

# Control Barrier Functions for Signal Temporal Logic Tasks

Lars Lindemann, *Student Member, IEEE*, and Dimos V. Dimarogonas, *Senior Member, IEEE*

**Abstract**—The need for computationally-efficient control methods of dynamical systems under temporal logic tasks has recently become more apparent. Existing methods are computationally demanding and hence often not applicable in practice. Especially with respect to multi-robot systems, these methods do not scale computationally. In this work, we propose a framework that is based on control barrier functions and signal temporal logic. In particular, time-varying control barrier functions are considered where the temporal properties are used to satisfy signal temporal logic tasks. The resulting controller is given by a switching strategy between a computationally-efficient convex quadratic program and a local feedback control law.

**Index Terms**—Autonomous systems, constrained control, hybrid systems.

## I. INTRODUCTION

THE classical objectives considered in automatic control are stabilization and tracking. Robotic applications, however, have shown the need for more complex objectives such as periodic, sequential, or reactive tasks that require new control methods and planning. These complex objectives are typically formulated in temporal logics [1]. Existing approaches that use linear temporal logic (LTL) rely on abstracting the system dynamics into a finite state transition system [2], [3]. This transition system is then used to obtain a discrete plan, which is subsequently executed by continuous-time control laws. These abstraction-based approaches are subject to large computational burdens due to the curse of dimensionality. Computationally-efficient frameworks for LTL have been presented in [4], [5] by considering a game theory-based approach and in [6] by considering a deductive proof system-based approach. In the former approach, however, the dynamical properties of the system need to be manually incorporated into the planning stage. The latter approach establishes a connection between discrete planning and continuous plan execution by low-level control guarantees such as those given by barrier functions. However, the focus in [6] is on deriving a deductive proof system for LTL tasks. In this paper, we also rely on low-level control guarantees, but without using deductive proof systems and for signal temporal logic (STL) tasks [7]. STL is interpreted over continuous-time signals and allows to formulate complex tasks that can have, opposed

This work was supported in part by the Swedish Research Council (VR), the European Research Council (ERC), the Swedish Foundation for Strategic Research (SSF), the EU H2020 Co4Robots project, the SRA ICT TNG project STaRT, and the Knut and Alice Wallenberg Foundation (KAW).

The authors are with the Department of Automatic Control, School of Electrical Engineering and Computer Science, Royal Institute of Technology (KTH), 100 44 Stockholm, Sweden. llindem@kth.se (L. Lindemann), dimos@kth.se (D.V. Dimarogonas)

to LTL, strict deadlines. The control of systems under STL tasks is known to be difficult and has been considered in [8], [9] where computationally demanding mixed-integer linear programs are obtained. Reinforcement learning-based control strategies for STL have been derived in [10].

Barrier functions guarantee forward invariance of a set with respect to the system dynamics of an unforced system [11]. Control barrier functions [12], [13], [14], on the other hand, consider systems with control inputs that are used to render a set forward invariant. Multi-objective control barrier functions have been considered in [15], [16]. Higher order and time-varying control barrier functions are used in [17] to establish forward invariance of a time-varying set.

In this paper, we consider systems under STL tasks. The main contribution is a control strategy that offers a good trade-off between computational efficiency and expressivity of the STL fragment under consideration. The proposed control strategy does not rely on an abstraction of the system and does not result into a separation of planning and continuous plan execution. In our previous work [18], we already proposed continuous-time feedback control laws that account for the aforementioned issues. In this paper, we continue along this line of research; however, with a different control approach that allows to formulate a broader range of complex tasks. In a first step, we define time-varying control barrier functions in a similar fashion as in [17]. In a second step, we use the time-varying control barrier functions to satisfy STL tasks. The temporal properties of the control barrier functions hence need to be designed in a way that accounts for the STL semantics. The resulting controller is given by a switching strategy between a computationally-efficient convex quadratic program and a local feedback control law.

Sec. II states preliminaries and the problem definition. Sec. III presents our proposed solution, which is verified by simulations in Sec. IV. Conclusions are given in Sec. V.

## II. PRELIMINARIES AND PROBLEM DEFINITION

Scalars and column vectors are denoted by non-bold letters  $x$  and bold letters  $\mathbf{x}$ , respectively;  $\mathbb{R}$  are the real numbers, while  $\mathbb{R}^n$  is the  $n$ -dimensional real vector space. The non-negative and positive real numbers are  $\mathbb{R}_{\geq 0}$  and  $\mathbb{R}_{> 0}$ , respectively. A class  $\mathcal{K}$  function  $\alpha : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$  is a continuous and strictly increasing function with  $\alpha(0) = 0$ . We denote by  $\mathbf{0}_n$  the  $n$ -dimensional zero vector.

Let  $\mathbf{x} \in \mathbb{R}^n$  and  $\mathbf{u} \in \mathcal{U} \subseteq \mathbb{R}^m$  be the state and input of a nonlinear input-affine control system

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})\mathbf{u}, \quad (1)$$

where the functions  $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$  and  $g : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times m}$  are locally Lipschitz continuous. For a reason that becomes apparent in Section III-B, we consider solutions to (1) with the initial condition  $\mathbf{x}(t_0)$  over fixed and compact time intervals  $[t_0, t_1] \subset \mathbb{R}_{\geq 0}$ . Given a control signal  $\mathbf{u} : [t_0, t_1] \rightarrow \mathcal{U}$ , the signal  $\mathbf{x} : [t_0, t_1] \rightarrow \mathbb{R}^n$  is a solution to (1) if  $\mathbf{x}$  is absolutely continuous and  $\mathbf{x}(t)$  satisfies (1) for all  $t \in [t_0, t_1]$ .

### A. Signal Temporal Logic (STL)

Signal temporal logic [7] is a predicate logic consisting of predicates  $\mu$  that are obtained after evaluation of a predicate function  $h : \mathbb{R}^n \rightarrow \mathbb{R}$  as  $\mu := \begin{cases} \text{True} & \text{if } h(\mathbf{x}) \geq 0 \\ \text{False} & \text{if } h(\mathbf{x}) < 0. \end{cases}$

The STL syntax defines an STL formula  $\phi$  and is given by

$$\phi ::= \text{True} \mid \mu \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \phi_1 U_{[a,b]} \phi_2,$$

where  $\phi_1, \phi_2$  are STL formulas and  $a, b \in \mathbb{R}_{\geq 0}$  with  $a \leq b$ . The satisfaction relation  $(\mathbf{x}, t) \models \phi$  denotes if the signal  $\mathbf{x} : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^n$ , e.g., a solution of (1), satisfies  $\phi$  at time  $t$ .

*Definition 1 (STL Semantics):* For a signal  $\mathbf{x} : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^n$ , the STL semantics [7] are recursively given by:

$$\begin{aligned} (\mathbf{x}, t) \models \mu & \quad \Leftrightarrow h(\mathbf{x}(t)) \geq 0 \\ (\mathbf{x}, t) \models \neg\phi & \quad \Leftrightarrow \neg((\mathbf{x}, t) \models \phi) \\ (\mathbf{x}, t) \models \phi_1 \wedge \phi_2 & \quad \Leftrightarrow (\mathbf{x}, t) \models \phi_1 \wedge (\mathbf{x}, t) \models \phi_2 \\ (\mathbf{x}, t) \models \phi_1 U_{[a,b]} \phi_2 & \quad \Leftrightarrow \exists t_1 \in [t+a, t+b] \text{ s.t. } (\mathbf{x}, t_1) \models \phi_2 \\ & \quad \wedge \forall t_2 \in [t, t_1], (\mathbf{x}, t_2) \models \phi_1 \\ (\mathbf{x}, t) \models F_{[a,b]} \phi & \quad \Leftrightarrow \exists t_1 \in [t+a, t+b] \text{ s.t. } (\mathbf{x}, t_1) \models \phi \\ (\mathbf{x}, t) \models G_{[a,b]} \phi & \quad \Leftrightarrow \forall t_1 \in [t+a, t+b], (\mathbf{x}, t_1) \models \phi. \end{aligned}$$

### B. Time-varying Control Barrier Functions (CBF)

Similarly to [17], we consider time-varying control barrier functions. Therefore, define the differentiable function  $\mathbf{b} : \mathcal{D} \times [t_0, t_1] \rightarrow \mathbb{R}$ , where  $\mathcal{D} \subseteq \mathbb{R}^n$ , and the set

$$\mathcal{C}(t) := \{\mathbf{x} \in \mathcal{D} \mid \mathbf{b}(\mathbf{x}, t) \geq 0\},$$

which explicitly depends on time. Next, conditions are given under which  $\mathbf{b}(\mathbf{x}, t)$  is a candidate control barrier function.

*Definition 2 (Candidate Control Barrier Function):* A differentiable function  $\mathbf{b} : \mathcal{D} \times [t_0, t_1] \rightarrow \mathbb{R}$  is a candidate control barrier function if for each  $\mathbf{x}_0 \in \mathcal{C}(t_0)$ , there exists an absolutely continuous function  $\mathbf{x} : [t_0, t_1] \rightarrow \mathbb{R}^n$  with  $\mathbf{x}(t_0) := \mathbf{x}_0$  such that  $\mathbf{x}(t) \in \mathcal{C}(t)$  for all  $t \in [t_0, t_1]$ .

A time-varying candidate control barrier function implies that  $\mathcal{C}(t)$  is non-empty for each  $t \in [t_0, t_1]$ , which is the standard assumption for a time-independent function to be a candidate control barrier function [12], [13]. For time-varying candidate control barrier functions, however, the set  $\mathcal{C}(t)$  needs to be such that there exists a function  $\mathbf{x} : [t_0, t_1] \rightarrow \mathbb{R}^n$  with  $\mathbf{x}(t) \in \mathcal{C}(t)$  for all  $t \in [t_0, t_1]$ . This function  $\mathbf{x}$  needs to be absolutely continuous so that it may possibly be a solution to (1) for a control signal  $\mathbf{u} : [t_0, t_1] \rightarrow \mathcal{U}$ .

*Remark 1:* The assumption of the existence of a function  $\mathbf{x} : [t_0, t_1] \rightarrow \mathbb{R}^n$  can be illustrated as follows. Assume that  $\mathcal{C}(t)$  is the union of two sets  $\mathcal{C}_1(t)$  and  $\mathcal{C}_2(t)$  that are not

connected for all  $t \in [t_0, t_1]$ , i.e.,  $\mathcal{C}(t) := \mathcal{C}_1(t) \cup \mathcal{C}_2(t)$  with  $\mathcal{C}_1(t) \cap \mathcal{C}_2(t) = \emptyset$  for all  $t \in [t_0, t_1]$ . If now  $\mathcal{C}_1(t)$  shrinks so that eventually  $\mathcal{C}_1(t) = \emptyset$  for all  $t \geq t'$ , where  $t' \in (t_0, t_1]$  is some constant, while  $\mathcal{C}_2(t) \neq \emptyset$  for all  $t \in [t_0, t_1]$ , then  $\mathcal{C}(t)$  is non-empty for all  $t \in [t_0, t_1]$ . If, however,  $\mathbf{x}_0 \in \mathcal{C}_1(t_0)$ , there exists no solution  $\mathbf{x} : [t_0, t_1] \rightarrow \mathbb{R}^n$  to (1) with initial condition  $\mathbf{x}(t_0) = \mathbf{x}_0$  so that  $\mathbf{x}(t) \in \mathcal{C}(t)$  for all  $t \in [t_0, t_1]$ .

We now define the notion of a valid control barrier function.

*Definition 3 (Valid Control Barrier Function):* A candidate control barrier function  $\mathbf{b}(\mathbf{x}, t)$  is a valid control barrier function for (1) if there exists a locally Lipschitz continuous class  $\mathcal{K}$  function  $\alpha$  such that, for all  $(\mathbf{x}, t) \in \mathcal{C}(t) \times [t_0, t_1]$ ,

$$\sup_{\mathbf{u} \in \mathcal{U}} \frac{\partial \mathbf{b}(\mathbf{x}, t)^T}{\partial \mathbf{x}} (f(\mathbf{x}) + g(\mathbf{x})\mathbf{u}) + \frac{\partial \mathbf{b}(\mathbf{x}, t)}{\partial t} \geq -\alpha(\mathbf{b}(\mathbf{x}, t)).$$

Definition 3 provides a first order condition, while [17, Definition 1] provides a higher order condition that allows  $\mathbf{b}(\mathbf{x}, t)$  to have a relative degree greater or equal than one. Definition 3, however, provides a more general first order condition by using a class  $\mathcal{K}$  function  $\alpha$ . Next, define  $S_{\mathbf{u}}(\mathbf{x}, t) := \{\mathbf{u} \in \mathcal{U} \mid \frac{\partial \mathbf{b}(\mathbf{x}, t)^T}{\partial \mathbf{x}} (f(\mathbf{x}) + g(\mathbf{x})\mathbf{u}) + \frac{\partial \mathbf{b}(\mathbf{x}, t)}{\partial t} \geq -\alpha(\mathbf{b}(\mathbf{x}, t))\}$  and forward invariance of  $\mathcal{C}(t)$  over the time interval  $[t_0, t_1]$ .

*Definition 4 (Forward Invariance):* A set  $\mathcal{C}(t)$  is forward invariant for a given control law  $\mathbf{u}$  if for each  $\mathbf{x}_0 \in \mathcal{C}(t_0)$  there exists a unique solution  $\mathbf{x} : [t_0, t_1] \rightarrow \mathbb{R}^n$  to (1) with  $\mathbf{x}(t_0) = \mathbf{x}_0$  such that  $\mathbf{x}(t) \in \mathcal{C}(t)$  for all  $t \in [t_0, t_1]$ .

The following lemma from [15] will be used later.

*Lemma 1:* Let  $\alpha : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$  be a locally Lipschitz continuous class  $\mathcal{K}$  function and  $\eta : [t_0, t_1] \rightarrow \mathbb{R}$  be an absolutely continuous function. If  $\dot{\eta}(t) \geq -\alpha(\eta(t))$  for every  $t \in [t_0, t_1]$ , and  $\eta(t_0) \geq 0$ , then  $\eta(t) \geq 0$  for all  $t \in [t_0, t_1]$ .

### C. Problem Formulation

In this paper, the following STL fragment is considered

$$\psi ::= \top \mid \mu \mid \neg\mu \mid \psi_1 \wedge \psi_2 \quad (2a)$$

$$\phi ::= G_{[a,b]} \psi \mid F_{[a,b]} \psi \mid \psi_1 U_{[a,b]} \psi_2 \mid \phi_1 \wedge \phi_2 \quad (2b)$$

where  $\psi_1, \psi_2$  are formulas of class  $\psi$  given in (2a), whereas  $\phi_1$  and  $\phi_2$  are formulas of class  $\phi$  given in (2b). To guarantee bounded trajectories, we pose the following assumption.

*Assumption 1:* For the formula  $\phi$  in (2b), there exists a  $C \geq 0$  such that  $(\mathbf{x}, 0) \models \phi$  implies  $\|\mathbf{x}(t)\| \leq C$  for all  $t \geq 0$ .

Note that the above assumption is not restrictive in practice and rather desired since a given formula  $\phi_{\text{task}}$  can always be combined with  $\phi_{\text{Ass.1}} := G_{[0, \infty)}(\|\mathbf{x}\| \leq C)$  for an arbitrarily large  $C$  so that  $\phi := \phi_{\text{task}} \wedge \phi_{\text{Ass.1}}$  satisfies Assumption 1. The problem under consideration is now given as follows.

*Problem 1:* Consider the dynamical system in (1) and an STL formula  $\phi$  as in (2b). Derive a control law  $\mathbf{u}(\mathbf{x}, t)$  so that the solution  $\mathbf{x} : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^n$  to (1) with initial condition  $\mathbf{x}(0)$  is such that  $(\mathbf{x}, 0) \models \phi$ .

## III. CONTROL APPROACH

In this section, we show that time-varying control barrier functions can be used to satisfy STL formulas by leveraging the temporal properties of  $\mathbf{b}(\mathbf{x}, t)$ . Our first result follows a similar proof as in [15] and is in line with the results of [17].

*Theorem 1:* Assume that  $\mathbf{u}(\mathbf{x}, t) \in S_{\mathbf{u}}(\mathbf{x}, t)$  is locally Lipschitz continuous in  $\mathbf{x}$  and piecewise continuous in  $t$  and that the (consequently) unique solutions to (1) are defined over  $[t_0, t_1]$ . Then the set  $\mathcal{C}(t)$  is forward invariant for the control law  $\mathbf{u}(\mathbf{x}, t)$  if  $\mathfrak{b}(\mathbf{x}, t)$  is a valid control barrier function.

*Proof:* Assume that  $\mathbf{x}(t_0) \in \mathcal{C}(t_0)$ , which implies that  $\mathfrak{b}(\mathbf{x}(t_0), t_0) \geq 0$ . By assumption  $\mathfrak{b}(\mathbf{x}, t)$  is a valid control barrier function and hence  $\mathbf{u}(\mathbf{x}, t) \in S_{\mathbf{u}}(\mathbf{x}, t) \neq \emptyset$  results in a solution  $\mathbf{x}$  to (1) with initial condition  $\mathbf{x}(t_0)$  that satisfies  $\dot{\mathfrak{b}}(\mathbf{x}(t), t) \geq -\alpha(\mathfrak{b}(\mathbf{x}(t), t))$  for all  $t \in [t_0, t_1]$ . Note that the solution  $\mathbf{x} : [t_0, t_1] \rightarrow \mathbb{R}^n$  exists by assumption. Now, let  $\eta(t) := \mathfrak{b}(\mathbf{x}(t), t)$ . Using Lemma 1, it can be concluded that  $\eta(t) \geq 0$  for all  $t \in [t_0, t_1]$ . It hence holds that  $\mathcal{C}(t)$  is forward invariant since  $\mathfrak{b}(\mathbf{x}(t), t) \geq 0$  implies  $\mathbf{x}(t) \in \mathcal{C}(t)$ . ■

### A. Control Barrier Functions for STL Tasks

Time-varying control barrier functions can now be used to satisfy STL tasks. We first pose the following assumption, which makes the system (1) feedback equivalent to  $\dot{\mathbf{x}} = \mathbf{u}$ .

*Assumption 2:* Let  $\mathcal{U} := \mathbb{R}^m$  and  $g(\mathbf{x})$  in (1) be such that  $g(\mathbf{x})g(\mathbf{x})^T$  is positive definite for all  $\mathbf{x} \in \mathcal{D}$ .

The main idea is next demonstrated in two examples.

*Example 1:* Consider the formula  $\phi := F_{[5,15]}(\|\mathbf{x} - [10 \ 0]^T\| \leq 5)$  and let  $t_0 := 0$  and  $t_1 := 15$ . Note that the corresponding predicate function is  $h(\mathbf{x}) := 5 - \|\mathbf{x} - [10 \ 0]^T\|$ . Also consider, without loss of generality, an initial condition  $\mathbf{x}(0) := [0 \ 0]^T$ . For the candidate control barrier function  $\mathfrak{b}(\mathbf{x}, t) := \gamma(t) - \|\mathbf{x} - [10 \ 0]^T\|$  with  $\gamma(t) := -\frac{5}{15}t + 10$ , it holds that  $\mathfrak{b}(\mathbf{x}(0), 0) = 0$  and hence  $\mathbf{x}(0) \in \mathcal{C}(0)$ . If there exists a control law  $\mathbf{u}(\mathbf{x}, t)$  so that the solution to (1) satisfies  $\mathfrak{b}(\mathbf{x}(t), t) \geq 0$  for all  $t \in [t_0, t_1]$ , e.g., when  $\mathfrak{b}(\mathbf{x}, t)$  is a valid control barrier function, then  $(\mathbf{x}, 0) \models \phi$  follows. Note therefore that  $\gamma(t_1) = 5$  and hence  $\mathfrak{b}(\mathbf{x}, t_1) = 5 - \|\mathbf{x} - [10 \ 0]^T\| = h(\mathbf{x})$ . This means that  $\mathfrak{b}(\mathbf{x}(t_1), t_1) \geq 0$  implies that  $\|\mathbf{x}(t_1) - [10 \ 0]^T\| \leq 5$ , which yields  $(\mathbf{x}, 0) \models \phi$  according to Definition 1.

We note that we later provide a control law  $\mathbf{u}(\mathbf{x}, t)$  so that the solution to (1) satisfies  $\mathfrak{b}(\mathbf{x}(t), t) \geq 0$  for all  $t \in [t_0, t_1]$ . Further note that  $\mathfrak{b}(\mathbf{x}, t)$  is associated with  $h(\mathbf{x})$  and ensures a temporal behavior that leads to satisfaction of  $\phi$ . In order to use conjunctions as in the semantics in (2), while at the same time avoiding nonsmooth analysis as in [15], [16], a smooth under-approximation of the min-operator is used. For  $p$  candidate control barrier functions  $\mathfrak{b}_i(\mathbf{x}, t)$ , note that

$$\min_{i \in \{1, \dots, p\}} \mathfrak{b}_i(\mathbf{x}, t) \approx -\ln \left( \sum_{i=1}^p \exp(-\mathfrak{b}_i(\mathbf{x}, t)) \right),$$

which satisfies some useful properties as shown next.

*Lemma 2:* Consider a conjunction of  $p$  candidate control barrier functions  $\mathfrak{b}_i(\mathbf{x}, t)$ . Then, it holds that

$$-\ln \left( \sum_{i=1}^p \exp(-\mathfrak{b}_i(\mathbf{x}, t)) \right) \leq \min_{i \in \{1, \dots, p\}} \mathfrak{b}_i(\mathbf{x}, t).$$

*Proof:* From [19, p.72], we know that

$$\max_{i \in \{1, \dots, p\}} \zeta_i \leq \ln \left( \sum_{i=1}^p \exp(\zeta_i) \right)$$

for  $\zeta_1, \dots, \zeta_p \in \mathbb{R}$ . By substituting  $\zeta_i := -\mathfrak{b}_i(\mathbf{x}, t)$  and since  $\max_{i \in \{1, \dots, p\}} \zeta_i = -\min_{i \in \{1, \dots, p\}} (-\zeta_i)$ , the result holds. ■

Lemma 2 is useful for the following reason: let  $\mathfrak{b}(\mathbf{x}, t) := -\ln(\exp(-\mathfrak{b}_1(\mathbf{x}, t)) + \exp(-\mathfrak{b}_2(\mathbf{x}, t)))$ , then  $\mathfrak{b}(\mathbf{x}, t) \geq 0$  implies  $\mathfrak{b}_1(\mathbf{x}, t) \geq 0$  and  $\mathfrak{b}_2(\mathbf{x}, t) \geq 0$ . In other words, the conjunction operator can be encoded by using the smooth approximation. This idea is illustrated next.

*Example 2:* Consider  $\phi := \phi_1 \wedge \phi_2$  with  $\phi_1 := F_{[5,15]}(\|\mathbf{x} - [10 \ 0]^T\| \leq 5)$  and  $\phi_2 := G_{[5,15]}(\|\mathbf{x} - [10 \ 5]^T\| \leq 10)$  with predicate functions  $h_1(\mathbf{x}) := 5 - \|\mathbf{x} - [10 \ 0]^T\|$  and  $h_2(\mathbf{x}) := 10 - \|\mathbf{x} - [10 \ 5]^T\|$ . Let again  $t_0 := 0$ ,  $t_1 := 15$ , and assume, without loss of generality, that  $\mathbf{x}(0) := [0 \ 0]^T$ . Select the candidate control barrier function  $\mathfrak{b}(\mathbf{x}, t) := -\ln(\exp(-\mathfrak{b}_1(\mathbf{x}, t)) + \exp(-\mathfrak{b}_2(\mathbf{x}, t)))$  where  $\mathfrak{b}_1(\mathbf{x}, t) := \gamma_1(t) - \|\mathbf{x} - [10 \ 0]^T\|$  and  $\mathfrak{b}_2(\mathbf{x}, t) := \gamma_2(t) - \|\mathbf{x} - [10 \ 5]^T\|$  with  $\gamma_1(t) := -t + 20$  and  $\gamma_2(t) := 11 \exp(-0.4796t) + 9$ . It holds that  $\mathfrak{b}(\mathbf{x}(0), 0) \geq 0$ ,  $\mathfrak{b}_1(\mathbf{x}, t_1) = 5 - \|\mathbf{x} - [10 \ 0]^T\| = h_1(\mathbf{x})$ , and  $\mathfrak{b}_2(\mathbf{x}, t') \leq h_2(\mathbf{x})$  for all  $t' \in [5, 15]$ . If there exists a control law  $\mathbf{u}(\mathbf{x}, t)$  so that the solution to (1) satisfies  $\mathfrak{b}(\mathbf{x}(t), t) \geq 0$  for all  $t \in [t_0, t_1]$ , e.g., when  $\mathfrak{b}(\mathbf{x}, t)$  is a valid control barrier function, then  $(\mathbf{x}, 0) \models \phi$ . According to Lemma 2 it follows that  $\mathfrak{b}_1(\mathbf{x}(t_1), t_1) \geq 0$  and  $\mathfrak{b}_2(\mathbf{x}(t'), t') \geq 0$  for all  $t' \in [5, 15]$ , which implies that  $\|\mathbf{x}(t_1) - [10 \ 0]^T\| \leq 5$  and  $\|\mathbf{x}(t') - [10 \ 5]^T\| \leq 10$  for all  $t' \in [5, 15]$ . This leads to  $(\mathbf{x}, 0) \models \phi$  due to the STL semantics in Definition 1.

We next formulate conditions that the function  $\mathfrak{b}(\mathbf{x}, t)$  has to satisfy to account for the semantics of an STL formula  $\phi$  in the fragment (2). These conditions on  $\mathfrak{b}(\mathbf{x}, t)$  are discussed in three steps (Step A, B, and C) and it always needs to hold that  $\mathfrak{b}(\mathbf{x}, t)$  is such that  $\mathfrak{b}(\mathbf{x}(0), 0) \geq 0$ . For negations on predicates  $\neg\mu$  as in (2a), let the corresponding predicate function be  $-h(\mathbf{x})$ . Furthermore, let  $h_1(\mathbf{x})$ ,  $h_2(\mathbf{x})$ ,  $h_3(\mathbf{x})$ , and  $h_4(\mathbf{x})$  correspond to  $\mu_1$ ,  $\mu_2$ ,  $\mu_3$ , and  $\mu_4$ , respectively.

**Step A)** Let us investigate single temporal operators (always, eventually, and until) in (2b) that *do not* contain conjunctions in the arguments, i.e.,  $G_{[a,b]}\mu_1$ ,  $F_{[a,b]}\mu_1$ , and  $\mu_1 U_{[a,b]}\mu_2$ . For  $G_{[a,b]}\mu_1$ , select  $\mathfrak{b}(\mathbf{x}, t)$  so that  $\mathfrak{b}(\mathbf{x}, t') \leq h_1(\mathbf{x})$  for all  $t' \in [a, b]$  (see Example 2). For  $F_{[a,b]}\mu_1$ , select  $\mathfrak{b}(\mathbf{x}, t)$  so that  $\mathfrak{b}(\mathbf{x}, t') \leq h_1(\mathbf{x})$  for some  $t' \in [a, b]$  (see Example 1 and 2). For  $\mu_1 U_{[a,b]}\mu_2$ , select  $\mathfrak{b}(\mathbf{x}, t) := -\ln(\exp(-\mathfrak{b}_1(\mathbf{x}, t)) + \exp(-\mathfrak{b}_2(\mathbf{x}, t)))$  so that  $\mathfrak{b}_2(\mathbf{x}, t') \leq h_2(\mathbf{x})$  for some  $t' \in [a, b]$  and  $\mathfrak{b}_1(\mathbf{x}, t'') \leq h_1(\mathbf{x})$  for all  $t'' \in [a, t']$ . Note that the conditions on  $\mathfrak{b}(\mathbf{x}, t)$  for an until operator are a conjunction of the conditions on  $\mathfrak{b}_1(\mathbf{x}, t)$  and  $\mathfrak{b}_2(\mathbf{x}, t)$  for an always and an eventually operator, respectively.

**Step B)** Let us now consider single temporal operators (always, eventually, and until) in (2b) that *do* contain conjunctions in the arguments, i.e.,  $G_{[a,b]}\psi_1$ ,  $F_{[a,b]}\psi_1$ , and  $\psi_1 U_{[a,b]}\psi_2$  where  $\psi_1$  and  $\psi_2$  may contain a conjunction of predicates as in (2a). Assume, without loss of generality,  $\psi_1 := \mu_1 \wedge \mu_2$  and  $\psi_2 := \mu_3 \wedge \mu_4$ . For  $G_{[a,b]}\psi_1$ , select  $\mathfrak{b}(\mathbf{x}, t) := -\ln(\exp(-\mathfrak{b}_1(\mathbf{x}, t)) + \exp(-\mathfrak{b}_2(\mathbf{x}, t)))$  so that  $\mathfrak{b}_1(\mathbf{x}, t') \leq h_1(\mathbf{x})$  and  $\mathfrak{b}_2(\mathbf{x}, t') \leq h_2(\mathbf{x})$  for all  $t' \in [a, b]$ . For  $F_{[a,b]}\psi_1$ , select  $\mathfrak{b}(\mathbf{x}, t) := -\ln(\exp(-\mathfrak{b}_1(\mathbf{x}, t)) + \exp(-\mathfrak{b}_2(\mathbf{x}, t)))$  so that  $\mathfrak{b}_1(\mathbf{x}, t') \leq h_1(\mathbf{x})$  and  $\mathfrak{b}_2(\mathbf{x}, t') \leq h_2(\mathbf{x})$  for some  $t' \in [a, b]$ .

$h_2(\mathbf{x})$  for some  $t' \in [a, b]$ . For  $\psi_1 U_{[a,b]} \psi_2$ , select  $\mathbf{b}(\mathbf{x}, t) := -\ln(\sum_{i=1}^4 \exp(-\mathbf{b}_i(\mathbf{x}, t)))$  so that  $\mathbf{b}_3(\mathbf{x}, t') \leq h_3(\mathbf{x})$  and  $\mathbf{b}_4(\mathbf{x}, t') \leq h_4(\mathbf{x})$  for some  $t' \in [a, b]$  and  $\mathbf{b}_1(\mathbf{x}, t'') \leq h_1(\mathbf{x})$  and  $\mathbf{b}_2(\mathbf{x}, t'') \leq h_2(\mathbf{x})$  for all  $t'' \in [a, t']$ .

Step C) Consider conjunctions of temporal operators as discussed in Step A and B. For instance, consider  $(G_{[a_1, b_1]} \psi_1) \wedge (F_{[a_2, b_2]} \psi_2) \wedge (\psi_3 U_{[a_3, b_3]} \psi_4)$ . Let  $\mathbf{b}(\mathbf{x}, t) := -\ln(\sum_{i=1}^3 \exp(-\mathbf{b}_i(\mathbf{x}, t)))$  where  $\mathbf{b}_1(\mathbf{x}, t)$ ,  $\mathbf{b}_2(\mathbf{x}, t)$ , and  $\mathbf{b}_3(\mathbf{x}, t)$  are associated with one temporal operator each and constructed as in Step A and B. Note that the available freedom in designing  $\mathbf{b}(\mathbf{x}, t)$  according to the above conditions results in a non-unique constructive procedure. We omit specific construction rules of the functions  $\mathbf{b}_i(\mathbf{x}, t)$  due to space limitations and refer to Examples 1 and 2 instead.

### B. A Hybrid Deletion Mechanism to overcome Conservatism

The function  $\mathbf{b}(\mathbf{x}, t)$  obtained in the previous section has the general form  $\mathbf{b}(\mathbf{x}, t) := -\ln(\sum_{i=1}^p \exp(-\mathbf{b}_i(\mathbf{x}, t)))$  where we denote by  $p$  the number of functions  $\mathbf{b}_i(\mathbf{x}, t)$  that arise due to the steps A, B, and C. Note that each  $\mathbf{b}_i(\mathbf{x}, t)$  with  $i \in \{1, \dots, p\}$  corresponds to either an always, eventually, or until operator with a corresponding time interval  $[a_i, b_i]$ . However, the function  $\mathbf{b}(\mathbf{x}, t)$  may be overly conservative and hence not a candidate control barrier function due to a large number of conjunctions. For instance, consider the formula  $\phi := F_{[5, 15]}(\|\mathbf{x}\| \leq 10) \wedge G_{[55, 60]}(\|\mathbf{x} - [100 \ 100]^T\| \leq 10)$ , where  $\mathbf{b}_1(\mathbf{x}, t)$  and  $\mathbf{b}_2(\mathbf{x}, t)$  are associated with the eventually and always operator, respectively. Clearly, the function  $\mathbf{b}(\mathbf{x}, t) := -\ln(\exp(-\mathbf{b}_1(\mathbf{x}, t)) + \exp(-\mathbf{b}_2(\mathbf{x}, t)))$  can not be a candidate control barrier function since  $\mathcal{C}(t) = \emptyset$  for all  $t \geq 55$  when  $\mathbf{b}_1(\mathbf{x}, t)$  is designed as in Example 1 with a decreasing function  $\gamma_1(t)$ . However, if  $\mathbf{b}(\mathbf{x}, t) := -\ln(\exp(-\mathbf{b}_1(\mathbf{x}, t)) + \exp(-\mathbf{b}_2(\mathbf{x}, t)))$  is considered within  $t \in [0, 15]$  and  $\mathbf{b}(\mathbf{x}, t) := \mathbf{b}_2(\mathbf{x}, t)$  is considered for  $t \in [15, 60]$ , then each function can be constructed so that they are candidate control barrier functions for the intervals  $[0, 15]$  and  $[15, 60]$ , respectively.

To make the approach less conservative, we hence deactivate the functions  $\mathbf{b}_i(\mathbf{x}, t)$  when they are not needed anymore, i.e., when the corresponding temporal operators are satisfied. For each temporal operator, the associated  $\mathbf{b}_i(\mathbf{x}, t)$  is deactivated at  $t = b_i$ . The function  $\mathbf{b}(\mathbf{x}, t) := -\ln(\sum_{i=1}^p \exp(-\mathbf{b}_i(\mathbf{x}, t)))$  is now piecewise continuous in  $t$  with switches at exactly those times where single barrier functions  $\mathbf{b}_i(\mathbf{x}, t)$  are deactivated, hence inducing a switching sequence, which we denote by  $\{t_0 := 0, t_1, t_2, \dots\}$ . This switching sequence defines a hybrid time domain as in [20, Chapter 2.2] and justifies the choice of solutions over  $[t_0, t_1]$  that we have adopted in the beginning. Note that this switching sequence is known in advance due to knowledge of  $[a_i, b_i]$ . In other words, at time  $t \geq t_j$  the next switch occurs at  $t_{j+1} := \operatorname{argmin}_{b_i \in \{b_1, \dots, b_p\}} \zeta(b_i, t)$  where  $\zeta(b_i, t) := b_i - t$  if  $b_i - t > 0$  and  $\zeta(b_i, t) := \infty$  otherwise.

### C. Efficient Solutions via Convex Quadratic Programming

Similarly to [12], [13], we now formulate a quadratic program that renders  $\mathcal{C}(t)$  forward invariant when  $\mathbf{b}(\mathbf{x}, t)$  is a

valid control barrier function. Therefore, consider

$$\begin{aligned} & \min_{\hat{\mathbf{u}} \in \mathcal{U}} \hat{\mathbf{u}}^T Q \hat{\mathbf{u}} & (3a) \\ \text{s.t. } & \frac{\partial \mathbf{b}(\mathbf{x}, t)}{\partial \mathbf{x}}^T (f(\mathbf{x}) + g(\mathbf{x})\hat{\mathbf{u}}) + \frac{\partial \mathbf{b}(\mathbf{x}, t)}{\partial t} \geq -\alpha(\mathbf{b}(\mathbf{x}, t)) & (3b) \end{aligned}$$

where  $Q \in \mathbb{R}^{m \times m}$  is a positive semi-definite matrix and where, for instance,  $\alpha(r) := r^2$ . This optimization problem is always feasible if  $\mathbf{b}(\mathbf{x}, t)$  is a valid control barrier function and needs to be solved in a sampled-data fashion. The optimization problem is convex and only depends on  $m$  decision variables.

The function  $\mathbf{b}(\mathbf{x}, t)$  has a relative degree greater than one if  $\frac{\partial \mathbf{b}(\mathbf{x}, t)}{\partial \mathbf{x}}^T g(\mathbf{x}) = \mathbf{0}_m^T$  for some  $(\mathbf{x}, t) \in \mathcal{C}(t) \times [t_0, t_1]$ . This implies that the constraint (3b) is only satisfied when  $\frac{\partial \mathbf{b}(\mathbf{x}, t)}{\partial t}^T f(\mathbf{x}) + \frac{\partial \mathbf{b}(\mathbf{x}, t)}{\partial t} \geq -\alpha(\mathbf{b}(\mathbf{x}, t))$ , which may in general not hold. Note that  $\frac{\partial \mathbf{b}(\mathbf{x}, t)}{\partial \mathbf{x}}^T g(\mathbf{x}) = \mathbf{0}_m^T$  if and only if  $g(\mathbf{x})^T \frac{\partial \mathbf{b}(\mathbf{x}, t)}{\partial \mathbf{x}} = \mathbf{0}_m$ . It holds that  $\operatorname{rank}(g(\mathbf{x})^T) = \operatorname{rank}(g(\mathbf{x})) = n$  due to Assumption 2 and consequently the nullspace of  $g(\mathbf{x})^T$  is empty due to the rank-nullity theorem. This means that  $\frac{\partial \mathbf{b}(\mathbf{x}, t)}{\partial \mathbf{x}}^T g(\mathbf{x}) = \mathbf{0}_m^T$  if and only if  $\frac{\partial \mathbf{b}(\mathbf{x}, t)}{\partial \mathbf{x}} = \mathbf{0}_n$ . Therefore, we next pose an assumption on  $\mathbf{b}(\mathbf{x}, t)$  via the set  $\mathcal{C}(t)$  to still be able to use the quadratic program in (3).

*Assumption 3:* For  $t_j \leq t \leq t_{j+1}$ , the set  $\mathcal{C}(t)$  is either non-decreasing, i.e.,  $\mathcal{C}(t_1) \subseteq \mathcal{C}(t_2)$  for  $t_j \leq t_1 < t_2 \leq t_{j+1}$ , or non-increasing, i.e.,  $\mathcal{C}(t_1) \supseteq \mathcal{C}(t_2)$  for  $t_j \leq t_1 < t_2 \leq t_{j+1}$ . In the latter case, it additionally holds that  $\frac{\partial \mathbf{b}(\mathbf{x}, t)}{\partial \mathbf{x}} \neq \mathbf{0}_n$  for all  $(\mathbf{x}, t) \in \mathcal{C}(t) \setminus \mathcal{C}(t_{j+1}) \times [t_j, t_{j+1}]$ , which means that  $\mathbf{b}(\mathbf{x}, t)$  is only allowed to have critical points in  $\mathcal{C}(t_{j+1})$ .

We next state the main theorem of this paper.

*Theorem 2:* Assume that the formula  $\phi$  as given in (2b) is encoded in  $\mathbf{b}(\mathbf{x}, t)$  by following the conditions given in Step A, B, and C in Section III-A (as applied in Examples 1 and 2) and that Assumptions 1 – 3 hold. If  $\mathbf{b}(\mathbf{x}, t)$  is a candidate control barrier function for each time interval  $[t_j, t_{j+1}]$ , then

$$\mathbf{u}(\mathbf{x}, t) := \begin{cases} -g(\mathbf{x})^T (g(\mathbf{x})g(\mathbf{x})^T)^{-1} f(\mathbf{x}), & \\ \text{if } \frac{\partial \mathbf{b}(\mathbf{x}, t)}{\partial \mathbf{x}} = \mathbf{0}_n \text{ for some } t' \in [t_j, t] & \\ \hat{\mathbf{u}} \text{ from (3)}, & \text{otherwise} \end{cases}$$

for  $t_j \leq t < t_{j+1}$  leads to  $(\mathbf{x}, 0) \models \phi$ .

*Proof:* According to the assumptions,  $\mathbf{b}(\mathbf{x}, t)$  is a candidate control barrier function for each time interval  $[t_j, t_{j+1}]$ . Note that at switches and by deleting single control barrier functions  $\mathbf{b}_i(\mathbf{x}, t)$  from  $\mathbf{b}(\mathbf{x}, t) := -\ln(\sum_{i=1}^p \exp(-\mathbf{b}_i(\mathbf{x}, t)))$ , the set  $\mathcal{C}(t)$  is non-decreasing at these switches. Hence, if  $\mathbf{b}(\mathbf{x}(t_{j+1}), t_{j+1}) \in \mathcal{C}(t_{j+1})$ , then  $\mathbf{b}^+(\mathbf{x}(t_{j+1}), t_{j+1}) \in \mathcal{C}^+(t_{j+1})$  where the superscript + indicates the values after the switch. This implies that it is sufficient to ensure forward invariance of  $\mathcal{C}(t)$  for each  $[t_j, t_{j+1}]$  separately. If then  $\frac{\partial \mathbf{b}(\mathbf{x}, t)}{\partial \mathbf{x}} \neq \mathbf{0}_n$  for all  $(\mathbf{x}, t) \in \mathcal{C}(t) \times [t_j, t_{j+1}]$ , then  $\mathbf{u}(\mathbf{x}, t) = \hat{\mathbf{u}}$  from (3) renders  $\mathcal{C}(t)$  forward invariant for  $[t_j, t_{j+1}]$ . Note therefore that (3) is always feasible since there always exists a control input  $g(\mathbf{x})^T (g(\mathbf{x})g(\mathbf{x})^T)^{-1} (-f(\mathbf{x}) + \mathbf{v})$  that leads to the closed-loop dynamics  $\dot{\mathbf{x}} = \mathbf{v}$  where  $\mathbf{v}$  allows full actuation. Hence,  $\mathbf{b}(\mathbf{x}, t)$  is a valid control barrier function for

the time interval  $[t_j, t_{j+1}]$ . Note also that  $\mathcal{U} := \mathbb{R}^m$  and that  $-g(\mathbf{x})^T(g(\mathbf{x})g(\mathbf{x})^T)^{-1}$  is always well-defined due to Assumption 2. Furthermore, [12, Theorem 3] guarantees that  $\mathbf{u}(\mathbf{x}, t)$  is locally Lipschitz continuous in  $\mathbf{x}$  and piecewise continuous in  $t$  and, due to Assumption 1,  $\mathbf{x}(t)$  will remain in the compact set  $\{\mathbf{x} \in \mathcal{D} \mid \|\mathbf{x}\| \leq C\}$  ensuring solutions over  $[t_j, t_{j+1}]$  so that Theorem 1 can be applied. If, however,  $\frac{\partial \mathbf{b}(\mathbf{x}(t'), t')}{\partial \mathbf{x}} = \mathbf{0}_n$  for some  $(\mathbf{x}(t'), t') \in \mathcal{C}(t') \times [t_j, t_{j+1}]$ , the quadratic program in (3) may not be feasible. In this case, the controller switches to the local control law  $\mathbf{u}(\mathbf{x}, t) := -g(\mathbf{x})^T(g(\mathbf{x})g(\mathbf{x})^T)^{-1}f(\mathbf{x})$  for all  $t \in [t', t_{j+1}]$  so that  $\dot{\mathbf{x}}(t) = \mathbf{0}_n$ . This situation, however, only occurs for the two cases stated in Assumption 3 and hence  $\mathcal{C}(t)$  is rendered forward invariant. This follows since  $\mathbf{x}(t')$  already fulfills  $\mathbf{x}(t') \in \mathcal{C}(t)$  for all  $t \in [t', t_{j+1}]$ , i.e., no more motion is required, and since  $\mathbf{u}(\mathbf{x}, t) = \hat{\mathbf{u}}$  from (3) rendered  $\mathcal{C}(t)$  forward invariant for  $[t_j, t']$ . Consequently, the closed loop trajectory results in  $(\mathbf{x}, 0) \models \phi$  due to the condition on  $\mathbf{b}(\mathbf{x}, t)$  in Section III-A that are in line with the STL semantics in Definition 1. Note that at most one control switch occurs within  $[t_j, t_{j+1}]$  so that chattering is avoided [21]. Assumption 1 guarantees complete solutions in the sense of [20, Chapter 2.2]. ■

*Remark 2:* The second case in Assumption 3, i.e., the fact that  $\frac{\partial \mathbf{b}(\mathbf{x}, t)}{\partial \mathbf{x}} \neq \mathbf{0}_n$  for all  $(\mathbf{x}, t) \in \mathcal{C}(t) \setminus \mathcal{C}(t_{j+1}) \times [t_j, t_{j+1}]$ , does not allow formulas like  $\phi := G_{[0,15]}(\|\mathbf{x} - [50 \ 50]^T\| \geq 15) \wedge F_{[5,15]}(\|\mathbf{x} - [90 \ 90]^T\| \leq 10)$  since it holds that  $\frac{\partial \mathbf{b}(\mathbf{x}, t)}{\partial \mathbf{x}} = \mathbf{0}_n$  for some  $(\mathbf{x}, t) \in \mathcal{C}(t) \setminus \mathcal{C}(t_{j+1}) \times [t_j, t_{j+1}]$ . At these points, (3) may not be feasible. For the above formula, this happens for some  $t \in [0, 15]$  when the state  $\mathbf{x}(t)$  lies on a line connecting  $[50 \ 50]^T$  and  $[90 \ 90]^T$ . The intuition is that the two temporal operators in  $\phi$  exert a force in exactly the opposite directions with the same magnitude so that  $\frac{\partial \mathbf{b}_1(\mathbf{x}, t)}{\partial \mathbf{x}} = -\frac{\partial \mathbf{b}_2(\mathbf{x}, t)}{\partial \mathbf{x}}$ . The level curves for this specific example and an appropriately designed candidate control barrier function are shown in Fig. 1. However, a broad range of formulas can still be expressed as will be shown in the simulation section. Example 2 also shows the case where  $\frac{\partial \mathbf{b}(\mathbf{x}, t)}{\partial \mathbf{x}} = \mathbf{0}_n$  for some  $(\mathbf{x}, t) \in \mathcal{C}(15) \times [5, 15]$  so that Theorem 2 can be applied since Assumption 3 still holds.

#### IV. SIMULATIONS

Consider a team of  $M := 5$  omnidirectional robots, where each robot is denoted by  $v_i$  with  $i \in \{1, \dots, M\}$  and modeled as in [22]. The state of each robot is denoted by  $\mathbf{x}_i := [\mathbf{p}_i^T \ x_{i,3}]^T \in \mathbb{R}^3$  where  $\mathbf{p}_i := [x_{i,1} \ x_{i,2}]^T$  indicates the robot's position and where  $x_{i,3}$  indicates the robot's orientation with respect to the first coordinate. Next, the states of all robots are stacked into  $\mathbf{x} := [\mathbf{x}_1^T \ \dots \ \mathbf{x}_M^T]^T \in \mathbb{R}^{3M}$ . We further include a collision avoidance mechanism into the dynamics of each robot by using  $f(\mathbf{x}) := [f_1(\mathbf{x})^T \ \dots \ f_M(\mathbf{x})^T]^T \in \mathbb{R}^{3M}$  with  $f_i(\mathbf{x}) := [f_{i,1}(\mathbf{x}) \ f_{i,2}(\mathbf{x}) \ 0]^T$  where, for  $i \in \{1, \dots, M\}$ ,

$$f_{i,1}(\mathbf{x}) := \sum_{j=1, j \neq i}^M k_i \frac{x_{i,1} - x_{j,1}}{\|\mathbf{p}_i - \mathbf{p}_j\| + 0.000001}$$

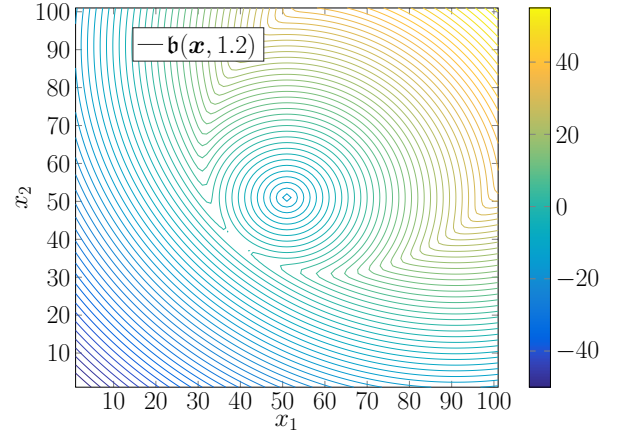


Fig. 1: Level curves of  $\mathbf{b}(\mathbf{x}, t)$  at  $t = 1.2$  showing a critical point at  $x_1 = x_2 \approx 38$ .

$$f_{i,2}(\mathbf{x}) := \sum_{j=1, j \neq i}^M k_i \frac{x_{i,2} - x_{j,2}}{\|\mathbf{p}_i - \mathbf{p}_j\| + 0.000001}$$

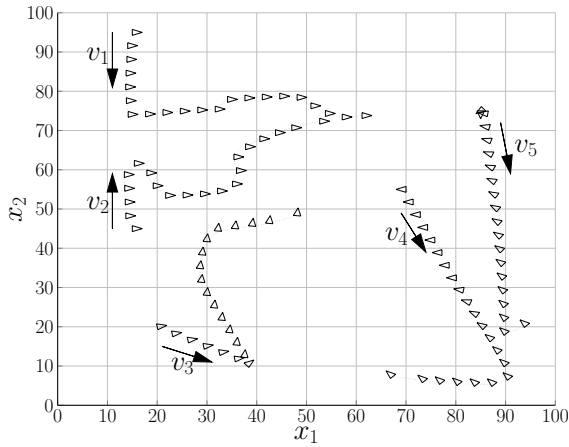
with  $k_i > 0$ . Note that  $f(\mathbf{x})$  is locally Lipschitz continuous as required. The dynamics for each robot are now given by

$$\begin{aligned} \dot{\mathbf{x}}_i &= f_i(\mathbf{x}) + \begin{bmatrix} \cos(x_{i,3}) & -\sin(x_{i,3}) & 0 \\ \sin(x_{i,3}) & \cos(x_{i,3}) & 0 \\ 0 & 0 & 1 \end{bmatrix} (B_i^T)^{-1} R_i \mathbf{u}_i \\ &= f_i(\mathbf{x}) + g_i(\mathbf{x}_i) \mathbf{u}_i \end{aligned}$$

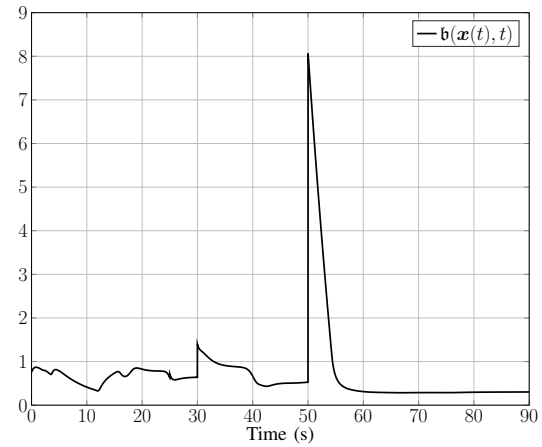
where  $\mathbf{u}_i$  is the angular velocity of the wheels,  $R_i := 0.02$  is the wheel radius, and  $B_i := \begin{bmatrix} 0 & \cos(\pi/6) & -\cos(\pi/6) \\ -1 & \sin(\pi/6) & \sin(\pi/6) \\ L_i & L_i & L_i \end{bmatrix}$  describes geometric constraints with  $L_i := 0.2$  being the radius of the robot body. By setting  $g(\mathbf{x}) := \text{diag}(g_1(\mathbf{x}_1), \dots, g_M(\mathbf{x}_M))$  and  $\mathbf{u} := [\mathbf{u}_1^T \ \dots \ \mathbf{u}_M^T]^T$ , the dynamics in (1) are obtained.

Consider the formula  $\phi := \phi_1 \wedge \phi_2 \wedge \phi_3 \wedge \phi_4$  where  $\phi_1 := F_{[10,30]}(\|\mathbf{p}_1 - \mathbf{p}_2\| \leq 10) \wedge F_{[25,50]}(\|\mathbf{p}_1 - [50 \ 75]^T\| \leq 10) \wedge \|\mathbf{p}_2 - [50 \ 65]^T\| \leq 10) \wedge F_{[10,90]}(\|\mathbf{p}_1 - \mathbf{p}_2\| \leq 10) \wedge \|\mathbf{p}_1 - [70 \ 70]^T\| \leq 10)$ , i.e.,  $\phi_1$  requires robots  $v_1$  and  $v_2$  to meet within some time intervals, e.g., to collaboratively work on a task, and also to be at specific locations within some time intervals for individual tasks. Let  $\phi_2 := (\|\mathbf{p}_3 - [40 \ 10]^T\| \leq 20) U_{[5,25]} |x_{3,3} - 90| \leq 10) \wedge F_{20,50}(\|\mathbf{p}_2 - \mathbf{p}_3\| \leq 15) \wedge G_{[70,90]}(\|\mathbf{p}_3 - [50 \ 50]^T\| \leq 10)$  define collaborative and individual tasks for robot  $v_3$  including requirements on its orientation. Furthermore, let  $\phi_3 := (\|\mathbf{p}_4 - [90 \ 10]^T\| \leq 10 \wedge |x_{4,3} - 135| \leq 10) U_{[10,50]}(\|\mathbf{p}_5 - [90 \ 10]^T\| \leq 10 \wedge |x_{4,3} - 135| \leq 10) \wedge F_{[50,90]}(\|\mathbf{p}_4 - [60 \ 10]^T\| \leq 10)$  define collaborative tasks for robot  $v_4$  and  $v_5$ ;  $\phi_3$  means that robot  $v_4$  is doing a task until robot  $v_5$  can take care of this task so that  $v_4$  can move on to its next individual task. The task  $\phi_4 := G_{[0,90]}(\|\mathbf{p}_1^T \ \mathbf{p}_2^T \ \mathbf{p}_3^T \ \mathbf{p}_4^T\|^T - [50 \ 10]^T\|_\infty \leq 50)$  is merely a safety task to remain within the workspace.

Fig. 2a shows the robot trajectory and as can be seen,  $\phi$  is satisfied. Fig. 2b shows the time evolution of  $\mathbf{b}(\mathbf{x}, t)$  where jumps occur due to the switching mechanism at the switching



(a) Robot trajectories where the triangles denote the orientation.



(b) Control barrier function  $b(\mathbf{x}(t), t)$ .

Fig. 2: Robot trajectories and time evolution of the control barrier function.

times  $t_1 = 25$ ,  $t_2 = 30$ , and  $t_3 = 50$ . Simulations have been performed on a two-core 1,8 GHz CPU with 4 GB of RAM and a sampling frequency of 100 Hz. On average, solving (3) with the MATLAB quadprog function took 5 ms.

## V. CONCLUSION

We proposed a computationally-efficient framework for control synthesis under signal temporal logic tasks. In particular, we established a link between time-varying control barrier functions and the semantics of signal temporal logic. The control barrier functions are designed such that they under-approximate the predicate functions of the signal temporal logic task in an appropriate way at some appropriately chosen points in time. The resulting control law is a switching controller between a convex quadratic program and a local feedback control law. In the future, we plan to derive robust results for the proposed method. Next steps are also to extend the proposed framework to decentralized multi-agent systems and to perform experiments. In this paper, sufficient conditions on the control barrier functions were provided without generically constructing them. Therefore, we plan to present a general construction procedure in the future that constructs the control barrier functions explicitly.

## REFERENCES

- [1] C. Baier and J.-P. Katoen, *Principles of Model Checking*, 1st ed. Cambridge, MA: The MIT Press, 2008.
- [2] G. E. Fainekos, A. Girard, H. Kress-Gazit, and G. J. Pappas, "Temporal logic motion planning for dynamic robots," *Automatica*, vol. 45, no. 2, pp. 343–352, 2009.
- [3] M. Kloetzer and C. Belta, "Automatic deployment of distributed teams of robots from temporal logic motion specifications," *IEEE Transactions on Robotics*, vol. 26, no. 1, pp. 48–61, 2010.
- [4] N. Piterman, A. Pnueli, and Y. Sa'ar, "Synthesis of reactive (1) designs," in *Proceedings of the International Workshop on VMCAI*, Charleston, SC, 2006, pp. 364–380.
- [5] H. Kress-Gazit, G. E. Fainekos, and G. J. Pappas, "Temporal-logic-based reactive mission and motion planning," *IEEE transactions on robotics*, vol. 25, no. 6, pp. 1370–1381, 2009.
- [6] R. Dimitrova and R. Majumdar, "Deductive control synthesis for alternating-time logics," in *Proceedings of the 14th Int. Conference on Embedded Software*, New Delhi, India, October 2014, p. 14.
- [7] O. Maler and D. Nickovic, "Monitoring temporal properties of continuous signals," in *Proceedings of the International Conference on FORMATS-FTRTFT*, Grenoble, France, September 2004, pp. 152–166.
- [8] V. Raman, A. Donz , M. Maasoumy, R. M. Murray, A. Sangiovanni-Vincentelli, and S. A. Seshia, "Model predictive control with signal temporal logic specifications," in *Proceedings of Conference on Decision and Control*, Los Angeles, CA, December 2014, pp. 81–87.
- [9] S. Sadraddini and C. Belta, "Robust temporal logic model predictive control," in *Proceedings of the 53rd Annual Allerton Conference on Communication, Control, and Computing*, Monticello, IL, September 2015, pp. 772–779.
- [10] D. Aksaray, A. Jones, Z. Kong, M. Schwager, and C. Belta, "Q-learning for robust satisfaction of signal temporal logic specifications," in *Proceedings of the Conference on Decision and Control*, Las Vegas, NV, December 2016, pp. 6565–6570.
- [11] K. P. Tee, S. S. Ge, and E. H. Tay, "Barrier lyapunov functions for the control of output-constrained nonlinear systems," *Automatica*, vol. 45, no. 4, pp. 918–927, 2009.
- [12] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, 2017.
- [13] A. D. Ames, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs with application to adaptive cruise control," in *Proceedings of the Conference on Decision and Control*, Los Angeles, CA., December 2014, pp. 6271–6278.
- [14] M. Z. Romdlony and B. Jayawardhana, "Stabilization with guaranteed safety using control lyapunov-barrier function," *Automatica*, vol. 66, pp. 39–47, 2016.
- [15] P. Glotfelter, J. Cort s, and M. Egerstedt, "Nonsmooth barrier functions with applications to multi-robot systems," *IEEE control systems letters*, vol. 1, no. 2, pp. 310–315, 2017.
- [16] L. Wang, A. D. Ames, and M. Egerstedt, "Multi-objective compositions for collision-free connectivity maintenance in teams of mobile robots," in *Proceedings of the Conference on Decision and Control*, Las Vegas, NV, December 2016, pp. 2659–2664.
- [17] X. Xu, "Constrained control of input-output linearizable systems using control sharing barrier functions," *Automatica*, vol. 87, pp. 195–201, 2018.
- [18] L. Lindemann, C. K. Verginis, and D. V. Dimarogonas, "Prescribed performance control for signal temporal logic specifications," in *Proceedings of the Conference on Decision and Control*, Melbourne, Australia, December 2017, pp. 2997–3002.
- [19] S. Boyd and L. Vandenberghe, *Convex optimization*, 1st ed. New York, NY: Cambridge university press, 2004.
- [20] R. Goebel, R. G. Sanfelice, and A. R. Teel, *Hybrid Dynamical Systems: modeling, stability, and robustness*, 1st ed. Princeton, NJ: Princeton University Press, 2012.
- [21] D. Liberzon, *Switching in systems and control*, 1st ed. New York, NY: Springer Science & Business Media, 2003.
- [22] Y. Liu, J. J. Zhu, R. L. Williams, and J. Wu, "Omni-directional mobile robot controller based on trajectory linearization," *Robotics and Autonomous Systems*, vol. 56, no. 5, pp. 461–479, 2008.