled dynamics, and the use of time-discretized models to design controllers for continuous-time dynamical systems. We discuss these issues in this paper. The main motivation is to demonstrate the possibility of accounting for the mismatches between a continuous-time control system and its various types of abstract models used for control synthesis. We do this by incorporating additional robustness measures in the abstract models. Such robustness measures are gained at the price of either increased nondeterminism in the abstracted models or relaxed versions of the specification being realized. Under a unified notion of abstraction, we provide concrete means of incorporating these robustness measures and establish results that demonstrate their effectiveness in dealing with the above mentioned issues.

Categories and Subject Descriptors

D.2.4 [**SOFTWARE ENGINEERING**]: Software/Program Verification—*Formal methods*; I.2.8 [**ARTIFICIAL INTELLIGENCE**]: Problem Solving, Control Methods, and Search—*Control theory*; G.4 [**MATHEMATICAL SOFTWARE**]: Reliability and robustness

General Terms

Theory, Verification

*This work is supported in part by a Marie Curie Career Integration Grant PCIG13-GA-2013-617377 (to J.L.) and by University of Michigan startup funds (to N.O.).

HSCC'14, April 15–17, 2014, Berlin, Germany.

ACM 978-1-4503-2732-9/14/04 ...$15.00.
http://dx.doi.org/10.1145/2562059.2562137.

Keywords

hybrid control; temporal logic; abstraction; discretization; robustness

1. INTRODUCTION

Abstraction-based, hierarchical approaches to control synthesis for dynamical systems from high-level specifications naturally lead to hybrid feedback controllers [21]. Such approaches have gained increased popularity over the last few years (see, e.g., [3, 8, 9, 11, 12, 15, 17, 21, 22, 26, 28]). The main workflow of these approaches consists of three steps: (i) construct finite abstractions of the dynamical control systems, (ii) solve a discrete synthesis problem based on the specification and abstraction and obtain a discrete control strategy, (iii) refine the discrete control strategy to a hybrid controller that renders the dynamical system satisfy the specification. As the first step in such approaches, how to construct finite abstractions of control systems, in particular, for nonlinear systems, received special attention (see [20, 23] and references therein).

One advantage of abstraction-based methods is that they provide a feedback solution, as opposed to open-loop trajectory generation strategies [7, 25]. Feedback has the potential to reduce the effects of disturbances and deal with sensing and modeling uncertainties. One of the motivations of this paper is to establish these in the context of temporal logic control. We present a unified abstraction framework equipped with certain robustness measures to account for imperfections in measurements and/or models. In particular, we show that, when the abstract system complies with these measures (with respect to a nominal concrete dynamical system), then a discrete control strategy synthesized using the abstract system is valid for (i.e., can be implemented with correctness guarantees on) a family of dynamical systems that can be represented as the nominal dynamical system subject to uncertainty.

We demonstrate the effectiveness of this abstraction framework on various problems commonly considered for control systems, including inter-sample behaviors of a sampled-data system, effects of imperfect state measurements and unmodeled dynamics, and the use of time-discretized models to design controllers for continuous-time dynamical systems. While these issues have been extensively dealt for stability analysis of control systems, they have not been discussed adequately in the context of control for temporal logic objectives. We present these as the main results of the paper.

2. PRELIMINARIES

Notation: $\mathbb{R}^n$ denotes the n-dimensional Euclidean space; $|x|$ denotes a given (but fixed) norm of x for $x \in \mathbb{R}^n$; $\mathbb{R}^+$ denotes the nonnegative real line; given $\delta \geq 0$ and $x \in \mathbb{R}^n$, $B_\delta(x) := \{x' \in \mathbb{R}^n : |x' - x| \leq \delta\}$; given an interval $I \subseteq \mathbb{R}^+$ and $U \subseteq \mathbb{R}^m$, U^I denotes the set of signals from I to U; given a function f, $\mathrm{dom}(f)$ denotes its domain; given $h > 0$, $\mathcal{C}_h$ denotes the space of $\mathbb{R}^n$-valued continuous functions on $[-h, 0]$.

2.1 Linear temporal logics

We use the stutter-invariant fragment of linear temporal logic (denoted by $\mathrm{LTL}_{\setminus\bigcirc}$) to specify system properties. The syntax of $\mathrm{LTL}_{\setminus\bigcirc}$ over a set of atomic propositions Π is defined inductively as:

$$\varphi := \pi \,|\, \neg\varphi \,|\, \varphi \vee \varphi \,|\, \varphi \mathcal{U} \varphi,$$

where $\pi \in \Pi$. Atomic propositions are statements on an observation space Y. A labeling function $L : Y \to 2^\Pi$ maps an observation to a set of propositions that hold true. Linear temporal logic formulas are interpreted over observed signals.

Negation Normal Form (NNF): All $\mathrm{LTL}_{\setminus\bigcirc}$ formulas can be transformed into negation normal form, where

- all negations appear only in front of the atomic propositions[1];
- only other logical operators `true`, `false`, $\wedge$, and $\vee$ can appear; and
- only the temporal operators $\mathcal{U}$ and $\mathcal{R}$ can appear, where $\mathcal{R}$ is defined by $\varphi_1 \mathcal{R} \varphi_2 \equiv \neg(\neg\varphi_1 \mathcal{U} \neg\varphi_2)$, called the *dual until* operator.

For syntactic convenience, we can define additional temporal operators $\Box$ and $\Diamond$ by $\Box\varphi \equiv \texttt{false}\mathcal{R}\varphi$ and $\Diamond\varphi \equiv \texttt{true}\mathcal{U}\varphi$.

Continuous semantics of $\mathrm{LTL}_{\setminus\bigcirc}$: Given a continuous-time signal $\xi \in Y^{\mathbb{R}^+}$, we define $\xi, t \vDash \varphi$ with respect to an $\mathrm{LTL}^+_{\setminus\bigcirc}$ formula φ at time t inductively as follows:

- $\xi, t \vDash \pi$ if and only if $\pi \in L(\xi(t))$;
- $\xi, t \vDash \neg\pi$ if and only if $\pi \notin L(\xi(t))$;
- $\xi, t \vDash \texttt{true}$ always holds;
- $\xi, t \vDash \texttt{false}$ never holds;
- $\xi, t \vDash \varphi_1 \vee \varphi_2$ if and only if $\xi, t \vDash \varphi_1$ or $\xi, t \vDash \varphi_2$;
- $\xi, t \vDash \varphi_1 \wedge \varphi_2$ if and only if $\xi, t \vDash \varphi_1$ and $\xi, t \vDash \varphi_2$;
- $\xi, t \vDash \varphi_1 \mathcal{U} \varphi_2$ if and only if there exists $t' \geq 0$ such that $\xi, t + t' \vDash \varphi_2$ and for all $t'' \in [0, t')$, $\xi, t + t'' \vDash \varphi_1$;
- $\xi, t \vDash \varphi_1 \mathcal{R} \varphi_2$ if and only if for all $t' \geq 0$ either $\xi, t + t' \vDash \varphi_2$ or there exists $t'' \in [0, t')$ such that $\xi, t + t'' \vDash \varphi_1$.

We write $\xi \vDash \varphi$ if $\xi, 0 \vDash \varphi$.

Discrete semantics of $\mathrm{LTL}_{\setminus\bigcirc}$: Given a sequence $\rho = \{y_i\}_{i=0}^\infty$ in Y, we define $\rho, i \vDash \varphi$ with respect to an $\mathrm{LTL}_{\setminus\bigcirc}$ formula φ inductively as follows:

- $\rho, i \vDash \pi$ if and only if $\pi \in L(h(y_i))$;
- $\rho, i \vDash \neg\pi$ if and only if $\pi \notin L(h(y_i))$;
- $\rho, i \vDash \texttt{true}$ always holds;
- $\rho, i \vDash \texttt{false}$ never holds;
- $\rho, i \vDash \varphi_1 \vee \varphi_2$ if and only if $\rho, i \vDash \varphi_1$ or $\rho, i \vDash \varphi_2$;
- $\rho, i \vDash \varphi_1 \wedge \varphi_2$ if and only if $\rho, i \vDash \varphi_1$ and $\rho, i \vDash \varphi_2$;
- $\rho, i \vDash \varphi_1 \mathcal{U} \varphi_2$ if and only if there exists $j \geq i$ such that $\rho, j \vDash \varphi_2$ and $\rho, k \vDash \varphi_1$ for all $k \in [i, j)$;
- $\rho, i \vDash \varphi_1 \mathcal{R} \varphi_2$ if and only if, for all $j \geq i$, either $\rho, j \vDash \varphi_2$ or there exists $k \in [i, j)$ such that $\rho, k \vDash \varphi_1$.

Similarly, we write $\rho \vDash \varphi$ if $\rho, 0 \vDash \varphi$.

[1]Hence all negations can be effectively removed by introducing new atomic propositions corresponding to the negations of current ones. We assume this has been done for all $\mathrm{LTL}_{\setminus\bigcirc}$ formulas involved in this paper.

2.2 Problem Statement

We consider both continuous-time control systems of the form

$$\dot{x} = f(x, u), \tag{1}$$

and discrete-time control systems of the form

$$x^+ = g(x, u), \tag{2}$$

where $x \in \mathbb{R}^n$, $u \in U \subseteq \mathbb{R}^m$, x^+ denotes the next state of x under the difference equation, and both f and g are functions from $\mathbb{R}^n \times \mathbb{R}^m$ to $\mathbb{R}^n$.

Given a control input signal $\mathbf{u} \in U^{[0,T]}$, we assume that there exists a unique solution x defined on $[0, T]$ such that $\dot{x}(s) = f(x(s), \mathbf{u}(s))$ for all $s \in [0, T]$. For the discrete-time system (2), given a sequence of control inputs u_0, u_1, u_2, $\cdots$ in U, a solution to (2) is a sequence x_0, x_1, x_2, $\cdots$ such that $x_{i+1} = g(x_i, u_i)$.

The objective is to design control strategies such that solutions of systems (1) or (2) satisfy a given $\mathrm{LTL}_{\setminus\bigcirc}$ specification. For continuous-time systems, we define a *control strategy* to be a partial function of the form:

$$\sigma(x(\tau_0), \cdots, x(\tau_i)) = \mathbf{u}_i \in U^{[0,\Delta_i]},\ \forall i = 0, 1, 2, \cdots.$$

The sampling times τ_0, τ_1, τ_2, $\cdots$ satisfy $\tau_{i+1} - \tau_i = \Delta_i$, which is the duration of the control input signal $\mathbf{u}_i$. The control sequence $\mathbf{u}_0$, $\mathbf{u}_1$, $\mathbf{u}_2$, $\cdots$ lead to a solution of (1), which satisfies $\dot{x}(t) = f(x(t), \mathbf{u}(t))$ for all $t \geq 0$, where $\mathbf{u} \in U^{\mathbb{R}^+}$ is the concatenation of the sequence of control input signals u_i's. For discrete-time systems, a *control strategy* is defined to be a partial function of the form:

$$\sigma(x_0, \cdots, x_i) = u_i \in U,\ \forall i = 0, 1, 2, \cdots.$$

In this paper, we consider systems with full state observations; that is, we let the observation space $Y = \mathbb{R}^n$. Solutions of (1) and (2) are interpreted as signals and sequences in Y, respectively.

Problem Statement (Continuous Synthesis): Given a continuous-time system (1) (or a discrete-time system (2)) and an $\mathrm{LTL}_{\setminus\bigcirc}$ specification φ, find a control strategy for the system such that all of its solutions satisfy φ.

It should be emphasized that φ is interpreted using the continuous semantics for solutions of (1) and discrete semantics for solutions of (2).

2.3 Transition systems

A *transition system* is a tuple $\mathcal{T} = (Q, Q_0, \mathcal{A}, \to_\mathcal{T}, Y, h, \Pi, L)$, where:

- Q is a (finite or infinite) set of states and Q_0 the initial states;
- $\mathcal{A}$ is a (finite or infinite) set of actions;
- $\to_\mathcal{T} \subseteq Q \times \mathcal{A} \times Q$ is a transition relation;
- Y is a (finite or infinite) set of observations;
- $h : Q \to Y$ is an observation map on the states;
- Π is a set of atomic propositions on Y;
- $L : Y \to 2^\Pi$ is a labeling function.

We often write $q \xrightarrow{a} q'$ to indicate $(q, a, q') \in \to$. $\mathcal{T}$ is said to be (i) *finite* if the cardinality of Q and $\mathcal{A}$ are finite, (ii) *infinite* otherwise, and (iii) *metric* if Y is equipped with a metric.

An *execution* of a transition system $\mathcal{T}$ is a sequence of pairs

$$\rho = (q_0, a_0)(q_1, a_1)(q_2, a_2)\cdots,$$

where $q_0 \in Q_0$ and $(q_i, a_i, q_{i+1}) \in \rightarrow_{\mathcal{T}}$ for all $i \geq 0$. A *control strategy* for a transition system $\mathcal{T}$ is a partial function $s : (q_0, a_0, \cdots, q_i) \mapsto a_i$ that maps the execution history to the next action. An *s-controlled execution* of a transition system $\mathcal{T}$ is an execution of $\mathcal{T}$, where for each $i \geq 0$, the action a_i is chosen according to the control strategy s.

2.3.1 Continuous-time control systems as transition systems

Continuous-time control systems can be formulated as transition systems. Given system (1), we define a transition system $\mathcal{T}_c = (Q, Q_0, \mathcal{A}, \rightarrow_{\mathcal{T}_c}, Y, h, \Pi, L)$ by

- $Q = X$ and $Q_0 = X_0$;
- $\mathcal{A} = \bigcup_{\tau \in \mathbb{R}^+} U^{[0,\tau]}$;
- $(q, \mathbf{u}, q') \in \rightarrow_{\mathcal{T}_c}$ if and only if there exists $\xi : [0, \tau] \rightarrow \mathbb{R}^n$ such that $\xi(0) = q$, $\xi(\tau) = q'$, and $\dot{\xi}(s) = f(\xi(s), \mathbf{u}(s))$ for all $s \in [0, \tau]$, where $\mathbf{u} \in U^{[0,\tau]} \subseteq \mathcal{A}$;
- $Y = Q$;
- $h : Q \rightarrow Y$ is defined by $\mathbf{id}_Q$, i.e., the identity map on Q;
- Π is a set of atomic propositions on Y;
- $L : Y \rightarrow 2^{\Pi}$ is a labeling function,

where the state space is restricted to $X \subseteq \mathbb{R}^n$, with initial states in $X_0 \subseteq X$. Each action in $\mathcal{A}$ is a control input signal in $\mathcal{U}^{[0,\tau]}$ for some τ. If we are interested in digital implementations of control systems with a fixed sampling period τ_s, the set of actions can be regarded as $\mathcal{A} = U$ and interpreted as a constant input signal defined on $[0, \tau_s]$ and taking value in U.

2.3.2 Discrete-time control systems as transition systems

Discrete-time control systems can also be formulated as transition systems. Given system (2), we define a transition system $\mathcal{T}_d = (Q, Q_0, \mathcal{A}, \rightarrow_{\mathcal{T}_d}, Y, h, \Pi, L)$ by

- $Q = X$ and $Q_0 = X_0$;
- $\mathcal{A} = U$;
- $(q, u, q') \in \rightarrow_{\mathcal{T}_d}$ if and only if $q' = g(q, u)$;
- $Y = Q$;
- $h : Q \rightarrow Y$ is defined by $\mathbf{id}_Q$, i.e., the identity map on Q;
- Π is a set of atomic propositions on Y;
- $L : Y \rightarrow 2^{\Pi}$ is a labeling function.

Each action in $\mathcal{A}$ is a control input in $\mathcal{U}$.

Having defined (exact) transition system models for (1) and (2), it is possible to rephrase the continuous synthesis problem defined earlier as follows.

Problem Restatement (Continuous Synthesis): Given the transition system $\mathcal{T}_c$ (or $\mathcal{T}_d$) and an LTL$_{\setminus\bigcirc}$ specification φ, find a control strategy s such that all s-controlled executions of $\mathcal{T}_c$ (or $\mathcal{T}_d$) lead to *solutions of (1)* (or (2)) that satisfy φ.

This is a *continuous* synthesis problem since the state space is still continuous (and hence infinite). Again, it is emphasized that φ, for executions of $\mathcal{T}_c$, is interpreted using the *continuous* semantics that involve solutions of (1). The motivation for doing so will become clear in Section 4.1.

3. ABSTRACTIONS WITH ROBUSTNESS MARGIN

Both the transition system $\mathcal{T}_c$ and $\mathcal{T}_d$ constructed above are infinite, with infinitely many states and actions. Under the assumption that the sets X and U are compact, we can construct finite abstractions of $\mathcal{T}_c$ and $\mathcal{T}_d$ as follows.

These abstractions are induced by an abstraction map. An *abstraction map* $\alpha : Q \rightarrow 2^{\hat{Q}}$ maps the states in Q into a subset of a finite set $\hat{Q}$. Without loss of generality, we assume $\hat{Q}$ is a subset of Q; if not, for each $\hat{q} \in \hat{Q}$, we can pick a point q such that $\hat{q} \in \alpha(q)$ to represent $\hat{q}$. This map α effectively introduces a finite covering of Q given by $\bigcup_{\hat{q} \in \hat{Q}} \alpha^{-1}(\hat{q})$.

Definition 1. Given the continuous-time control system (1), its transition systems representation

$$\mathcal{T}_c = (Q, Q_0, \mathcal{A}, \rightarrow_{\mathcal{T}_c}, Y, h, \Pi, L),$$

and a tuple of positive constants $(\eta, \gamma_1, \gamma_2, \delta)$ satisfying $\gamma_i \geq \eta$ $(i = 1, 2)$, a finite transition system

$$\hat{\mathcal{T}}_c = (\hat{Q}, \hat{Q}_0, \hat{\mathcal{A}}, \rightarrow_{\hat{\mathcal{T}}_c}, \hat{Y}, \hat{h}, \hat{\Pi}, \hat{L})$$

is called an $(\eta, \gamma_1, \gamma_2, \delta)$-*abstraction* of $\mathcal{T}_c$ if there exists an abstraction map $\alpha : Q \rightarrow 2^{\hat{Q}}$ such that

(i) $\hat{Q}$, $\hat{Q}_0$, and $\hat{\mathcal{A}}$ are finite subsets of Q, Q_0 and $\mathcal{A}$;
(ii) $|q - \hat{q}| \leq \eta$ for all $(q, \hat{q}) \in Q \times \hat{Q}$ such that $\hat{q} \in \alpha(q)$;
(iii) $(\hat{q}, \mathbf{u}, \hat{q}') \in \rightarrow_{\hat{\mathcal{T}}_c}$ if there exists $\xi : [0, \tau] \rightarrow \mathbb{R}^n$ such that $|\xi(0) - \hat{q}| \leq \gamma_1$, $|\xi(\tau) - \hat{q}'| \leq \gamma_2$, and $\dot{\xi}(s) = f(\xi(s), \mathbf{u}(s))$ for all $s \in [0, \tau]$, where $\mathrm{dom}(\mathbf{u}) = [0, \tau]$;
(iv) $\hat{h} = h|_{\hat{Q}}$, i.e., h restricted on $\hat{Q}$, $\hat{Y} = \hat{Q}$, and $\hat{\Pi} = \Pi$;
(v) $\hat{L} : \hat{Y} \rightarrow 2^{\hat{\Pi}}$ is defined by

$$\pi \in \hat{L}(y),\ y \in \hat{Y} \Longleftrightarrow \pi \in L(y'),\ \forall y' \in B_\delta(y). \quad (3)$$

We write $\mathcal{T}_c \preceq_{(\eta,\gamma_1,\gamma_2,\delta)} \hat{\mathcal{T}}_c$.

Remark 1. Since each action in $\hat{\mathcal{A}}$ is a control input signal with some finite duration τ and $\hat{\mathcal{A}}$ is a finite set, there exists a maximum duration for signals in $\hat{\mathcal{A}}$, denoted by $\Delta(\hat{\mathcal{A}})$ or Δ. If we restrict the actions to constant signals of a fixed duration τ_s (e.g., due to periodic sampling and zero-order hold), we have $\Delta = \tau_s$.

Similarly, we can define an $(\eta, \gamma_1, \gamma_2, \delta)$-abstraction of $\mathcal{T}_d$.

Definition 2. Given the discrete-time control system (2), its transition systems representation

$$\mathcal{T}_d = (Q, Q_0, \mathcal{A}, \rightarrow_{\mathcal{T}_c}, Y, h, \Pi, L),$$

and a tuple of positive constants $(\eta, \gamma_1, \gamma_2, \delta)$ satisfying $\gamma_i \geq \eta$ $(i = 1, 2)$, a finite transition system

$$\hat{\mathcal{T}}_d = (\hat{Q}, \hat{Q}_0, \hat{\mathcal{A}}, \rightarrow_{\hat{\mathcal{T}}_d}, \hat{Y}, \hat{h}, \hat{\Pi}, \hat{L})$$

is called an $(\eta, \gamma_1, \gamma_2, \delta)$-*abstraction* of $\mathcal{T}_d$ if there exists an abstraction map $\alpha : Q \rightarrow 2^{\hat{Q}}$ such that

- $(\hat{q}, u, \hat{q}') \in \rightarrow_{\hat{\mathcal{T}}_c}$ if there exists ξ and ξ' such that $|\xi - \hat{q}| \leq \gamma_1$, $|\xi' - \hat{q}'| \leq \gamma_2$, and $\xi' = g(\xi, u)$,

and (i), (iii), (iv) and (v) in Definition 2 hold for $\hat{\mathcal{T}}_d$. We write $\mathcal{T}_d \preceq_{(\eta,\gamma_1,\gamma_2,\delta)} \hat{\mathcal{T}}_d$.

The abstraction relations defined above can be seen as an over-approximation of the system dynamics with discretization granularity η and parameters γ_i $(i = 1, 2)$ to account for mismatches in abstraction. It is, at the same time, an under-approximation on the control actions in the sense that these are restricted to a subset of all available actions. The parameter δ provides a robustness margin which plays an important role in preserving the correctness of executions with respect to a given LTL$_{\setminus\bigcirc}$ specification (closely related to robust interpretations of temporal logic formulas [4]). The above relation essentially defines an alternating simulation from $\mathcal{T}_c$ to $\hat{\mathcal{T}}_c$ ($\mathcal{T}_d$ to $\hat{\mathcal{T}}_d$) [1] that takes into account approximation errors (cf. [19]) and provides robustness margins to accommodate these errors. Here, additional robustness measures are given by γ_i $(i = 1, 2)$, where, as shall be demonstrated by the main results of this paper, γ_1 is useful in accounting for imperfect state measurements, while γ_2 is useful in dealing with uncertainties/mismatches in the models used for controller synthesis.

Example 1. We give concrete examples of $\hat{\mathcal{T}}_c$ and $\hat{\mathcal{T}}_d$ by constructing $\hat{Q}$ as a state discretization of Q. Given a positive integer k, let $\mathbb{Z}^n$ denote the n-dimensional integer lattice, which is the lattice in $\mathbb{R}^n$ whose lattice points are k-tuples of integers. For a given parameter $\mu > 0$, define

$$[X]_\mu := \mu\mathbb{Z}^n \cap X, [X_0]_\mu := \mu\mathbb{Z}^n \cap X_0,$$

where $\mu\mathbb{Z}^n := \{\mu z : z \in \mathbb{Z}^n\}$. Clearly, $[X]_\mu$ and $[X_0]_\mu$ are finite sets given that X and X_0 are compacts. As for actions, instead of looking at an infinite set of actions, we consider control input signals of durations within a finite set $\hat{T} := \{k\tau_s : k \in K\}$ and taking values in a finite subset $\hat{U}$ of U, where K is a finite subset of positive integers and τ_s is the minimum sampling period. For example, one can choose $K = \{2^s : s = 0, \cdots N\}$ for some integer $N \geq 0$ (cf. [15]). Finite abstractions for $\mathcal{T}_c$ and $\mathcal{T}_d$ can be defined as in Definitions 1 and 2 with $\hat{Q} = [X]_\mu$, $Q_0 = [X_0]_\mu$, and $\hat{\mathcal{A}} = \bigcup_{\tau\in\hat{T}} \hat{U}^{[0,\tau]}$. Note that this discretization result in conditions (ii) in Definitions 1 and 2 being satisfied with $\eta = \mu/2$. □

3.1 Discrete synthesis

The main reason to construct finite abstractions such as $\hat{\mathcal{T}}_c$ and $\hat{\mathcal{T}}_d$ is to formulate discrete synthesis problems that can be used to solve the continuous synthesis problems previously defined for $\mathcal{T}_c$ and $\mathcal{T}_d$.

Given a set of atomic propositions Π on Y, an LTL$_{\setminus\bigcirc}$ formula over Π can be interpreted over executions of $\hat{\mathcal{T}}_c$ and $\hat{\mathcal{T}}_d$ using the discrete semantics. We can now formulate the discrete problems as follows.

Problem Statement (Discrete Synthesis): Given the transition system $\hat{\mathcal{T}}_c$ (or $\hat{\mathcal{T}}_d$) and an LTL$_{\setminus\bigcirc}$ specification φ, find a control strategy s such that all s-controlled executions of $\hat{\mathcal{T}}_c$ (or $\hat{\mathcal{T}}_d$) satisfy φ.

If there exists a control strategy for $\hat{\mathcal{T}}_c$ (or $\hat{\mathcal{T}}_d$) such that all controlled executions of $\hat{\mathcal{T}}_c$ (or $\hat{\mathcal{T}}_d$) satisfy φ, we say φ is *realizable* on $\hat{\mathcal{T}}_c$ (or $\hat{\mathcal{T}}_d$). We may call control strategies for $\hat{\mathcal{T}}_c$ (or $\hat{\mathcal{T}}_d$) *discrete* (control) strategies and those for $\mathcal{T}_c$ (or $\mathcal{T}_d$) *continuous* (control) strategies.

For the finite abstractions $\hat{\mathcal{T}}_c$ and $\hat{\mathcal{T}}_d$ to be useful, we need to guarantee two things: (i) every discrete control strategy for $\hat{\mathcal{T}}_c$ (or $\hat{\mathcal{T}}_d$) can be implemented to form a control strategy for the continuous system $\mathcal{T}_c$ (or $\mathcal{T}_d$); and (ii) if the discrete strategy solves the discrete synthesis problem for $\hat{\mathcal{T}}_c$ (or $\hat{\mathcal{T}}_d$), then the corresponding continuous strategy solves the discrete synthesis problem for $\mathcal{T}_c$ (or $\mathcal{T}_d$). Establishing these under various scenarios will be the main results of this paper.

3.2 Computation of transitions

A question that remains is how to compute the transitions in $\rightarrow_{\hat{\mathcal{T}}_c}$ and $\rightarrow_{\hat{\mathcal{T}}_d}$. One way to do this is by simulating a trajectory starting from each of the point in $\hat{Q}$ and estimating the state evolution under the dynamics of (1) and (2).

In the continuous-time case, we rely on the following condition:

$$|x(t; \mathbf{u}, \xi) - x(t; \mathbf{u}, \zeta)| \leq \beta(|\xi - \zeta|, t), \tag{4}$$

for all $\tau \in \mathbb{R}^+$, $\mathbf{u} \in U^{[0,\tau]}$, and $t \in [0, \tau]$, where $x(t; \mathbf{u}, \xi)$ and $x(t; \mathbf{u}, \zeta)$ denote solutions of $\dot{x} = f(x, u)$ starting from ξ and ζ and with control input $\mathbf{u}$, respectively, and $\beta : \mathbb{R}^+ \times \mathbb{R}^+ \rightarrow \mathbb{R}^+$ is a continuous function such that for each fixed t, $\beta(\cdot, t)$ is a class-$\mathcal{K}_\infty$ function[2]. Such a condition is a special case of the notion of incremental forward completeness defined in [28]. Condition (4) essentially defines a continuous dependence condition of (1) on its initial conditions that is uniform for all $\mathbf{u}$ taking values in U. An explicit form of β can usually be obtained using Lyapunov-type techniques. In addition, if (1) is incrementally asymptotically stable [2], β can be chosen as a $\mathcal{KL}$-function[3].

Proposition 1. *Suppose (4) holds. If* $(\hat{q}, \mathbf{u}, \hat{q}') \in \rightarrow_{\hat{\mathcal{T}}_c}$ *whenever* $(\hat{q}, \mathbf{u}, \hat{q}') \in \hat{Q} \times \hat{\mathcal{A}} \times \hat{Q}$ *and* $|\hat{q}' - x(\tau; \mathbf{u}, \hat{q})| \leq \beta(\gamma_1, \tau) + \gamma_2$, *then* $\mathcal{T}_c \preceq_{(\eta,\gamma_1,\gamma_2,\delta)} \hat{\mathcal{T}}_c$, *where* $dom(\mathbf{u}) = [0, \tau]$.

Proof. We have to show that $\rightarrow_{\hat{\mathcal{T}}_c}$ constructed above contains all transitions $(\hat{q}, \mathbf{u}, \hat{q}')$ whenever there exists $\xi : [0, \tau] \rightarrow \mathbb{R}^n$ such that $|\xi(0) - \hat{q}| \leq \gamma_1$, $|\xi(\tau) - \hat{q}'| \leq \gamma_2$, and $\dot{\xi}(s) = f(\xi(s), \mathbf{u}(s))$ for all $s \in [0, \tau]$.

Consider $x(t; \mathbf{u}, \hat{q})$ and $x(t; \mathbf{u}, \xi(0))$. We have

$$\begin{aligned} |x(\tau; \mathbf{u}, \hat{q}) - \xi(\tau)| &= |x(\tau; \mathbf{u}, \hat{q}) - x(\tau; \mathbf{u}, \xi(0))| \\ &\leq \beta(|\xi(0) - \hat{q}|, \tau) \leq \beta(\gamma_1, \tau). \end{aligned}$$

It follows that

$$\begin{aligned} |x(\tau; \mathbf{u}, \hat{q}) - \hat{q}'| &\leq |x(\tau; \mathbf{u}, \hat{q}) - \xi(\tau)| + |\xi(\tau) - \hat{q}'| \\ &\leq \beta(\gamma_1, \tau) + \gamma_2, \end{aligned}$$

which implies that $(\hat{q}, \mathbf{u}, \hat{q}') \in \rightarrow_{\hat{\mathcal{T}}_c}$. □

For discrete-time systems, we replace (4) with the following condition:

$$|g(u, \xi) - g(u, \zeta)| \leq \beta(|\xi - \zeta|), \tag{5}$$

where $u \in U$ and β is class-$\mathcal{K}_\infty$ function.

Proposition 2. *Suppose (5) holds. If* $(\hat{q}, u, \hat{q}') \in \rightarrow_{\hat{\mathcal{T}}_d}$ *whenever* $(\hat{q}, u, \hat{q}') \in \hat{Q} \times \hat{\mathcal{A}} \times \hat{Q}$ *and* $|\hat{q}' - g(u, \hat{q})| \leq \beta(\gamma_1) + \gamma_2$, *then* $\mathcal{T}_d \preceq_{(\eta,\gamma_1,\gamma_2,\delta)} \hat{\mathcal{T}}_d$.

[2] A function $\kappa : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ is called a *class-$\mathcal{K}$ function* if it is strictly increasing and $\kappa(0) = 0$; it is called a *class-$\mathcal{K}_\infty$ function* if it is a class-$\mathcal{K}$ function and $\kappa(r) \rightarrow \infty$ as $r \rightarrow \infty$.

[3] A function $\kappa : \mathbb{R}^+ \times \mathbb{R}^+ \rightarrow \mathbb{R}^+$ is called a *class-$\mathcal{KL}$ function* if, for each fixed t, $\kappa(\cdot, t)$ is a class-$\mathcal{K}_\infty$ function and, for each fixed r, $\kappa(r, t) \rightarrow 0$ as $t \rightarrow \infty$.

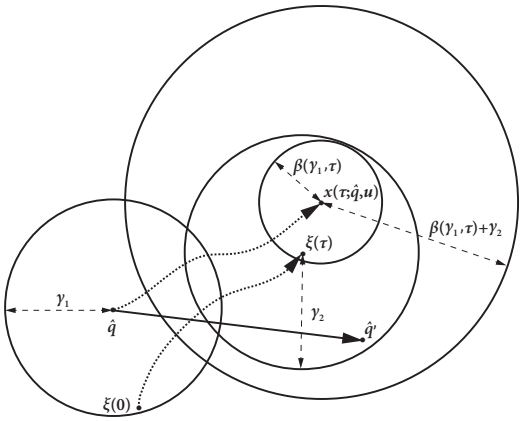

Figure 1: Proposition 1 provides concrete means to over-approximate $\rightarrow_{\hat{\mathcal{T}}_c}$ by adding $(\hat{q},\mathbf{u},\hat{q}')$ to $\rightarrow_{\hat{\mathcal{T}}_c}$ whenever $|\hat{q}'-x(\tau;\mathbf{u},\hat{q})| \leq \beta(\gamma_1,\tau)+\gamma_2$. In view of condition (4), the ball $B_{\beta(\gamma_1,\tau)}(x(\tau;\hat{q},\mathbf{u}))$ includes $\xi(\tau)$ for all $\xi:[0,\tau]\to\mathbb{R}^n$ such that $|\xi(0)-\hat{q}| \leq \gamma_1$ and $\dot{\xi}(s)=f(\xi(s),\mathbf{u}(s))$ for all $s\in[0,\tau]$. Hence, the ball $B_{\beta(\gamma_1,\tau)+\gamma_2}(x(\tau;\hat{q},\mathbf{u}))$ contains all $\hat{q}'\in\hat{Q}$ that is γ_2-close to $\xi(\tau)$ for some ξ defined above; that is, all $\hat{q}'\in\hat{Q}$ such that $(\hat{q},\mathbf{u},\hat{q}')\in\rightarrow_{\hat{\mathcal{T}}_c}$ as required by Definition 1.

Proposition 1 essentially says that, for each state in $\hat{q}\in\hat{Q}$ and $\mathbf{u}\in\hat{A}$, if we add $(\hat{q},\mathbf{u},\hat{q}')$ to $\rightarrow_{\hat{\mathcal{T}}_c}$ for each $\hat{q}'\in\hat{Q}\cap B_\gamma(x(\tau;\mathbf{u},\hat{q}))$, where $\gamma=\beta(\gamma_1,\tau)+\gamma_2$, then we obtain an $(\eta,\gamma_1,\gamma_2,\delta)$-abstraction of $\mathcal{T}_c$ in the sense of Definition 1. Figure 1 illustrates how transitions in $\rightarrow_{\hat{\mathcal{T}}_c}$ can be computed in Proposition 1. The intuition behind Proposition 2 is similar.

Augmented Progress Properties.

In view of comments above, if τ is sufficiently small compared with γ, then the ball $B_\gamma(x(\tau;\mathbf{u},\hat{q}))$ will almost always include $\hat{q}$ itself, which introduces a self-transition $(\hat{q},\mathbf{u},\hat{q})$ for almost all $\hat{q}\in\hat{Q}$. As we shall treat all non-determinism as adversary when solving the discrete synthesis problem, these self-transitions can render the problem unrealizable if the specification involves making progress. In addition to self-transitions, non-determinism can potentially induce spurious cyclic executions in the abstract system that do not exist in the continuous system (1). To deal with these issues, we can use augmented finite transition systems [17] to enforce additional progress assumptions when solving the discrete synthesis problem. Such progress assumptions can be captured by the following $\text{LTL}_{\setminus\bigcirc}$ formula:

$$\varphi_g \doteq \bigwedge_{\mathbf{u}\in\hat{A}} \bigwedge_{G\in\mathcal{G}(\mathbf{u})} \neg\Diamond\Box((\vee_{\hat{q}\in G}\hat{q}) \wedge \mathbf{u}), \tag{6}$$

where each $G\in\mathcal{G}(\mathbf{u})$ represents a progress group. That is, the set $\bigcup_{\hat{q}\in G}\alpha^{-1}(\hat{q})$ does not contain any invariant sets for system (1) when a fixed $\mathbf{u}$ is repeatedly executed. Such progress groups can be trivially computed for affine or incrementally stable dynamics. It is also possible to approximate them when the dynamics are polynomial [17]. Appropriately encoding these progress properties is essential for achieving certain specifications (e.g., reachability).

Remark 2. Without additional assumption, the estimate $\gamma(\tau)=\beta(\gamma_1,\tau)+\gamma_2$ used by Proposition 1 and illustrated in Figure 1 can be conservative and may lead to too much nondeterminism that renders the discrete synthesis problem unrealizable. One way to overcome this is to assume (1) to be incrementally stable, in which case β can be chosen as a $\mathcal{KL}$ function. We can then choose τ sufficiently large such that

$$\beta(\gamma(\tau),\tau)=\beta(\beta(\gamma_1,\tau)+\gamma_2,\tau)\leq\gamma(\tau),$$

which is always possible since β is a $\mathcal{KL}$ function and $\gamma_2<\gamma(\tau)\leq\beta(\gamma_1,0)+\gamma_2$. The above inequality essentially captures how nondeterminism is bounded within two steps of transitions.

4. MAIN RESULTS—IMPLICATIONS OF THE ROBUSTNESS MARGIN

The main objective of this section is to show that, with the notions of abstractions defined in Definitions 1 and 2, we are able to reason about the qualitative properties of solutions of (1) and (2) in a number of different scenarios, including inter-sample behaviors of a sampled-data system, effects of imperfect state measurements and unmodeled dynamics, and the use of time-discretized models to design controllers for continuous-time dynamical systems.

4.1 Continuous correctness by discrete reasoning

When implementing a discrete strategy, perhaps obtained from solving a discrete synthesis problem, we are effectively implementing a hybrid feedback controller such that solutions of (1) (or (2)) satisfy a given specification.

In general, the existence of a discrete control strategy for the discrete synthesis problem for $\hat{\mathcal{T}}_c$ (or $\hat{\mathcal{T}}_d$) with an $\text{LTL}_{\setminus\bigcirc}$ formula φ *does not* guarantee the existence of a control strategy that solves the continuous synthesis problem for (1) (or (2)) with the same specification φ. In fact, using discretization-based (or grid point-based), rather than proposition-preserving partition-based, abstractions, we need extra conditions to ensure correctness of continuous executions from discrete reasoning. This motivates (3) in defining abstractions, which essentially captures the idea of contracting and expanding atomic propositions as used in [3, 12]. This extra condition is needed to account for inter-sample behaviors as illustrated in the following simple example.

Example 2. Consider a two dimensional system given in polar coordinates $\dot{r}=-r$ and $\dot{\theta}=\omega$. This is an asymptotically stable linear system, hence incrementally stable. Trajectories of this systems are spiraling towards the origin, such as the trajectory x illustrated in Figure 2. Suppose we are interested in verifying that all trajectories starting from the set A and eventually reach the set B, while not intersecting the set C, which can be captured by the specification $\varphi\equiv(A\rightarrow\Diamond B)\wedge\Box C^c)$, where C^c is the complement of the set C. Suppose that we are using sampled values of x to verify whether $x\models\varphi$ and the sampling period is τ_s. For any $\tau_s>0$, if we choose $\omega=2\pi/\tau_s$, the trajectory x starting from $(a_0,0)\in A$ will lead to a sampled sequence of $(a_0,0)(a_1,0)(a_2,0)\cdots$, which clearly satisfies φ. However, $x\not\models\varphi$ as it intersects with C. This simple example illustrates that extra conditions are needed to account for inter-

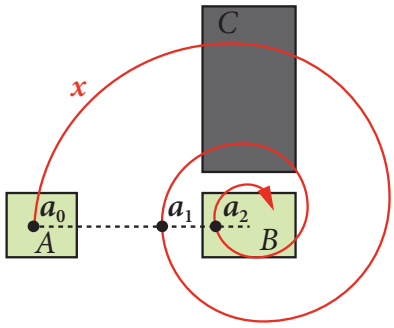

Figure 2: **A simple illustration of inter-sample behaviors that violate a given specification, while a sampled sequence satisfies the same specification.**

sample behaviors and these conditions will have to depend on system dynamics. □

We let $M = \sup_{x\in X, u\in U} |f(x,u)|$ and Δ be the maximum duration of actions in $\hat{\mathcal{A}}$.

THEOREM 1. *If $\mathcal{T}_c \preceq_{(\eta,\gamma_1,\gamma_2,\delta)} \hat{\mathcal{T}}_c$ with $\gamma_i \geq \eta$ $(i = 1, 2)$ and $\delta \geq M\Delta/2 + \eta$, then, given any LTL$_{\setminus\bigcirc}$ formula φ, φ being realizable for $\hat{\mathcal{T}}_c$ implies that φ is realizable for $\mathcal{T}_c$.*

PROOF. By the definition of an $(\eta, \gamma_1, \gamma_2, \delta)$-abstraction, to every control strategy $\hat{f}$ for $\hat{T}_c$, there corresponds a control strategy f for $\mathcal{T}_c$ such that, to each f-controlled execution of $\mathcal{T}_c$, there corresponds a $\hat{f}$-controlled execution in $\hat{\mathcal{T}}_c$. In fact, this is ensured by the fact that $|q - \hat{q}| \leq \eta$ for all $(q, \hat{q}) \in Q \times \hat{Q}$ such that $\hat{q} \in \alpha(q)$ and the condition $\gamma_i \geq \eta$ $(i = 1, 2)$.

We denote this correspondence by ρ to $\hat{\rho}$, where

$$\rho = (q_0, \mathbf{u}_0)(q_1, \mathbf{u}_1)(q_2, \mathbf{u}_2)\cdots$$

and

$$\hat{\rho} = (\hat{q}_0, \mathbf{u}_0)(\hat{q}_1, \mathbf{u}_1)(\hat{q}_2, \mathbf{u}_2)\cdots$$

Each $\hat{q}_i$ is an abstract state corresponding to q_i and hence $|q_i - \hat{q}_i| \leq \eta$ for all $i \geq 0$. Furthermore, corresponding to ρ, there is the solution x with $x(\tau_i) = q_i$ for all $i \geq 0$, where $\tau_0 = 0$ and $\tau_{i+1} = \tau_i + \Delta_i$, where Δ_i is the duration of $\mathbf{u}_i$. We have to show that $\hat{\rho} \vDash \varphi$ implies $x \vDash \varphi$. We prove this by proving a stronger statement: $\hat{\rho}, i \vDash \varphi$ for $i \geq 0$ implies $x, t \vDash \varphi$ for all $t \in J_i = [\tau_i - \Delta/2, \tau_i + \Delta/2] \cap \mathbb{R}^+$.

The proof is by induction on the structure of an LTL$_{\setminus\bigcirc}$ formula.

Case $\varphi = \pi$: To show $x, t \vDash \pi$ for all $t \in J_i$, we have to show that $\pi \in L(x(t))$. This follows from $q_i = x(\tau_i)$, $\pi \in \hat{L}(\hat{q}_i)$, and

$$|x(t) - \hat{q}_i| \leq |x(t) - x(\tau_i)| + |q_i - \hat{q}_i| \leq M\Delta/2 + \eta \leq \delta. \quad (7)$$

Case $\varphi = \varphi_1 \mathcal{U} \varphi_2$: To show $x, t \vDash \varphi$ for all $t \in J_i$, we need to show that, for each fixed $t \in J_i$, there exists $t' \geq 0$ such that $x, t + t' \vDash \varphi_2$ and for all $t'' \in [0, t')$, $x, t + t'' \vDash \varphi_1$. We have $\hat{\rho}, i \vDash \varphi$; that is, there exists $j > i$ such that $\hat{\rho}, j \vDash \varphi_2$ and $\hat{\rho}, k \vDash \hat{\varphi}_1$ for all $k \in [i, j)$. It follows from the inductive assumption that $x, s \vDash \varphi_2$ for all $s \in J_j$ and $x, s \vDash \varphi_1$ for all $s \in J_k$ and all $k \in [i, j)$. Take $t' = \max(\tau_j - \Delta/2, t) - t$. Then $t + t' \in J_j$ and hence $x, t + t' \vDash \varphi_2$. In addition, for all $t'' \in [0, t')$, we have $t + t'' \in J_k$ for some $k \in [i, j)$ and hence $x, t + t'' \vDash \varphi_1$. In fact, $\cup_{i\leq k\leq j-1} J_k = [\tau_i - \Delta/2, \tau_{j-1} + \Delta/2] \cap \mathbb{R}^+ \supseteq [t, \tau_j - \Delta/2) = [t, t + t') \ni t + t''$.

Case $\varphi = \varphi_1 \mathcal{R} \varphi_2$: To show $x(t) \vDash \varphi$ for all $t \in J_i$, we need to show that, for each fixed $t \in J_i$, we have, for all $t' \geq 0$ either $\xi, t + t' \vDash \varphi_2$ or that there exists $t'' \in [0, t')$ such that $x, t + t'' \vDash \varphi_1$. We have $\hat{\rho}, i \vDash \varphi$; that is, for all $j \geq i$, either $\hat{\rho}, j \vDash \varphi_2$ or there exists $k \in [i, j)$ such that $\hat{\rho}, k \vDash \varphi_1$. Given $t' \geq 0$, let τ_j be such that $t + t' \in J_j$, where $j \geq i$. For this j, we have either $\hat{\rho}, j \vDash \varphi_2$ or that there exists $k \in [i, j)$ such that $\hat{\rho}, k \vDash \varphi_1$. It follows from the inductive assumption that either $x, s \vDash \varphi_2$ for all $s \in J_j$ or there exists $k \in [i, j)$ such that $x, s \vDash \varphi_1$ for all $s \in J_k$. If the former holds, since $t + t' \in J_j$, we get $\xi, t + t' \vDash \varphi_2$. If the latter holds, since $t + t' \geq \tau_j - \Delta/2 > \tau_k - \Delta/2$ and $\tau_k + \Delta/2 \geq \tau_i + \Delta/2 \geq t$, we know $[t, t + t') \cap J_k \neq \emptyset$. Thus, there exists $t'' \in [0, t')$ such that $\xi, t + t'' \vDash \varphi_1$.

The other cases are straightforward. □

REMARK 3. The condition $\delta \geq \eta + M\Delta$ can be relaxed by considering a one-step version of it; that is, the relation holds for every single transition $(\hat{q}, \mathbf{u}, \hat{q}') \in \rightarrow_{\hat{\mathcal{T}}_c}$. This will use a non-uniform, state-dependent error specification (η becomes a function on $\hat{Q}$) and a state-dependent robustness margin (δ becomes a function on $\hat{Q}$). The bounded M can be taken on the set of concrete states corresponding to $\hat{q}$ and $\hat{q}'$ and the set of inputs u taken by the signal $\mathbf{u}$. Moreover, we can use precise information of the duration of an action $\mathbf{u}$ in each transition (denoted by τ), instead of using a global bound Δ for such τ's.

THEOREM 2. *If $\mathcal{T}_d \preceq_{(\eta,\gamma_1,\gamma_2,\delta)} \hat{\mathcal{T}}_d$ with $\gamma_i \geq \eta$ $(i = 1, 2)$ and $\delta \geq \eta$, then, given any LTL$_{\setminus\bigcirc}$ formula φ, φ being realizable for $\hat{\mathcal{T}}_d$ implies that φ is realizable for $\mathcal{T}_d$.*

PROOF. The proof is similar to that for Theorem 1. The only difference is that we do not need to account for inter-sample behaviors. Hence, the condition is weaken to $\delta \geq \eta$, which essentially says that all concrete states corresponding to the same discrete states should satisfy the same propositions. □

4.2 Imperfect state measurement: bounded errors or delays

In practice, state measurements are not perfect, often subject to measurement noise or quantization. Furthermore, delays are ubiquitous in control systems. In this subsection, we consider the robustness of a hybrid controller for (1) that realizes a temporal logic objective with respect to imperfect state measurements. The details of the problem are illustrated in Figure 3.

THEOREM 3. *Suppose that (1) is to be controlled under the situation illustrated in Figure 3. If $\mathcal{T}_c \preceq_{(\eta,\gamma_1,\gamma_2,\delta)} \hat{\mathcal{T}}_c$ with $\gamma_1 \geq hM + \varepsilon + \eta$, $\gamma_2 \geq \varepsilon + \eta$, and $\delta \geq (h + \Delta)M/2 + (\varepsilon + \eta)$, then, given any LTL$_{\setminus\bigcirc}$ formula φ, φ being realizable for $\hat{\mathcal{T}}_c$ implies that φ is realizable for $\mathcal{T}_c$.*

PROOF. Let x_0, x_1, x_2, $\cdots$, be the measurements taken at the plant at times τ_0, τ_1, τ_2, $\cdots$; that is $x_i = x(\tau_i)$ for all $i \geq 0$. As shown in Figure 3, we assume it takes time delay h_1 for the hybrid controller to receive a perhaps noisy measurement given by $\hat{x}_i = x(\tau_i) + e_i$ at time $\tau_i + h_1$. The controller makes a decision and passes on a suggested input

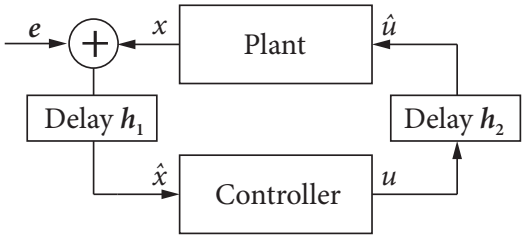

Figure 3: Illustration of a controller that takes delayed (by h_1) and imperfect measurement (subject to measurement errors bounded by ε) from a plant and sends a control input that is received by the plant after another delay h_2 (measured from when the controller receives the measurement and to when the control input has been actuated by the plant). The total round-trip delay $h_1 + h_2$ is not assumed to be constant, but assumed to be bounded by some constant h. While the plant is waiting for the next control input, it keeps on executing the previous one.

$\mathbf{u}_i$ (which includes the duration of $\mathbf{u}_i$ denoted by Δ_i). The plant will receive this input subject to another delay h_2 at time $\tau_i+h_1+h_2$. From this point on, the control input is set to $\mathbf{u}_i$. Between τ_i and $\tau_i + h$, the plant will keep executing the previous input $\mathbf{u}_{i-1}$; initially, between τ_0 and $\tau_0 + h$, assume this input is set to some initial value. We need to be clear how τ_i's are defined: we set $\tau_0 = 0$ and the rest of the sampling times τ_i ($i \geq 1$) are defined by $\tau_i = \tau_{i-1}+\Delta_{i-1}+h$.

There are two things to prove: (1) every measured states (with delays and noise) are accounted for in the abstraction, so that the discrete control strategy can be implemented. Put more straightforwardly, every measured states should be expected by the controller so that it can make a decision based on the strategy automaton; (2) the plant trajectory $x(t)$, $t \geq 0$, should satisfy the desired specification φ.

The first is ensured by that the transition from $\hat{x}_i$ to $\hat{x}_{i+1}$ is captured by the a transition $\hat{q}_i$ to $\hat{q}_{i+1}$ in the abstraction. We only need to verify that there exists a trajectory ξ of (1) under input signals $\mathbf{u}_i$ such that $|\xi(0) - \hat{q}_i| \leq \gamma_1$ and $|\xi(\Delta_i) - \hat{q}_{i+1}| \leq \gamma_2$ for all $i \geq 0$. We know that $|\hat{x}_i - \hat{q}_i| \leq \eta$ and $|\hat{x}_{i+1} - \hat{q}_{i+1}| \leq \eta$. We also know that $|\hat{x}_i - x(\tau_i)| \leq \varepsilon$, $\tau_{i+1} = \tau_i + \Delta_i + h$ for all $i \geq 0$, and $\mathbf{u}_i$ is activated on $[\tau_i + h, \tau_i + h + \Delta_i]$. Letting $\xi(s) = x(\tau_i + h + s)$ for $s \in [0, \Delta_i]$, then $\xi(0) = x(\tau_i + h)$ and $\xi(\Delta_i) = x(\tau_i+\Delta_i+h)$. It is easy to verify that $|\xi(0) - \hat{q}_i| \leq |x(\tau_i + h) - x(\tau_i)|+|x(\tau_i) - \hat{x}_i|+|\hat{x}_i - \hat{q}_i| \leq hM+\varepsilon+\eta \leq \gamma_1$ and $|\xi(\Delta_i) - \hat{q}_{i+1}| \leq |x(\tau_{i+1}) - \hat{x}_{i+1}| + |\hat{x}_{i+1} - \hat{q}_{i+1}| \leq \varepsilon + \eta \leq \gamma_2$.

Let $x(\tau_i) = q_i$. We have $|q_i - \hat{q}_i| \leq \varepsilon + \eta$ and $\tau_{i+1} - \tau_i = \Delta_i + h$ for all $i \geq 0$. We can prove $x \vDash \varphi$ following the proof of Theorem 1 with η replaced by $\eta + \varepsilon$ and Δ_i replaced by Δ_i+h. The result follows from $\delta \geq (h+\Delta)M/2+(\eta+\varepsilon)$. □

For discrete-time systems, we do not consider delays in this paper, but the following result gives robustness with respect to measurement errors. The proof is omitted.

THEOREM 4. *Suppose that (2) is to be controlled subject to measurement errors bounded by ε. If $\mathcal{T}_d \preceq_{(\eta,\gamma_1,\gamma_2,\delta)} \hat{\mathcal{T}}_d$ with $\gamma_i \geq \varepsilon + \eta$ ($i = 1, 2$), and $\delta \geq \varepsilon + \eta$, then, given any LTL$_{\setminus\bigcirc}$ formula φ, φ being realizable for $\hat{\mathcal{T}}_d$ implies that φ is realizable for $\mathcal{T}_d$.*

4.3 Unmodeled dynamics: bounded disturbance or delays

We can also apply the same methodology to prove the effectiveness of the design in the situation where systems (1) and (2) contain unmodeled dynamics that can be treated as bounded disturbance in the right-hand side of (1) and (2).

A general time-delay system can be written as a functional differential equation:

$$\dot{x} = F(x_t, u), \quad t \geq 0, \tag{8}$$

where $F : \mathcal{C}_h \times U \to \mathbb{R}^n$ is a functional with control input $u \in U$, and $x_t(s) = x(t + s)$ for all $s \in [-h, 0]$. We assume that, given any initial condition $x_0 \in \mathcal{C}_h$, (8) has a unique global solution.

We can rewrite F such that it has an *ordinary part* and a *functional part*:

$$F(x_t, u) = f(x, u) + g(x_t, u), \tag{9}$$

where $f : \mathbb{R}^+ \times \mathbb{R}^n \to \mathbb{R}^n$ and $g : \mathcal{C}_h \times U \to \mathbb{R}^n$. This form can be obtained, for example, from (8) by letting $g(x_t, u) := F(x_t, u) - f(x, u)$. The idea is to design controllers for system (8), based on the delay-free model (1). The results rely on the following assumption:

Assumption (Boundedness).
There exists a constant $D_h > 0$ such that $|g(x_t, u)| \leq D_h$ for all $u \in U$ and all solutions x_t of (8) that completely lies in X; that is, $x_t(s) \in X$ for all $s \in [-h, 0]$. □

In most delay models, $D_h \to 0$ as $h \to 0$ for compact sets X and U. We will treat $g(x_t, u)$ as additive disturbances to the right-hand side of (1). Therefore, the results also work for general disturbances satisfying a boundedness condition as in the above assumption. Similar to that for previous results, we let M be such that $|F(x_t, u)| \leq M$ for all $u \in U$ and all solutions x_t of (8) that completely lies in X.

THEOREM 5. *Suppose the boundedness assumption holds and that (8) is to be controlled with a hybrid controller that is designed using the delay-free model (1). If $\mathcal{T}_c \preceq_{(\eta,\gamma_1,\gamma_2,\delta)} \hat{\mathcal{T}}_c$ with $\gamma_1 \geq \eta$, $\gamma_2 \geq (e^{L\Delta} - 1)D_h/L + \eta$, where L is the uniform Lipschitz constant of $f(\cdot, u)$ on X for all $u \in U$, and $\delta \geq \Delta M/2 + \eta$, then, given any LTL$_{\setminus\bigcirc}$ formula φ, φ being realizable for $\hat{\mathcal{T}}_c$ implies that φ is realizable for $\mathcal{T}_c$.*

PROOF. Let x_0, x_1, x_2, $\cdots$, be the measurements taken for the system (8) at times τ_0, τ_1, τ_2, $\cdots$; that is $x_i = x(\tau_i)$ for all $i \geq 0$, where $\tau_0 = 0$ and $\tau_{i+1} = \tau_i + \Delta_i$ for all $i \geq 0$ and $\mathbf{u}_i$ is activated on $[\tau_i, \tau_i + \Delta_i]$ for each $i \geq 0$. The only thing that needs to be proved is that the abstraction based on model 1 actually accounts for all possible behaviors of solutions of (8). That is, each transition from x_i to x_{i+1} is captured by a transition $\hat{q}_i$ to $\hat{q}_{i+1}$ in the abstraction. We only need to verify that there exists a trajectory ξ of (1) under inputs $\mathbf{u}_i$ such that $|\xi(0) - \hat{q}_i| \leq \gamma_1$ and $|\xi(\Delta_i) - \hat{q}_{i+1}| \leq \gamma_2$. Let ξ be a solution of (1) starting from x_i. We have $\xi(0) = \xi(\tau_i)$ and $\dot{\xi}(s) = f(\xi(s), \mathbf{u}_i(s))$ for all $s \in [0, \Delta_i]$. Define $y(s) = x(\tau_i + s)$ for $s \in [-h, \Delta_i]$. Then $y(0) = x(\tau_i)$ and $\dot{y}(s) = F(y_s, \mathbf{u}_i(s)) = f(y(s), \mathbf{u}_i(s)) + g(y_s, \mathbf{u}_i(s))$ for all $s \in [0, \Delta_i]$. Let $z(s) = y(s) - \xi(s)$ for $s \in [-h, \Delta_i]$. It follows that $|\dot{z}| \leq L\,|z| + D_h$ and $z(0) = 0$, where L is the uniform Lipschitz constant of $f(\cdot, u)$ on X for all $u \in U$ and D_h is the bound on g specified in the assumption. Using a differential inequality on $|z|$, it is easy to establish that

$|z(s)| \leq (e^{Ls}-1)d/L$ for $s \in [0, \Delta_i]$. Therefore, $|\xi(0) - \hat{q}_i| \leq |z(0)| + |x(\tau_i) - \hat{q}_i| \leq \eta \leq \gamma_1$ and $|\xi(\Delta_i) - \hat{q}_{i+1}| \leq |z(\Delta_i)| + |x(\tau_i + \Delta_i) - \hat{q}_{i+1}| \leq (e^{L\Delta} - 1)d/L + \eta \leq \gamma_2$. □

For discrete-time systems, we do not consider delays in this paper, but the following result gives robustness with respect to bounded additive disturbances. The proof is omitted.

THEOREM 6. *Suppose that (2) is subject to additive disturbances bounded by d. If $\mathcal{T}_d \preceq_{(\eta,\gamma_1,\gamma_2,\delta)} \hat{\mathcal{T}}_d$ with $\gamma_1 \geq \eta$, $\gamma_2 \geq d + \eta$, and $\delta \geq \eta$, then, given any LTL$_{\setminus\bigcirc}$ formula φ, φ being realizable for $\hat{\mathcal{T}}_d$ implies that φ is realizable for $\mathcal{T}_d$.*

4.4 Justification of time-discretization-based design

There are situations one would like to use a time-discretized model to design controllers for a continuous-time system, for example, when there is already a design methodology proved to be effective for discretized systems. What are the issues that need to be considered to ensure the performance of the resulted controller? This is a standard question in the design of stabilizing controllers (e.g., [5]). Here we consider it in the context of hybrid control for temporal logic objectives.

Let (2) be a time-discretized model for (1), which could be an exact model (e.g., available in the case where f is linear) or an approximate model (such as obtained from applying a numerical scheme). For example, $g(x, u)$ can be defined by $g(x, u) = x + \Delta f(x, u)$ as in a forward Euler scheme with a constant step size Δ. We only consider the case of constant step size and write the time-discretized control system as

$$x^+ = g_\Delta(x, u), \tag{10}$$

where $x \in X \subseteq \mathbb{R}^n$ and $u \in U \subseteq \mathbb{R}^m$ and g_Δ is a suitable one-step numerical scheme with a constant step size Δ.

Assumption (Consistency).
The numerical scheme g_Δ satisfies

$$|x(\Delta; x_0) - g_\Delta(x_0, u)| \leq \Delta C(\Delta),$$

for all $x_0 \in X$ and $u \in U$, where $C(\Delta) \to 0$ as $\Delta \to 0$. □

For example, for the forward Euler scheme with a fixed step size Δ, the above assumption holds with $C(\Delta) = (e^{L\Delta} - 1)/L$, where L is the uniform Lipschitz constant of $f(\cdot, u)$ on X for all $u \in U$.

THEOREM 7. *Suppose the consistency assumption holds and that (1) is to be controlled with a hybrid controller synthesized using the time-discretized model (10). If $\mathcal{T}_c \preceq_{(\eta,\gamma_1,\gamma_2,\delta)} \hat{\mathcal{T}}_c$ with $\gamma_1 \geq \eta$, $\gamma_2 \geq \Delta C(\Delta) + \eta$, and $\delta \geq \Delta M/2 + \eta$, then, given any LTL$_{\setminus\bigcirc}$ formula φ, φ being realizable for $\hat{\mathcal{T}}_c$ implies that φ is realizable for $\mathcal{T}_c$ and the controlled executions of $\mathcal{T}_c$ lead to solutions of (1) that satisfy φ.*

PROOF. Let x_0, x_1, x_2, $\cdots$, be the measurements taken for the system (1) at times τ_0, τ_1, τ_2, $\cdots$; that is $x_i = x(\tau_i)$ for all $i \geq 0$, where $\tau_0 = 0$ and $\tau_{i+1} = \tau_i + \Delta_i$ for all $i \geq 0$, $\mathbf{u}_i \equiv u_i$ on $[\tau_i, \tau_i + \Delta_i]$ for each $i \geq 0$, and u_i is a control input given by the discrete strategy. We need to show that: (1) every measured state is accounted for in the abstraction (computed from the discretized model), so that the discrete control strategy can be implemented; (2) the plant trajectory $x(t)$, $t \geq 0$, should satisfy the desired specification φ. Let $\{\hat{q}_i\}$ denote a sequence of abstract states corresponding to $\{x_i\}$.

To prove (1): for each i, we need to show that there exists ξ and ξ' such that $|\xi - \hat{q}_i| \leq \gamma_1$, $|\xi' - \hat{q}_{i+1}| \leq \gamma_2$, and $\xi' = g_\Delta(\xi, u_i)$. We let $\xi = x_i$ and $\xi' = g_\Delta(x_i, u_i)$. Then $|\xi - \hat{q}_i| \leq \eta \leq \gamma_1$. Moreover, it follows from the one-step consistency assumption that $|x_{i+1} - g_\Delta(x_i, u_i)| \leq \Delta C(\Delta)$ and $|\xi' - \hat{q}_{i+1}| \leq |x_{i+1} - g_\Delta(x_i, u_i)| + |x_{i+1} - \hat{q}_{i+1}| \leq \Delta C(\Delta) + \eta \leq \gamma$.

To prove (2): We can prove $x \vDash \varphi$ following the proof of Theorem 1. □

5. EXAMPLE

We consider a simple cruise control example where the goal is to regulate the vehicle's velocity to a desired range while respecting speed limits. The longitudinal dynamics of the car is given by

$$\dot{v} = u - c_0 - c_1 v^2 \tag{11}$$

where $v \in [v_{\min}, v_{\max}]$ is the velocity of the car, $u \in [-3a, 2a]$ is the scaled input acceleration and c_i for $i = 1, 2$ are proper constants to account for rolling resistance, air drag and headwind [16], which are chosen as $c_0 = 0.1$, $c_1 = 0.00016$, $a = 0.5$. The unit of velocity is in meters per second (m/s).

We consider a specification of the form

$$\varphi \equiv \Box(v \leq 30) \wedge \Diamond\Box(v \in [22, 24]),$$

which respects a speed limit of 30 and eventually reaches and stays within the desired range $[22, 24]$. To demonstrate the results in Section 4, we assume that the measurement of v involves a bounded error in the range $[-\varepsilon, \varepsilon]$ with $\varepsilon = 0.1$ and there is a round-trip delay in sensing, computation, and actuation, as illustrated in Figure 3, that is bounded by a constant $h = 0.01$. For $[v_{\min}, v_{\max}] = [20, 30]$ and $[-3a, 2a] = [-1.5, 1]$, $M = \sup_{v \in [v_{\min}, v_{\max}], u \in [-3a, 2a]} |f(x, u)| = 3a + c_0 + 900c_1 = 1.744$. Therefore, according to Theorem 3, we can choose an $(\eta, \gamma_1, \gamma_2, \delta)$-abstraction with $\eta = 0.05$, $\gamma_1 = 0.1674$, $\gamma_2 = 0.15$, and $\delta = 0.5947$ to formulate a discrete synthesis problem. We compute such an abstraction by discretizing $[v_{\min}, v_{\max}]$ with grid size 0.1. To compute transitions, it is easy to show that the estimate (4) holds with $\beta(r, t) = re^{-40c_1 t}$ on $[v_{\min}, v_{\max}]$ for $u \in [-3a, 2a]$. Proposition 1 is then used to compute transitions. The resulting discrete synthesis problem is solved using TuLiP [27]. Simulation results that illustrate the implementations of the discrete strategies are shown in Figure 4, which demonstrate that it is important to account for measurement errors and delays within the abstractions used for controller synthesis.

6. RELATED WORK

There are two common ways to construct finite abstractions. One is to partition the state space into a finite number of "proposition-preserving" regions (see, e.g., [17, 26]). This approach has the advantage of resulting in a small number of abstract states (given by elements in the partition) and also do not require any stability assumptions on the system dynamics. However, the fact that the computation of transitions in this type of abstraction relies heavily on the geometry of the vector fields with respect to the partition makes it difficult to incorporate robustness measures, especially those to deal with imperfect state information except for some special cases [14].

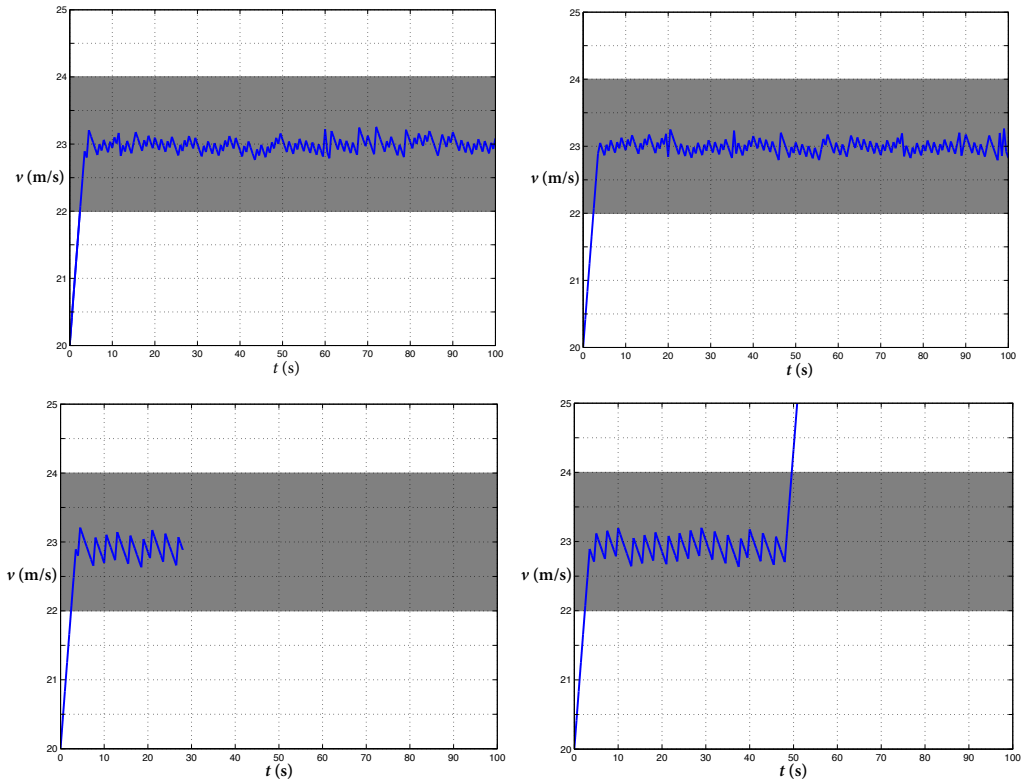

Figure 4: **Simulation results for the cruise control example (11), where the system is subject to measurement errors bounded by $\varepsilon = 0.1$ and a delay in sensing, computation, and actuation bounded by $h = 0.01$. We use an $(\eta, \gamma_1, \gamma_2, \delta)$-abstraction (given by Definition 1) of (11) to synthesize a hybrid control strategy. The upper two figures show simulated trajectories generated by a controller synthesized using an $(\eta, \gamma_1, \gamma_2, \delta)$-abstraction with $\eta = 0.05$, $\gamma_1 = 0.1674$, $\gamma_2 = 0.15$, and $\delta = 0.5947$, where γ_i $(i = 1, 2)$ are used to account for measurement errors and delays as shown in Theorem 3. The grey band indicates the desired speed range $[22, 24]$. The delays and measurements are randomly generated, which clearly led to somewhat random trajectories as opposed to periodic ones. The lower two figures show what could happen if measurement errors and delays are not accounted for within the abstractions, where we have chosen $\gamma_1 = \gamma_2 = \eta = 0.05$, while δ is kept the same. The lower left figure shows a termination of simulation when the measured system state, due to uncertainties in measurements, is mapped to an unexpected discrete state in the controlling automaton. One may keep executing the previous control input if the measurement is unexpected, but this may lead to violation of system specification as shown in the lower right figure.**

Another approach is to discretize the state space. This has been extensively used for constructing approximate symbolic models for control systems (see, e.g., [15, 18, 19, 21, 28]) based on the notion of approximate (bi)simulation [6]. In these approaches, a finite transition system model is constructed by discretizing the time, the input space, and the continuous state space. Under certain incremental stability assumptions, the resulting finite system can be shown to be approximately bisimilar to the time-discretized model of a continuous-time control system. The stability assumption can be relaxed [28] if one is interested in constructing simulations instead of bisimulations. The advantage of this approach is that it provides a quantitative measure of the fidelity of abstractions using metric transition systems. However in above mentioned papers, the approximation is between the finite abstraction and the time-discretized model of a continuous-time control system and it is unclear how to handle imperfect state information. In this paper we considered a discretization-based approach and addressed these shortcomings. In particular, we introduced abstractions with robustness margins to rigorously reason about the inter-sample behaviors and to account for imperfections in measurements and models.

The type of robustness considered in this paper is related to but distinct from that of [13, 24]. The focus of [13, 24] is on the design of discrete controllers for finite transition systems (namely, discrete synthesis) against unmodeled disturbances or transitions, whereas the current paper aims to establish robustness of discrete controllers against imperfect measurements and unmodeled dynamics in the continuous plants.

Our work is also related to control of hybrid systems with imperfect state information. In [10], the author considered stability of switched systems with limited information under slow switching. Limited information refers to the situation where the state measurements are taken only at sampling times and quantized using a finite alphabet. This is exactly how the hybrid controller is implemented in this paper: it takes measurements at sampling times, maps it to discrete states in the finite abstractions, and looks for appropriate control actions, based on an automaton that represents a discrete control strategy.

7. CONCLUSIONS

In this paper we presented a notion of abstractions with robustness margins and showed that it is possible to synthesize provably-correct robust feedback controllers based on such abstractions. This allows us to handle various types of imperfections in the models or measurements and to reason about implementation artifacts in a unified fashion. The idea can be naturally generalized to multi-scale abstractions where the abstract states are non-uniformly distributed around the continuous state space. Future work will include investigating such abstractions and combining them with automated refinement procedures to mitigate potential state explosion problem.

8. REFERENCES

[1] R. Alur, T. A. Henzinger, O. Kupferman, and M. Y. Vardi. Alternating refinement relations. In *Proc. International Conference on Concurrency Theory (CONCUR)*, pages 163–178, 1998.

[2] D. Angeli. A Lyapunov approach to incremental stability properties. *IEEE Trans. on Automatic Control*, 47(3):410–421, 2002.

[3] G. E. Fainekos, A. Girard, H. Kress-Gazit, and G. J. Pappas. Temporal logic motion planning for dynamic robots. *Automatica*, 45:343–352, 2009.

[4] G. E. Fainekos and G. J. Pappas. Robustness of temporal logic specifications for continuous-time signals. *Theoretical Computer Science*, 410(42):4262–4291, 2009.

[5] G. F. Franklin, M. L. Workman, and D. Powell. *Digital Control of Dynamic Systems*. Addison-Wesley Longman Publishing Co., Inc., 1997.

[6] A. Girard and G. Pappas. Approximation metrics for discrete and continuous systems. *IEEE Trans. on Automatic Control*, 52:782–798, 2007.

[7] S. Karaman and E. Frazzoli. Sampling-based algorithms for optimal motion planning with deterministic μ-calculus specifications. In *Proc. of American Control Conference (ACC)*, pages 735–742, 2012.

[8] M. Kloetzer and C. Belta. A fully automated framework for control of linear systems from temporal logic specifications. *IEEE Trans. Automatic Control*, 53:287–297, 2008.

[9] H. Kress-Gazit, G. E. Fainekos, and G. J. Pappas. Temporal-logic-based reactive mission and motion planning. *IEEE Trans. Robotics*, 25:1370–1381, 2009.

[10] D. Liberzon. Limited-information control of hybrid systems via reachable set propagation. In *Proc. of the 16th International Conference on Hybrid Systems: Computation and Control (HSCC)*, pages 11–20, 2013.

[11] J. Liu, N. Ozay, U. Topcu, and R. Murray. Synthesis of reactive switching protocols from temporal logic specifications. *IEEE Trans. on Automatic Control*, 58(7):1771–1785, 2013.

[12] J. Liu, U. Topcu, N. Ozay, and R. M. Murray. Reactive controllers for differentially flat systems with temporal logic constraints. In *Proc. of the 51st IEEE Conference on Decision and Control (CDC)*, pages 7664–7670, 2012.

[13] R. Majumdar, E. Render, and P. Tabuada. Robust discrete synthesis against unspecified disturbances. In *Proc. of the 14th International Conference on Hybrid Systems: Computation and Control (HSCC)*, pages 211–220, 2011.

[14] O. Mickelin, N. Ozay, and R. M. Murray. Synthesis of correct-by-construction control protocols for hybrid systems using partial state information. In *Proc. of American Control Conference (ACC)*, 2014.

[15] S. Mouelhi, A. Girard, and G. Gössler. Cosyma: a tool for controller synthesis using multi-scale abstractions. In *Proc. of the 16th International Conference on Hybrid Systems: Computation and Control (HSCC)*, pages 83–88, 2013.

[16] G. Orosz and S. P. Shah. A nonlinear modeling framework for autonomous cruise control. In *Proc. of ASME Annual Dynamic Systems and Control Conference joint with JSME Motion & Vibration Conference*, pages 467–471, 2012.

[17] N. Ozay, J. Liu, P. Prabhakar, and R. M. Murray. Computing augmented finite transition systems to synthesize switching protocols for polynomial switched systems. In *Proc. of American Control Conference (ACC)*, 2013.

[18] G. Pola, P. Pepe, M. D. Di Benedetto, and P. Tabuada. Symbolic models for nonlinear time-delay systems using approximate bisimulations. *Systems & Control Letters*, 59(6):365–373, 2010.

[19] G. Pola and P. Tabuada. Symbolic models for nonlinear control systems: Alternating approximate bisimulations. *SIAM J. Control Optim.*, 48:719–733, 2009.

[20] G. Reiszig. Computing abstractions of nonlinear systems. *IEEE Trans. Automatic Control*, 56:2583–2598, 2011.

[21] P. Tabuada. *Verification and Control of Hybrid Systems: A Symbolic Approach*. Springer, 2009.

[22] P. Tabuada and G. J. Pappas. Linear time logic control of discrete-time linear systems. *IEEE Trans. Automatic Control*, 51:1862–1877, 2006.

[23] Y. Tazaki and J. Imura. Discrete abstractions of nonlinear systems based on error propagation analysis. *IEEE Trans. Automatic Control*, 57:550–564, 2012.

[24] U. Topcu, N. Ozay, J. Liu, and R. M. Murray. On synthesizing robust discrete controllers under modeling uncertainty. In *Proc. of the 15th ACM International Conference on Hybrid Systems: Computation and Control (HSCC)*, pages 85–94, 2012.

[25] E. M. Wolff and R. M. Murray. Optimal control of nonlinear systems with temporal logic specifications. In *Proc. of the International Symposium on Robotics Research (ISRR)*, 2013.

[26] T. Wongpiromsarn, U. Topcu, and R. M. Murray. Receding horizon temporal logic planning. *IEEE Trans. Automatic Control*, 57:2817–2830, 2012.

[27] T. Wongpiromsarn, U. Topcu, N. Ozay, H. Xu, and R. M. Murray. TuLiP: a software toolbox for receding horizon temporal logic planning. In *Proc. of the 14th International Conference on Hybrid Systems: Computation and Control (HSCC)*, 2011.

[28] M. Zamani, G. Pola, M. Mazo Jr, and P. Tabuada. Symbolic models for nonlinear control systems without stability assumptions. *IEEE Trans. Automatic Control*, 57:1804–1809, 2012.