Handling Disjunctions in Signal Temporal Logic Based Control Through Nonsmooth Barrier Functions

Adrian Wiltz and Dimos V. Dimarogonas

***Abstract*— For a class of spatio-temporal tasks defined by a fragment of Signal Temporal Logic (STL), we construct a nonsmooth time-varying control barrier function (CBF) and develop a controller based on a set of simple optimization problems. Each of the optimization problems invokes constraints that allow to exploit the piece-wise smoothness of the CBF for optimization additionally to the common gradient constraint in the context of CBFs. In this way, the conservativeness of the control approach is reduced in those points where the CBF is nonsmooth. Thereby, nonsmooth CBFs become applicable to time-varying control tasks. Moreover, we overcome the problem of vanishing gradients for the considered class of constraints which allows us to consider more complex tasks including disjunctions compared to approaches based on smooth CBFs. As a well-established and systematic method to encode spatio-temporal constraints, we define the class of tasks under consideration as an STL-fragment. The results are demonstrated in a relevant simulation example.**

I. Introduction

In applications, one often encounters spatio-temporal constraints which impose both state- and time-constraints on a system. Logic expressions can be used to express such constraints. For example, one can form out of elementary rules *Robot 1 must move within 5 seconds to region A* (R1), or *Robot 2 must move within 5 seconds to region B* (R2), and *Robot 1 and Robot 2 must keep a distance of at most d to each other* (R3) the overall rule $(\mathrm{R1} \vee \mathrm{R2}) \wedge \mathrm{R3}$ where $\wedge, \vee$ denote logic AND and OR, respectively. Temporal logics like STL (Signal Temporal Logic) [14] allow the specification of such spatio-temporal constraints and increase the expressiveness of boolean logic by the temporal aspect. In the sequel, we call a composition of various spatio-temporal constraints by logic operators a *task*. Although STL originates from the field of formal verification in computer science, it is becoming increasingly popular as a well-established and systematic method to formulate spatio-temporal tasks in the field of control. Therefore, we also define the class of tasks under consideration as an STL-fragment in this paper. Most available control approaches for spatio-temporal tasks as [3], [6], [13] are based upon automata theory, which is often computationally expensive due to state discretization. Thereby, potential field based methods can be a computationally efficient alternative for some classes of spatio-temporal constraints [10], [11].

This work was supported by the ERC Consolidator Grant LEAFHOUND, the Swedish Foundation for Strategic Research (SSF) COIN, the Swedish Research Council (VR) and the Knut and Alice Wallenberg Foundation.

The authors are with the Division of Decision and Control Systems, KTH Royal Institute of Technology, SE-100 44 Stockholm, Sweden {wiltz, dimos}@kth.se.

Potential field based methods have a long tradition in control theory and have been successfully applied to tasks as collision- [4] and obstacle-avoidance [16] as well as spatio-temporal tasks [10], [11]. In the latter case, smooth time-varying Control Barrier Functions (CBF) are employed. CBFs, introduced in [17] and [19], are a control concept for ensuring the invariance of sets and proved to be a suitable tool for guaranteeing the satisfaction of state constraints on control problems [1]. By now, a broad range of results on CBFs can be found in the literature. Whereas first approaches on CBFs [19], [2] consider systems with relative degree one, [15], [21] also consider systems with higher relative degree.

With view to constraints specified via logic expressions in the context of CBFs, especially two approaches must be named: For state-constraints specified via boolean logic, [9] employs nonsmooth CBFs. On the other hand, [10] constructs a smooth time-varying CBF that ensures the satisfaction of specified spatio-temporal tasks defined via an STL-fragment. However, since [10] uses a smoothed approximation of maximum and minimum operators, there exist points where the gradient of the CBF vanishes. This may be problematic when considering disjunctions (logic OR) in the context of time-varying CBFs.

In this paper, we resolve this problem by using a nonsmooth CBF approach and can therefore take also disjunctions into account when considering spatio-temporal tasks. In contrast to [9], it is too conservative to require that a control action results in an ascend on multiple "active" CBFs at the same time. Therefore, although inspired by [7], [8], we do not base our control approach on the Filippov-operator and differential inclusions as [9], and employ a somewhat different approach in those points where the CBF is nonsmooth. In fact, we can circumvent the usage of differential inclusions by basing our controller on a set of optimization problems that exploit the piecewise smoothness of the CBF and we can show that the solutions to the closed-loop system are Carathéodory solutions. Thereby, we make the nonsmooth approach less conservative and thus applicable to time-varying CBFs.

The sequel is structured as follows: Section II introduces the considered dynamics, nonsmooth time-varying CBFs, and reviews STL; Section III constructs a nonsmooth CBF candidate for the STL-fragment under consideration, presents the control approach and proves set invariance; Section IV presents a relevant simulation example and demonstrates applicability of the proposed control scheme; Section V summarizes the conclusions of this paper. Proofs to the derived theoretic results can be found in the appendix of [20].

Notation: Sets are denoted by calligraphic letters. Let $\mathcal{A} \subseteq \mathbb{R}^n, \mathcal{B} \subseteq \mathbb{R}^m$, and let $d(\cdot,\cdot)$ define a metric on $\mathcal{A}$. The ε-neighborhood of $x \in \mathcal{A}$ is $B_\varepsilon(x) := \{y \in \mathcal{A} \,|\, d(x,y) < \varepsilon\}$, $\operatorname{Int}\mathcal{A}$ the interior, $\partial\mathcal{A}$ the boundary of $\mathcal{A}$; the Lebesgue measure of $\mathcal{A}' \subseteq \mathcal{A}$ is $\mu(\mathcal{A}')$, and if a property of a function $f : \mathcal{A} \to \mathcal{B}$ holds everywhere on $\mathcal{A} \setminus \mathcal{A}'$ with $\mu(\mathcal{A}') = 0$, we say that it holds almost everywhere (a.e.). Let $\mathcal{I} \subset \mathbb{N}$ be a finite index set with cardinality $|\mathcal{I}|$ and $\{a_i\}_{i\in\mathcal{I}} := \{a_i \,|\, i \in \mathcal{I}\}$. Let $f_i : \mathcal{A} \to \mathcal{B}$. The maximum and minimum operators are denoted by $\min_{i\in\mathcal{I}} f_i$ and $\max_{i\in\mathcal{I}} f_i$, respectively, and we define $\min_{i\in\mathcal{I}} f_i := 0$, $\max_{i\in\mathcal{I}} f_i := 0$ for $\mathcal{I} = \emptyset$. A function $\alpha : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is a class $\mathcal{K}$ function if it is continuous, strictly increasing and $\alpha(0) = 0$. The left and right sided derivatives of a function $f(t)$ with respect to t where $f : \mathbb{R} \to \mathbb{R}$ are defined as $d_{t^-} f(t) := \lim_{\nu\to 0^-} \frac{f(t+\nu)-f(t)}{\nu}$ and $d_{t^+} f(t) := \lim_{\nu\to 0^+} \frac{f(t+\nu)-f(t)}{\nu}$, respectively. For a vector-field g and a smooth real-valued scalar function h, we denote the Lie-derivative by $L_g h$. The inverse unit step is $\sigma^{-1}(x) := \begin{cases} 1 & x \leq 0 \\ 0 & x > 0 \end{cases}$. Logic *and* and *or* are denoted by $\wedge$ and $\vee$, respectively.

II. Preliminaries

At first, we introduce the system dynamics under consideration, redefine the concept of control barrier functions (CBF) in order to suit the control problem, and review STL-formulas as a formalism for defining complex spatio-temporal constraints on a control problem.

A. System Dynamics

We consider the input-affine system

$$\dot{x} = f(x) + g(x)u, \quad x(t_0) = x_0 \tag{1}$$

on the closed time-interval $\mathcal{T} = [t_1, t_2] \subseteq \mathbb{R}$, $t_0 \in \mathcal{T}$, where $x \in \mathcal{X} \subseteq \mathbb{R}^n$, $u \in \mathbb{R}^m$, and f, g are continuous functions with respective dimensions. Besides, we say that a time-varying set $\mathcal{C}(t) \subseteq \mathcal{X}$ is *forward time-invariant* for system (1), if $x(t) \in \mathcal{C}(t)\ \forall t \geq t_0$ for $x(t_0) = x_0 \in \mathcal{C}(t_0)$.

B. Non-Smooth Time-Varying Control Barrier Functions

Let $b(t,x)$ be a real-valued function $b : \mathcal{T} \times \mathcal{X} \to \mathbb{R}$, and we define $T \in \mathcal{T}$ as the time for which $b(t,x) \equiv 0$ for all $t > T$[1]. We assume that $b(t,x)$ is continuous and piecewise continuously differentiable in x, almost everywhere continuously differentiable in t, and

$$\lim_{\tau\to t^-} b(\tau, x) < \lim_{\tau\to t^+} b(\tau, x) \tag{2}$$

holds for discontinuities at $t < T$. Due to their role in what follows, we call a function $b(t,x)$ with the aforementioned properties a *barrier function* (BF).

Definition 1 (Safe Set)**.** Let $\mathcal{T}$ be the time-interval where (1) is defined. The set-valued function $\mathcal{C}(t) := \{x \in \mathcal{X} \,|\, b(t,x) \geq 0\}$ is called a time-varying *safe set* where b is a barrier function. If $x(t) \in \mathcal{C}(t)$ for all times $t \in \mathcal{T}$, we call x *feasible*.

Note that $\mathcal{C}(t) \equiv \mathcal{X}$ for $t > T$, i.e., the condition $b(t,x) \geq 0$ is trivially satisfied as $b(t,x) \equiv 0$. Next, we derive from the continuity properties of $b(t,x)$ the following continuity properties of the set-valued function $\mathcal{C}(t)$.

[1]If there is no $t \in \mathcal{T}$ such that $b(t,x) \equiv 0$ holds, then we can still set $T = t_2$.

Lemma 1. The time-varying superlevel sets $\mathcal{C}'(t) := \{x \in \mathcal{X} \,|\, b(t,x) \geq c\}$, $c \in \mathbb{R}$, of $b(t,x)$ are continuous a.e. with respect to t, and at a discontinuity at time t it holds

$$\lim_{\tau\to t^-} \mathcal{C}'(\tau) \subseteq \lim_{\tau\to t^+} \mathcal{C}'(\tau). \tag{3}$$

For details on the continuity of set-valued functions, we refer to [18, Ch. 5B]. By (3) it is ensured that for a discontinuity at time t it holds $x \in \lim_{\tau\to t^-} \mathcal{C}(\tau) \Rightarrow x \in \lim_{\tau\to t^+} \mathcal{C}(\tau)$, i.e., a feasible state stays feasible. Finally, we define control barrier functions for the nonsmooth time-varying case as follows.

Definition 2 (Control Barrier Function (CBF))**.** A barrier function $b(t,x)$ is a *control barrier function* for system (1) if there exists a class $\mathcal{K}$ function α such that

$$\sup_u d_{\delta^+} b(t+\delta, x+\delta(f(x)+g(x)u))\Big|_{\delta=0} \geq -\alpha(b(t,x)) \tag{4}$$

for all $x \in \mathcal{C}(t)$ and all $t \in \mathcal{T}$ where b is continuous.

Remark 1. The derivative on the left is a right sided directional derivative similar to [8, p. 155]. The advantage of this formulation is that it is well-defined for Lipschitz-continuous functions b which are not necessarily everywhere differentiable. As we are concerned with forward invariance, we consider the right-sided directional derivative. In the proof of Proposition 7, we relate (4) to other commonly used CBF gradient conditions as the one in [10].

C. Signal Temporal Logic (STL)

Next, we briefly review Signal Temporal Logic (STL), and specify the considered class of tasks as an STL-fragment. STL is a predicate logic with temporal operators. A predicate p has a truth value which is defined by $p := \begin{cases} \top & \text{if } h(x) \geq 0 \\ \bot & \text{if } h(x) < 0 \end{cases}$ where $\top$ and $\bot$ denote *True* and *False*, respectively, and $h : \mathcal{X} \to \mathbb{R}$ a predicate function. The grammar of a general STL formula is given as [14]

$$\theta ::= \top | p | \neg\theta | \theta_1 \vee \theta_2 | \theta_1 \mathcal{U}_{[a,b]} \theta_2 \tag{5}$$

where θ_1, θ_2 are STL formulas, and $0 \leq a \leq b$ with $a, b \in \mathbb{R}_{\geq 0}$. The satisfaction relation $(x,t) \vDash \theta$ indicates that a time-dependent function x satisfies θ from time t onwards. It is inductively defined as

$$(x,t) \vDash p \quad \Leftrightarrow h(x(t)) \geq 0 \tag{6a}$$

$$(x,t) \vDash \neg\theta \quad \Leftrightarrow \neg((x,t) \vDash \theta) \Leftrightarrow (x,t) \nvDash \theta \tag{6b}$$

$$(x,t) \vDash \theta_1 \vee \theta_2 \quad \Leftrightarrow (x,t) \vDash \theta_1 \text{ or } (x,t) \vDash \theta_2 \tag{6c}$$

$$(x,t) \vDash \theta_1 \mathcal{U}_{[a,b]} \theta_2 \Leftrightarrow \exists t' \in [t+a, t+b] \text{ s.t. } (s,t') \vDash \theta_2 \text{ and } (x,t'') \vDash \theta_1,\ \forall t'' \in [t,t']. \tag{6d}$$

Instead of $(x,0) \vDash \theta$, we also write $x \vDash \theta$ in the sequel. Due to De Morgans law, this grammar also includes conjunctions, defined as $\theta_1 \wedge \theta_2 := \neg(\neg\theta_1 \vee \neg\theta_2)$. Moreover starting with the *until* operator, the *eventually* and *always* operators can be defined as $(x,t) \vDash \mathcal{F}_{[a,b]}\theta := \top\mathcal{U}_{[a,b]}\theta$ and $(x,t) \vDash \mathcal{G}_{[a,b]}\theta := \neg\mathcal{F}_{[a,b]}\neg\theta$, respectively, and it equivalently holds

$$(x,t) \vDash \mathcal{F}_{[a,b]}\theta \Leftrightarrow \exists t' \in [t+a, t+b] \text{ s.t. } (x,t') \vDash \theta, \tag{7a}$$

$$(x,t) \vDash \mathcal{G}_{[a,b]}\theta \Leftrightarrow (x,t') \vDash \theta,\ \forall t' \in [t+a, t+b]. \tag{7b}$$

In the sequel, we consider the STL-fragment

$$\psi ::= \top|p|\psi_1 \vee \psi_2|\psi_1 \wedge \psi_2 \tag{8a}$$

$$\phi ::= \phi_1 \vee \phi_2|\phi_1 \wedge \phi_2|\mathcal{F}_{[a,b]}\psi|\mathcal{G}_{[a,b]}\psi|\psi_1\mathcal{U}_{[a,b]}\psi_2 \tag{8b}$$

where $a, b \in \mathcal{T}$ and $a \leq b$, which is the fragment considered in [10] extended by disjunctions. Below we denote STL-formulas satisfying grammar (8a) or (8b) by ψ_i or ϕ_i, respectively. As we see later, the problem of vanishing gradients in the presence of disjunctions as encountered in [10], [11] can be resolved with a non-smooth approach.

Assumption 1. We assume that $h : \mathcal{X} \to \mathbb{R}$ is a continuously differentiable and concave function. Let $\mathcal{H}$ be the set of all maximum points of $h(x)$, i.e., $\mathcal{H} := \{x_0 \in \mathcal{X} \,|\, \exists\varepsilon > 0 : ||x_0 - x|| < \varepsilon \Rightarrow h(x) < h(x_0)\}$. We additionally assume that $L_g h(x) \neq 0\ \forall x \in \mathcal{X} \setminus \mathcal{H}$ and $L_g h(x) \neq 0$ if $L_f h(x) \neq 0$ for $x \in \mathcal{H}$ (first-order condition on h).

III. Main Results

In the sequel, we present the construction of a BF which parallels [9], [10] in parts, and show that it satisfies the properties assumed in Section II-B. Thereafter, we outline the proposed control approach and prove the invariance of safe sets for the closed-loop system.

A. Construction of BFs

Consider an STL-formula ϕ_0 that satisfies grammar (8) and comprises predicates $\{p_i\}_{i\in\mathcal{I}^e}$ where $\mathcal{I}^e \subset \mathbb{N}$ is an index set; the corresponding predicate functions are $\{h_i(x)\}_{i\in\mathcal{I}^e}$. In this section, our goal is to construct a BF $b_0(t, x)$ for the STL-formula ϕ_0 such that $b_0(t, x(t)) \geq 0\ \forall t \in \mathcal{T}$ implies $x \vDash \phi_0$; then we say that b_0 *implements* the STL-formula ϕ_0.

In a first step, we construct the BFs $\{b_i\}_{i\in\mathcal{I}^e}$ for each of the predicates $\{p_i\}_{i\in\mathcal{I}^e}$ which we call *elementary barrier functions*:

R0: For $\psi_i = p_i$, the corresponding BFs are $b_i(t, x) := h_i(x)$.

Using the set of elementary BFs as a starting point, we can recursively construct a BF b_0 implementing ϕ_0. Therefore, we introduce rules for the construction of BFs b_i which implement STL-formulas ψ_i and ϕ_i satisfying grammar (8a) and (8b), respectively, as a composition of already constructed BFs $\{b_{i'}\}_{i'\in\mathcal{B}_i}$ where $\mathcal{B}_i \subset \mathbb{N}$ is a finite index set. The construction rules are given as follows:

R1: If $\psi_i = \bigwedge_{i'\in\mathcal{B}_i} \psi_{i'}$, choose $b_i(t, x) = \min_{i'\in\mathcal{B}_i} b_{i'}(t, x)$.

R2: If $\psi_i = \bigvee_{i'\in\mathcal{B}_i} \psi_{i'}$, choose $b_i(t, x) = \max_{i'\in\mathcal{B}_i} b_{i'}(t, x)$.

R3: For $\phi_i = \mathcal{F}_{[a,b]}\psi_{i'}$, we have $\mathcal{B}_i = \{i'\}$ and choose $b_i(t, x) = (b_{i'}(t, x) + \gamma_i(t))\sigma^{-1}(t - \beta_i)$ where σ^{-1} is the inverse unit step as defined in the notation section, $\gamma_i : \mathcal{T} \to \mathbb{R}$ is a continuously differentiable function such that $\exists t' \in [a, b] : \gamma_i(t') \leq 0$, and time $\beta_i := \min\{t' | \gamma_i(t') \leq 0\}$.

R4: For $\phi_i = \mathcal{G}_{[a,b]}\psi_{i'}$, we have $\mathcal{B}_i = \{i'\}$ and choose $b_i(t, x) = (b_{i'}(t, x) + \gamma_i(t))\sigma^{-1}(t - \beta_i)$ where σ^{-1} is the inverse unit step, $\gamma_i : \mathcal{T} \to \mathbb{R}$ is continuously differentiable, $\gamma_i(t') \leq 0$ for all $t' \in [a, b]$, and time $\beta_i := b$.

R5: If $\phi_i = \bigwedge_{i'\in\mathcal{B}_i} \phi_{i'}$, choose $b_i(t, x) = \min_{i'\in\widetilde{\mathcal{B}}_i(t)} b_{i'}(t, x)$ where $\widetilde{\mathcal{B}}_i(t) := \{i' \in \mathcal{B}_i | t \leq \beta_{i'}\}$ which means that $b_{i'}$ is *deactivated* at time $\beta_{i'}$, i.e., b_i does not depend on $b_{i'}$ for $t > \beta_{i'}$ anymore. In addition, set time $\beta_i := \max_{i'\in\mathcal{B}_i} \beta_{i'}$.

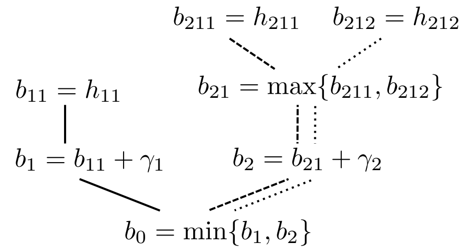


Fig. 1. Illustration of the recursive construction of b_0.

R6: If $\phi_i = \bigvee_{i'\in\mathcal{B}_i} \phi_{i'}$, choose $b_i(t,x) = \max_{i'\in\widetilde{\mathcal{B}}_i(t)} b_{i'}(t,x)\sigma^{-1}(t - \beta_i)$ with $\widetilde{\mathcal{B}}_i(t) = \{i' \in \mathcal{B}_i \,|\, b_{i'}(\tau, x(\tau)) \geq 0\ \forall\tau \in [t_1, t]\}$ and time $\beta_i := \min_{i'\in\widetilde{\mathcal{B}}_i(t)} \beta_{i'}$.

R7: For $\phi_i = \psi_{i'}\mathcal{U}_{[a,b]}\psi_{i''}$, note that $(x, t) \vDash \psi_{i'}\mathcal{U}_{[a,b]}\psi_{i''} \Leftrightarrow \exists t' \in [t + a, t + b]$ s.t. $(x, t) \vDash \mathcal{G}_{[0,t']}\psi_{i'} \wedge \mathcal{F}_{[a,t']}\psi_{i''}$ and construction rules R2, R3 and R5 can be applied.

For BFs b_i constructed in R0, we define $\mathcal{B}_i = \emptyset$; for b_i constructed in R0, R1, R2, R3 and R4, we set $\gamma_i(t) \equiv 0$ and $\widetilde{\mathcal{B}}_i(t) \equiv \mathcal{B}_i$; and for b_i constructed in R0, R1 and R2, we set $\beta_i = \infty$. We call a scalar β_i *deactivation time* of b_i. Deactivation times reduce the conservativeness of b_0, cf. [10]. The set containing the indices of all BFs b_i constructed in intermediate steps of the construction of b_0 is denoted by $\mathcal{I}$ and must be distinguished from $\mathcal{I}^e$. Besides, we require that $\mathcal{B}_{i_1} \cap \mathcal{B}_{i_2} = \emptyset$ for all $i_1, i_2 \in \mathcal{I}$, $i_1 \neq i_2$, i.e., every BF might be used for the construction of at most one other BF and hence b_0 assumes a tree structure as illustrated by Figure 1. We illustrate the application of the construction rules R0-R7 with an example.

Example 1. Consider $\phi_0 = \mathcal{G}_{[a,b]}(h_{11}(x) \geq 0) \wedge \mathcal{F}_{[c,d]}(h_{211}(x) \geq 0 \vee h_{212}(x) \geq 0)$; here, we have $\mathcal{I}^e = \{11, 211, 212\}$. According to R0, define $\psi_{11} := p_{11}$, $\psi_{211} := p_{211}$, and $\psi_{212} := p_{212}$ with h_{11}, h_{211}, h_{212} as the respective predicate functions of predicates p_{11}, p_{211}, p_{212}. Then, we start the recursive construction by choosing the elementary BFs as $b_{11}(t, x) := h_{11}(x)$, $b_{211}(t, x) := h_{211}(x)$, $b_{212}(t, x) := h_{212}(x)$. Moreover, we define $\phi_1 := \mathcal{G}_{[a,b]}\psi_1$, $\psi_{21} := \psi_{211} \vee \psi_{212}$, $\phi_2 := \mathcal{F}_{[a,b]}\psi_{21}$, and construct b_1, b_{21}, b_2 by applying R4, R2, R3, respectively. Since $\phi_0 = \phi_1 \wedge \phi_2$, we finally obtain $b_0(t, x) = \min\{h_{11}(x) + \gamma_1(t), \max\{h_{211}(x), h_{212}(x)\} + \gamma_2(t)\}$ by applying R5. Besides, $\mathcal{I} = \{0, 1, 2, 11, 21, 211, 212\}$. Figure 1 illustrates the successive construction of b_0; the chosen indices emphasize the relation of the BFs among each other.

Now, we show that b_0 exhibits the assumed properties from Section II-B and thus constitutes a BF.

Lemma 2. The function $b_0(t, x)$ is a BF.

Next, we prove that the satisfaction of the time-dependent state-constraint $b_0(t, x(t)) \geq 0$ for all $t \in \mathcal{T}$ implies the satisfaction of the STL-formula ϕ_0. In the next theorem, let b_ψ implement an STL-formula ψ satisfying grammar (8a), and b_ϕ implement ϕ satisfying grammar (8b).

Theorem 3. If $b_\phi(t, x(t)) \geq 0$ for all $t \in \mathcal{T}$, then $x \vDash \phi$. Besides, $b_\psi(t, x(t)) \geq 0 \Rightarrow (x, t) \vDash \psi$.

In the sequel, we require the following assumption in addition to the fact that b_0 is a BF.

Assumption 2. Let x be a maximum point of b_0 at a given time t, i.e., $b_0(t,x) \geq b_0(t,x') \; \forall x' \in B_\varepsilon(x)$ for some $\varepsilon > 0$. We assume that there exists a constant $b_{\min} \in \mathbb{R}$ such that $b_0(t,x) > b_{\min} > 0$ for any maximum point x of b_0 at any given time $t > T$.

Remark 2. Assumption 2 excludes STL-formulas that require predictions in order to ensure forward invariance. Such control tasks are beyond the scope of this paper. In particular, Assumption 2 implies that there exist connected sets $\mathcal{C}_{i'}(t)$ such that $\mathcal{C}(t) = \bigcup_{i'} \mathcal{C}_{i'}(t)$ where $\operatorname{Int}(\mathcal{C}_{i'}(t)) \neq \emptyset$ for all $t \in \mathcal{T}$.

In the next section, we present a control scheme based on the constructed BF b_0 and show that it is a CBF.

B. Controller Design

In related CBF literature [1], [9], [10], an optimization problem is solved that ensures the satisfaction of a CBF gradient condition. However, directly solving

$$u^* = \operatorname*{argmin}_u u^T Q u \tag{9a}$$

$$\text{s.t. } d_{\delta^+} b_0(t+\delta, x+\delta(f(x)+g(x)u))\Big|_{\delta=0} \geq -\alpha(b_0(t,x)), \tag{9b}$$

where (9b) ensures the satisfaction of the CBF gradient condition (4), is numerically difficult as b_0 is nonsmooth. Therefore, we subdivide (9) into multiple basic optimization problems with a simplified gradient condition which can be numerically easily solved.

At first, we define some index sets that help us to describe the tree structure of the BFs constructed in Section III-A. Recall that we denote the index set of all BFs as $\mathcal{I}$ and the index set of elementary BFs as $\mathcal{I}^e$. For any $i,k \in \mathcal{I}^e$, we define the index set $\mathcal{Q}_i^k(t) := \begin{cases} \{k\} & \text{if } i = k \\ \emptyset & \text{otherwise} \end{cases}$, and for any $k \in \mathcal{I}^e$ and $i \in \mathcal{I} \setminus \mathcal{I}^e$ we define

$$\mathcal{Q}_i^k(t) := \begin{cases} \bigcup_{i' \in \widetilde{\mathcal{B}}_i(t)} \mathcal{Q}_{i'}^k(t) \cup \{i\} & \text{if } \exists i' \in \widetilde{\mathcal{B}}_i(t) \colon \mathcal{Q}_{i'}^k(t) \neq \emptyset \\ \emptyset & \text{otherwise} \end{cases} \tag{10}$$

Note that there exists *at most* one $i' \in \widetilde{\mathcal{B}}_i(t)$ such that $\mathcal{Q}_{i'}^k(t) \neq \emptyset$ because $\mathcal{B}_{i_1} \cap \mathcal{B}_{i_2} = \emptyset \; \forall i_1, i_2 \in \mathcal{I}$ with $i_1 \neq i_2$ which is due to the tree structure of all BFs. Therefore, index set $\mathcal{Q}_i^k$ can be interpreted as a branch in the tree structure that connects the BFs with indices i and k. Moreover, we call $b_{i'}(t,x)$, $i' \in \widetilde{\mathcal{B}}_i(t)$ with $i \in \mathcal{I}$, an *active* BF of b_i at $(t,x) \in \mathcal{T} \times \mathcal{X}$ if $b_i(t,x) = b_{i'}(t,x) + \gamma_i(t)$. Correspondingly, we define the *active index set* of a BF $b_i(t,x)$ as

$$\mathcal{I}_i^a(t,x) := \{i' \in \widetilde{\mathcal{B}}_i(t) \,|\, b_i(t,x) = b_{i'}(t,x) + \gamma_i(t)\}. \tag{11}$$

The *active elementary BF index set* for $i \in \mathcal{I}$ is defined as

$$\begin{aligned} \mathcal{I}_i^{e,a}(t,x) := \{ & k \in \mathcal{I}^e \,|\, \mathcal{Q}_k^i(t) \neq \emptyset \wedge \\ & b_i(t,x) = h_k(x) + \textstyle\sum_{i' \in \mathcal{Q}_i^k(t)} \gamma_{i'}(t)\} \end{aligned} \tag{12}$$

Index sets $\mathcal{I}_i^a$ and $\mathcal{I}_i^{e,a}$ allow to simplify b_0 for a given time t and state x, and in the following we take advantage of the fact that for $k \in \mathcal{I}_i^{e,a}(t,x)$ the BF b_i can be written as $b_i(t,x) = h_k(x) + \sum_{i' \in \mathcal{Q}_i^k(t)} \gamma_{i'}(t)$. Furthermore for each pair $k,l \in \mathcal{I}_0^{e,a}$, there exist unique indices i_{kl}, j_{kl} and q_{kl} such that $i_{kl}, j_{kl} \in \mathcal{I}_{q_{kl}}^a$, $k \in \mathcal{I}_{i_{kl}}^{e,a}$, $l \notin \mathcal{I}_{i_{kl}}^{e,a}$, $l \in \mathcal{I}_{j_{kl}}^{e,a}$ and $k \notin \mathcal{I}_{j_{kl}}^{e,a}$. The tree structure of the BFs and the recursive definition of the index sets allow for their efficient computation. We illustrate the meaning of the definitions by revisiting Example 1.

Example 2. Consider ϕ_0 in Example 1. As all expressions are considered for the same time t and state x, we omit state and time arguments. According to (10), $\mathcal{Q}_0^{211} = \{0, 2, 21, 211\}$ specifies the indices of the branch connecting BFs with indices 0 and 211. Correspondingly, $\mathcal{Q}_0^{212} = \{0, 2, 21, 212\}$, $\mathcal{Q}_{211}^{211} = \{211\}$, $\mathcal{Q}_{212}^{212} = \{212\}$. Let $\mathcal{I}_0^a = \{2\}$, $\mathcal{I}_2^a = \{21\}$, $\mathcal{I}_{21}^a = \{211, 212\}$ be active index sets. Then, as it can be seen from Figure 1, we have $\mathcal{I}_0^{e,a} = \mathcal{I}_2^{e,a} = \mathcal{I}_{21}^{e,a} = \{211, 212\}$, $\mathcal{I}_{211}^{e,a} = \{211\}$ and $\mathcal{I}_{212}^{e,a} = \{212\}$ according to (12). From these active elementary index sets, we can determine $i_{kl} = 211$, $j_{kl} = 212$, $q_{kl} = 21$ for $k = 211$ and l=212. Loosely speaking, $b_{q_{kl}}$ denotes the BF from which the branches leading to b_k and b_l emanate, and i_{kl}, j_{kl} are chosen such that $\mathcal{Q}_{i_{kl}}^k$ and $\mathcal{Q}_{j_{kl}}^l$ do not contain common indices.

Next, we define subsets $\mathcal{S}_k(t) \subseteq \mathcal{X}$ on the state space as

$$\mathcal{S}_k(t) := \{x \,|\, k \in \mathcal{I}_0^{e,a}(t,x)\} \tag{13}$$

which can be equivalently written as $\mathcal{S}_k(t) = \{x \,|\, b_0(t,x) = b_0^k(t,x) := h_k(x) + \sum_{i' \in \mathcal{Q}_0^k(t)} \gamma_{i'}(t)\}$. Since for all times $t \leq T$ and all states $x \in \mathcal{X}$ there exists at least one active elementary BF $k \in \mathcal{I}_0^{e,a}(t,x)$, it holds that $\mathcal{X} = \bigcup_{k \in \mathcal{I}^e} \mathcal{S}_k(t)$. The sets $\mathcal{S}_k(t)$ enjoy the favorable property that $b_0(t,x)$ is continuously differentiable with respect to x in the interior $\operatorname{Int} \mathcal{S}_k(t)$ and possibly non-smooth only on $\partial\mathcal{S}_k(t) \cap \partial\mathcal{S}_l(t)$ for some $l \in \mathcal{I}^e$. Furthermore, it holds for some $k,l \in \mathcal{I}^e$ that $b_0^k(t,x) = b_0^l(t,x)$ for all $x \in \mathcal{S}_k(t) \cap \mathcal{S}_l(t)$, and in particular for $x \in \partial\mathcal{S}_k(t) \cap \partial\mathcal{S}_l(t)$.

Given $x(t) \in \partial\mathcal{S}_k(t) \cap \partial\mathcal{S}_l(t)$ for some $t \in \mathcal{T} \setminus \{\beta_i\}_{i \in \mathcal{I}}$, we can therefore formulate the condition $\frac{\partial}{\partial(t,x)}(b_0^k(t,x) - b_0^l(t,x))[1, (f(x)+g(x)u)^T]^T \gtreqless 0$ to constrain u such that $x(\tau) \in \mathcal{S}_k(\tau)$ or $x(\tau) \in \mathcal{S}_l(\tau)$ (depending on the choice of $\geq$ or $\leq$) for all $\tau \in [t, t+\delta]$ and a $\delta > 0$. By ensuring that x stays for some time in the interior of one of the subsets $\mathcal{S}_k(t)$ or $\mathcal{S}_l(t)$, we can take advantage of the piecewise differentiability of b_0. In the sequel, we generalize this idea for $x(t) \in \bigcap_{k \in \mathcal{I}_0^{e,a}} \mathcal{S}_k(t)$ and formulate an optimization problem for each $k \in \mathcal{I}_0^{e,a}$ with a gradient condition which is simplified in comparison to (9b).

For determining the boundary between $\mathcal{S}_k(t)$ and $\mathcal{S}_l(t)$, we define

$$s_{kl}(t,x) := \begin{cases} b_{j_{kl}}^l(t,x) - b_{i_{kl}}^k(t,x) & \text{if } b_{q_{kl}} = \min_{i' \in \widetilde{\mathcal{B}}_{q_{kl}}} b_{i'} \\ b_{i_{kl}}^k(t,x) - b_{j_{kl}}^l(t,x) & \text{if } b_{q_{kl}} = \max_{i' \in \widetilde{\mathcal{B}}_{q_{kl}}} b_{i'} \end{cases} \tag{14}$$

where $b_i^k(t,x) := h_k(x) + \sum_{i' \in \mathcal{Q}_i^k(t)} \gamma_{i'}(t)$ for $i \in \mathcal{I}$, $k \in \mathcal{I}_i^{e,a}(t,x)$. Then for a given time t, $s_{kl}(t,x) = 0$ determines those x on the boundary between $\mathcal{S}_k(t)$ and $\mathcal{S}_l(t)$, i.e., $x \in \partial\mathcal{S}_k \cap \mathcal{S}_l$, $x \in \mathcal{S}_k \cap \partial\mathcal{S}_l$, or $x \in \partial\mathcal{S}_k \cap \partial\mathcal{S}_l$ if $\frac{\partial s_{kl}}{\partial x}(t,x) \neq 0$. Especially the set $\partial\mathcal{S}_k \cap \partial\mathcal{S}_l$ is of interest as it can be shown that only there b_0 is non-differentiable in x. The directional derivative of s_{kl} along the trajectory of (1) is given as

$$s_{kl}'(t,x,u) := \frac{\partial s_{kl}}{\partial(t,x)}(t,x) \begin{bmatrix} 1 \\ f(x)+g(x)u \end{bmatrix}. \tag{15}$$

For an illustration of (15), consider Figure 2. If $x(t) \in \partial\mathcal{S}_1(t) \cap \partial\mathcal{S}_2(t)$, $s_{12}'(t,x,u) \geq 0$ only admits inputs u such that $\dot{x} = f(x) + g(x)u$ points into $\mathcal{S}_1(t)$, thus $x(\tau) \in \mathcal{S}_1(\tau)$

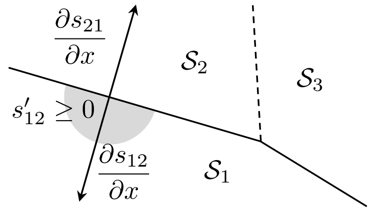


Fig. 2. Illustration of $b_0(t,x)$ for a given t, $x \in \mathcal{X} \subseteq \mathbb{R}^2$ and $\phi_0 = \mathcal{G}_{[a,b]}(h_1(x) \geq 0) \wedge \mathcal{F}_{[c,d]}(h_2(x) \geq 0 \vee h_3(x) \geq 0)$. The lines between the sections $\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3$ indicate $s_{kl} = 0$ for $k, l \in \mathcal{I}^e = \{1,2,3\}$.

for all $\tau \in [t, t+\delta]$ and a $\delta > 0$.

Using this insight, we define optimization problems with a simplified gradient constraint as

$$u_k^*(t) = \underset{u}{\operatorname{argmin}}\ u^T Q u \tag{16a}$$

$$\text{s.t. } \frac{\partial b_0^k}{\partial x}(t,x)(f(x)+g(x)u) + \frac{\partial b_0^k}{\partial t}(t,x) \geq -\alpha(b_0(t,x)) \tag{16b}$$

$$s'_{kl}(t,x,u) \geq 0 \quad \forall l \in \mathcal{I}_0^{e,a}(t,x),\ l \neq k. \tag{16c}$$

for $k \in \mathcal{I}_0^{e,a}(t,x)$ where $Q \in \mathbb{R}^{m\times m}$ is a positive-definite matrix and α a class $\mathcal{K}$ function. If (16) has no feasible solution, we set $u_k^* = \infty$. Finally,

$$u^*(t) = \underset{u_k^* \text{ with } k \in \mathcal{I}_0^{e,a}(t,x)}{\operatorname{argmin}} u_k^*(t)^T Q u_k^*(t) \tag{17}$$

is applied as control input to (1). Note that in (15) and (16b) the time derivatives of s_{kl} and b_0^k only exist on $\mathcal{T} \setminus \{\beta_i\}_{i\in\mathcal{Q}_0^k}$. At times $t \in \{\beta_i\}_{i\in\mathcal{Q}_0^k}$, s_{kl} and b_0^k might be discontinuous and we consider the left sided derivative instead, i.e., $d_{t^-}s_{kl}$ and $d_{t^-}b_0^k$, respectively. As it can be seen from the proof in Theorem 8, this choice is arbitrary and does not impact the invariance result. Next, we show that there always exists a feasible solution to (17) if the class $\mathcal{K}$ function α satisfies the following condition.

Assumption 3. It holds $\alpha(b_{\text{min}}) > -\frac{\partial \gamma_i}{\partial t}(t)\ \forall t \in \mathcal{T}, \forall i \in \mathcal{I}$.

Note that since γ_i is continuously differentiable and defined on a closed interval, $\frac{\partial \gamma_i}{\partial t}$ is bounded. As the class $\mathcal{K}$ function α can be freely chosen, there always exists a class $\mathcal{K}$ function α such that Assumption 3 is fulfilled.

Lemma 4. Let Assumption 3 hold. For all $(t,x) \in \mathcal{T}\times\mathcal{C}(t)$, there exist $k \in \mathcal{I}_0^{e,a}(t,x)$ such that (16) has a feasible and finite solution.

As it can be seen from the proof, in Assumption 3 the class $\mathcal{K}$ function α is chosen such that no increase in b_0 is required in (16b) by varying the system's state x if $b_0(t,x) \geq b_{\text{min}}$. This is especially important when x is already a maximum point of b_0 for a given time t and a variation of x does not lead to an increase on b_0. Loosely speaking, α determines when $b_0(t,x)$ is sufficiently close to zero such that the controller reacts in order ensure the invariance of the safe set $\mathcal{C}(t)$, whereas functions γ_i determine how "quickly" state x has to change. Thereby, γ_i is decisive for the magnitude of the control input u.

In the remainder of this section, we prove the forward invariance of $\mathcal{C}(t)$. Therefore, consider a solution $\varphi : \mathcal{T} \to \mathcal{X}$ to the closed-loop system (1) when control input (17) is applied. At first, we investigate the continuity properties of the control input trajectory $u : \mathcal{T} \to \mathbb{R}^m$ of the closed-loop system and show that φ is a Carathéodory solution. However, this is not obvious as $u(t,x)$ is discontinuous both in t and x.

Lemma 5. The control input trajectory $u^* : \mathcal{T} \to \mathbb{R}^m$ of the closed-loop system (1) with controller (16)-(17) is continuous a.e.

Corollary 6. A solution $\varphi : \mathcal{T} \to \mathcal{X}$ to the closed-loop system (1) with controller (16)-(17) is a Carathéodory solution. Moreover, it holds for the right sided derivative that $d_{t^+}\varphi(t) = f(\varphi(t)) + g(\varphi(t))u^*(t)$ for $t \notin \{\beta_i\}_{i\in\mathcal{I}}$.

Remark 3. Solution φ is not necessarily unique. If multiple u_k^* minimize the objective in (17), then any of the inputs could be applied. Depending on which input is chosen, different solutions φ are obtained.

Next, we compare the controllers (9) and (16)-(17). This result facilitates the proof of forward invariance of the safe set in Theorem 8, and allows us to compare the proposed non-smooth CBF approach to other CBF-based controllers.

Proposition 7. The optimization problem (9) is equivalent to (16)-(17) for all $t \in \mathcal{T} \setminus \{\beta_i\}_{i\in\mathcal{I}}$.

In contrast to (9), the constraints in (16) can be more easily evaluated since b_0^k is differentiable contrary to b_0. Based on the previous results, we can prove forward invariance of $\mathcal{C}(t)$.

Theorem 8. The control law (16)-(17) renders $\mathcal{C}(t)$ forward invariant for all $t \in \mathcal{T}$, and b_0 is a CBF.

IV. Simulations

Similarly to [10], we consider a multiagent system comprising three omnidirectional robots which are modeled as in [12] and use a collision avoidance mechanism as in [10]. The state of agent i is given as $x_i = [p_i^T, \rho_i]^T$ where $p_i = [x_{i,1}, x_{i,2}]$ denotes its position and ρ_i its orientation; the state of all agents together is given as $x = [x_1^T, x_2^T, x_3^T]^T$. The dynamics of agent i are

$$\dot{x}_i = f_i(x) + \begin{bmatrix} \cos(\rho_i) & -\sin(\rho_i) & 0 \\ \sin(\rho_i) & \cos(\rho_i) & 0 \\ 0 & 0 & 1 \end{bmatrix} (B_i^T)^{-1} R_i u_i$$

where $f_i(x) = [f_{i,1}(x), f_{i,2}(x), 0]^T$ with $f_{i,k}(x) = \sum_{j=1, i\neq j}^{3} k_i \frac{x_{i,k}-x_{j,k}}{||p_i-p_j||^2+0.00001}$, $k_i > 0$, $B_i = \begin{bmatrix} 0 & \cos(\pi/6) & -\cos(\pi/6) \\ -1 & \sin(\pi/6) & \sin(pi/6) \\ L_i & L_i & L_i \end{bmatrix}$ with $L_i = 0.2$ as the radius of the robot body, $R_i = 0.02$ is the wheel radius, and u_i is the angular velocity of the wheels and serves as control input. As required, the system is input-affine and f_i, g_i are continuous. Besides, we admit sufficiently large inputs to the system.

The task for the three agents comprises three parts: (1) approaching each other: $\phi_1 := \mathcal{F}_{[10,20]}(||p_1 - p_2|| \leq 10 \vee ||p_1 - p_3|| \leq 10 \vee ||p_2 - p_3|| \leq 10) \wedge \mathcal{G}_{[20,60]}(||p_2 - p_3|| \leq 15)$; (2) moving to given points: $\phi_2 := (||p_3 - [-5,-5]^T|| \leq 10)\mathcal{U}_{[5,20]}(||p_1 - p_2|| \leq 10) \wedge \mathcal{F}_{[10,20]}(||p_1 - [0,30]^T|| \leq 10) \wedge \mathcal{G}_{[50,60]}(||p_1 - [30,0]^T|| \leq 10) \wedge (\mathcal{G}_{[50,60]}(||p_2 - [-30,-30]^T|| \leq 10) \vee \mathcal{G}_{[50,60]}(||p_3 - [30,-30]^T|| \leq 10))$; and (3) staying within a defined area: $\phi_3 := \mathcal{G}_{[0,60]}(||[p_1^T, p_2^T, p_3^T]^T||_\infty \leq 40)$. The norms $||\cdot||$ and $||\cdot||_\infty$ denote the euclidean and the maximum norm,

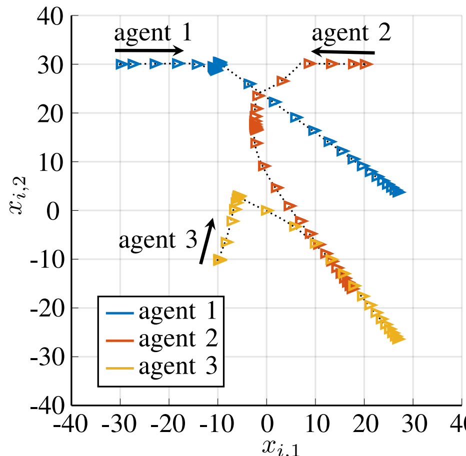
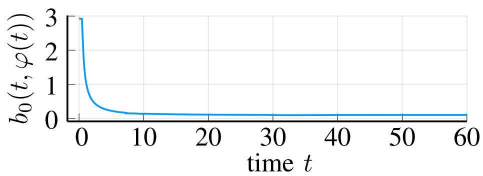
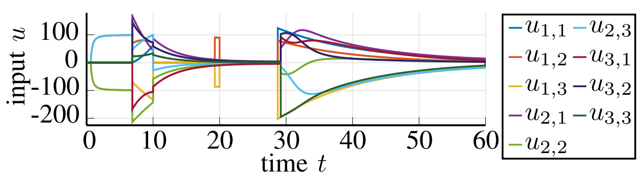


Fig. 3. Trajectories of the mobile agents. Orientation is denoted by the triangles' orientation.



(a)



(b)

Fig. 4. Control barrier function $b_0(t, \varphi(t))$ and control inputs u over time.

respectively. The overall task is given as the conjunction $\phi_0 := \phi_1 \wedge \phi_2 \wedge \phi_3$. In contrast to [10], the considered task also contains disjunctions.

For the construction of the CBF $b_0(t, x)$, all rules from Section III-A (R0-R7) are applied; the controller is designed according to Section III-B. The simulation is implemented in Julia using Jump [5] and run on an Intel Core i5-10310U with 16GB RAM. The controller is evaluated with 50Hz and the control input is applied using a zero-order hold; the computation of the control input took 16ms on average. The trajectories of the agents resulting from the simulation are depicted in Figure 3, the evolution of b_0 and the applied inputs in Figure 4. Since $b_0(t, \varphi(t)) \geq 0 \ \forall t \in [0, 60]$, we conclude that the specified constraints are satisfied. Besides, the inputs are indeed continuous a.e.

V. Conclusion

In this paper, we constructed a nonsmooth time-varying CBF for Signal Temporal Logic tasks including disjunctions and derived a controller that ensures their satisfaction. By using a nonsmooth approach, we avoided the problem of vanishing gradients on the CBF that occurs when employing smoothed approximations of the minimum and maximum operators. Moreover, by partitioning the state space into sections and designing an optimization problem for each of them, we could determine the respective elementary barrier function "of relevance". This allowed us to avoid the usage of differential inclusions in our derivation, thereby to reduce the conservativeness of our results and to apply a non-smooth approach to time-varying CBFs.

References

[1] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *18th European Control Conference (ECC)*, 2019, pp. 3420–3431.

[2] A. D. Ames, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs with application to adaptive cruise control," in *53rd IEEE Conference on Decision and Control*, 2014, pp. 6271–6278.

[3] C. Belta, A. Bicchi, M. Egerstedt, E. Frazzoli, E. Klavins, and G. J. Pappas, "Symbolic planning and control of robot motion [grand challenges of robotics]," *IEEE Robotics Automation Magazine*, vol. 14, no. 1, pp. 61–70, 2007.

[4] D. V. Dimarogonas, S. G. Loizou, K. J. Kyriakopoulos, and M. M. Zavlanos, "A feedback stabilization and collision avoidance scheme for multiple independent non-point agents," *Automatica*, vol. 42, no. 2, pp. 229–243, 2006.

[5] I. Dunning, J. Huchette, and M. Lubin, "Jump: A modeling language for mathematical optimization," *SIAM Review*, vol. 59, no. 2, pp. 295–320, 2017.

[6] G. Fainekos, H. Kress-Gazit, and G. Pappas, "Temporal logic motion planning for mobile robots," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, 2005, pp. 2020–2025.

[7] A. F. Filippov, "Differential equations with discontinuous right-hand side," in *Fifteen Papers on Differential Equations*, ser. American Mathematical Society Translations, vol. 42, no. 2. American Mathematical Society, 1964, pp. 199–231.

[8] ——, *Differential equations with discontinuous righthand sides*, ser. Mathematics and its applications Soviet series, 18. Dordrecht: Kluwer Academic Publishers, 1988.

[9] P. Glotfelter, J. Cortes, and M. Egerstedt, "A nonsmooth approach to controller synthesis for boolean specifications," *IEEE Transactions on Automatic Control*, pp. 5160–5174, 2020.

[10] L. Lindemann and D. V. Dimarogonas, "Control barrier functions for signal temporal logic tasks," *IEEE Control Systems Letters*, vol. 3, no. 1, pp. 96–101, 2019.

[11] L. Lindemann and D. V. Dimarogonas, "Barrier function based collaborative control of multiple robots under signal temporal logic tasks," *IEEE Transactions on Control of Network Systems*, vol. 7, no. 4, pp. 1916–1928, 2020.

[12] 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.

[13] S. Loizou and K. Kyriakopoulos, "Automatic synthesis of multi-agent motion tasks based on ltl specifications," in *43rd IEEE Conference on Decision and Control (CDC)*, vol. 1, 2004, pp. 153–158.

[14] O. Maler and D. Nickovic, "Monitoring temporal properties of continuous signals," in *Formal Techniques, Modelling and Analysis of Timed and Fault-Tolerant Systems*, Y. Lakhnech and S. Yovine, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 152–166.

[15] Q. Nguyen and K. Sreenath, "Exponential control barrier functions for enforcing high relative-degree safety-critical constraints," in *American Control Conference (ACC)*, 2016, pp. 322–328.

[16] D. Panagou, D. M. Stipanovič, and P. G. Voulgaris, "Multi-objective control for multi-agent systems using lyapunov-like barrier functions," in *52nd IEEE Conference on Decision and Control*, 2013, pp. 1478–1483.

[17] S. Prajna and A. Jadbabaie, "Safety verification of hybrid systems using barrier certificates," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 477–492.

[18] R. T. Rockafellar, *Variational analysis*, 1st ed., ser. Grundlehren der mathematischen Wissenschaften. Springer Berlin, Heidelberg, 2009.

[19] P. Wieland and F. Allgöwer, "Constructive safety using control barrier functions," *IFAC Proceedings Volumes*, vol. 40, no. 12, pp. 462–467, 2007, 7th IFAC Symposium on Nonlinear Control Systems.

[20] A. Wiltz and D. V. Dimarogonas, "Handling disjunctions in signal temporal logic based control through nonsmooth barrier functions," 2022. [Online]. Available: https://arxiv.org/abs/2208.10803

[21] W. Xiao and C. Belta, "High order control barrier functions," *IEEE Transactions on Automatic Control*, 2021.