



Formal Synthesis of Control Strategies for Positive Monotone Systems

Sadra Sadraddini , *Student Member, IEEE*, and Calin Belta , *Fellow, IEEE*

Abstract—We design controllers from formal specifications for positive discrete-time monotone systems that are subject to bounded disturbances. Such systems are widely used to model the dynamics of transportation and biological networks. The specifications are described using signal temporal logic (STL), which can express a broad range of temporal properties. We formulate the problem as a mixed-integer linear program (MILP) and show that under the assumptions made in this paper, which are not restrictive for traffic applications, the existence of open-loop control policies is sufficient and almost necessary to ensure the satisfaction of STL formulas. We establish a relation between satisfaction of STL formulas in infinite time and set-invariance theories and provide an efficient method to compute robust control invariant sets in high dimensions. We also develop a robust model predictive framework to plan controls optimally while ensuring the satisfaction of the specification. Illustrative examples and a traffic management case study are included.

Index Terms—Formal synthesis and verification, monotone systems, transportation networks.

I. INTRODUCTION

IN RECENT years, there has been a growing interest in using formal methods for specification, verification, and synthesis in control theory. Temporal logics [1] provide a rich, expressive framework for describing a broad range of properties, such as safety, liveness, and reactivity. In formal synthesis, the goal is to control a dynamical system from such a specification. For example, in an urban traffic network, a synthesis problem can be to generate traffic light control policies that ensure gridlock avoidance and fast enough traffic through a certain road, for all times.

Control synthesis for linear and piecewise affine systems from linear temporal logic (LTL) specifications was studied in [2]–[4]. The automata-based approach used in these works requires constructing finite abstractions that (bi)simulate the original system. Approximate finite bisimulation quotients for nonlinear systems were investigated in [5] and [6]. The main limitations of finite

Manuscript received February 26, 2017; revised September 12, 2017 and December 14, 2017; accepted February 23, 2018. Date of publication March 9, 2018; date of current version January 28, 2019. This work was partially supported by the National Science Foundation (NSF) under Grant CPS-1446151 and Grant CMMI-1400167. Recommended by Associate Editor A. Girard. (*Corresponding author: Sadra Sadraddini.*)

The authors are with the Department of Mechanical Engineering, Boston University, Boston, MA 02215 USA (e-mail: sadra@bu.edu; cbelta@bu.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TAC.2018.2814631

abstraction approaches are the large computational burden of discretization in high dimensions and conservativeness when exact bisimulations are impossible or difficult to construct. As an alternative approach, LTL optimization-based control of mixed-logical dynamical (MLD) systems [7] using mixed-integer programs was introduced in [8] and [9], and was recently extended to model predictive control (MPC) from signal temporal logic (STL) specifications in [10]–[12]. However, these approaches are unable to guarantee infinite-time safety, and the results are fragile in the presence of nondeterministic disturbances.

In some applications, the structural properties of the system and the specification can be exploited to consider alternative approaches to formal control synthesis. We are interested in systems in which the evolution of the state exhibits a type of *order preserving law* known as *monotonicity*, which is common in models of transportation, biological, and economic systems [13]–[16]. Such systems are also *positive* in the sense that the state components are always nonnegative. Control of positive systems have been widely studied in the literature [17]–[19]. Positive linear systems are always monotone [20].

In this paper, we study optimal STL control of discrete-time positive monotone systems (i.e., systems with state partial order on the positive orthant) with bounded disturbances. STL allows designating time intervals for temporal operators, which makes it suitable for describing requirements with deadlines. Moreover, STL is equipped with quantitative semantics, which provides a measure to quantify how strongly the specification is satisfied/violated. The quantitative semantics of STL can also be used as cost for maximization in an optimal control setting. The STL specifications in this paper are restricted to a particular form that favors smaller values for the state components. We assume that there exists a maximal disturbance element that characterizes a type of upper-bound for the evolution of the system. These assumptions are specifically motivated by the dynamics of traffic networks, where the disturbances represent the volume of exogenous vehicles entering the network and the maximal disturbance characterizes the rush hour exogenous flow. Our optimal control study is focused on STL formulae with infinite-time safety/persistence properties, which is relevant to optimal and correct traffic control in the sense that the vehicular flow is always free of congestion while the associated delay is minimized.

The key contributions of this paper are as follows. First, for finite-time semantics, we prove that the existence of open-loop control policies is necessary and sufficient for maintaining STL correctness. For the correctness of infinite-time semantics, we show that the existence of open-loop control sequences is sufficient and almost necessary, in a sense that is made clear in the paper. Implementing open-loop control policies is very simple since online state measurements are not required, which can

prove useful in applications where the state is difficult to access. We use a robust MPC approach to optimal control. The main contribution of our MPC framework is guaranteed recursive feasibility, a property that was not established in prior STL MPC works [10]–[12]. We show via a case study that our method is applicable to systems with relatively high dimensions.

This remainder of the paper is organized as follows. We introduce the necessary notation and background on STL in Section II. The problems are formulated in Section III. The technical details for control synthesis from finite and infinite-time specifications are given in Section IV and Section V, respectively. The robust MPC framework is explained in Section VI. Finally, we introduce a traffic network model and explain its monotonicity properties in Section VII, where a case study is also presented.

A. Related Work

This paper is an extension of the conference version [21], where we studied safety control of positive monotone systems. Here, we significantly enrich the range of specifications to STL, provide complete proofs, and include optimal control.

Monotone dynamical systems have been extensively investigated in the mathematics literature [22]–[24]. Early studies mainly focused on stability properties and characterization of limit sets for autonomous, deterministic continuous-time systems [22], [25]. The results do not generally hold for discrete-time systems, as discussed in [23]. In particular, attractive periodic orbits are proven to be nonexistent for continuous-time autonomous systems [25], but may exist for discrete-time autonomous systems. Here, we present a similar result for controlled systems, where we show that a type of attractive periodic orbit exists for certain control policies.

Angeli and Sontag [26] extended the notion of monotonicity to deterministic continuous-time control systems and provided results on interconnections of these systems. However, they assumed monotonicity with respect to both state and controls. We do not require monotonicity with respect to controls, which enables us to consider a broader class of systems. In particular, we do not require controls to belong to a partially ordered set.

Switching policies for exponential stabilization of switched positive linear systems were studied in [27] and [28]. Stabilization is closely related to set-invariance, which is thoroughly studied in this paper. Apart from richer specifications, we can handle more complex systems. We consider hybrid systems in which the mode is either determined directly by the control input or indirectly by the state (e.g., signalized traffic networks).

Recently, there has been some interest on formal verification and synthesis for monotone systems. Safety control of cooperative systems was investigated in [29]–[31]. However, these work, like [26], assumed monotonicity with respect to the control inputs as well. Computational benefits gained from monotonicity for reachability analysis of hybrid systems were highlighted in [32]. More recently, Coogan and Arcak [33] provided an efficient method to compute finite abstractions for mixed-monotone systems (a more general class than monotone systems). Kim *et al.* [34] exploited monotonicity to compute finite-state abstractions that are used for compositional LTL control. While the approaches in [33] and [34] can consider systems and specifications beyond the assumptions in this paper, they still require state-space discretization, which is a severe limitation in high dimensions. Moreover, they are conservative since the finite abstractions are often not bisimilar with the

original system—whereas our approach provides a notion of (almost) completeness. Finally, as opposed to the all mentioned works, our framework is amenable to optimal temporal logic control.

II. PRELIMINARIES

A. Notation

For two integers a and b , we use $\text{rem}(a, b)$ to denote the remainder of division of a by b . Given a set \mathcal{S} and a positive integer K , we use the shorthand notation \mathcal{S}^K for $\prod_{i=1}^K \mathcal{S}$. A signal is defined as an infinite sequence $\mathbf{s} = s_0 s_1 \dots$, where $s_k \in \mathcal{S}$, $k \in \mathbb{N}$. Given $s_1, s_2, \dots, s_K \in \mathcal{S}$, the repetitive infinite-sequence $s_1 s_2 \dots s_K s_1 s_2 \dots s_K \dots$ is denoted by $(s_1 s_2 \dots s_K)^\omega$. The set of all signals that can be generated from \mathcal{S} is denoted by \mathcal{S}^ω . We use $\mathbf{s}[k] = s_k s_{k+1} \dots$ and $\mathbf{s}[k_1 : k_2] = s_{k_1} s_{k_1+1} \dots s_{k_2}$, $k_1 < k_2$, to denote specific portions of \mathbf{s} . A real signal is $\mathbf{r} = r_0 r_1 r_2 \dots$, where $r_k \in \mathbb{R}^n$, $\forall k \in \mathbb{N}$. A vector of all ones in \mathbb{R}^n is denoted by $\mathbf{1}_n$. We use the notation $\mathbf{1}_n[0 : K] := \mathbf{1}_n \dots \mathbf{1}_n$, where $\mathbf{1}_n$ is repeated $K + 1$ times. The positive closed orthant of the n -dimensional Euclidian space is denoted by $\mathbb{R}_+^n := \{x \in \mathbb{R}^n | x_{[i]} \geq 0, i = 1, \dots, n\}$, where $x = (x_{[1]}, x_{[2]}, \dots, x_{[n]})^T$. For $a, b \in \mathbb{R}^n$, the nonstrict partial order relation \preceq is defined as: $a \preceq b \Leftrightarrow b - a \in \mathbb{R}_+^n$.

Definition 1 ([35]): A set $\mathcal{X} \subset \mathbb{R}_+^n$ is a lower set if $\forall x \in \mathcal{X}, L(x) \subseteq \mathcal{X}$, where $L(x) := \{y \in \mathbb{R}_+^n | y \preceq x\}$.

It is straightforward to verify that if \mathcal{X}_1 and \mathcal{X}_2 are lower sets, then $\mathcal{X}_1 \cup \mathcal{X}_2$ and $\mathcal{X}_1 \cap \mathcal{X}_2$ are also lower sets. We extend the usage of notation \preceq to equal-length real signals. For two real signals \mathbf{r}, \mathbf{r}' , we denote $\mathbf{r}'[t'_1 : t'_2] \preceq \mathbf{r}[t_1 : t_2]$, $t_2 - t_1 = t'_2 - t'_1$, if $\mathbf{r}'_{t'_1+k} \preceq \mathbf{r}_{t_1+k}$, $k = 0, 1, \dots, t_2 - t_1$. Moreover, if $\mathbf{r}, \mathbf{r}' \in (\mathbb{R}_+^n)^\omega$, we are also allowed to write $\mathbf{r}'[t'_1 : t'_2] \in L(\mathbf{r}[t_1 : t_2])$.

B. Signal Temporal Logic (STL)

In this paper, STL [36] formulas are defined over discrete-time real signals. The syntax of negation-free STL is

$$\varphi := \pi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \mathbf{U}_I \varphi_2 \mid \mathbf{F}_I \varphi \mid \mathbf{G}_I \varphi \quad (1)$$

where $\pi = (p(r) \leq c)$ is a predicate on $r \in \mathbb{R}^n$, $p : \mathbb{R}^n \rightarrow \mathbb{R}$, $c \in \mathbb{R}$; \wedge and \vee are Boolean connectives for conjunction and disjunction, respectively; \mathbf{U}_I , \mathbf{F}_I , and \mathbf{G}_I are the timed *until*, *eventually*, and *always* operators, respectively, and $I = [t_1, t_2]$ is a time interval, $t_1, t_2 \in \mathbb{N} \cup \{\infty\}$, $t_2 \geq t_1$. When $t_1 = t_2$, we use the shorthand notation $\{t_1\} := [t_1, t_1]$. Exclusion of negation does not restrict expressivity of temporal properties. It can be easily shown that any temporal logic formula can be brought into *negation normal form* (where all negation operators apply to the predicates) [12], [37]. We deliberately omit negation from STL syntax for laying out properties that are later exploited in the paper. For simplicity, in the rest of the paper, we will refer to negation-free STL simply as STL. The semantics of STL is inductively defined as

$$\begin{aligned} \mathbf{r}[t] \models \pi & \Leftrightarrow p(r_t) \leq c \\ \mathbf{r}[t] \models \varphi_1 \vee \varphi_2 & \Leftrightarrow \mathbf{r}[t] \models \varphi_1 \vee \mathbf{r}[t] \models \varphi_2 \\ \mathbf{r}[t] \models \varphi_1 \wedge \varphi_2 & \Leftrightarrow \mathbf{r}[t] \models \varphi_1 \wedge \mathbf{r}[t] \models \varphi_2 \\ \mathbf{r}[t] \models \varphi_1 \mathbf{U}_I \varphi_2 & \Leftrightarrow \exists t' \in t + I \text{ s.t. } \mathbf{r}[t'] \models \varphi_2 \\ & \quad \wedge \forall t'' \in [t, t'], \mathbf{r}[t''] \models \varphi_1 \\ \mathbf{r}[t] \models \mathbf{F}_I \varphi & \Leftrightarrow \exists t' \in t + I \text{ s.t. } \mathbf{r}[t'] \models \varphi \\ \mathbf{r}[t] \models \mathbf{G}_I \varphi & \Leftrightarrow \forall t' \in t + I \text{ s.t. } \mathbf{r}[t'] \models \varphi \end{aligned} \quad (2)$$

where \models is read as *satisfies*. The *language* of φ is the set of all signals such that $\mathbf{r}[0] \models \varphi$. The *horizon* of an STL formula φ , denoted by h^φ , is defined as the time required to decide the satisfaction of φ , which is recursively computed as [38]

$$\begin{aligned} h^\pi &= 0 \\ h^{\varphi_1 \wedge \varphi_2} &= h^{\varphi_1 \vee \varphi_2} = \max(h^{\varphi_1}, h^{\varphi_2}) \\ h^{\mathbf{F}_{[t_1, t_2]} \varphi} &= h^{\mathbf{G}_{[t_1, t_2]} \varphi} = t_2 + h^\varphi \\ h^{\varphi_1 \mathbf{U}_{[t_1, t_2]} \varphi_2} &= t_2 + \max(h^{\varphi_1}, h^{\varphi_2}). \end{aligned} \quad (3)$$

Definition 2: An STL formula φ is *bounded* if $h^\varphi < \infty$.

Definition 3 ([39]): A *safety STL* formula is an STL formula in which all “until” and “eventually” intervals are bounded.

The satisfaction of φ by $\mathbf{r}[t]$ is decided only by $\mathbf{r}[t : t + h^\varphi]$ and the rest of the signal values are irrelevant. Therefore, instead of $\mathbf{r}[t] \models \varphi$, we occasionally write $\mathbf{r}[t : t + h^\varphi] \models \varphi$ with the same meaning. The *STL robustness score* $\rho(\mathbf{r}, \varphi, t) \in \mathbb{R}$ is a measure indicating how strongly φ is satisfied by $\mathbf{r}[t]$, which is recursively computed as [36]

$$\begin{aligned} \rho(\mathbf{r}, \pi, t) &= c - p(r_t) \\ \rho(\mathbf{r}, \varphi_1 \vee \varphi_2, t) &= \max(\rho(\mathbf{r}, \varphi_1, t), \rho(\mathbf{r}, \varphi_2, t)) \\ \rho(\mathbf{r}, \varphi_1 \wedge \varphi_2, t) &= \min(\rho(\mathbf{r}, \varphi_1, t), \rho(\mathbf{r}, \varphi_2, t)) \\ \rho(\mathbf{r}, \varphi_1 \mathbf{U}_I \varphi_2, t) &= \max_{t' \in t+I} (\min(\rho(\mathbf{r}, \varphi_2, t') \\ &\quad \min_{t'' \in [t, t']} \rho(\mathbf{r}, \varphi_1, t''))) \\ \rho(\mathbf{r}, \mathbf{F}_I \varphi, t) &= \max_{t' \in t+I} \rho(\mathbf{r}, \varphi, t') \\ \rho(\mathbf{r}, \mathbf{G}_I \varphi, t) &= \min_{t' \in t+I} \rho(\mathbf{r}, \varphi, t'). \end{aligned} \quad (4)$$

Positive (respectively, negative) robustness indicates satisfaction (respectively, violation) of the formula.

Example 1: Consider signal $\mathbf{r} \in \mathbb{R}^\omega$, where $r_k = k, k \in \mathbb{N}$, and $\pi = (r^2 \leq 10)$. We have $\rho(\mathbf{r}, \mathbf{G}_{[0,3]} \pi, 0) = \min(10 - 0, 10 - 1, 10 - 4, 10 - 9) = 1$ (satisfaction) and $\rho(\mathbf{r}, \mathbf{F}_{[4,6]} \pi, 0) = \max(10 - 16, 10 - 25, 10 - 36) = -6$ (violation).

Remark 1: There are minor differences between the original STL introduced in [36] and the one used in this paper. In [36], STL was developed as an extension of metric interval temporal logic [39] for real-valued continuous-time signals. Here, without any loss of generality, we apply STL to discrete-time signals. Our STL is based on metric temporal logic (similar to [38]). Thus, we allow the intervals of temporal operators to be singletons (punctual) or unbounded. It is worth to note that any STL formula in this paper can be translated into an LTL formula by appropriately replacing the time intervals of temporal operators with LTL “next” operator. However, the LTL representation of STL formulas can be very inefficient. We prefer STL for convenience of specifying requirements for systems with real-valued states. We also exploit the STL quantitative semantics.

III. PROBLEM STATEMENT AND APPROACH

We consider discrete-time systems of the following form:

$$x_{t+1} = f(x_t, u_t, w_t) \quad (5)$$

where $x_t \in \mathcal{X}$ is the state, $\mathcal{X} \subset \mathbb{R}^n$, $u_t \in \mathcal{U}$ is the control input, $\mathcal{U} = \mathbb{R}^{m_r} \times \{0, 1\}^{m_b}$, and $w_t \in \mathcal{W}$ is the disturbance (adversarial input) at time $t, t \in \mathbb{N}$, $\mathcal{W} = \mathbb{R}^{q_r} \times \{0, 1\}^{q_b}$. The sets \mathcal{U}

and \mathcal{W} may include real and binary values. For instance, the set of controls in the traffic model developed in Section VII includes binary values for decisions on traffic lights and real values for ramp meters. These types of systems are positive as all state components are nonnegative. We also assume that \mathcal{X} is bounded.

Definition 4: System (5) is *monotone* (with partial order on \mathbb{R}_+^n) if for all $x, x' \in \mathcal{X}$, $x' \preceq x$, we have $f(x', u, w) \preceq f(x, u, w), \forall u \in \mathcal{U}, \forall w \in \mathcal{W}$.

The systems considered in this paper are positive and monotone with partial order on \mathbb{R}_+^n . For the remainder of the paper, we simply refer to systems in Definition 4 as monotone.¹ Although the results of this paper are valid for any general $f : \mathcal{X} \times \mathcal{U} \times \mathcal{W} \rightarrow \mathcal{X}$, we focus on systems that can be written in the form of MLD systems [7], which are defined in Section IV. It is well known that a wide range of systems involving discontinuities (hybrid systems), such as piecewise affine systems, can be transformed into MLDs [40].

Assumption 1: There exist $w^* \in \mathcal{W}$ such that

$$\forall x \in \mathcal{X} \quad \forall u \in \mathcal{U}, f(x, u, w) \preceq f(x, u, w^*) \quad \forall w \in \mathcal{W}. \quad (6)$$

We denote $f(x, u, w^*)$ by $f^*(x, u)$ and refer to f^* as the *maximal system*. As it will be further explained in this paper, the behavior of monotone system (5) is mainly characterized by its maximal f^* . Assumption 1 is restrictive but holds for many compartmental systems where the disturbances are additive and the components are independent. Therefore, the maximal system corresponds to the situation that every component takes its most extreme value. We also note that if Assumption 1 is removed, overestimating f by some f^* such that $f(x, u, w) \preceq f^*(x, u), \forall w \in \mathcal{W}$ is always possible for a bounded f . By overestimating f the control synthesis methods of this paper remain correct, but become conservative.

We describe the desired system behavior using specifications written as STL formulas over a finite set of predicates. We assume that each predicate π is in the following form:

$$\pi = (a_\pi^T x \leq b_\pi) \quad (7)$$

where $a_\pi \in \mathbb{R}_+^n, b_\pi \in \mathbb{R}_+$. It is straightforward to verify that the closed half-space defined by (7) is a lower set in \mathbb{R}_+^n . By restricting the predicates into the form (7), we ensure that a predicate remains true if the values of state components are decreased (Note that this is true for any lower set. We require linearity in order to decrease the computational complexity.). This restriction is motivated by monotonicity. For example, in a traffic network, the state is the vector representation of vehicular densities in different segments of the network. The satisfaction of a “sensible” traffic specification has to be preserved if the vehicular densities are not increased all over the network. Otherwise, the specification encourages large densities and congestion.

Definition 5: A control policy $\mu := \bigcup_{t \in \mathbb{N}} \mu_t$ is a set of functions $\mu_t : \mathcal{X}^{t+1} \rightarrow \mathcal{U}$, where

$$u_t = \mu_t(x_0, x_1, \dots, x_t).$$

An *open-loop* control policy takes the simpler form $u_t = \mu_t(x_0)$, i.e., the decision on the sequence of control inputs is made using only the initial state x_0 . On the other hand, in a (history dependent) *feedback* control policy, $u_t =$

¹The term *cooperative* in dynamics systems theory is used specifically to refer to systems that are monotone with partial order defined on the positive orthant. We avoid using this term here as it might generate confusion with the similar terminology used for multiagent control systems.

$\mu_t(x_0, x_1, \dots, x_t)$, the controller implementation requires real-time access to the state and its history.

An infinite sequence of admissible disturbances is $\mathbf{w} = w_0 w_1 \dots$, where $w_k \in \mathcal{W}$, $k \in \mathbb{N}$. Following the notation introduced in Section II-A, the set of all infinite-length sequences of admissible disturbances is denoted by \mathcal{W}^ω . Given an initial condition x_0 , a control policy μ and $\mathbf{w} \in \mathcal{W}^\omega$, the *run* of the system is defined as the following signal:

$$\mathbf{x} = \mathbf{x}(x_0, \mu, \mathbf{w}) := x_0 x_1 x_2 \dots$$

where $x_{t+1} = f(x_t, u_t, w_t), \forall t \in \mathbb{N}$. Now, we formulate the problems studied in this paper. In all problems, we assume that a monotone system (5) is given, Assumption 1 holds, and all the predicates are in the form of (7).

Problem 1 (Bounded STL Control): Given a bounded STL formula φ , find a set of initial conditions $\mathcal{X}_0 \subset \mathcal{X}$ and a control policy μ such that

$$\mathbf{x}(x_0, \mu, \mathbf{w})[0] \models \varphi \quad \forall \mathbf{w} \in \mathcal{W}^\omega \quad \forall x_0 \in \mathcal{X}_0.$$

As mentioned in the previous section, the satisfaction of φ solely depends on $\mathbf{x}[0 : h^\varphi]$, where h^φ is obtained from (3). The horizon h^φ can be viewed as the time when the specification ends. In many engineering applications, the system is required to uphold certain behaviors for all times. Therefore, guaranteeing infinite-time safety properties is important. We formulate *bounded-global* STL formulas in the form of

$$\varphi_b \wedge \mathbf{G}_{[\Delta, \infty]} \varphi_g \quad (8)$$

where φ_b and φ_g are bounded STL formulas, $\mathbf{G}_{[\Delta, \infty]}$ stands for unbounded temporal “always”—as defined in Section II-B, and $\Delta \geq h^{\varphi_b}$ is a positive integer. Formula (8) states that first, φ_b is satisfied by the signal from time 0 to Δ , and afterwards, φ_g holds for all times.

Problem 2 (Bounded-global STL Control): Given bounded STL formulas φ_b, φ_g , $\Delta \in [h^{\varphi_b}, \infty)$ find a set of initial conditions $\mathcal{X}_0 \subset \mathcal{X}$ and a control policy μ such that

$$\mathbf{x}(x_0, \mu, \mathbf{w})[0] \models \varphi_b \wedge \mathbf{G}_{[\Delta, \infty]} \varphi_g \quad \forall \mathbf{w} \in \mathcal{W}^\omega \quad \forall x_0 \in \mathcal{X}_0. \quad (9)$$

As a special case, we allow φ_b to be logical truth so Problem 2 reduces to *global STL* control problem of satisfying $\mathbf{G}_{[\Delta, \infty]} \varphi_g$. Note that if φ_g is replaced by logical truth, Problem 2 reduces to Problem 1. We have distinguished Problem 1 and Problem 2 as we use different approaches to solve them.

It can be shown that (see Appendix) a large subset of safety STL formulas—as in Definition 3—can be written as $\bigvee_{i=1}^{n_\phi} \phi_i$, where each $\phi_i, i = 1, \dots, n_\phi$, is a bounded-global formula. Therefore, the framework for solutions to Problem 2 can also be used for safety STL control as it leads to n_ϕ instances of Problem 2, where a solution to any of the instances is also a solution to the original safety STL control problem. The drawback to this approach is that n_ϕ can be very large.

Remark 2: We avoid separate problem formulations for STL formulas containing unbounded “eventually” or “until” operators as their unbounded intervals can be safely underapproximated by bounded intervals. However, bounded underapproximation is not sound for the unbounded “always” operator. A safety formula can be satisfied (respectively, violated) with infinite-length (respectively, finite-length) signals [39].

In the presence of disturbances, feedback controllers obviously outperform open-loop controllers. We show that the

existence of open-loop control policies for guaranteeing the STL correctness of monotone systems in Problem 1 (respectively, Problem 2) is sufficient and (respectively, almost) necessary. The online knowledge of state is not necessary for STL correctness. But it can be exploited for planning controls optimally. While our framework can accommodate optimal control versions of Problem 1 and Problem 2, the focus of this paper is on robust optimal control problem for global STL formulas—of the form $\mathbf{G}_{[0, \infty]} \varphi$, where φ is a bounded formula. These type of problems are of practical interest for optimal traffic management (as discussed in Section VII).

We use a MPC approach, which is a popular, powerful approach to optimal control of constrained systems. Given a planning horizon of length H ,² a sequence of control actions starting from time t is denoted by $u_t^H := u_{0|t} u_{1|t} \dots u_{H-1|t}$. Given u_t^H and x_t , we denote the predicted H -step system response by

$$x_t^H(x_t, u_t^H, w_t^H) := x_{1|t} x_{2|t} \dots x_{H|t}$$

where $x_{k+1|t} = f(x_{k|t}, u_{k|t}, w_{k|t}), k = 0, 1, \dots, H-1, x_{0|t} = x_t$ and $w_t^H := w_{0|t} w_{1|t} \dots w_{H-1|t}$. At each time, u_t^H is found such that it optimizes a cost function $J(x_t^H, u_t^H), J : \mathcal{X}^H \times \mathcal{U}^H \rightarrow \mathbb{R}$, subject to system constraints. When u_t^H is computed, only the first control action $u_{0|t}$ is applied to the system and given the next state, the optimization problem is resolved for u_{t+1}^H . Thus, the implementation is closed loop.

Problem 3 (Robust STL MPC): Given a bounded STL formula φ , an initial condition x_0 , a planning horizon H , and a cost function $J(x_t^H, u_t^H)$, find a control policy such that $u_t = \mu(x_0, \dots, x_t) = u_{0|t}^{\text{opt}}$, where $u_t^{H, \text{opt}} := u_{0|t}^{\text{opt}} \dots u_{H-1|t}^{\text{opt}}$, and $u^{H, \text{opt}}$ is the following minimizer:

$$\begin{aligned} \operatorname{argmin}_{u_t^H} \quad & \max_{w_t^H} J(x_t^H(x_t, u_t^H, w_t^H), u_t^H) \\ \text{s.t.} \quad & \mathbf{x}(x_0, \mu, \mathbf{w})[0] \models \mathbf{G}_{[0, \infty]} \varphi \quad \forall \mathbf{w} \in \mathcal{W}^\omega \\ & x_{k+1} = f(x_k, u_k, w_k) \quad \forall k \in \mathbb{N}. \end{aligned} \quad (10)$$

The primary challenge of robust STL MPC is guaranteeing the satisfaction of the global STL formula while the controls are planned in a receding horizon manner [see the constraints in (10)]. Our approach takes the advantage of the results from Problem 2 to design appropriate terminal sets for the MPC algorithm such that the generated runs are guaranteed to satisfy the global STL specification while the online control decisions are computed (sub)optimally. Due to the temporal logic constraints, our MPC setup differs from the conventional one. The details are explained in Section VI.

For computational purposes, we assume that J is a piecewise affine function of the state and controls. Moreover, the cost functions in our applications are nondecreasing with respect to the state in the sense that $x'_{k|t} \preceq x_{k|t}, k = 1, 2, \dots, H \Rightarrow J(x_t^H, u_t^H) \preceq J(x_t^H, u_t^H), \forall u_t^H \in \mathcal{U}^H$. As it will become clear later in the paper, we will exploit this property to simplify the worst case optimization problem in (10) to an optimization problem for the maximal system.

As mentioned earlier, a natural objective is maximizing STL robustness score. It follows from the linearity of the predicates in (7) and max and min operators in (4) that STL robustness score is a piecewise affine function of finite-length signals. We

²The MPC horizon H should not be confused with the STL horizon h^φ .

can also consider optimizing a weighted combination of STL robustness score and a given cost function. We use this cost formulation for traffic application in Section VII.

IV. FINITE HORIZON SEMANTICS

In this section, we explain the solution to Problem 1. First, we exploit monotonicity to characterize the properties of the solutions. Next, we explain how to synthesize controls using a mixed-integer linear programming (MILP) solver.

Lemma 1: Consider runs \mathbf{x} and \mathbf{x}' and an STL formula φ . If for some t, t' , we have $\mathbf{x}'[t' : t' + h^\varphi] \preceq \mathbf{x}[t : t + h^\varphi]$, then $\mathbf{x}[t] \models \varphi$ implies $\mathbf{x}'[t'] \models \varphi$.

Proof: Since all predicates denote lower sets in the form of (7), we have $x'_i \preceq x_i \Rightarrow a_\pi^T x'_i \leq a_\pi^T x_i$, $\mathbf{x}[t] \models \pi \Rightarrow \mathbf{x}'[t'] \models \pi$. Thus, all predicates that were true by the valuations in \mathbf{x} remain true for \mathbf{x}' . The negation-free semantics in (2) implies that without falsifying any predicate, a formula cannot be falsified. Therefore, $\mathbf{x}[t] \models \varphi$ implies $\mathbf{x}'[t'] \models \varphi$. ■

The largest set of admissible initial conditions is defined as:

$$\mathcal{X}_0^{\max} := \left\{ x_0 \in \mathcal{X} \mid \exists \mu \text{ s.t. } \mathbf{x}(x_0, \mu, \mathbf{w}) \models \varphi \quad \forall \mathbf{w} \in \mathcal{W}^{\mathbf{w}} \right\}.$$

The set \mathcal{X}_0^{\max} is a union of polyhedra. Finding the half-space representation of all polyhedral sets in \mathcal{X}_0^{\max} may not be possible for high dimensions. Therefore, we find a half-space representation for a subset of \mathcal{X}_0^{\max} . The following result states how to check whether $x_0 \in \mathcal{X}_0^{\max}$.

Theorem 1: We have $x_0 \in \mathcal{X}_0^{\max}$ if and only if there exists an open-loop control sequence

$$u_0^{ol, x_0} u_1^{ol, x_0} \dots u_{h^\varphi-1}^{ol, x_0}$$

such that $\mathbf{x}^{ol, x_0}[0 : h^\varphi] \models \varphi$, where $\mathbf{x}^{ol, x_0}[0 : h^\varphi] = x_0^{ol, x_0} x_1^{ol, x_0} \dots x_{h^\varphi}^{ol, x_0}$, and $x_{k+1}^{ol, x_0} = f^*(x_k^{ol, x_0}, u_k^{ol, x_0})$, $k = 0, \dots, h^\varphi - 1$, $x_0^{ol, x_0} = x_0$.

Proof: (*Necessity*) Satisfaction of φ with $\mathbf{w} \in \mathcal{W}^{\mathbf{w}}$ requires at least one satisfying run for the maximal system, hence a corresponding control sequence exists. Denote it by $u_0^{ol, x_0} u_1^{ol, x_0} \dots u_{h^\varphi-1}^{ol, x_0}$. (*Sufficiency*) Consider any run generated by the original system $x_{k+1} = f(x_k, u_k^{ol, x_0}, w_k)$. We prove that $x_k \preceq x_k^{ol, x_0}$, $k = 0, 1, \dots, h^\varphi$ by induction over k . The base case $x_0 \preceq x_0^{ol, x_0}$ is trivial ($x_0 = x_0^{ol, x_0}$). The inductive step is verified from monotonicity: $x_{k+1} = f(x_k, u_k^{ol, x_0}, w_k) \preceq f^*(x_k^{ol, x_0}, u_k^{ol, x_0}) = x_{k+1}^{ol, x_0}$. Therefore, $\mathbf{x}[0 : h^\varphi] \preceq \mathbf{x}^{ol, x_0}[0 : h^\varphi]$, $\forall \mathbf{w}[0 : h^\varphi-1] \in \mathcal{W}^{h^\varphi}$. It follows from Lemma 1 that $\mathbf{x}[0 : h^\varphi] \models \varphi$, $\forall \mathbf{w}[0 : h^\varphi-1] \in \mathcal{W}^{h^\varphi}$. ■

Corollary 1: The set \mathcal{X}_0^{\max} is a lower set.

Proof: Consider any $x'_0 \in L(x_0)$, $x_0 \in \mathcal{X}_0^{\max}$. Let $x'_{k+1} = f(x'_k, u_k^{ol, x_0}, w_k)$, $k = 0, 1, \dots, h^\varphi - 1$. It follows from monotonicity that $x'_k \preceq x_k^{ol, x_0}$, $k = 0, 1, \dots, h^\varphi$, $\forall \mathbf{w}[0 : h^\varphi-1] \in \mathcal{W}^{h^\varphi}$. By the virtue of Lemma 1, $\mathbf{x}'[0 : h^\varphi] \preceq \mathbf{x}^{ol, x_0}[0 : h^\varphi]$. Therefore, we have $\forall x_0 \in \mathcal{X}_0^{\max}$, $x'_0 \in L(x_0) \Rightarrow x'_0 \in \mathcal{X}_0^{\max}$, which indicates that \mathcal{X}_0^{\max} is a lower set. ■

Corollary 2: If $x_0 \in \mathcal{X}_0^{\max}$ and μ^{ol} is the following open-loop control policy

$$\mu_t^{ol}(x_0) = u_t^{ol, x_0}, t = 0, 1, \dots, h^\varphi - 1$$

then $\mathbf{x}(x'_0, \mu, \mathbf{w})[0 : h^\varphi] \models \varphi$, $\forall \mathbf{w} \in \mathcal{W}^{h^\varphi}$, $\forall x'_0 \in L(x_0)$.

Proof: Follows from the proof of Corollary 1. ■

Now that we have established the properties of the solutions to Problem 1, we explain how to compute the admissible initial conditions and their corresponding open-loop control sequences. The approach is based on formulating the conditions in Theorem 1 as a set of constraints that can be incorporated into a feasibility solver. We convert all the constraints into a set of mixed-integer linear constraints and use off-the-shelf MILP solvers to check for feasibility. Converting logical properties into mixed-integer constraints is a common procedure, which was employed for MLD systems in [7]. Karaman *et al.* [8] and Raman *et al.* [10] extended this technique to a framework for time-bounded model checking of temporal logic formulas. A variation of this method is explained here.

First, the STL formula is recursively translated into a set of mixed-integer constraints. For each predicate $\pi = (a_\pi^T x \leq b_\pi)$, as in (7), we define a binary variable $z_k^\pi \in \{0, 1\}$ such that 1 (respectively, 0) stands for true (respectively, false). The relation between z^π , robustness ρ , and x are encoded as

$$a_\pi^T x - M(1 - z^\pi) + \rho \leq b_\pi \quad (11a)$$

$$a_\pi^T x + Mz^\pi + \rho \geq b_\pi. \quad (11b)$$

The constant M is a sufficiently large number such that $M \geq \max\{a_\pi^T K, b_\pi\}$, where $K \in \mathbb{R}_+^n$ is the upper bound for the state values, $x_k \preceq K$, $k = 0, 1, \dots, h^\varphi$. In practice, M is chosen sufficiently large such that the constraint $x \preceq K$ is never active. Note that the largest value of ρ for which $z^\pi = 1$ is $b_\pi - a_\pi^T x$, which is equal to the robustness of π .

Now, we encode the truth table relations. For instance, we desire to capture $1 \wedge 0 = 0$ and $1 \vee 0 = 1$ using mixed-integer linear equations. Disjunction and conjunction connectives are encoded as the following constraints:

$$z = \bigwedge_{i=1}^{n_z} z_i \Rightarrow z \leq z_i, i = 1, \dots, n_z \quad (12a)$$

$$z = \bigvee_{i=1}^{n_z} z_i \Rightarrow z \leq \sum_{i=1}^{n_z} z_i \quad (12b)$$

where $z \in [0, 1]$ is declared as a continuous variables. However, it only can take binary values as evident from (12). Similarly, define $z_k^\varphi \in [0, 1]$ as the variable indicating whether $\mathbf{x}[k] \models \varphi$. An STL formula is recursively translated as:

$$\varphi = \bigwedge_{i=1}^{n_\varphi} \varphi_i \Rightarrow z_k^\varphi = \bigwedge_{i=1}^{n_\varphi} z_k^{\varphi_i}$$

$$\varphi = \bigvee_{i=1}^{n_\varphi} \varphi_i \Rightarrow z_k^\varphi = \bigvee_{i=1}^{n_\varphi} z_k^{\varphi_i}$$

$$\varphi = \mathbf{G}_I \psi \Rightarrow z_k^\varphi = \bigwedge_{k' \in I} z_{k'}^\psi$$

$$\varphi = \mathbf{F}_I \psi \Rightarrow z_k^\varphi = \bigvee_{k' \in I} z_{k'}^\psi$$

$$\varphi = \psi_1 \mathbf{U}_I \psi_2 \Rightarrow z_k^\varphi = \bigvee_{k' \in I} \left(z_{k'}^{\psi_2} \wedge \bigwedge_{k'' \in [k, k']} z_{k''}^{\psi_1} \right). \quad (13)$$

Finally, we add the following constraints:

$$z_0^\varphi = 1, \rho \geq 0. \quad (14)$$

Proposition 1: The set of constraints in (11)–(14) has the following properties.

- i) We have $\mathbf{x}[0] \models \varphi$ if the set of constraints is feasible.
- ii) We have $\mathbf{x}[0] \not\models \varphi$ if the set of constraints is infeasible.
- iii) The largest ρ such that the set of constraints, while “ $\rho \geq 0$ ” is removed from (14), is feasible is equal to $\rho(\mathbf{x}, \varphi, 0)$.

Proof: i) We provide the proof for (12), as the case for more complex STL formulas are followed in a recursive manner from (13). If $z = 1$, we have from (12a) that $z_i = 1, i = 1, \dots, n_z$, which correctly encodes conjunctions. Similarly, $z = 1$ in (12b) indicates that not all $z_i, i = 0, \dots, n_z$ can be zero, or, $\exists i \in \{1, \dots, n_z\}$ such that $z_i = 1$, which correctly encodes disjunctions. ii) Infeasibility can be recursively traced back into (12). For both (12a) and (12b), if $z = 1$ is infeasible, it indicates that $z_i = 0, i = 1, \dots, n_z$. iii) We also prove this statement for (12) as it is the base of recursion for general STL formulas. Let $z_i = (a_{\pi_i}^T x + \rho \leq b_{\pi_i}), i = 1, \dots, n_z$. Consider (12a) and the following optimization problem:

$$\begin{aligned} \rho^{\max} &= \operatorname{argmax} \quad \rho \\ \text{s.t.} \quad &a_{\pi_i}^T x + \rho \leq b_{\pi_i}, i = 1, \dots, n_z \end{aligned}$$

where its solution is $\min_{i=1, \dots, n_z} (b_{\pi_i} - a_{\pi_i}^T x)$, which is identical to the quantitative semantics for conjunction [see (4)]. Similarly, consider (12b) and the following optimization problem:

$$\begin{aligned} \rho^{\max} &= \operatorname{argmax} \quad \rho \\ \text{s.t.} \quad &\exists i \in \{1, \dots, n_z\}, a_{\pi_i}^T x + \rho \leq b_{\pi_i} \end{aligned}$$

where the solution is $\max_{i=1, \dots, n_z} (b_{\pi_i} - a_{\pi_i}^T x)$, which is identical to the quantitative semantics for disjunction. \square

Our integer formulation for Boolean connectives slightly differs from the formulation in [8] and [10], where lower bound constraints for the z s are required. For example, for translating $z = \bigwedge_{i=1}^{n_z} z_i$, it is required to add $z \geq \sum_{i=1}^{n_z} z_i - n_z + 1$ to impose a lower bound for z . However, these additional constraints become necessary only when the negation operator is present in the STL formula. Hence, they are removed in our formulation. This reduces the constraint redundancy and degeneracy of the problem. By doing so, we observed computation speed gains (up to reducing the computation time by 50%) in our case studies. Moreover, we encode quantitative semantics in a different way than [10], where a separate STL robustness-based encoding is developed, which introduces additional integers. Due to property iii) in Proposition 1, our encoding does not require additional integers to capture robustness, hence it is computationally more efficient.

Definition 6: System (5) is in MLD form [7] if written as

$$x_{t+1} = Ax_t + B_u u_t + B_w w_t + D_\delta \delta_t + D_r r_t \quad (15a)$$

$$E_\delta \delta_t + E_r r_t \leq E_x x_t + E_u u_t + E_w w_t + e \quad (15b)$$

where $\delta_t \in \{0, 1\}^{n_\delta}$ and $r_t \in \mathbb{R}^{n_r}$ are auxiliary variables and $A, B_u, B_w, D_\delta, D_r, E_\delta, E_r, E_x, E_u, E_w, e$ are appropriately defined constant matrices such that (15) is well posed in the sense that given x_t, u_t, w_t , the feasible set for x_{t+1} is a single point equal to $f(x_t, u_t, w_t)$. Introducing auxiliary variables and enforcing (15b) can capture nonlinear f [7].

The system equations are brought into mixed-integer linear constraints by transforming system (5) into its MLD form. As mentioned earlier, any piecewise affine system can be transformed into an MLD. In the case studies of this paper, the construction of (15) from a piecewise affine (5) is not explained as the procedure is well documented in [40].

Finally, the set of constraints in Theorem 1 can be cast as

$$\begin{cases} x_0^{ol, x_0} = x_0, & \text{Initial condition} \\ x_{k+1}^{ol, x_0} = f^*(x_k^{ol, x_0}, u_k^{ol, x_0}), & \text{System constraints} \\ z_k^\pi = (a_{\pi_i}^T x_k^{ol, x_0} \leq b_{\pi_i}), & \text{Predicates} \\ z_0^\varphi = 1, \rho \geq 0, & \text{STL satisfaction.} \end{cases} \quad (16)$$

Checking the satisfaction of the set of constraints in (16) can be formulated as a MILP feasibility problem, which is handled using powerful off-the-shelf solvers. For a fixed initial condition x_0 , the feasibility of the MILP indicates whether $x_0 \in \mathcal{X}_o^{\max}$. An explicit representation of \mathcal{X}_o^{\max} requires variable elimination from (16), which is computationally intractable for a large MILP. Alternatively, we can set x_0 as a free variable while maximizing a cost function (e.g., norm of x_0) such that a large $L(x_0)$ is obtained. Another natural candidate is maximizing $\rho(x^{ol, x_0}, \varphi, 0)$. It is worth to note that by finding a set of distinct initial conditions and taking the union of all $L(x_0)$, we are able to find a representation for an underapproximation of \mathcal{X}_o^{\max} .

MILPs are NP-complete. The complexity of solving (16) grows exponentially with respect to the number of binary variables and polynomially with respect to the number of continuous variables. The number of binary variables in our framework is $\mathcal{O}(h^\varphi(n_\pi + m_b + q_b + n_\delta)) - n_\pi$ is the number of predicates—and the number of continuous variables is $\mathcal{O}(h^\varphi(n + m_r + q_r + n_r))$. In other words, the exponential growth builds upon the intricacy of the specification and the number of modes demonstrated by the hybrid nature of the system. However, the complexity is polynomial with respect to the dimension of the state.

Example 2: Consider the following switched system:

$$x^+ = e^{A_u \tau} x + A_u^{-1}(I - e^{-A_u \tau})w$$

where $x = (x_{[1]}, x_{[2]})^T \in \mathbb{R}_+^2$, $u \in \mathcal{U}$ is the control input (switch), $\mathcal{U} = \{1, 2\}$, and

$$A_1 = \begin{pmatrix} 1 & 1 \\ 1 & -5 \end{pmatrix}, A_2 = \begin{pmatrix} -8 & 1 \\ 1 & 2 \end{pmatrix}.$$

The (additive) disturbance w is bounded to $L(w^*)$, where $w^* = (1.5, 1)^T$ and $\tau = 0.1$. This system is the discrete-time version of $\dot{x} = A_u x + w$ with sample time τ . Both matrices are Metzler (all off-diagonal terms are nonnegative, hence all the elements of its exponential are positive) and non-Hurwitz, hence constant input results in unbounded trajectories. The system is desired to satisfy the following STL formula:

$$\varphi = \bigvee_{T=0}^{10} (\mathbf{F}_{[0, T]} p_1 \wedge \mathbf{F}_{\{T\}} p_2)$$

where $p_1 = ((x_{[1]} \leq 1) \wedge (x_{[2]} \leq 5))$ and $p_2 = ((x_{[1]} \leq 5) \wedge (x_{[2]} \leq 1))$. In plain English, φ states that “within ten time units, the trajectory visits the box characterized by p_1 first and then the box corresponding to p_2 ” (see Fig. 1). We transformed this system into its MLD form (15). We formulated the constraints

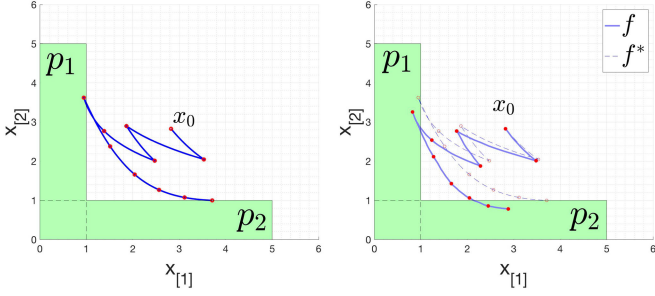


Fig. 1. Example 2: [Left] The trajectory of the maximal system f^* that satisfies the specification. [Right] For the same controls, the trajectory of the original system f with w drawn from a uniform distribution over $L(w^*)$.

in (16) as a MILP and set the cost function to maximize $\|x_0\|_\infty$ and used the Gurobi³ MILP solver. The solution was obtained in less than 0.05 s on a 3-GHz Dual Core MacBook Pro. We obtained $x_0 = (2.82 \ 2.82)^T$ and the following open-loop control sequence: 1 2 1 2 2 1 1 1 1. By applying this control sequence, we sampled a trajectory of the original system f with values of w drawn from a uniform distribution over $L(w^*)$. Both the trajectories of f and f^* satisfy the specification. The results are shown in Fig. 1.

V. INFINITE HORIZON SEMANTICS

In this section, we provide a solution to Problem 2. We show that the infinite-time property in (8) can be guaranteed using repetitive control sequences. First, we consider global specifications and extend the results from our previous work [21] in Section V-A. Next, we show how to find controls for bounded-global STL formulas (see Problem 2) in Section V-B. Solution completeness is discussed in Section V-C.

A. Global Formulas: S-Sequences and Inductive Invariance

Consider the global specification $\mathbf{G}_{[0,\infty]}\varphi$, where φ is a bounded formula. We introduce some additional notation.

Definition 7: Given a bounded STL formula φ over predicates in the form (7), the *language realization set* (LRS) [41] is

$$\mathcal{L}^\varphi := \{x_0 x_1 \dots x_{h^\varphi} \in \mathcal{X}^{h^\varphi} \mid x_0 x_1 \dots x_{h^\varphi} \models \varphi\}. \quad (17)$$

Proposition 2: The set \mathcal{L}^φ is a lower set.

Proof: For all $x_0 x_1 \dots x_{h^\varphi} \in \mathcal{L}^\varphi$ and $x'_0 x'_1 \dots x'_{h^\varphi} \preceq x_0 x_1 \dots x_{h^\varphi}$, it follows from Lemma 1 that $x'_0 x'_1 \dots x'_{h^\varphi} \models \varphi$. Thus, $x'_0 x'_1 \dots x'_{h^\varphi} \in \text{LRS}(\varphi)$, hence $\text{LRS}(\varphi)$ is a lower set. ■

It follows from the semantics of global operator in (2) that $\mathbf{x} \models \mathbf{G}_{[0,\infty]}\varphi$ is equivalent to $\mathbf{x}[t : t + h^\varphi] \in \mathcal{L}^\varphi, \forall t \in \mathbb{N}$.

Definition 8: A set $\Omega_{\mathcal{L}^\varphi} \subseteq \mathcal{L}^\varphi$ is a *robust control invariant* (RCI) set if:

$$\begin{aligned} \forall x_0 x_1 \dots x_{h^\varphi} \in \Omega_{\mathcal{L}^\varphi}, \exists u \in \mathcal{U}, \text{ s.t.} \\ x_1 x_2 \dots x_{h^\varphi} f(x_{h^\varphi}, u, w) \in \Omega_{\mathcal{L}^\varphi} \quad \forall w \in \mathcal{W}. \end{aligned} \quad (18)$$

Satisfaction of $\mathbf{G}_{[0,\infty]}\varphi$ is accomplished by finding a RCI set in \mathcal{L}^φ . Note that unlike traditional definitions of RCI sets

(e.g., [42]), where the set is defined in the state-space \mathcal{X} , our RCI set is defined in an augmented form of the state-space \mathcal{X}^{h^φ} . The LRS can also be interpreted as the “safe” set in the $(h^\varphi + 1)$ -length trajectory space. The *maximal* RCI set inside \mathcal{L}^φ provides a complete solution to the set-invariance problem. The computation of maximal RCI set requires implementing an iterative fixed-point algorithm, which is computationally intensive for MLD systems and nonconvex sets (see [43] and [44] for discussion). We use monotonicity to provide an alternative approach. The following result is a more general version of the one in [21].

Theorem 2: Given a bounded formula φ , if there exists $\mathbf{x}^s[0 : h^\varphi] \in \mathcal{L}^\varphi$, and a sequence of controls: u_0^s, \dots, u_{T-1}^s , where T is a positive integer determining the length of the sequence, such that:

- 1) $\mathbf{x}^s[k : k + h^\varphi] \in \mathcal{L}^\varphi, k = 0, 1, \dots, T$, where $x_{h^\varphi+k+1}^s = f^*(x_{h^\varphi+k}^s, u_k^s)$,
- 2) $\mathbf{x}^s[T : T + h^\varphi] \preceq \mathbf{x}^s[0 : h^\varphi]$,

then the following set is a RCI set in \mathcal{L}^φ :

$$\Omega_{\mathcal{L}^\varphi} := \bigcup_{k=0}^{T-1} L(\mathbf{x}^s[k : k + h^\varphi]). \quad (19)$$

Proof: For any $x'_0 x'_1 \dots x'_{h^\varphi} \in \Omega_{\mathcal{L}^\varphi}$, there exists $i \in \{0, 1, \dots, T-1\}$ such that $x'_0 x'_1 \dots x'_{h^\varphi} \in L(\mathbf{x}^s[i : i + h^\varphi])$. On one hand, we have $x_{i+1}^s \dots x_{i+h^\varphi}^s f^*(x_{i+h^\varphi}^s, u_i^s) \in \Omega_{\mathcal{L}^\varphi}$. On the other hand, we have $x'_1 \preceq x_{i+1}^s, \dots, x'_{h^\varphi} \preceq x_{i+h^\varphi}^s$. By applying u_i^s , monotonicity implies

$$\begin{aligned} f(x'_{h^\varphi}, u_i^s, w) \preceq f^*(x_{i+h^\varphi}^s, u_i^s) = x_{i+1+h^\varphi}^s \quad \forall w \in \mathcal{W} \\ \Rightarrow x'_1 x'_2 \dots x'_{h^\varphi} f(x'_{h^\varphi}, u_i^s, w) \in \\ L(x_{i+1}^s x_{i+2}^s \dots x_{i+1+h^\varphi}^s) \quad \forall w \in \mathcal{W}. \end{aligned}$$

The proof is complete from the fact that $x_{i+1}^s \dots x_{i+1+h^\varphi}^s \in \Omega_{\mathcal{L}^\varphi}$ for all $i \in \{0, 1, \dots, T-1\}$. ■

Corollary 3: Let the conditions in Theorem 2 hold and $\mathbf{x}[t_0 : t_0 + h^\varphi] \in L(\mathbf{x}^s[0 : h^\varphi])$ for some $t_0 \in \mathbb{N}$. Consider the following control sequence starting from time $t_0 + h^\varphi$:

$$\mathbf{u}^s := (u_0^s u_1^s \dots u_{T-1}^s)^\omega \quad (20)$$

i.e., $u_t^s = u_{\text{rem}(t-t_0-h^\varphi, T)}^s, t \geq t_0 + h^\varphi$. Let $x_{k+1} = f^*(x_k, u_k), k = t_0 + h^\varphi, \dots$. Then, we have $\mathbf{x}[t : t + h^\varphi] \in \Omega_{\mathcal{L}^\varphi}, \forall t \geq t_0$.

Proof: We prove by induction that $\mathbf{x}[t : t + h^\varphi] \in L(\mathbf{x}^s[\text{rem}(t-t_0, T) : \text{rem}(t-t_0, T) + h^\varphi]), \forall t \geq t_0$. The base case for $t=t_0$ is true. In order to prove the inductive step $\mathbf{x}[t+1 : t+1 + h^\varphi] \in L(\mathbf{x}^s[\text{rem}(t+1-t_0, T) : \text{rem}(t+1-t_0, T) + h^\varphi])$, we need to prove that $x_{t+k+1} \preceq x_{\text{rem}(t+1-t_0, T)+k}^s, k = 0, \dots, h^\varphi$, for which we need to only prove the case for $k = h^\varphi$ as previous inequalities are already assumed by induction. We show $x_{t+h^\varphi+1} \preceq x_{\text{rem}(t+1-t_0, T)+h^\varphi}^s$ through monotonicity and the induction assumption that $x_{t+h^\varphi} \preceq x_{\text{rem}(t-t_0, T)+h^\varphi}^s$:

$$\begin{aligned} x_{t+h^\varphi+1} &= f^*(x_{t+h^\varphi}, u_{\text{rem}(t-t_0, T)}^s) \\ &\preceq f^*(x_{\text{rem}(t-t_0, T)+h^\varphi}^s, u_{\text{rem}(t-t_0, T)}^s) \\ &= x_{\text{rem}(t-t_0, T)+1+h^\varphi}^s \preceq x_{\text{rem}(t+1-t_0, T)+h^\varphi}^s. \end{aligned}$$

³www.gurobi.com

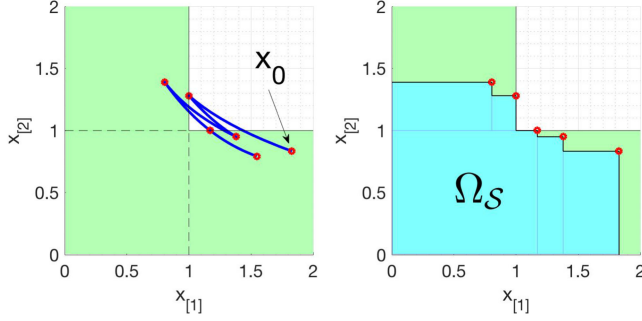


Fig. 2. Example 3: [Left] The trajectory satisfying the conditions of s-sequences. [Right] The corresponding RCI set inside \mathcal{S} .

Note that $x_{T+h^\varphi}^s \preceq x_{h^\varphi}^s$. The “ \preceq ” in the last line can be replaced by “ $=$ ” when $\text{rem}(t - t_0, T) + 1 \neq T$. ■

We refer to the repetitive sequence of controls in (20) as an *s-sequence*. An s-sequence is an invariance inducing open-loop control policy. Once the latest $h^\varphi + 1$ -length of system state are brought into $\Omega_{\mathcal{L}^\varphi}$, an s-sequence keeps the $h^\varphi + 1$ -length trajectory of the system in $\Omega_{\mathcal{L}^\varphi}$ for all subsequent times.

The computation of an s-sequence requires solving an MILP for $\mathbf{x}^s[h^\varphi : T] \models \mathbf{G}_{[0, T]} \varphi$ (an instance of Problem 1) with an additional set of constraints in $\mathbf{x}^s[0 : h^\varphi] \models \varphi$ (again, an instance of Problem 1, but without the dynamical constraints. In other words, $\mathbf{x}^s[0 : h^\varphi]$ does not need to be a trajectory of the maximal system), and $\mathbf{x}^s[T : T + h^\varphi] \preceq \mathbf{x}^s[0 : h^\varphi]$ (linear constraints). We are usually interested in the shortest s-sequence since its computation requires the smallest MILP. Algorithmically, we start from $T = 1$ and implement $T \leftarrow T + 1$ until the MILP formulating the conditions in Theorem 2 becomes feasible and an s-sequence is found. As it will be implied from results in Section V-C, existence of an s-sequence is almost necessary for existence of a RCI set.

Example 3: Consider the system in Example 2. We wish to keep the trajectory in the set characterized by $p_1 \vee p_2$, i.e., $\mathcal{S} = L((1, 5)^T) \cup L((5, 1)^T)$. Note that this set is nonconvex. We set the cost function to maximize $\|x_0\|_1$. The shortest s-sequence has $T = 5$ and is $(2 \ 1 \ 2 \ 1 \ 1)^\omega$. The resulting trajectory satisfying the definition of s-sequence is shown in Fig. 2 [Left]. The corresponding RCI set is shown in Fig. 2 [Right] (cyan region), which is characterized by the $x_0^s, x_1^s, \dots, x_4^s$ (red dots) that lie inside \mathcal{S} (green region). Note that the $[0, 2] \times [0, 2]$ portion of the coordinates in Fig. 1 is shown here for a clearer representation of the details.

B. Bounded-Global Specifications: ϕ -Sequences

Now, we consider general bounded-global formulas—as in Problem 2—and generalize the paradigm used for s-sequences. We provide the key result of this section.

Theorem 3: Given a bounded-global STL formula $\phi = \varphi_b \wedge \mathbf{G}_{[\Delta, \infty]} \varphi_g$, an initial condition x_0 , a control sequence $u_0^\phi \dots u_{\Delta+T+h^\varphi-1}^\phi$, where T is a positive integer, and a non-negative integer $T_0 < T$, let the following conditions hold:

- 1) $\mathbf{x}^\phi[0 : \Delta + T + h^\varphi] \models \varphi_b \wedge \mathbf{G}_{[\Delta, \Delta+T]} \varphi_g$,
- 2) $\mathbf{x}^\phi[\Delta + T : \Delta + T + h^\varphi] \preceq \mathbf{x}^\phi[\Delta + T_0 : \Delta + T_0 + h^\varphi]$,

where $x_{k+1}^\phi = f^*(x_k^\phi, u_k^\phi)$, $k \in [0, \Delta + T + h^\varphi - 1]$, $x_0^\phi = x_0$. Let μ^{ol} be the open-loop control policy corresponding to the following control sequence:

$$\mathbf{u}^\phi := u_0^\phi \dots u_{\Delta+T+h^\varphi-1}^\phi \left(u_{\Delta+T_0+h^\varphi}^\phi \dots u_{\Delta+T+h^\varphi-1}^\phi \right)^\omega \quad (21)$$

Then, $\mathbf{x}(x_0^\phi, \mu^{ol}, \mathbf{w}) \models \phi, \forall \mathbf{w} \in \mathcal{W}^\omega, \forall x_0^\phi \in L(x_0)$. Moreover, the following set is a RCI set in \mathcal{L}^φ :

$$\Omega_{\mathcal{L}^\varphi} := \bigcup_{i=0}^{T-T_0-1} L(\mathbf{x}^\phi[\Delta + T_0 + i : \Delta + T_0 + h^\varphi + i]). \quad (22)$$

Proof: We need to prove that $\mathbf{x}(x_0^\phi, \mu^{ol}, \mathbf{w}^*) \models \phi$, where $\mathbf{w}^* = (w^*)^\omega$. The fact that $\mathbf{x}(x_0^\phi, \mu^{ol}, \mathbf{w})[0] \models \phi, \forall x_0^\phi \in L(x_0), \forall \mathbf{w} \in \mathcal{W}^\omega$ follows from monotonicity and Lemma 1. The fact that $\Omega_{\mathcal{L}^\varphi}$ is a RCI set follows from Theorem 2 as (19) is obtained from replacing $\Delta = T_0 = 0$ in (22). It follows that $(u_{\Delta+T_0+h^\varphi}^\phi \dots u_{\Delta+T+h^\varphi-1}^\phi)^\omega$ is an s-sequence. For all $t \geq \Delta + T + h^\varphi$, let

$$x_{t+1}^\phi = f^* \left(x_t^\phi, u_{\Delta+T_0+h^\varphi+\text{rem}(t-\Delta-T_0-h^\varphi, T-T_0)}^\phi \right). \quad (23)$$

Using Corollary 3, we have $\mathbf{x}^\phi[k + \Delta + T_0 : k + \Delta + T_0 + h^\varphi] \in \mathcal{L}^\varphi, \forall k \in \mathbb{N}$, and the proof is complete. ■

We refer to the sequence of controls in (21) as a ϕ -sequence. The computation of a ϕ -sequence requires solving an MILP for $\mathbf{x}^\phi[0 : \Delta + T + h^\varphi] \models \varphi_b \wedge \mathbf{G}_{[\Delta, \Delta+T]} \varphi_g$ (an instance of Problem 1) with an additional set of constraints in $\mathbf{x}^\phi[\Delta + T : \Delta + T + h^\varphi] \preceq \mathbf{x}^\phi[\Delta + T_0 : \Delta + T_0 + h^\varphi]$ (linear constraints). Thus, similar to s-sequences, the computation of a ϕ -sequence is based on feasibility checking of a MILP. We have two parameters T and $T_0 < T$ to search over. We start from $T = 1$ and implement $T \leftarrow T + 1$, while checking for all $T_0 < T$, until the corresponding MILP gets feasible. In Section V-C, we discuss the necessity of existence of a feasible solution for some T_0, T .

Another interpretation of a ϕ -sequence is a sequence that consists of an initialization segment of length $\Delta + h^\varphi$ to bring the latest h^φ states of the system into $\Omega_{\mathcal{L}^\varphi} \subseteq \mathcal{L}^\varphi$ and a repetitive segment of length T to stay in $\Omega_{\mathcal{L}^\varphi}$. The repetitive segment is an s-sequence. Since control inputs eventually becoming periodic, the long-term behavior is expected to demonstrate periodicity, which leads to the following result based on Theorem 3.

Corollary 4: The ω -limit set of the run given by (23) is nonempty and corresponds to the following periodical orbit:

$$\left(x_0^{\phi, \infty} x_1^{\phi, \infty} \dots x_{T-T_0-1}^{\phi, \infty} \right)^\omega \quad (24)$$

where $x_k^{\phi, \infty} := \lim_{c \rightarrow \infty} x_{k+\Delta+T_0+c(T-T_0)}^\phi, k = 0, \dots, T - T_0 - 1$.

Proof: We show that $x_t^\phi \preceq x_{t+T-T_0}^\phi, \forall t \geq \Delta + T_0$. Similar to the proof of Corollary 3, we use induction. The base case for $t = \Delta + T_0$ is already in the second condition in Theorem 3. The inductive step is proven as follows:

$$x_{t+1+T-T_0}^\phi = f^*(x_{t+T-T_0}^\phi, u_{T-T_0+t}) \preceq f^*(x_t^\phi, u_t) = x_{t+1}^\phi$$

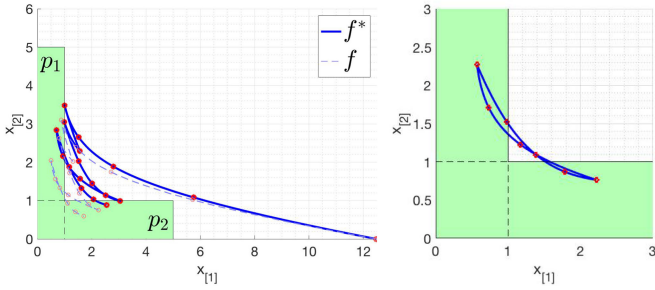


Fig. 3. Example 4: [Left] The first 19 points of the trajectory of the maximal system f^* that satisfy the conditions in Theorem 3. A sample trajectory of f is also shown. [Right] The ω -limit set (red dots) of f^* is a seven-periodic orbit.

where from (21), we have

$$u_{t+T-T_0} = u_t = u_{t+\Delta+T_0+h^{\varphi_g} + \text{rem}(t-\Delta-T_0-h^{\varphi_g}, T-T_0)}.$$

Thus, each component of the sequence $x_{\Delta+T_0+k}^{\phi}, x_{\Delta+T+k}^{\phi}, x_{\Delta+2T-T_0+k}^{\phi}, \dots, k = 0, \dots, T-T_0$ is monotonically decreasing. *Monotone convergence theorem* [45] explains that a lower-bounded monotonically decreasing sequence converges (in this case, all values are lower-bounded by zero). Thus, $\lim_{c \rightarrow \infty} x_{\Delta+T_0+k+c(T-T_0)}^{\phi}, k = 0, \dots, T-T_0$, exists and the proof is complete. ■

Example 4: Consider the system in Example 2. We wish to satisfy

$$\phi = \mathbf{F}_{[0,5]} p_1 \wedge \mathbf{G}_{[5,\infty)} (\mathbf{F}_{[0,6]} p_1 \wedge \mathbf{F}_{[0,6]} p_2).$$

The specification is in the form in (9) with $h^{\varphi_b} = \Delta = 5$, $h^{\varphi_g} = 6$. This specification requires that p_1 is visited at least once until $t = 5$ and, afterwards, p_1 and p_2 are persistently visited while the maximum time between two subsequent visits is not greater than 6. We find a ϕ -sequence solving a MILP for $T = 7, T_0 = 0$, while maximizing $\|x_0\|_1$. The obtained ϕ -sequence is $\mathbf{u}^{\phi} = 2 \ 2 \ 2 \ 2 \ 1 \ 2 \ 1 \ 1 \ 1 \ 1 \ 2 \ (2 \ 2 \ 1 \ 1 \ 1 \ 1 \ 2)^{\omega}$ for $x_0 = (12.4, 0)^T$. The first $\Delta + T + h^{\varphi_g} + 1 = 5 + 7 + 6 + 1 = 19$ time points of the trajectory of the maximal system f^* satisfying the conditions in Theorem 3 are shown in Fig. 4 [Left]. A sample trajectory of f with values of \mathbf{w} chosen uniformly from \mathcal{W} is also shown. Both trajectories satisfy ϕ . The limit-set of f^* , which is a seven-periodical orbit, is shown in Fig. 4 [Right].

C. Necessity of Open-Loop Strategies

We showed that if there exists an initial condition and a finite length control sequence such that the statements in Theorem 2 hold, an open-loop control sequence is sufficient for satisfying of a bounded-global formula, as was formulated in Problem 2. In this section, we address the necessity conditions. We show that the existence of open-loop control strategies for satisfying a bounded-global specifications is *almost necessary* in the sense that if a ϕ -sequence is not found using Theorem 3 for large values of T , then it is almost certain that no correct control policy (including feedback policies) exists, or, if exists any, it is *fragile* in the sense that a slight increase in the effect of the disturbances makes the policy invalid. We characterize the necessity conditions based on hypothetical perturbations in the disturbance set.

Theorem 4: Suppose system (5) is strongly monotone with respect to the maximal disturbance in the sense that for all $\epsilon > 0$, there exists a perturbed disturbance set \mathcal{W}_p with maximal disturbance w_p^* such that

$$\forall x \in \mathcal{X} \quad \forall u \in \mathcal{U}, f(x, u, w^*) + 1_n \epsilon \preceq f(x, u, w_p^*). \quad (25)$$

Consider the bounded-global formula $\phi = \varphi_b \wedge \mathbf{G}_{[\Delta, \infty)} \varphi$. Given $\epsilon > 0$, the disturbance set is altered to \mathcal{W}_p such that (25) holds. If there exists a control policy μ and an initial condition x_0 such that $\mathbf{x}(x_0, \mu, \mathbf{w}_p) \models \phi, \forall \mathbf{w}_p \in \mathcal{W}_p^{\omega}$, then there exists at least one open-loop control policy μ^{ol} in the form of a ϕ -sequence in (21) for the original system such that

$$T \leq A/\epsilon^{n(h^{\varphi_g} + 1)} \quad (26)$$

where A is a constant depending on \mathcal{L}^{φ_g} .

Proof: Given a bounded set $\mathcal{C} \subset \mathbb{R}^{n(h^{\varphi_g} + 1)}$, we define the diameter $d(\mathcal{C}) := \inf\{d|s_1 \preceq s_2 + d1_{n(h^{\varphi_g} + 1)}, \forall s_1, s_2 \in \mathcal{C}\}$ (e.g., the diameter of an axis-aligned hyper-box is equal to the length of its largest side). Consider a partition of \mathcal{L}^{φ_g} by a finite number of cells, where the diameter of each cell is less than ϵ . The maximum number of cells required for such a partition is $A/\epsilon^{n(h^{\varphi_g} + 1)}$, where A is a constant dependent on the shape and volume of \mathcal{L}^{φ_g} . A conservative upper bound on A can be given as follows. Define $a^* \in \mathbb{R}_+$ as

$$\text{argmin}_{a \in \mathbb{N}} \left\{ \mathbf{x}[0 : h^{\varphi_g}] \preceq a \mathbf{1}_n[0 : h^{\varphi_g}] \quad \forall \mathbf{x}[0 : h^{\varphi_g}] \in \mathcal{L}^{\varphi_g} \right\}.$$

Since \mathcal{L}^{φ_g} is bounded and closed, a^* exists. We have $\mathcal{L}^{\varphi_g} \subseteq L(a^* \mathbf{1}_n(h^{\varphi_g} + 1))$. Let A be $a^{*n(h^{\varphi_g} + 1)}$ —the volume of $L(a^* \mathbf{1}_n(h^{\varphi_g} + 1))$, which is a hyper-box. Partition $L(a^* \mathbf{1}_n(h^{\varphi_g} + 1))$ into $N := A/\epsilon^{n(h^{\varphi_g} + 1)}$ number of equally sized cubic cells with side length of ϵ . Such a partition also partitions \mathcal{L}^{φ_g} to at most N number of cells where the diameter of each cell is not greater than ϵ .

Since there exists μ such that $\mathbf{x}(x_0, \mu, \mathbf{w}_p) \models \phi, \forall \mathbf{w}_p \in \mathcal{W}_p^{\omega}$, then there exist at least one run satisfying ϕ for system $x_{k+1} = f(x_k, u_k, w_p^*)$. Let $x_0, \dots, x_{\Delta+h^{\varphi_g}+N}$ be the first $\Delta + h^{\varphi_g} + N + 1$ time points of such a run. We have $\mathbf{x}[k : k + h^{\varphi_g}] \in \mathcal{L}^{\varphi_g}, k = \Delta, \dots, \Delta + N$. Consider the sequence $\mathbf{x}[\Delta : \Delta + h^{\varphi_g}] \mathbf{x}[\Delta + 1 : \Delta + 1 + h^{\varphi_g}] \dots \mathbf{x}[\Delta + N : \Delta + N + h^{\varphi_g}]$. Consider a partition of \mathcal{L}^{φ_g} with cells such that for all cells the diameter is less than ϵ . By the virtue of *pigeonhole principle*, there exists a cell that contains at least two time points $\mathbf{x}[k_1 : k_1 + h^{\varphi_g}]$ and $\mathbf{x}[k_2 : k_2 + h^{\varphi_g}]$, $\Delta \leq k_1 \leq k_2 \leq \Delta + N$. From the assumption on the diameter of the cells, we have

$$\mathbf{x}[k_2 : k_2 + h^{\varphi_g}] \preceq \mathbf{x}[k_1 : k_1 + h^{\varphi_g}] + \epsilon \mathbf{1}_n[0 : h^{\varphi_g}]. \quad (27)$$

Now consider system $x'_{k+1} = f(x'_k, u_k, w^*)$ —the original maximal system—with $x'_{k_1+h^{\varphi_g}} = x_{k_1+h^{\varphi_g}}$. We prove that

$$x'_k + 1_n \epsilon \preceq x_k \quad \forall k > k_1 + h^{\varphi_g}. \quad (28)$$

We use induction. The base case for $k = k_1 + h^{\varphi_g} + 1$ is verified using (25):

$$\begin{aligned} x'_{k_1+1+h^{\varphi_g}} + 1_n \epsilon &= f(x'_{k_1+h^{\varphi_g}}, u_{k_1+h^{\varphi_g}}, w^*) + 1_n \epsilon \\ &\leq f(x'_{k_1+h^{\varphi_g}}, u_{k_1+h^{\varphi_g}}, w_p^*) \\ &= x_{k_1+1+h^{\varphi_g}}. \end{aligned}$$

The inductive step is verified using monotonicity and (25):

$$\begin{aligned} x'_{k+1+h^{\varphi_g}} + 1_n \epsilon &= f(x'_{k+h^{\varphi_g}}, u_{k+h^{\varphi_g}}, w^*) + 1_n \epsilon \\ &\leq f(x'_{k+h^{\varphi_g}}, u_{k+h^{\varphi_g}}, w_p^*) \\ &\leq f(x_{k+h^{\varphi_g}}, u_{k+h^{\varphi_g}}, w_p^*) = x_{k+1+h^{\varphi_g}}. \end{aligned}$$

It immediately follows from (28) that

$$\mathbf{x}'[k_2 : k_2 + h^{\varphi_g}] + \epsilon \mathbf{1}_n[0 : h^{\varphi_g}] \leq \mathbf{x}[k_2 : k_2 + h^{\varphi_g}]. \quad (29)$$

Since the left-hand side of (27) is the right-hand side of (29), we have

$$\mathbf{x}'[k_2 : k_2 + h^{\varphi_g}] \leq \mathbf{x}[k_1 : k_1 + h^{\varphi_g}]. \quad (30)$$

This is reminiscent of the conditions in Theorem 2. Now by defining $x'_k := x_k, k = k_1, \dots, k_1 + h^{\varphi_g} - 1$, we conclude that

$$\Omega'_{\mathcal{L}^{\varphi_g}} := \bigcup_{k=k_1}^{k_2-1} L(\mathbf{x}'[k : k + h^{\varphi_g}])$$

is a RCI set for system with adversarial disturbance set \mathcal{W} and $(u_{k_1} \dots u_{k_2+h^{\varphi_g}-1})^\omega$ is an s-sequence.

Now, once again, consider the original system $x'_{k+1} = f(x'_k, u_k, w^*)$ with $x'_0 = x_0$. Monotonicity implies $\mathbf{x}'[0 : k_1 + h^{\varphi_g}] \leq \mathbf{x}[0 : k_1 + h^{\varphi_g}]$. Thus, by applying $u_0, \dots, u_{k_1+h^{\varphi_g}-1}$ and using Lemma 1, we have $\mathbf{x}'[0 : k_1 + h^{\varphi_g}] \models \varphi_b \wedge \mathbf{G}_{[\Delta, k_1]} \varphi_g$. Corollary 3 implies $\mathbf{x}'[k_1 + h^{\varphi_g}] \models \mathbf{G}_{[0, \infty]} \varphi_g$, if $(u_{k_1}, \dots, u_{k_2-1})^\omega$ is applied starting from time k_1 . Finally, monotonicity and Lemma 1 immediately indicate that $\mathbf{x}(x''_0, \mu^{ol}, \mathbf{w}) \models \phi, \forall x''_0 \in L(x_0), \forall \mathbf{w} \in \mathcal{W}^\omega$, where μ^{ol} is the following open-loop control strategy producing the following control sequence:

$$u_0 \dots u_{k_1-1} (u_{k_1} u_{k_2+h^{\varphi_g}-1})^\omega$$

which is in the form of (21) with $k_1 = \Delta + T_0 + h^{\varphi_g}$ and $k_2 = \Delta + T$. Since $k_2 \leq \Delta + N$, we also have $T \leq N, N = A/\epsilon^{n(h^{\varphi_g}+1)}$, and the proof is complete. ■

Corollary 5: Suppose that for all $T \leq T^{\max}, T_0 < T$, there does not exist an initial condition and a control sequence such that the conditions in Theorem 3 hold. Then there does not exist any solution to Problem 2 given that the maximal disturbance is w_p^* such that (25) holds with $\epsilon > \sqrt[n(h^{\varphi_g}+1)]{T^{\max}}$.

The relation between the fragility in Theorem 4 and the length of the ϕ -sequence suggests that by performing the search for longer ϕ -sequences (which are computationally more difficult), the bound for fragility becomes smaller, implying that a correct control policy (if exists) is close to the limits (i.e., robustness score is close to zero, or the constraints are barely satisfied in the case with maximal disturbance). In practice, the bounds in Theorem 4 are very conservative and one may desire to find tighter bounds for specific applications.

Example 5: Consider Example 3. Suppose that there does not exist an s-sequence of length smaller than 144 with maximal disturbance w^* . The constant A (area in this 2-D case, see proof of Theorem 4) of region corresponding to $p_1 \vee p_2$ is 9. Therefore, \mathcal{S} can be partitioned into 144 equally sized square cells with side length 0.25. Note that we have $\epsilon^2 \geq 9/T$. Since the disturbances are additive, it follows that if $A_u^{-1}(I - e^{-A_u t})(w_p^* - w^*) > (0.25, 0.25)^T, u = 1, 2$, then there does not exist any control strategy μ and $x_0 \in \mathbb{R}_+^n$ such that $\mathbf{x}(x_0, \mu, \mathbf{w}_p) \models \mathbf{G}_{[0, \infty]}(x \in \mathcal{S}), \forall \mathbf{w}_p \in \mathcal{W}_p^\omega$.

VI. MODEL PREDICTIVE CONTROL

In this section, we provide a solution to Problem 3. We assume full knowledge of the history of state. As mentioned in Section III, the cost function J is assumed to be nondecreasing with respect to the state values, hence the system constraints are replaced with those of the maximal system. First, we explain the MPC setup for global STL formulas. Next, we prove that the proposed framework is guaranteed to generate runs that satisfy the global STL specification (8).

Let $t \geq h^\varphi - 1$. The case of $t < h^\varphi - 1$ is explained later. Given planning horizon H , the states that are predictable at time t using controls in u_t^H are $x_{1|t}, x_{2|t}, \dots, x_{H|t}$. Given predictions $x_{1|t}, x_{2|t}, \dots, x_{H|t}$, we need to enforce $\mathbf{x}[t - h^\varphi + 1, t + H] \models \mathbf{G}_{[0, H-1]} \varphi$ at time t . Notice that

$$\mathbf{x}_{|t}[t - h^\varphi + 1, t + H] := x_{t-h^\varphi+1} \dots x_t x_{1|t} \dots x_{H|t} \quad (31)$$

i.e., the first h^φ time points are actual values, the rest are predictions. Also, note that the values in $\mathbf{x}[\tau : \tau + h^\varphi]$ are independent of the values in x_t^H for $\tau \leq t - h^\varphi$ and are not fully available for $\tau > t + H - h^\varphi$. Thus, $[t - h^\varphi + 1, t + H - h^\varphi]$ is the time window for imposing constraints at time t [12].

The MPC optimization problem is initially written as (we do not use it for control synthesis as explained shortly):

$$\begin{aligned} \text{minimize} \quad & J(x_t^H, u_t^H) \\ \text{s.t.} \quad & x_{k+1|t} = f^*(x_{k|t}, u_{k|t}), k = 0, \dots, H-1 \\ & \mathbf{x}_{|t}[t - h^\varphi + 1, t + H] \models \mathbf{G}_{[0, H-1]} \varphi. \end{aligned} \quad (32)$$

The set of constraints in (32) requires the knowledge of $x_{t-h^\varphi+1} x_{t-h^\varphi+2} \dots x_t$. Thus, the proposed control policy requires a finite memory for the history of last h^φ states. As it will be shown in Proposition 3, persistent feasibility of the constraints in (32) leads to fulfilling $\mathbf{G}_{[0, \infty]} \varphi$. However, persistent feasibility of the MPC setup in (32) is not guaranteed. We address this issue for the remainder of this section.

Definition 9: An MPC strategy is *recursively feasible* if, for all $t \in \mathbb{N}$, the control at time t is selected such that the MPC optimization problem at $t + 1$ becomes feasible.

Our goal is to modify (32) such that it becomes recursively feasible. It is known that adding a (the maximal) RCI set acting as a terminal constraint is sufficient (and necessary) to guarantee recursive feasibility [46]. We add the terminal constraint $\mathbf{x}[t + H - h^\varphi : t + H] \in \Omega_{\mathcal{L}^\varphi}$ to (32) to obtain

$$\begin{aligned} u_t^{H, \text{opt}} &= \underset{u_t^H \in \mathcal{U}^H}{\text{argmin}} J(x_t^H, u_t^H) \\ \text{s.t.} \quad & x_{k+1|t} = f^*(x_{k|t}, u_{k|t}), k = 0, \dots, H-1 \\ & \mathbf{x}_{|t}[t - h^\varphi + 1, t + H] \models \mathbf{G}_{[0, H-1]} \varphi \\ & \mathbf{x}_{|t}[t + H - h^\varphi : t + H] \in \Omega_{\mathcal{L}^\varphi}. \end{aligned} \quad (33)$$

Proposition 3: Let $\mu_t(x_0, \dots, x_t) = \mu_t(x_{t-h^\varphi+1}, \dots, x_t) = u_{0|t}^{H, \text{opt}}$, where $u_{0|t}^{H, \text{opt}} = u_{0|t}^{H, \text{opt}} \dots u_{H-1|t}^{H, \text{opt}}$ is given by (33). If the optimization problem (33) is feasible for all $t \geq h^\varphi - 1$, then $\mathbf{x}(x_0, \mu, \mathbf{w})[0] \models \mathbf{G}_{[0, \infty]} \varphi, \forall \mathbf{w} \in \mathcal{W}^\omega$.

Proof: We show that $\mathbf{x}(x_0, \mu, \mathbf{w})[0 : k + h^\varphi] \models \mathbf{G}_{[0, k]} \varphi, \forall \mathbf{w} \in \mathcal{W}^*, \forall k \in \mathbb{N}$, using induction over k . Consider (33) for $t = k + h^\varphi - 1$ for any $k \in \mathbb{N}$. The second constraint in (33) requires $\mathbf{x}_{|t}[k, k + h^\varphi] \models \varphi$, or equivalently,

$x_k \dots x_{k+h^\varphi-1} x_{1|k+h^\varphi-1} \models \varphi$. By applying $u_{0|t}^{\text{opt}}$, monotonicity implies $x_{k+h^\varphi} = f(x_{k+h^\varphi-1}, u_{0|t}^{\text{opt}}, w) \preceq x_{1|k+h^\varphi-1} = f^*(x_{k+h^\varphi-1}, u_{0|t}^{\text{opt}}), \forall w \in \mathcal{W}$. From Lemma 1, we have $\mathbf{x}[k : k+h^\varphi] \models \varphi$. Thus, we have shown $\mathbf{x}[k : k+h^\varphi] \models \varphi, \forall k \in \mathbb{N}$, and the proof is complete. ■

Proposition 4: The MPC strategy corresponding to (33) is recursively feasible.

Proof: Suppose $u_t^H = u_{0|t} \dots u_{H-1|t}$ and $x_t^H = x_{t+1|t} \dots x_{t+H-1|t}$ is a feasible solution for (33) at time t . Since $\Omega_{\mathcal{L}^\varphi}$ is a RCI set, there exist $u^r \in \mathcal{U}$ such that $\mathbf{x}_{|t}[t+H+1-h^\varphi : t+H+1] = x_{H-h^\varphi+1|t} x_{H-h^\varphi+2|t} \dots x_{H|t} f(x_{H|t}, u^r, w) \in \Omega_{\mathcal{L}^\varphi}, \forall w \in \mathcal{W}$. Suppose $u_{0|t}$ is applied to the system. We have $x_{t+1} = f(x_t, u_{0|t}, w) \preceq f^*(x_t, u_{0|t}) = x_{1|t}, \forall w \in \mathcal{W}$.

Now, we prove that the optimization problem at time $t+1$ is feasible by showing that at least one feasible solution exists. Let $u_{t+1}^H = u_{1|t} u_{2|t} \dots u_{H|t} u^r$. We already showed that $x_{t+1} = x_{0|t+1} \preceq x_{1|t}$. By induction and using monotonicity, it follows that $x_{k|t+1} \preceq x_{k+1|t}, k = 1, \dots, H-2$. Therefore, we have $x_{t-h^\varphi+2} \dots x_{t+1} x_{1|t+1} \dots x_{H-1|t+1} \preceq x_{t-h^\varphi+2} \dots x_{1|t} x_{2|t} \dots x_{H|t}$, which using Lemma 1 establishes $x_{t-h^\varphi+2} \dots x_{t+1} x_{1|t+1} \dots x_{H-1|t+1} \models \mathbf{G}_{[0, H-1]} \varphi$. In order to complete the proof, it remains to show that $\mathbf{x}[t+H+1-h^\varphi : t+H+1] = x_{H+1-h^\varphi|t} \dots x_{H|t+1} \models \varphi$. This follows from invariance. Note that $x_{H|t+1} = f^*(x_{H|t}, u^r)$. Therefore $x_{H+1-h^\varphi|t} = \dots x_{H|t+1} \in \Omega_{\mathcal{L}^\varphi}$, and since $\Omega_{\mathcal{L}^\varphi} \in \mathcal{L}^\varphi$, we have $x_{H+1-h^\varphi|t} \dots x_{H|t+1} \models \varphi$, and the proof is complete. ■

The MPC optimization problem is also converted into a MILP problem. It is computationally easier to solve the optimization problem in (33) by solving T MILPs:

$$\begin{aligned} u_t^{\text{opt}, H} &= \operatorname{argmin}_{u_t^H \in \mathcal{U}^H, i=0, \dots, T-1} J(x_t^H, u_t^H) \\ \text{s.t. } &x_{k+1|t} = f^*(x_{k|t}, u_{k|t}), k = 0, \dots, H-1 \\ &\mathbf{x}_{|t}[t-h^\varphi+1, t+H] \models \mathbf{G}_{[0, H-1]} \varphi \\ &\mathbf{x}_{|t}[t+H-h^\varphi : t+H] \in L(\mathbf{x}^\varphi, x_0 [i : i+h^\varphi]). \end{aligned} \quad (34)$$

Note that all MILPs can be aggregated into a single large MILP in the expense of additional constraints for capturing nonconvexities of the terminal condition.

Finally, consider $t < h^\varphi$. In this case, we require $H \geq h^\varphi$ and replace the interval $[t-h^\varphi+1, t+H-h^\varphi]$ with $[0, t+H-h^\varphi]$ for $t < h^\varphi$ in (34). For applications where initialization is not important in the long term (like traffic management), a simpler approach is to initialize the MPC from $t = h^\varphi - 1$ and assume that all previous state values are zero (hence, all the past predicates are evaluated as true).

Remark 3: In our previous work on STL MPC of linear systems [12], we did not establish recursive feasibility. In order to recover from possible infeasibility issues, we proposed maximizing the STL robustness score (a negative value) whenever the MPC optimization problem became infeasible. Although recursive feasibility is guaranteed here, unmodeled disturbances and initial conditions outside $\mathcal{X}_0^{\text{max}}$ can lead to infeasibility. The formalism in [12] can be used to recover from infeasibility with minimal violation of the specification.

VII. APPLICATION TO TRAFFIC MANAGEMENT

In this section, we explain how to apply our methods to traffic management. First, the model that we use for traffic networks is explained, which is similar to the one in [47] but freeways are also modeled. Next, the monotonicity properties of the model are discussed. We show that there exists a congestion-free set in the state space in which the traffic dynamics is monotone. Finally, a case study on a mixed urban and freeway network is presented.

A. Model

The topology of the network is described by a directed graph $(\mathcal{V}, \mathcal{L})$, where \mathcal{V} is the set of nodes and \mathcal{L} is the set of edges. Each $l \in \mathcal{L}$ represents a one-way traffic link from tail node $\tau(l) \in \mathcal{V} \cup \emptyset$ to head node $\eta(l) \in \mathcal{V}$, where $\tau(l) = \emptyset$ stands for links originating from outside of the network. We distinguish between the following three types of links based on their control actuations:

- 1) \mathcal{L}_r : road links actuated by traffic lights;
- 2) \mathcal{L}_o : freeway on-ramps actuated by ramp meters; and
- 3) \mathcal{L}_f : freeway segments that are not directly controlled.

Freeway off-ramps are treated in the same way as the roads. Uncontrolled roads are also treated the same as freeways. We have $\mathcal{L}_r \cup \mathcal{L}_o \cup \mathcal{L}_f = \mathcal{L}$.

Remark 4: Some works, e.g., [14], consider control over freeway links by varying speed limits, which adds to the control power but requires the existence of such a control architecture within the infrastructure. We do not consider this type of control actuation in this paper but it can easily be incorporated into our model by modeling freeways links the same way as on-ramps, where the speed limit becomes analogous to the ramp meter input.

The number of vehicles on link l at time t is represented by $x_{[l],t} \in [0, c_l]$, which is assumed to be a continuous variable, and c_l is the capacity of l . In other words, vehicular movements are treated as fluid-like flow in our model. The number of vehicles that are able to flow out of l in one time step, if link l is actuated, is

$$q_{[l],t} := \min \left\{ x_{[l],t}, \bar{q}_l, \min_{\{l'| \tau(l')=\eta(l)\}} \frac{\alpha_{l:l'}}{\beta_{l:l'}} (c_{l'} - x_{[l'],t}) \right\} \quad (35)$$

where \bar{q}_l is the maximum outflow of link l in one time step, which is physically related to the speed of the vehicles. The last argument in the minimizer determines the minimum supply available in the downstream links of l , where $\alpha_{l:l'} \in [0, 1]$ is the capacity ratio of link l' available to vehicles arriving from link l (typically portion of the lanes), $\beta_{l:l'} \in [0, 1]$ is the ratio of the vehicles in l that flow into l' (turning ratio). For simplicity, we assume that capacity ratios and turning ratios are constants. System state is represented by $x \in \mathbb{R}_+^n : \{x_{[l]}\}_{l \in \mathcal{L}}$, where n is the number of the links in the network. The state space is $\mathcal{X} := \prod_{l \in \mathcal{L}} [0, c_l]$.

A schematic diagram illustrating the behavior of $q_{[l]}$ with respect to the state variables $x_{[l]}, x_{[l']}$ —which is known as the *fundamental diagram* in the traffic literature [48]—is shown in Fig. 4. The link flow drops if one (or more) of its downstream links do not have enough capacity to accommodate the incoming flow. In this case [when the last argument in (35) is the minimizer], we say that the traffic flow is *congested*. Otherwise, the traffic flow is *free*. This motivates the following definition.

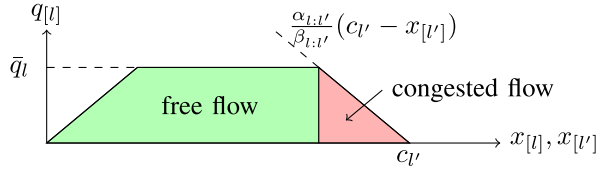


Fig. 4. Fundamental diagram. The flow out of link l drops if the number of vehicles on the immediate downstream link l' is close to its capacity. The congestion is defined by this blocking behavior.

Definition 10: The *congestion-free set*, denoted by Π , is defined as the following region in the state space:

$$\Pi := \left\{ x \in \mathcal{X} \mid \min\{x_{[l]}, \bar{q}_l\} \leq \frac{\alpha_{l:l'}}{\beta_{l:l'}} (c_{l'} - x_{[l']}) \right. \\ \left. \forall l, l' \in \mathcal{L}, \tau(l') = \eta(l) \right\}. \quad (36)$$

Proposition 5: The congestion-free set is a lower set.

Proof: Consider $x \in \Pi$ and any $x' \in L(x)$. For all $l, l' \in \mathcal{L}, \tau(l') = \eta(l)$, we have $\min\{x'_l, \bar{q}_l\} \leq \min\{x_{[l]}, \bar{q}_l\}$ and $(c_{l'} - x_{[l']}) \leq (c_{l'} - x'_{[l']})$. Therefore, $\min\{x'_l, \bar{q}_l\} \leq \frac{\alpha_{l:l'}}{\beta_{l:l'}} (c_{l'} - x'_{[l']})$. Thus, $x' \in \Pi$, which indicates Π is a lower set. ■

Note that Π is, in general, nonconvex. The predicate ($x \in \Pi$) can be written as a Boolean logic formula over predicates in the form of (7) as

$$\bigwedge_{l, l' \in \mathcal{L}, \tau(l') = \eta(l)} \left(\left((x_{[l]} \leq \bar{q}_l) \wedge \left(x_{[l]} + \frac{\alpha_{l:l'}}{\beta_{l:l'}} x_{[l']} \leq \frac{\alpha_{l:l'}}{\beta_{l:l'}} c_{l'} \right) \right) \vee \left(q_{[l]} + \frac{\alpha_{l:l'}}{\beta_{l:l'}} x_{[l']} \leq \frac{\alpha_{l:l'}}{\beta_{l:l'}} c_{l'} \right) \right). \quad (37)$$

Notice how the minimizer in (36) is translated to a disjunction in (37).

Now, we explain the controls. The actuated flow of link l at time t is denoted by $\vec{q}_{[l],t}$, where we have the following relations:

$$\vec{q}_{[l],t} = \begin{cases} s_{[l],t} q_{[l],t}, & l \in \mathcal{L}_r \\ \min\{q_{[l],t}, r_{[l],t}\}, & l \in \mathcal{L}_o \\ q_{[l],t}, & l \in \mathcal{L}_f \end{cases} \quad (38)$$

where $s_{[l],t} \in \{0, 1\}$ is the traffic light for link l , where 1 (respectively, 0) stands for green (respectively, red) light, and $r_{[l],t} \in \mathbb{R}_+$ is the ramp meter input for on-ramp l at time t . Ramp meter input limits the number of vehicles that are allowed to enter the freeway in one time step. In order to disallow simultaneous green lights for links l, l' (which are typically pair of links pointing toward a common intersection in perpendicular directions), we add the additional constraints $s_{[l],t} + s_{[l'],t} \leq 1$. In simple gridded networks, as in our case study network illustrated in Fig. 5, it is more convenient to define phases for actuation in north–south (NS) or east–west (EW) directions that are unambiguously mapped to traffic lights for each individual link. The evolution of the network is given by the following:

$$x_{[l],t+1} = x_{[l],t} - \vec{q}_{[l],t} + w_{[l],t} + \sum_{l', \eta(l') = \tau(l)} \beta_{l:l'} \vec{q}_{[l'],t} \quad (39)$$

where $w_{[l],t}$ is the number of exogenous vehicles entering link l at time t , which is viewed as the adversarial input. The evolution relation mentioned above can be compacted into the form (5) as

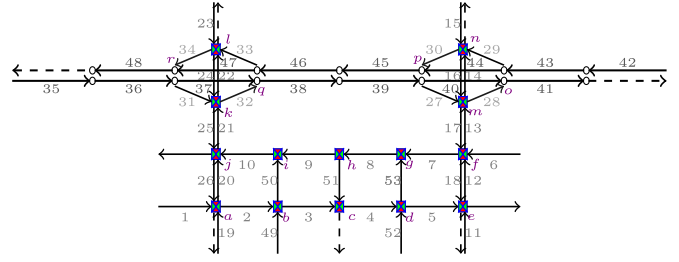


Fig. 5. Traffic management case study: A network of freeways and urban roads. There are 14 intersections controlled by traffic lights and 4 ramp meters.

follows:

$$x_{t+1} = f_{\text{traffic}}(x_t, u_t, w_t) \quad (40)$$

where u_t and w_t are the vector representations for control inputs (combination of traffic lights and ramp meters) and disturbances inputs, respectively. Note that f_{traffic} represents a hybrid system in which each mode is affine. The mode is determined by the control inputs and state (which determines the minimizer arguments). Some works consider nonlinear representations for the fundamental diagram (see Fig. 4), but they still can be approximated using piecewise affine functions.

B. Monotonicity

Theorem 5: System (40) is monotone in Π .

Proof: Consider $x', x \in \Pi, x \leq x'$. We show that $f_{\text{traffic}}(x, u, w) \leq f_{\text{traffic}}(x', u, w), \forall w \in \mathcal{W}, \forall u \in \mathcal{U}$. Observe in (39) that we only need to verify is proving that $x_{[l]} - \bar{q}_{[l]}$ is a non-decreasing function of $x_{[l]}$ as all other terms are additive and nondecreasing with respect to x . Since $x, x' \in \Pi$, the last argument in (35) is never the minimizer. Thus, for all $l \in \mathcal{L}$, we have $x_{[l]} - \bar{q}_{[l]} \in \{0, x_{[l]} - r_{[l]}, x_{[l]} - c_{[l]}, x_{[l]}\}$, depending on the mode of the system and actuations, which all are nondecreasing functions of $x_{[l]}$. Thus, f_{traffic} is monotone in Π . ■

The primary objective in our traffic management approach is finding control policies such that the state is restricted to Π , which not only eliminates congestion, but also ensures that the system is monotone, hence the methods of this paper become applicable. It is worth to note that the traffic system becomes nonmonotone when flow is congested in diverging junctions, as shown in [49]. This phenomenon is attributed to the first-in-first-out (FIFO) nature of the model. By assuming fully non-FIFO models, system becomes monotone in the whole state space. For a more thorough discussion on physical aspects of monotonicity in traffic networks see [15].

The maximal system in (40) corresponds to the scenario where each w_l is equal to its maximum allowed value w_l^* .

C. Case Study

1) Network: Consider the network in Fig. 5, which consists of urban roads (see links 1–26, 27, 29, 31, 33 and 49–53), freeway segments (see links 35–48), and freeway on-ramps (see links 28, 30, 32, 34). The layout of the network illustrates a freeway passing by an urban area, which is common in many realistic traffic layouts. There are 14 intersections (nodes a–n) controlled by traffic lights. Each intersection has two modes of actuation: NS and EW. There are four entries to the freeway

TABLE I
PARAMETERS OF THE NETWORK IN FIG. 5

links	parameters
1–26, 49–53	$\bar{q}_l = 15, c_l = 40$
27–34	$\bar{q}_l = 15, c_l = 30$
35–48	$\bar{q}_l = 40, c_l = 60$
Turning ratios	value
$\beta_{2:50}, \beta_{4:53}, \beta_{8:51}, \beta_{12:7}, \beta_{13:28}, \beta_{15:30}, \beta_{16:28},$ $\beta_{21:32}, \beta_{24:32}, \beta_{26:2}, \beta_{36:31}, \beta_{36:33}, \beta_{39:27}, \beta_{43:29}$	0.2
$\beta_{5:12}, \beta_{6:13}, \beta_{6:18}, \beta_{10:21}, \beta_{10:26}$	0.3
$\beta_{1:20}, \beta_{6:7}$	0.4
$\beta_{1:2}, \beta_{11:12}, \beta_{14:30}, \beta_{17:7}, \beta_{17:18}, \beta_{19:2}, \beta_{19:20},$ $\beta_{22:34}, \beta_{23:24}, \beta_{23:34}, \beta_{27:14}, \beta_{27:17}, \beta_{29:16},$ $\beta_{31:22}, \beta_{31:25}, \beta_{33:24}, \beta_{49:3}, \beta_{49:50}, \beta_{51:4},$ $\beta_{52:5}, \beta_{52:53},$	0.5
$\beta_{2:3}, \beta_{3:4}, \beta_{4:5}, \beta_{8:9}, \beta_{12:13}, \beta_{13:14}, \beta_{15:16}, \beta_{16:17},$ $\beta_{20:21}, \beta_{21:22}, \beta_{24:25}, \beta_{25:26}, \beta_{36:37}, \beta_{39:40},$ $\beta_{43:44}, \beta_{46:47},$	0.8
Capacity ratios	value
$\alpha_{19:2}, \alpha_{26:2}, \alpha_{17:7}, \alpha_{12:7}, \alpha_{13:28}, \alpha_{16:28}$	0.5
$\alpha_{14:30}, \alpha_{15:30}, \alpha_{21:32}, \alpha_{24:32}, \alpha_{22:34}, \alpha_{23:34}$	0.5
Disturbances (arrival rates)	
$w_1^* = w_6^* = 4.5, w_{11}^* = w_{15}^* = w_{19}^* = 5, w_{23}^* = 6$ $w_{35}^* = w_{42}^* = 20, w_{49}^* = w_{52}^* = 2$	

TABLE II
EXISTENCE OF ϕ -SEQUENCES FOR THE NETWORK IN FIG. 5

Demand Changes from Table I	T	Existence	Comp. Time (s)
–	5	yes	6
–	6	no	4
–	7	no	10
–	8	no	75
–	9	no	11
–	10	yes	36
$w_1^* = w_6^* = 3, w_{11}^* = w_{15}^* = w_{19}^* = 6$	5	yes	5
$w_1^* = w_6^* = 4, w_{11}^* = w_{15}^* = w_{19}^* = 6$	5	no	0.5
$w_1^* = w_6^* = 1.5, w_{49}^* = w_{52}^* = 3.5$	5	yes	16
$w_1^* = w_6^* = 7.5, w_{11}^* = w_{15}^* = w_{19}^* = w_{23}^* = 2$	6	yes	9
$w_1^* = w_6^* = 9, w_{11}^* = w_{15}^* = w_{19}^* = w_{23}^* = 1$	5	yes	4
$w_1^* = w_6^* = 10, w_{11}^* = w_{15}^* = w_{19}^* = w_{23}^* = 0$	30	no	3.5
$w_{15}^* = w_{23}^* = 8, w_{35}^* = w_{42}^* = 10$	6	yes	23
$w_{15}^* = w_{23}^* = 0, w_{35}^* = w_{42}^* = 30$	5	yes	4

(nodes o-r) that are regulated by ramp meters. We have $n = 53$ and $\mathcal{U} = \mathbb{R}_+^4 \times \{0, 1\}^{14}$. Vehicles arrive from links 1, 6, 11, 15, 19, 23, 35, 42, 49, and 52. The parameters of the network are shown in Table I.

2) Specification: As mentioned earlier, the primary objective is keeping the state in the congestion-free set. In addition, since the demand for the NS side roads (see links 49–53) is smaller than the traffic in the EW roads, we add a timed liveness requirement for the traffic flow on links 49–53:

$$\psi = \bigwedge_{l=49,50,\dots,53} (x_{[l]} \geq 5) \Rightarrow \mathbf{F}_{[0,3]}(x_{[l]} \leq 5)$$

which states that “if the number of vehicles on any of the NS side roads exceeds 5, their flow is eventually actuated within three time units ahead.” The global specification is given as

$$\phi = \mathbf{G}_{[0,\infty]}((x \in \Pi) \wedge \psi). \quad (41)$$

TABLE III
 ϕ -SEQUENCE IN THE CASE STUDY

node	Initialization			Repetitive Controls				
	u_0^ϕ	u_1^ϕ	u_2^ϕ	u_3^ϕ	u_4^ϕ	u_5^ϕ	u_6^ϕ	u_7^ϕ
a	EW	NS	NS	NS	EW	EW	NS	NS
b	NS	EW	EW	EW	NS	NS	EW	EW
c	EW	NS	NS	EW	EW	EW	NS	NS
d	EW	NS	EW	NS	EW	EW	NS	EW
e	EW	EW	NS	NS	NS	EW	EW	NS
f	NS	EW	NS	EW	NS	NS	EW	NS
g	NS	EW	NS	EW	EW	NS	EW	NS
h	EW	NS	EW	EW	EW	EW	NS	EW
i	EW	NS	EW	EW	NS	EW	NS	EW
j	EW	EW	NS	NS	EW	EW	EW	NS
k	EW	EW	NS	NS	NS	EW	EW	NS
l	NS	EW	EW	NS	NS	NS	EW	EW
m	EW	NS	NS	EW	NS	EW	NS	NS
n	NS	NS	EW	NS	EW	NS	NS	EW
o	0.0	0.0	0.0	12.8	0.0	0.0	0.0	0.0
p	4.0	14.0	0.0	9.5	0.0	4.0	11.5	0.0
q	0.0	0.0	10.0	0.0	2.5	0.0	0.0	10.0
r	5.5	0.0	4.0	14.0	11.5	5.5	0.0	4.0

Note that $h^\varphi = 3$, $\varphi = (x \in \Pi) \wedge \psi$.

3) Open-Loop Control Policy: We use Theorem 3. The shortest ϕ -sequence that we found for this problem has $T = 5, T_0 = 0$. The corresponding MILP had 2357 variables (of which 1061 were binary) and 4037 constraints,⁴ which is solved using the Gurobi MILP solver in less than 6 s on a dual core 3.0 GHz MacBook Pro. The cost is set to zero in order to just check for feasibility. Even though finding an optimal solution and checking for feasibility of a MILP have the same theoretical complexity, the latter is executed much faster in practice. For instance, finding a ϕ -sequence, while minimizing or maximizing $\sum_{k=0}^7 \|x_k^\phi\|_1$ both took more than 20 min. Note that it is virtually intractable to attack a problem of this size (53 dimensional state) using any method that involves state-space discretization, such as the method in [33] (e.g., if each state-component is partitioned into two intervals, the finite-state problem size will be 2^{53}).

Monotonicity implies that any demand set \mathcal{W} for which there exists a solution to Problem 2 is a lower set. The set corresponding to the values at the bottom of Table I is one of them. Table II shows results on existence of ϕ -sequences for some other demand scenarios. Computation times for solving a MILP do not demonstrate a generic behavior. For the rest of this section, the numerical examples are reported for the values in Table I.

The control values in the ϕ -sequence are shown in Table III. As stated in Theorem 3, starting from an initial condition in $L(x_0)$, applying the open-loop control policy (21) guarantees satisfaction of the specification. In other words, after applying the initialization segment, the repetitive controls in Table III become a fixed timetable for the inputs of the traffic lights and the ramp meters. Starting from x_0 , which is a 53-dimensional vector, we apply (21) using the values in Table III. The trajectory of the maximal system is shown in Fig. 6 [Top]. The traffic signals are coordinated such that the traffic flows free of congestion. The black dashed lines represent the capacity of the links, and

⁴The scripts for this case study are available in <http://blogs.bu.edu/sadra/format-monotone>

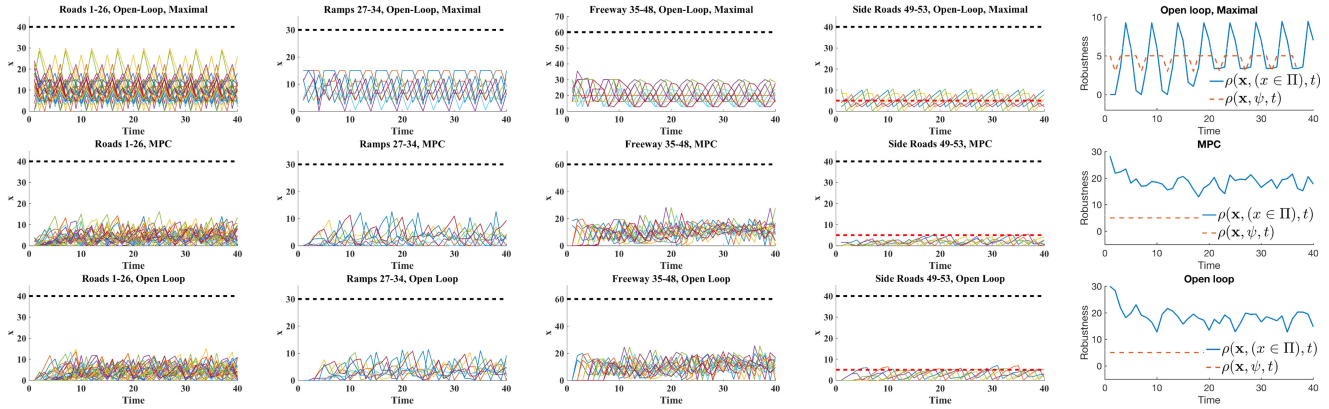


Fig. 6. Traffic management case study: [Top Row] the trajectory of the maximal system obtained from applying the open-loop control policy (21) with initial condition x_0^φ [Middle Row] robust MPC generated trajectory with zero initial condition with disturbances chosen uniformly $L(w^*)$ [Bottom Row] trajectory generated from applying the open-loop control policy (21) with zero initial conditions and the same disturbances as in [Middle Row].

the dashed line in the fourth figure (from the left) represents the threshold for the liveness subspecification (ψ). It is observed that all the state values for side road links (49–53) persistently fall below the threshold. The robustness values for $(x \in \Pi)$ and ψ are shown in the fifth figure. As mentioned earlier, robustness corresponds to the minimum volume of vehicles that the system is away from congestion, or violating the specification. The robustness values are always positive, indicating satisfaction.

As stated in Theorem 4, the trajectory of the maximal system converges to a periodic orbit. It is worth to note that the number of vehicles on freeway links is significantly smaller than its capacity, which is attributed to the fact that the number designated for \bar{q} (related to the maximum speed) of freeway links is relatively large (30, as opposed to 15 for roads). Therefore, freeway links are utilized in a way that there is enough space for high-speed noncongested flow.

4) Robust MPC: Here, it is assumed that the controller has full state knowledge. We apply the techniques developed in Section VI. Using the result from the previous section, the set $\Omega_{\mathcal{L}^\varphi}$ is constructed in \mathbb{R}_+^{212} ($= \mathbb{R}^{n(h^\varphi+1)}$, $n = 53$, $h^\varphi = 3$). The cost criteria that we use in this case study is the total delay induced in the network over the planning horizon H . A vehicle is delayed by one time unit if it cannot flow out of a link in one time step, which may be because of the actuation (e.g., red light) or waiting for the flow of other vehicles in the same link (i.e., we have $x_{[l]} \geq c_l$). We are also interested in maximizing the STL robustness score. The cost function is

$$J_{\text{traffic}}(x^H, u^H) := -\zeta \rho(\mathbf{x}, \mathbf{G}_{[0, H-1]} \varphi, t - h^\varphi + 1) + \sum_{k=0}^{H-1} \gamma^k \sum_{l \in \mathcal{L}} (x_{[l], t+k} - \bar{q}_{[l], t+k}) \quad (42)$$

where $\bar{q}_{[l]}$, given by (38), is the amount of vehicles that flow out of link l , γ is the discount factor for delays predicted in future, and ζ is a positive weight for robustness. Notice the connection between the time window of STL robustness score in (42) and MPC constraint enforcement in (33). It follows from Theorem 5 and STL quantitative semantics (4) that the cost function mentioned above is nondecreasing with respect to the state in Π . Therefore, in order to minimize the worst case

cost, the maximal system is considered in the MPC optimization problem.

Starting from zero initial conditions, we implement the MPC algorithm (34) with $H = 3$ for 40 time steps. We set $\zeta = 1000$, $\gamma = 0.5$ in (42). The disturbances at each time step were randomly drawn from $L(w^*)$ using a uniform distribution. The maximum computation time for each MPC step time step was less than 0.8 s (less than 0.5 s on average). The resulting trajectory is shown in Fig. 6 [Middle]. For the same sequence of disturbances, the trajectory resulted from applying the open-loop control policy (21) (using the values in Table III) is shown in Fig. 6 [Bottom]. Both trajectories satisfy the specification. However, robust MPC has obviously better performance when costs are considered. The total delay accumulated over 40 time steps is

$$J_{40} = \sum_{\tau=0}^{40} \sum_{l \in \mathcal{L}} (x_{[l], \tau} - \bar{q}_{[l], \tau}).$$

The cost above obtained from applying robust MPC was $J_{40} = 1843$, while the one for the open-loop control policy was $J_{40} = 2299$, which demonstrates the usefulness of the state knowledge in planning controls in a more optimal way. An optimal tuning of parameters η and γ requires an experimental study which is out of scope of this paper. We only remark that we usually obtained larger delays with nonzero η , which shows that including STL robustness score in the MPC cost function may be useful even though the ultimate goal is minimizing the total delay.

It is worth to note that we also tried implementing the MPC algorithm (for the case $\mathbf{w} = (w^*)^\omega$, or the maximal system) without the terminal constraints, as in (32). The MPC got infeasible at $t = 8$. The violating constraints were those in $x \in \Pi$. This observation indicates that the myopic behavior of MPC in (32), when no additional constraints are considered, can lead to congestion in the network.

VIII. CONCLUSION AND FUTURE WORK

We developed methods to control positive monotone discrete-time systems from STL specifications. We showed that open-loop control sequences are sufficient and (almost) necessary for guaranteeing the correctness of STL specifications. A robust MPC method was introduced to plan controls optimally,

while guaranteeing global STL specifications. We showed the usefulness of our results on traffic management.

Future work will focus on nonmonotone systems with parametric uncertainty whose state evolution can be overapproximated in an appropriate way using monotone systems. We will develop adaptive control schemes to tune parameters automatically using the data gathered from the evolution of the system. This will eventually lead to data-driven control techniques for transportation networks with formal guarantees.

APPENDIX

Theorem 6: Let \mathbb{S} be the set of all STL formulas that can be written in the form:

$$\phi = \bigvee_{i=1}^{n_\phi} \varphi_{b,i} \wedge \mathbf{G}_{[\Delta_i, \infty]} \varphi_{g,i} \quad (43)$$

where $\varphi_{b,i}, \Delta_i \geq h^{\varphi_{b,i}}, \varphi_{g,i}, i = 1, \dots, n_\phi$ are bounded STL formulas. Then, \mathbb{S} is a subset of safety STL formulas that is closed under STL syntax with bounded temporal operators.

Proof: First, a quick inspection of (43) verifies that it is a safety STL formula. A predicate π is a bounded formula (with zero horizon) and is a special case of (43), hence $\pi \in \mathbb{S}$.

We also have the following property that relaxes the form in (43): *For all bounded STL formulas φ_1, φ_2 , we have $\varphi_1 \wedge \mathbf{G}_{[\Gamma, \infty]} \varphi_2 \in \mathbb{S}, \forall \Gamma \in \mathbb{N}$.* Proof: The case for $\Gamma \geq h^{\varphi_1}$ is already in the form (43) with $n_\phi = 1$. If $\Gamma < h^{\varphi_1}$, we write $\mathbf{G}_{[\Gamma, \infty]} \varphi_2 = \mathbf{G}_{[\Gamma, h^{\varphi_1}]} \varphi_2 \wedge \mathbf{G}_{[h^{\varphi_1}, \infty]} \varphi_2$. Now, define $\varphi_1 \wedge \mathbf{G}_{[\Gamma, h^{\varphi_1}]} \varphi_2$ as the new bounded formula and retain the form in (43) with $n_\phi = 1$.

We show that \mathbb{S} is closed under STL syntax with bounded operators. The distributivity properties of Boolean connectives and temporal operators (see, e.g., [50]) imply that: $\phi_1 \vee (\phi_2 \wedge \phi_3) = (\phi_1 \vee \phi_2) \wedge (\phi_1 \vee \phi_3)$, $\phi_1 \wedge (\phi_2 \vee \phi_3) = (\phi_1 \wedge \phi_2) \vee (\phi_1 \wedge \phi_3)$, $\mathbf{F}_I(\phi_1 \vee \phi_2) = (\mathbf{F}_I \phi_1) \vee (\mathbf{F}_I \phi_2)$, and $\mathbf{G}_I(\phi_1 \wedge \phi_2) = (\mathbf{G}_I \phi_1) \wedge (\mathbf{G}_I \phi_2)$, where ϕ_1, ϕ_2, ϕ_3 are temporal logic formulas and I is an interval.

- 1) $\phi_1, \phi_2 \in \mathbb{S} \Rightarrow \phi_1 \wedge \phi_2 \in \mathbb{S}, \phi_1 \vee \phi_2 \in \mathbb{S}$: This result easily follows from the distributivity properties of Boolean connectives mentioned above.
- 2) $\phi \in \mathbb{S} \Rightarrow \mathbf{F}_{\{t\}} \phi \in \mathbb{S}$: We use $\mathbf{F}_{\{t\}} \mathbf{G}_{[a,b]} = \mathbf{G}_{[t+a, t+b]}$ and distributivity to have (note that $\mathbf{F}_{\{t\}} = \mathbf{G}_{\{t\}}$)

$$\begin{aligned} \mathbf{F}_{\{t\}} \left(\bigvee_{i=1}^{n_\phi} (\varphi_{b,i} \wedge \mathbf{G}_{[\Gamma_i, \infty]} \varphi_{g,i}) \right) \\ = \bigvee_{i=1}^{n_\phi} (\mathbf{F}_{\{t\}} \varphi_{b,i} \wedge \mathbf{G}_{[t+\Gamma_i, \infty]} \varphi_{g,i}). \end{aligned}$$

Introducing $\mathbf{F}_{\{t\}} \varphi_{b,i}, i = 1, \dots, n_\phi$ as new bounded STL formulas leads to the form in (43).

- 3) $\phi \in \mathbb{S} \Rightarrow \mathbf{F}_{[a,b]} \phi \in \mathbb{S}, \mathbf{G}_{[a,b]} \phi \in \mathbb{F}$: Use $\mathbf{F}_{[a,b]} = \bigvee_{t \in [a,b]} \mathbf{F}_{\{t\}}$ and $\mathbf{G}_{[a,b]} = \bigwedge_{t \in [a,b]} \mathbf{F}_{\{t\}}$ to convert temporal operators to Boolean connectives.
- 4) $\phi_1, \phi_2 \in \mathbb{S} \Rightarrow \phi_1 \mathbf{U}_{[a,b]} \phi_2 \in \mathbb{S}$: Use the STL semantics (2) to substitute the bounded “until” operator using

bounded “eventually” and bounded “always” operators

$$\phi_1 \mathbf{U}_{[a,b]} \phi_2 = \bigvee_{t \in [a,b]} (\mathbf{G}_{[a,t]} \phi_1 \wedge \mathbf{F}_{\{t\}} \phi_2).$$

Example 6: The “reach and stay” formula $\mathbf{F}_I \mathbf{G}_{[0, \infty]} \varphi$, where φ is a bounded formula, is equivalent to $\bigvee_{t \in I} \mathbf{G}_{[t, \infty]} \varphi$.

Remark 5: What remains to show that \mathbb{S} is equivalent to the set of all safety STL formulas is having that $\phi \in \mathbb{S} \Rightarrow \mathbf{G}_{[\Gamma, \infty]} \phi \in \mathbb{S}, \forall \Gamma \in \mathbb{N}$, which is not true by restricting n_ϕ in (43) to be finite. Formulas that involve nested unbounded “always” operator and cannot be further simplified, such as $\mathbf{G}_{[\Gamma', \infty]}(\varphi_1 \vee \mathbf{G}_{[\Gamma, \infty]} \varphi_2)$, are rarely encountered in applications. ■

REFERENCES

- [1] C. Baier and J.-P. Katoen, *Principles of Model Checking*. Cambridge, MA, USA: MIT Press, 2008.
- [2] P. Tabuada and G. J. Pappas, “Linear time logic control of discrete-time linear systems,” *IEEE Trans. Autom. Control*, vol. 51, no. 12, pp. 1862–1877, Dec. 2006.
- [3] M. Kloetzer and C. Belta, “A fully automated framework for control of linear systems from temporal logic specifications,” *IEEE Trans. Autom. Control*, vol. 53, no. 1, pp. 287–297, Feb. 2008.
- [4] B. Yordanov, J. Tumova, I. Cerna, J. Barnat, and C. Belta, “Temporal logic control of discrete-time piecewise affine systems,” *IEEE Trans. Autom. Control*, vol. 57, no. 6, pp. 1491–1504, Jun. 2012.
- [5] G. Pola and P. Tabuada, “Symbolic models for nonlinear control systems: Alternating approximate bismulations,” *SIAM J. Control Optim.*, vol. 48, no. 2, pp. 719–733, 2009.
- [6] M. Zamani, G. Pola, M. Mazo, and P. Tabuada, “Symbolic models for nonlinear control systems without stability assumptions,” *IEEE Trans. Autom. Control*, vol. 57, no. 7, pp. 1804–1809, Jul. 2012.
- [7] A. Bemporad and M. Morari, “Control of systems integrating logic, dynamics, and constraints,” *Automatica*, vol. 35, no. 3, pp. 407–427, 1999.
- [8] S. Karaman, R. G. Sanfelice, and E. Frazzoli, “Optimal control of mixed logical dynamical systems with linear temporal logic specifications,” in *Proc. 47th IEEE Conf. Decision Control*, 2008, pp. 2117–2122.
- [9] E. M. Wolff, U. Topcu, and R. M. Murray, “Optimization-based trajectory generation with linear temporal logic specifications,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2014, pp. 5319–5325.
- [10] V. Raman, A. Donzé, M. Maasoumy, R. M. Murray, A. Sangiovanni-Vincentelli, and S. A. Seshia, “Model predictive control with signal temporal logic specifications,” in *Proc. 53rd IEEE Conf. Decision Control*, 2014, pp. 81–87.
- [11] V. Raman, A. Donzé, D. Sadigh, R. M. Murray, and S. A. Seshia, “Reactive synthesis from signal temporal logic specifications,” in *Proc. ACM 18th Int. Conf. Hybrid Syst., Comput. Control*, 2015, pp. 239–248.
- [12] S. Sadraddini and C. Belta, “Robust temporal logic model predictive control,” in *Proc. 53rd Annu. Allerton Conf. Commun., Control Comput.*, Sep. 2015, pp. 772–779.
- [13] R. May, *Theoretical Ecology: Principles and Applications*. London, U.K.: Oxford Univ. Press, 2007.
- [14] G. Como, E. Lovisari, and K. Savla, “Throughput optimality and overload behavior of dynamical flow networks under monotone distributed routing,” *IEEE Trans. Control Netw. Syst.*, vol. 2, no. 1, pp. 57–67, Mar. 2015.
- [15] S. Coogan, M. Arcak, and A. A. Kurzhanskiy, “On the mixed monotonicity of FIFO traffic flow models,” in *Proc. 55th IEEE Conf. Decision Control*, 2016, pp. 7611–7616.
- [16] E. S. Kim, M. Arcak, and S. A. Seshia, “Directed specifications and assumption mining for monotone dynamical systems,” in *Proc. 19th ACM Int. Conf. Hybrid Syst., Comput. Control*, Vienna, Austria, 2016, pp. 21–30.
- [17] W. M. Haddad, V. Chellaboina, and Q. Hui, *Nonnegative and Compartmental Dynamical Systems*. Princeton, NJ, USA: Princeton Univ. Press, 2010.
- [18] A. Rantzer, “Distributed control of positive systems,” in *Proc. 50th IEEE Conf. Decision Control Eur. Control Conf.*, Dec. 2011, pp. 6608–6611.
- [19] P. De Leenheer and D. Aeyels, “Stabilization of positive linear systems,” *Syst. Control Lett.*, vol. 44, no. 4, pp. 259–271, 2001.

- [20] A. Rantzer, "Distributed control of positive systems," *2011 50th IEEE Conference on Decision and Control and European Control Conference*, Orlando, FL, 2011, pp. 6608–6611.
- [21] S. Sadraddini and C. Belta, "Safety control of monotone systems with bounded uncertainties," in *Proc. IEEE 55th Conf. Decision Control*, Dec. 2016, pp. 4874–4879.
- [22] M. W. Hirsch, "Systems of differential equations that are competitive or cooperative II: Convergence almost everywhere," *SIAM J. Math. Anal.*, vol. 16, no. 3, pp. 423–439, 1985.
- [23] M. W. Hirsch and H. Smith, "Monotone maps: A review," *J. Difference Equations Appl.*, vol. 11, no. 4/5, pp. 379–398, 2005.
- [24] H. Smith, *Monotone Dynamical Systems: An Introduction to the Theory of Competitive and Cooperative Systems*. Providence, RI, USA: American Mathematical Society, 2008, no. 41.
- [25] K. P. Hadeler and D. Glas, "Quasimonotone systems and convergence to equilibrium in a population genetic model," *J. Math. Anal. Appl.*, vol. 95, no. 2, pp. 297–303, 1983.
- [26] D. Angeli and E. D. Sontag, "Monotone control systems," *IEEE Trans. Autom. Control*, vol. 48, no. 10, pp. 1684–1698, Oct. 2003.
- [27] F. Blanchini, P. Colaneri, and M. E. Valcher, "Co-positive Lyapunov functions for the stabilization of positive switched systems," *IEEE Trans. Autom. Control*, vol. 57, no. 12, pp. 3038–3050, Dec. 2012.
- [28] E. Fornasini and M. E. Valcher, "Stability and stabilizability criteria for discrete-time positive switched systems," *IEEE Trans. Autom. Control*, vol. 57, no. 5, pp. 1208–1221, May 2012.
- [29] M. R. Hafner and D. Del Vecchio, "Computational tools for the safety control of a class of piecewise continuous systems with imperfect information on a partial order," *SIAM J. Control Optim.*, vol. 49, no. 6, pp. 2463–2493, 2011.
- [30] R. Ghaemi and D. Del Vecchio, "Control for safety specifications of systems with imperfect information on a partial order," *IEEE Trans. Autom. Control*, vol. 59, no. 4, pp. 982–995, Apr. 2014.
- [31] P.-J. Meyer, A. Girard, and E. Witrant, "Robust controlled invariance for monotone systems: Application to ventilation regulation in buildings," *Automatica*, vol. 70, pp. 14–20, 2016.
- [32] N. Ramdani, N. Meslem, and Y. Candau, "Computing reachable sets for uncertain nonlinear monotone systems," *Nonlinear Anal., Hybrid Syst.*, vol. 4, no. 2, pp. 263–278, 2010.
- [33] S. Coogan and M. Arcak, "Efficient finite abstraction of mixed monotone systems," in *Proc. ACM 18th Int. Conf. Hybrid Syst., Comput. Control*, 2015, pp. 58–67.
- [34] E. S. Kim, M. Arcak, and S. A. Seshia, "Symbolic control design for monotone systems with directed specifications," *Automatica*, vol. 83, pp. 10–19, 2017.
- [35] B. A. Davey and H. A. Priestley, *Introduction to Lattices and Order*. Cambridge, U.K.: Cambridge Univ. Press, 2002.
- [36] O. Maler and D. Nickovic, "Monitoring temporal properties of continuous signals," in *Formal Techniques, Modelling and Analysis of Timed and Fault-Tolerant Systems*. Berlin, Germany: Springer, 2004, pp. 152–166.
- [37] J. Ouaknine and J. Worrell, "Some recent results in metric temporal logic," in *Formal Modeling and Analysis of Timed Systems*. Berlin, Germany: Springer, 2008, pp. 1–13.
- [38] A. Dokhanchi, B. Hoxha, and G. Fainekos, "On-line monitoring for temporal logic robustness," in *Runtime Verification*. Berlin, Germany: Springer, 2014, pp. 1–20.
- [39] J. Ouaknine and J. Worrell, "Safety metric temporal logic is fully decidable," in *Proc. Int. Conf. Tools Algorithms Construction Anal. Syst.*, 2006, pp. 411–425.
- [40] W. Heemels, B. D. Schutter, and A. Bemporad, "Equivalence of hybrid dynamical models," *Automatica*, vol. 37, no. 7, pp. 1085–1091, 2001.
- [41] S. Sadraddini and C. Belta, "Feasibility envelopes for metric temporal logic specifications," in *Proc. IEEE 55th Conf. Decision Control*, Dec. 2016, pp. 5732–5737.
- [42] F. Blanchini, "Set invariance in control—A survey," *Automatica*, vol. 35, no. 11, pp. 1747–1767, 1999.
- [43] E. C. Kerrigan, "Robust constraint satisfaction: Invariant sets and predictive control," Ph.D. dissertation, Dept. of Eng., Univ. of Cambridge, Cambridge, U.K., 2000.
- [44] S. V. Raković, P. Grieder, M. Kvasnica, D. Q. Mayne, and M. Morari, "Computation of invariant sets for piecewise affine discrete time systems subject to bounded disturbances," in *Proc. 43rd IEEE Conf. Decision Control*, 2004, vol. 2, pp. 1418–1423.
- [45] W. A. Sutherland, *Introduction to Metric and Topological Spaces*. London, U.K.: Oxford Univ. Press, 1975.
- [46] E. C. Kerrigan and J. M. Maciejowski, "Robust feasibility in model predictive control: Necessary and sufficient conditions," in *Proc. IEEE Conf. Decision Control*, 2001, vol. 1, pp. 728–733.
- [47] S. Coogan, E. A. Gol, M. Arcak, and C. Belta, "Traffic network control from temporal logic specifications," *IEEE Trans. Control Netw. Syst.*, vol. 3, no. 2, pp. 162–172, Jun. 2016.
- [48] N. Geroliminis and C. F. Daganzo, "Existence of urban-scale macroscopic fundamental diagrams: Some experimental findings," *Transp. Res. Part B, Methodological*, vol. 42, no. 9, pp. 759–770, 2008.
- [49] S. Coogan and M. Arcak, "Dynamical properties of a compartmental model for traffic networks," in *Proc. Amer. Control Conf.*, 2014, pp. 2511–2516.
- [50] M. Huth and M. Ryan, *Logic in Computer Science: Modelling and Reasoning About Systems*. Cambridge, U.K.: Cambridge Univ. Press, 2004.



Sadra Sadraddini (S'16) received the B.Sc. degrees in mechanical engineering and in aerospace engineering (dual majors) from the Sharif University of Technology, Tehran, Iran, in 2013. He is currently working toward the Ph.D. degree in mechanical engineering at Boston University, Boston, MA, USA.

His research interests include formal methods to control theory with various applications in cyber-physical systems.



Calin Belta (F'17) received B.Sc. and M.Sc. degrees from the Technical University of Iasi, Romania in 1995 and 1997 and M.Sc. and Ph.D. degrees from the University of Pennsylvania, Philadelphia, USA in 2001 and 2003. He is a Professor in the Department of Mechanical Engineering at Boston University, where he holds the Tegan family Distinguished Faculty Fellowship. He is the Director of the BU Robotics Lab, and is also affiliated with the Department of Electrical and Computer Engineering, the Division of Systems Engineering at Boston University, the Center for Information and Systems Engineering (CISE), and the Bioinformatics Program. His research focuses on dynamics and control theory, with particular emphasis on hybrid and cyber-physical systems, formal synthesis and verification, and applications in robotics and systems biology. He received the Air Force Office of Scientific Research Young Investigator Award and the National Science Foundation CAREER Award. He is a fellow of IEEE.

Prof. Belta was the recipient of the Air Force Office of Scientific Research Young Investigator Award and the National Science Foundation CAREER Award.