Extent-Compatible Control Barrier Functions

Mohit Srinivasan[a,*], Matthew Abate[a,b], Gustav Nilsson[a], Samuel Coogan[a,c]

[a]*School of Electrical and Computer Engineering, Georgia Institute of Technology, USA*
[b]*School of Mechanical Engineering, Georgia Institute of Technology, USA*
[c]*School of School of Civil and Environmental Engineering, Georgia Institute of Technology, USA*

Abstract

Safety requirements in dynamical systems are commonly enforced with set invariance constraints over a safe region of the state space. Control barrier functions, which are Lyapunov-like functions for guaranteeing set invariance, are an effective tool to enforce such constraints and guarantee safety when the system is represented as a point in the state space. In this paper, we introduce extent-compatible control barrier functions as a tool to enforce safety for the system explicitly accounting for its volume (extent) within an ambient workspace. In order to implement the extent-compatible control barrier functions framework, we first propose a sum-of-squares optimization program that is solved pointwise in time to ensure safety. Since sum-of-squares programs can be computationally prohibitive, we next propose an approach that instead considers a finite number of points sampled on the extent boundary. The result is a quadratic program for guaranteed safety that retains the computational advantage of traditional barrier functions. While this alternative is generally more conservative than the sum-of-squares approach, we show that conservatism is reduced by increasing the number of sampled points. Simulation and robotic implementation results are provided.

*Corresponding author

Email addresses: `mohit.srinivasan@gatech.edu` (Mohit Srinivasan), `matt.abate@gatech.edu` (Matthew Abate), `gustav.nilsson@gatech.edu` (Gustav Nilsson), `sam.coogan@gatech.edu` (Samuel Coogan)

[1]This work was partially supported by the National Science Foundation under Grant 1749357.

Keywords: Control barrier functions, quadratic programs, set invariance

1. Introduction

A controlled dynamical system is considered *safe* if it can be ensured that a given set of safe states is forward invariant under the action of a controller, i.e., the system state remains within the safe set for all time when initialized within the safe set. For example, collision avoidance between robots, obstacle avoidance during waypoint navigation, or lane changing for autonomous vehicles can be cast as invariance constraints. Techniques for enforcing safety of dynamical systems via invariance constraints include level-set methods [1], methods leveraging reachability analysis [2], and model-predictive control methods [3].

When a nominal but possibly unsafe controller is available, control barrier functions (CBFs), introduced in [4], act as a particularly effective tool to enforce safety for control-affine dynamical systems. CBFs have been applied for collision avoidance in multi-robot systems [5], adaptive cruise control [4], motion planning [6], and safety for robotic manipulators [7]. Traditionally, CBFs are used in conjunction with quadratic programs (QPs) to compute at each time instant a safe control input.

Existing CBF-based approaches focus on establishing forward invariance for the system state, also called the system configuration. The system's volume (or extent) in its ambient workspace must therefore be implicitly included in defining the set of safe states via a CBF. In some cases, this is straightforward and can be achieved by, e.g., shrinking the safe set [8, 9]. For example, consider an application of adaptive cruise control for a vehicle [4]. In this case, the vehicle's length is accounted for when defining a safe following distance, and ultimately when defining an appropriate CBF. In other cases, incorporating the physical extent when defining an appropriate closed-form barrier is considerably more challenging or not possible, even when the system's extent and safe workspace are geometrically simple. This is because CBF-based methods rely on characterizing the safe set as a level-set of a function that is known in closed form.

For example, consider a vehicle maneuvering in a two dimensional region such as a parking lot. The vehicle's state is defined by its two dimensional position and its heading angle, and its extent is described by its physical footprint. The safe region of the workspace is defined as a two dimensional region excluding obstacles. In this case, collision of the vehicle with an obstacle depends on the geometric relationship between the vehicle's position, heading, extent, and obstacles. Even when the vehicle's extent and the workspace are simple geometric shapes, as in the case study in this paper, this relationship is complex and makes it difficult or impossible to define an appropriate classical CBF as a function of system state that is exactly the set of safe states. At best, it may be possible to obtain a closed-form approximate safe set, but this approximation will generally be conservative. Moreover, if the system's extent changes—for example, a vehicle docks with a trailer—then the classical CBF must be redesigned.

In this paper, we propose a novel CBF-based approach for ensuring safety constraints of a control-affine dynamical system that explicitly accounts for extent in an ambient workspace. We then define an extent-compatible CBF that uses a modified CBF constraint to ensure that the extent set always remains within the safe set of the workspace. We first propose implementing the resulting constraint using a sum-of-squares (SOS) optimization program [10]. Since SOS programs can be computationally difficult for high dimensional systems and are only applicable when the safe and extent sets can be represented as polynomials, we next prove that the guarantee on system safety can be retained by considering only a finite set of sampled points on the boundary of the extent set, and we propose a QP-based controller using the sampled points. This sampling-based approach relies on bounds of Lipschitz constants of functions appearing in the barrier function formulation and therefore may be more conservative than an SOS approach, however, the conservatism can be controlled by increasing the number of sample points. The proposed framework is demonstrated with a case study of a vehicle navigating in a two dimensional region, as motivated above.

This paper is organized as follows: Section 2 presents background on CBFs. Section 3 proposes the extent-compatible control barrier function formulation that is the main contribution of this paper. Section 4 introduces the QP-based controller and presents two solution methods to guarantee safety: a method utilizing SOS programming and a method that uses a finite number of points sampled on the extent set boundary. Section 5 consists of a case study which implements the proposed framework both in numerical simulations and on a differential drive robot. Section 6 provides concluding remarks.

2. Mathematical Background

In this section, we provide background on the traditional control barrier function formulation. To that end, consider a control-affine dynamical system

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

where f and g are locally Lipschitz continuous, $x \in \mathcal{D} \subset \mathbb{R}^n$ is the state of the system, $\mathcal{D}$ is assumed to be open, and $u \in \mathbb{R}^m$ denotes the control input.

Associated with the system (1) is a *safe set* $\mathcal{C} \subset \mathcal{D}$, defined as the super zero level set of a continuously differentiable function $h : \mathcal{D} \to \mathbb{R}$, i.e., $\mathcal{C} = \{x \in \mathcal{D} \mid h(x) \geq 0\}$. We call h the *safe function.* To ensure forward complete trajectories, we assume throughout that the safe set $\mathcal{C}$ is bounded. If forward completeness can be guaranteed in other ways, e.g., by ensuring that the control inputs are such that the vector field in (1) is globally Lipschitz, the assumption that $\mathcal{C}$ is bounded is not needed for the presented theory to hold true.

As presented in [11, 12], one can use zeroing control barrier functions (ZCBFs) in order to guarantee forward invariance of a safe set. In particular, a continuously differentiable function $h : \mathcal{D} \to \mathbb{R}$ satisfying the regularity condition $\frac{\partial}{\partial x} h(x) \neq 0$ for all x such that $h(x) = 0$ is a *Zeroing Control Barrier Function (ZCBF)* if there exists a locally Lipschitz extended class $\mathcal{K}$ function α such that for all $x \in \mathcal{D}$,

$$\sup_{u \in \mathbb{R}^m} \left\{ \frac{\partial h(x)}{\partial x}(f(x) + g(x)u) + \alpha(h(x)) \right\} \geq 0\,, \tag{2}$$

where we recall that a continuous function $\alpha : \mathbb{R} \to \mathbb{R}$ is *extended class* $\mathcal{K}$ if $\alpha(0) = 0$ and it is strictly increasing.

In the instance that a safe function h defining a safe set $\mathcal{C}$ is also a ZCBF, choosing a control input at each state x from the set

$$U(x) = \left\{ u \in \mathbb{R}^m \,\middle|\, \frac{\partial h(x)}{\partial x}(f(x) + g(x)u) + \alpha(h(x)) \geq 0 \right\} \tag{3}$$

guarantees that the safe set $\mathcal{C}$ is forward invariant [11]. Specifically, if $x(0) \in \mathcal{C}$ and $U(x)$ as in (3) is non-empty for all $x \in \mathcal{D}$, then, as shown in [13, Theorem 4], any continuous feedback controller $u : \mathcal{D} \to \mathbb{R}^m$ such that $u(x) \in U(x)$ for all $x \in \mathcal{D}$ is such that $x(t) \in \mathcal{C}$ for all $t \geq 0$.

3. Extent Compatible Control Barrier Functions

In this section, we first formalize the problem statement, followed by the notion of extent-compatible control barrier functions for systems of the form (1). We then prove that such functions enable guaranteed safe control of the system, including its extent.

3.1. Problem Statement

Given a ZCBF, [13, Theorem 4] guarantees that the system state will remain within the safe set $\mathcal{C}$ when control inputs are continuous in x and chosen according to $u(x) \in U(x)$ for all $x \in \mathcal{D}$, where $U(x)$ is as defined in (3). This notion of safety, however, requires the extent of the system to be implicitly accounted for in the characterization of the safe set within the system domain. In this section, we define a notion of system safety that *explicitly* includes the physical volume of the system within an ambient workspace that is distinct from the domain of the statespace as described in, *e.g.*, [14, Ch. 3]. To that end, let $\mathcal{W} \subset \mathbb{R}^w$ denote this workspace and assume $\mathcal{W}$ is open; when the system state is $x \in \mathcal{D}$, the system physically occupies some subset of $\mathcal{W}$ that depends on x, and the set of safe states is now characterized as a subset $\mathcal{C} \subset \mathcal{W}$. We encapsulate this notion of volume with an extent function E such that $E(x, y) \leq 0$ means the point $y \in \mathcal{W}$ is within the system's extent when the state of the system is $x \in \mathcal{D}$. Additional mild technical assumptions avoid pathological cases, as formalized in the following definition.

Definition 1 (Extent Function)**.** An *extent function* $E : \mathcal{D} \times \mathcal{W} \to \mathbb{R}$ is a continuously differentiable function such that:

1. $\mathcal{E}(x) = \{y \in \mathcal{W} \mid E(x, y) \leq 0\}$ is nonempty for all $x \in \mathcal{D}$,
2. $\frac{\partial}{\partial y} E(x, y) \neq 0$ for all (x, y) such that $E(x, y) = 0$, and
3. for all $\delta > 0$, there exists $\epsilon > 0$ such that for all $x \in \mathcal{D}$ and all $y \in \mathcal{W}$, if $|E(x, y)| \leq \epsilon$ then $\inf_{\hat{y} \in \partial\mathcal{E}(x)} \|y - \hat{y}\| \leq \delta$ where $\partial\mathcal{E}(x) = \{y \in \mathcal{W} \mid E(x, y) = 0\}$.

Condition 1 is the key condition of the definition. In particular, the set $\mathcal{E}(x) \subset \mathcal{W}$ above defines the system's extent when its state is $x \in \mathcal{D}$, and $\partial\mathcal{E}(x) = \{y \in \mathcal{W} \mid E(x, y) = 0\}$ denotes the extent boundary. As a simple example, if the extent contains all points within a distance $d > 0$ of the system state x, then $\mathcal{W} = \mathcal{D}$ and we may choose $E(x, y) = \|y - x\|^2 - d^2$. Conditions 2 and 3 are mild technical conditions. In particular, Condition 2 ensures that the gradient with respect to y of E does not vanish on the extent set boundary $\mathcal{E}(x)$, which is entirely analogous to standard regularity assumptions on traditional control barrier functions as is made in, e.g., [11], and Condition 3 ensures that $E(x, y)$ only approaches 0 at the extent boundary. For example, Condition 2 is violated for $E(x, y) = (y - x)^3$ with $\mathcal{D} = \mathcal{W} = \mathbb{R}$ because $E(x, y)$ as a function of y has zero slope when $x = y$, and $E(x, y) = -\cos(y - x)$ with $\mathcal{D} = \mathcal{W} = (-3\pi/2, 3\pi/2)$ violates Condition 3 when $x = 0$ because $E(x, y)$ approaches zero as y approaches the domain boundary $\{-3\pi/2, 3\pi/2\}$, which is far from the extent set boundary $\partial\mathcal{E}(x) = \{-\pi/2, \pi/2\}$. In our experience, extent functions of practical use always satisfy these technical conditions.

Given an extent function, we aim to ensure that the extent of the system is contained within the safe set for all time, i.e., $\mathcal{E}(x(t)) \subset \mathcal{C}$ for all $t \geq 0$ along trajectories of (1). An example of such a problem setup is shown in Fig. 1.

Problem 1. *Given a control affine dynamical system as in* (1) *with extent function* $E(x, y)$*, synthesize a controller* u *which guarantees* $\mathcal{E}(x(t)) \subset \mathcal{C}$ *for all* $t \geq 0$ *whenever* $\mathcal{E}(x(0)) \subset \mathcal{C}$*.*

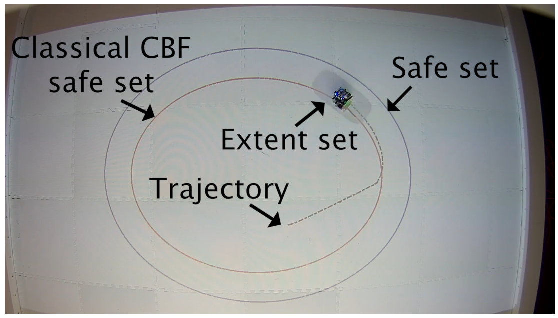


Figure 1: A motivating example, where a differential drive robot with a system volume covered by an extent set must stay within the ellipsoidal safe set when a nominal unsafe controller tries to drive it outside. This paper proposes controller frameworks to guarantee safety of the system including its volume under such situations, without introducing unnecessary conservatism by shirking the safe set, something that a classical CBF solution would have do. The image is from a Robotarium [15] implementation of the framework introduced in this paper, which we also describe in Section 5.

As a motivating example, consider a differential drive robot (*i.e.*, vehicle) with a superellipsoidal extent set and ellipsoidal safe set as shown in Fig 1. The robot including its extent set must stay within the safe set under the action of a controller that tries to drive it outside the set. This is the main intuition for Problem 1. We revisit this motivating example in Section 5.

3.2. Extent-Compatible Control Barrier Function (Ec-CBF)

We now introduce extent-compatible control barrier functions (Ec-CBFs) which are analogous to ZCBFs but guarantee that the entire extent set remains within the safe set under the action of a suitable control input. For a ZCBF h, the condition (2) ensures that, as the system state approaches the boundary of the safe set, a control action is available that limits the rate at which h decreases. This ensures that $h(x(t))$ remains nonnegative and the system remains safe. The rate at which the barrier h is allowed to decrease is dictated by the extended class $\mathcal{K}$ function α in (2). In implementations, the choice of α serves a practical role in restricting how quickly the system is allowed to approach the safe set boundary. To extend this idea to systems with extent, we introduce the following definition, which, informally, uses two extended class $\mathcal{K}$ functions to ensure that the boundary of the extent set does not approach the boundary of the safe set too quickly, thus ensuring safety.

Definition 2 (Extent-Compatible Control Barrier Function (Ec-CBF))**.** A continuously differentiable function h satisfying the regularity condition $\frac{\partial}{\partial y}h(y) \neq 0$ for all $y \in \mathcal{W}$ such that $h(y) = 0$ is an *extent-compatible control barrier function* (Ec-CBF) for the system (1) with extent function E if there exists locally Lipschitz extended class $\mathcal{K}$ functions α_1 and α_2 such that for all $x \in \mathcal{D}$ with $\mathcal{E}(x) \subset \mathcal{C}$ and for all $y \in \mathcal{C}$, defining

$$\mathcal{M}(x, y, u) := \frac{\partial E(x, y)}{\partial x}(f(x) + g(x)u) + \alpha_1(E(x, y)) + \alpha_2(h(y)),$$

it holds that $\sup_{u \in \mathbb{R}^m}\{\mathcal{M}(x, y, u)\} \geq 0$.

Given an Ec-CBF h, for all $x \in \mathcal{D}$, define the set

$$\mathcal{U}(x) = \{u \in \mathbb{R}^m \mid \mathcal{M}(x, y, u) \geq 0 \text{ for all } y \in \mathcal{C}\}. \tag{4}$$

Assuming that the extent of the system initially begins inside the safe region, choosing a control input $u(x) \in \mathcal{U}(x)$ at any given state $x \in \mathcal{D}$ guarantees that $\mathcal{E}(x(t)) \subset \mathcal{C}$ for all $t \geq 0$, as formalized in the following theorem.

Theorem 1. *Consider system* (1) *with initial state* $x(0)$*, an extent function* E*, an Ec-CBF* h *with associated safe set* $\mathcal{C} = \{y \in \mathcal{W} \mid h(y) \geq 0\} \subset \mathcal{W}$*, and* $\mathcal{U}(x)$ *as defined in* (4)*. If* $\mathcal{E}(x(0)) \subset \mathcal{C}$*, then any continuous feedback controller* $u : \mathcal{D} \to \mathbb{R}^m$ *such that* $u(x) \in \mathcal{U}(x)$ *for all* $x \in \mathcal{D}$ *guarantees that* $\mathcal{E}(x(t)) \subset \mathcal{C}$ *for all* $t \geq 0$*.*

Proof. Suppose by contradiction that the assumptions of the theorem hold but there exists a time $t' > 0$ such that $\mathcal{E}(x(t')) \not\subset \mathcal{C}$, that is, the system is unsafe at time t'. By the defininition of $\mathcal{E}$ and $\mathcal{C}$, this means there exists a point $y' \in \mathcal{W}$ such that $E(x(t'), y') \leq 0$ and $h(y') < 0$. The first step of the proof is to establish that, in fact, there exists a time $0 < t^\dagger \leq t'$ and a point $y^\dagger \in \mathcal{W}$ such that $E(x(t^\dagger), y^\dagger) < 0$ and $h(y^\dagger) = 0$, *i.e.*, $y^\dagger$ is in the interior of the extent set at time $t^\dagger$ and on the boundary of the safe set $\mathcal{C}$. To see that this is true, first, without loss of generality, we can assume $E(x(t'), y') < 0$ because, if instead $E(x(t'), y') = 0$, then since $\frac{\partial}{\partial y} E(x(t'), y') \neq 0$ by the definition of extent function, for small enough ϵ, it holds that $E(x(t'), y'') < 0$ and $h(y'') < 0$ where $y'' = y' - \epsilon \frac{\partial}{\partial y} E(x(t'), y')$, and we could consider y'' instead of y'. Next, since the system is assumed to be initialized in safe conditions with $\mathcal{E}(x(0)) \subset \mathcal{C}$, and since trajectories are continuous, there must exist a time $t^\dagger$ and a point $y^\dagger$ when the system becomes unsafe, *i.e.*, there exists $0 < t^\dagger \leq t'$ and $y^\dagger \in \mathcal{W}$ such that $E(x(t^\dagger), y^\dagger) < 0$ and $h(y^\dagger) = 0$, as desired.

Now, let $w(t) = E(x(t), y^\dagger)$, *i.e.*, $w(t)$ is the value of the extent function at the point $y^\dagger$ over time. Since the system is assumed to be initially safe, it holds that $w(0) = E(x(0), y^\dagger) \geq 0$, and by construction, $w(t^\dagger) = E(x(t^\dagger), y^\dagger) < 0$. But, for all $t \geq 0$,

$$\begin{aligned} \dot{w}(t) &= \frac{\partial E(x(t), y^\dagger)}{\partial x} \big(f(x(t)) + g(x(t)) u(x(t)) \big) \\ &\geq -\alpha_1(w(t)) - \alpha_2(h(y^\dagger)) \\ &= -\alpha_1(w(t)) \,, \end{aligned}$$

where the first inequality holds since h is an Ec-CBF and the second equality follows because $-\alpha_2(h(y^\dagger)) = 0$.

Now, consider the initial value problem $\dot{\eta}(t) = -\alpha_1(\eta(t))$ with $\eta(0) = w(0)$. Note that $\eta(0) \geq 0$ since $w(0) \geq 0$. The comparison lemma [16, Lemma 3.4] then implies $w(t) \geq \eta(t) \geq 0$ for all $t \geq 0$. But this contradicts that $w(t^\dagger) < 0$. Hence, $\mathcal{E}(x(t)) \subset \mathcal{C}$ for all $t \geq 0$. □

Observe that when x is such that a point of the boundary of the extent set is on the boundary of the safe set, the condition (4) ensures that the control input u makes the system stay in the extent-set and the functions α_1 and α_2 are used to make this control action smoother. The only feature of α_2 used in the proof of Theorem 1 is that $\alpha_2(0) = 0$. The additional properties on α_2 imposed in Definition 2 are useful for practical implementation and exploited in Theorem 3.

In contrast to traditional CBFs, Ec-CBFs are defined over the workspace $\mathcal{W}$ and explicitly account for the system extent within the workspace via the the condition $\sup_{u\in\mathbb{R}^m}\{\mathcal{M}(x,y,u)\} \geq 0$, which depends on the system extent function E. Moreover, if the system extent changes (e.g., because a different robot is used or a vehicle docks with a trailer), then only the system extent function E need change, and the Ec-CBF h remains the same.

Next, we propose an optimization scheme for selecting inputs from the set $\mathcal{U}(x)$ to guarantee safety while minimally deviating from some prescribed nominal controller.

4. Minimally Invasive Quadratic Program Controller

In the scenario where a system designer would like to employ some possibly unsafe nominal feedback control policy $k : \mathcal{D} \to \mathbb{R}^m$ on the system (1) with extent, we propose incorporating (4) as a constraint at runtime to obtain a safe controller as a minimally invasive quadratic program (QP) using a Ec-CBF, similar to the technique proposed in [4] for ZCBFs. This procedure leads to a control law which ensures that the extent set of the system is contained within the safe set $\mathcal{C}$ for all $t \geq 0$, given that $\mathcal{E}(x(0)) \subset \mathcal{C}$. In particular, we propose a quadratic program solved for each x of the form

$$u_{\mathrm{QP}}(x) = \operatorname*{arg\,min}_{u\in\mathcal{U}(x)} \ \|u - k(x)\|_2^2 \,. \tag{5}$$

The above QP is minimally invasive in the sense that it guarantees safety of the system including its extent, while following the nominal input k with minimal deviation. However, for fixed x, $\mathcal{U}(x)$ is defined from (4) and requires a given inequality to hold for all y, leading to an infinite number of linear constraints on u. In the remainder of this section, we present two approaches that retain safety guarantees. The first one is an exact solution, but may not necessarily be computationally efficient. The second one is an approximate solution which still guarantees safety and is amenable for online implementation.

4.1. Optimization Over Sum-of-Squares Polynomials

In the first approach, we recast (5) as a sum-of-squares (SOS) optimization problem in the independent variable y. Recall that x is the *a priori* fixed current state of the system and y is a free variable denoting any point in the workspace $\mathcal{W}$.

Definition 3 (Sum-of-Squares (SOS) Polynomials)**.** A polynomial $s(y)$ is a *sum-of-squares* polynomial if it can be written as $s(y) = \sum_{i=1}^{\ell} p_i(y)^2$ for some natural number ℓ where each $p_i(y)$ is a polynomial. Let $\Sigma[y]$ denote the set of all SOS polynomials. Note that if $s(y) \in \Sigma[y]$, then $s(y) \geq 0$ for all $y \in \mathbb{R}^n$.

Theorem 2. *Consider system* (1) *with initial state* $x(0)$*, an extent function* E*, and an Ec-CBF* h *with associated safe set* $\mathcal{C} = \{y \in \mathcal{W} \mid h(y) \geq 0\} \subset \mathcal{W}$*. Further assume* $E(x,y)$ *and* $\alpha_1(E(x,y))$ *are polynomial in* y *for any fixed* x *and that* $h(y)$ *and* $\alpha_2(h(y))$ *are polynomial in* y*. Let* $k : \mathcal{D} \to \mathbb{R}$ *be a continuous nominal controller and suppose* $\mathcal{E}(x(0)) \subset \mathcal{C}$*. If the set*

$$\widetilde{\mathcal{U}}(x) = \Bigg\{ u \in \mathbb{R}^m \;\Bigg|\; \frac{\partial E(x,y)}{\partial x}(f(x) + g(x)u) + \alpha_1(E(x,y)) + \alpha_2(h(y)) - s(y)h(y) \in \Sigma[y] \text{ for some } s(y) \in \Sigma[y] \Bigg\}$$

is non-empty for all $x \in \mathcal{D}$*, then the solution* $x(t)$ *of system* (1) *with*

$$u(x) = u_{SOS}(x) := \operatorname*{arg\,min}_{u \in \widetilde{\mathcal{U}}(x)} \|u - k(x)\|_2^2 \tag{6}$$

is such that $\mathcal{E}(x(t)) \subset \mathcal{C}$ *for all* $t \geq 0$*.*

Proof. For each x, the optimization problem (6) is feasible by hypothesis, and the fact that $u \in \widetilde{\mathcal{U}}(x)$ implies

$$\frac{\partial E(x,y)}{\partial x}(f(x) + g(x)u) + \alpha_1(E(x,y)) + \alpha_2(h(y)) - s(y)h(y) \geq 0 \tag{7}$$

for all $y \in \mathbb{R}^n$, since the left hand side of the inequality is required to be an SOS polynomial. Next, observe that $s(y)h(y) \geq 0$ for all $y \in \mathcal{C}$ since $s(y)$ is a SOS polynomial, and for all points of the safe set, i.e., $y \in \mathcal{C}$, we have $h(y) \geq 0$

as per the definition of the safe set in Section 2. Hence the requirement that $u \in \widetilde{\mathcal{U}}(x)$ implies

$$\frac{\partial E(x,y)}{\partial x}(f(x)+g(x)u)+\alpha_1(E(x,y))+\alpha_2(h(y))\geq 0 \tag{8}$$

for all $y \in \mathcal{C}$, i.e., $u \in \mathcal{U}(x)$ as defined in (4). In addition, since for all $x \in \mathcal{D}$, the constraint in $\widetilde{\mathcal{U}}(x)$ is convex in u, $\|u\|$ is convex in u, and k is continuous in x, using [13, Theorem 5], we conclude that the controller is continuous. From Theorem 1, $\mathcal{E}(x(t)) \subset \mathcal{C}$ for all $t \geq 0$. □

To implement the SOS controller in, e.g., SOSTOOLS [10], the degree of the SOS decision polynomial $s(y)$ in the constraint defining $\widetilde{\mathcal{U}}(x)$ is fixed a priori. In addition, the quadratic cost in (6) is recast in epigraph form to obtain an equivalent problem with linear cost and an additional semidefinite constraint via *Schur complement* [17]; in particular, the initial formulation (6) is equivalent to

$$u_{\text{SOS}}(x) = \underset{u\in\widetilde{\mathcal{U}}(x)}{\arg\min}\ \min_{\delta\in\Delta(u,x)} \delta\,,$$

with

$$\Delta(u,x) = \left\{ \delta \in \mathbb{R} \;\middle|\; \begin{bmatrix} I & u \\ u^T & \delta + 2k(x)^T u - k(x)^T k(x) \end{bmatrix} \succeq 0 \right\}.$$

The above SOS approach allows us to adopt a tractable method to guarantee safety for the system. However, this approach has two drawbacks. First, it requires extent sets and safe sets to be defined by polynomial functions. Second, with increasing system dimensionality, SOS programs are known to become computationally difficult. Hence, we next propose a computationally efficient approach that replaces the infinite number of linear constraints of (5) with a finite number of constraints induced by a finite number of points sampled on the extent boundary.

4.2. A Sampling-Based Approach to Set Invariance with Extent

In this subsection, we propose an alternative relaxation of (5) which retains the computational advantages of the original QP formulation for ZCBFs. The intuition is to enforce the constraint in (4) on the boundary of the extent set, but only for a finite number of sampled points. To obtain a finite number of sampled points, we discretize the boundary of the extent set $\partial\mathcal{E}(x)$. The main technical difficulty is ensuring that a barrier condition at each sample point is sufficient to guarantee safety for the entire safe set. The following theorem formally guarantees safety of this sampling approach by supposing a constraint on the magnitude of the control input for each sampled point and utilizing bounds of the Lipschitz constants obtained from the Ec-CBF h and the extent function E to guarantee that the entire extent set boundary, and hence the entire extent set, remains within the safe set.

Theorem 3. *Consider system* (1) *with initial state* $x(0)$*, an extent function* E*, and an Ec-CBF* h *with associated safe set* $\mathcal{C} = \{y \in \mathcal{W} \mid h(y) \geq 0\} \subset \mathcal{W}$*. Further assume that the domain* $\mathcal{D}$ *and the workspace* $\mathcal{W}$ *are bounded,* $\mathcal{E}(x)$ *is bounded for all* $x \in \mathcal{D}$*, let* $M > 0$ *be a bound on the magnitude of the control input, and for some* $\tau > 0$ *let* $\partial\mathcal{E}_\tau(x) \subset \partial\mathcal{E}(x)$ *be a finite set such that for all* $y \in \partial\mathcal{E}(x)$*, it holds that* $\min_{\widetilde{y}\in\partial\mathcal{E}_\tau(x)} \|\widetilde{y} - y\| \leq \tau/2$*. Additionally, let*

$$A \geq \sup_{y\in\mathcal{W}} \left\| \frac{\partial}{\partial y} h(y) \right\|, \tag{9}$$

$$B \geq \sup_{x\in\mathcal{D}, y\in\mathcal{W},\, \|u\|<M} \left\| \frac{\partial^2 E(x,y)}{\partial x \partial y} (f(x) + g(x)u) \right\|. \tag{10}$$

Consider the set

$$\widehat{\mathcal{U}}(x) = \Bigg\{ u \in \mathbb{R}^m \;\Big|\; \|u\| \leq M \text{ and } \frac{\partial E(x,y^*)}{\partial x}(f(x) + g(x)u) + \gamma \cdot h(y^*) \geq (B + \gamma A)\tau \text{ holds for all } y^* \in \partial\mathcal{E}_\tau(x) \Bigg\} \tag{11}$$

where $\gamma > 0$ *is a constant. Let* $k : \mathcal{D} \to \mathbb{R}$ *be a continuous nominal control input. If for all* $x \in \mathcal{D}$ *the set* $\widehat{\mathcal{U}}(x)$ *is non-empty and* $\mathcal{E}(x(0)) \subset \mathcal{C}$*, then the solution* $x(t)$ *to the system* (1) *with* $u = u_{sampled}(x)$*, where*

$$u_{sampled}(x) = \underset{u\in\widehat{\mathcal{U}}(x)}{\arg\min} \|u - k(x)\|_2^2, \tag{12}$$

is such that $\mathcal{E}(x(t)) \subset \mathcal{C}$ for all $t \geq 0$. Moreover, the controller $u_{sampled}(x)$ is continuous with respect to x for all $x \in \mathcal{D}$.

Proof. Introduce the open set

$$\mathcal{B} = \bigcup_{y^* \in \partial\mathcal{E}_\tau(x)} B^o_{\frac{3\tau}{4}}(y^*)\,,$$

where $B^o_{\frac{3\tau}{4}}(y^*)$ denotes an open ball with radius $\frac{3\tau}{4}$ centered around y^*. Choose $\epsilon > 0$ such that for all x, $\bar{\mathcal{E}}(x) = \{y \in \mathcal{W} \mid |E(x,y)| \leq \epsilon\} \subseteq \mathcal{B}$. Such a choice of ϵ is possible due to part 3 of the definition of the extent function (Definition 1). Clearly $\partial\mathcal{E}_\tau(x) \subset \partial\mathcal{E}(x) \subset \bar{\mathcal{E}}(x)$. From the mean value theorem, for all $y \in \bar{\mathcal{E}}(x)$ and $y^* = \arg\min_{\bar{y} \in \partial\mathcal{E}_\tau(x)} \|\bar{y} - y\|$, it follows that

$$h(y^*) - h(y) \leq A\|y^* - y\|\,, \tag{13}$$

and

$$\left(\frac{\partial E(x,y^*)}{\partial x} - \frac{\partial E(x,y)}{\partial x}\right)(f(x) + g(x)u) \leq B\|y^* - y\| \tag{14}$$

whenever $\|u\| \leq M$. Now, observe that $\|y^* - y\| \leq 3\tau/4$. Multiplying (13) with $-\gamma$ and (14) with -1 and then adding the inequalities yields

$$\begin{aligned}
&\frac{\partial E(x,y)}{\partial x}(f(x) + g(x)u) + \gamma \cdot h(y) \\
&\geq \frac{\partial E(x,y^*)}{\partial x}(f(x) + g(x)u) + \gamma \cdot h(y^*) - (B + \gamma A)3\tau/4 \\
&\geq (B + \gamma A)\tau/4
\end{aligned}$$

for any $u \in \widehat{\mathcal{U}}(x)$ where the last inequality follows from the definition of $\widehat{\mathcal{U}}(x)$.

Choose $\alpha_2(s) = \gamma s$ for all $s \geq 0$ and $\alpha_1(s) = (B + \gamma A)\tau/(4\epsilon)s$ for all $s \leq |\epsilon|$. Then, for any $u \in \widehat{\mathcal{U}}(x)$, we have $\mathcal{M}(x,y,u) \geq 0$ whenever $|E(x,y)| \leq \epsilon$ where $\mathcal{M}$ is as in Definition 2. Moreover, it is straightforward to see that α_1 can be chosen to be sufficiently large for $s > |\epsilon|$ so that $\mathcal{M}(x,y,u) \geq 0$ for $u \in \widehat{\mathcal{U}}(x)$ also when $|E(x,y)| > \epsilon$. Thus, h is a Ec-CBF and $u_{\text{sampled}}(x) \in \mathcal{U}(x)$ for all $x \in \mathcal{D}$. Since for all $y^* \in \partial\mathcal{E}_\tau(x)$, in the definition of $\widehat{\mathcal{U}}(x)$, the constraint

$$\frac{\partial E(x,y^*)}{\partial x}(f(x) + g(x)u) + \gamma \cdot h(y^*) \geq (B + \gamma A)\tau$$

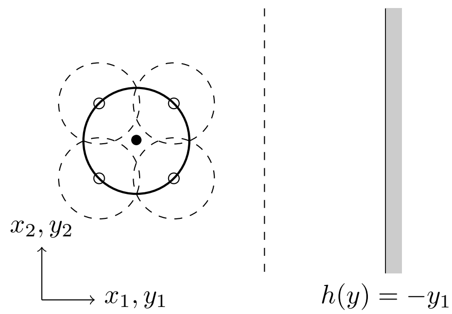


Figure 2: Example of the different discretizations in Example 1. The black dot is the system and the solid circle its extent. The four discretion points are with circles and the distances with dashed circles. The dashed line show how close the discretizations points can be to the barrier and still satisfy the inequality.

is convex in u, $\|u\|$ is convex in u and k is continuous in x, using [13, Theorem 5], we conclude that the controller $u_{\text{sampled}}(x)$ is continuous. Thus, from Theorem 1, the extent set $\mathcal{E}(x)$ is contained within the safe set for all $t \geq 0$ and for all $x \in \mathcal{D}$ such that $E(x(0)) \subset \mathcal{C}$; that is, $\mathcal{E}(x(t)) \subset \mathcal{C}$ for all $t \geq 0$. □

The constants (9) and (10) are interpreted as upper bounds of the Lipschitz constants for functions appearing in the definition of Ec-CBF. Whenever the domain $\mathcal{D}$ and workspace $\mathcal{W}$ are bounded, as is usually the case in practice, such constants will exist. Moreover, since the constants A and B could potentially be large, the constant τ must be chosen small enough so that $\widehat{\mathcal{U}}(x)$ is nonempty and, as we show in the following example, choosing τ to be large can result in unwanted conservatism.

Example 1. Consider the system $\dot{x} = u$, where $x = \begin{bmatrix} x_1 & x_2 \end{bmatrix}^T \in \mathbb{R}^2$ is the system state and $u \in \mathbb{R}^2$ is a bounded control input such that $\|u\|_2 \leq M$ where $M = 1$. We take $\mathcal{D} = \mathcal{W} = \mathbb{R}^2$ and $E(x, y) = (x_1 - y_1)^2 + (x_2 - y_2)^2 - 1$ and encode the safety constraint with the safe function $h(y) = -y_1$. As noted earlier, although the safe set is not bounded in this example, forward completeness can still be guaranteed due to bounded control input, and hence the presented results still hold true in this setting. This problem setting is depicted in Fig. 2. We take the upper bound of the Lipschitz constants as $A = 1$ and $B = 2$ satisfying (9) and (10). Observe that for this specific choice of extent function, the constants can be determined although the domain and workspace are unbounded. In practice, systems operate in a compact domain, and hence, the constants A and B exist and can be computed.

To demonstrate the conservatism with few sampled points, we consider a sampling-based controller which uses four samples; that is, we take $\partial\mathcal{E}_\tau(x) = \{(x_1 \pm 1/\sqrt{2}, x_2 \pm 1/\sqrt{2}), (x_1 \pm 1/\sqrt{2}, x_2 \mp 1/\sqrt{2})\}$, such that $\tau = 2\sqrt{2-\sqrt{2}}$. The controller (12) then has four constraints,

$$\sqrt{2}(-u_1 \pm u_2) - \gamma\left(x_1 - \sqrt{2}\right) \geq (B + \gamma A)\tau\,, \tag{15}$$

$$\sqrt{2}(u_1 \pm u_2) - \gamma\left(x_1 + \sqrt{2}\right) \geq (B + \gamma A)\tau\,. \tag{16}$$

In the instance that $x_1 = -(\frac{B}{\gamma} + A)\tau$, the only feasible solution is $(u_1, u_2) = (-\gamma, 0)$ with $\gamma \leq M$, which will effectively steer the system away from the barrier. Observe that γ plays a key role in the behavior of the system. A higher value of γ will allow for the system to get closer to the barrier, but once the system is close to the boundary, a more aggressive control action, i.e., $u_1 = \gamma$ is applied.

Intuitively, the sampling-based technique essentially covers the boundary of the extent set with balls around the discretized points, and ensures that the balls do not ever cross over into the unsafe set, as is visualized in Fig 2.

5. Experimental Results

In this section, we present a case study[2] of the proposed framework implemented in the Robotarium testbed [15] on a differential drive robot with dynamics

$$\dot{x}_1 = v \cdot \cos(\phi)\,, \quad \dot{x}_2 = v \cdot \sin(\phi)\,, \quad \dot{\phi} = \omega\,,$$

[2] Source code for the implementation is available at `https://github.com/gtfactslab/ExtentCBF`

where $x_1 \in \mathbb{R}$, $x_2 \in \mathbb{R}$ are the position coordinates of the robot, $\phi \in [-\pi, \pi)$ is the orientation, and $v \in \mathbb{R}$ and $\omega \in \mathbb{R}$ are the linear velocity input and angular velocity input. Define the system state by $x = \begin{bmatrix} x_1 & x_2 & \phi \end{bmatrix}^T$. The Robotarium workspace is taken to be $\mathcal{W} = [-1.6, 1.6] \times [-1, 1] \subseteq \mathbb{R}^2$ and the system domain $\mathcal{D}$ is defined by $x_1 \in [-1.6, 1.6]$, $x_2 \in [-1, 1]$, and $\phi \in [-\pi, \pi)$, that is, the domain $\mathcal{D}$ consists of the workspace $\mathcal{W}$ augmented with the robot's orientation. We consider an ellipsoidal safe set $\mathcal{C} = \{y \in \mathbb{R}^2 \mid h(y) \geq 0\}$ where $h(y) = 1 - y^T P_{\text{safe}} y$ with $P_{\text{safe}} = \text{diag}(1^{-2}, 0.8^{-2}) \in \mathbb{R}^{2\times 2}$, and the extent set as a fourth-order superellipse described by the extent function

$$E(x, y) = (5)^4 (\Delta_1 \cos(\phi) + \Delta_2 \sin(\phi))^4 + (10)^4 (-\Delta_1 \sin(\phi) + \Delta_2 \cos(\phi))^4 - 1\,, \quad (17)$$

where $\Delta_i = (x_i - y_i)$ for $i = 1, 2$.

We take the nominal control as $u_{\text{nom}} = \begin{bmatrix} 1 & 0.4 \end{bmatrix}^T$, with a simulation horizon of 1000 iterations. To highlight the benefit of the proposed method, we also implement a classical CBF solution, where the controller uses a shrunken safe set to account for the extent of the system. In implementation, the CBF controller enforced the forward invariance of $\tilde{h}(y) = 1 - y^T \tilde{P}_{\text{safe}} y$, where $\tilde{P}_{\text{safe}} = \text{diag}(0.8^{-2}, 0.6^{-2})$, which is a subset of the true safe set $\mathcal{C}$. We compare the trajectories generated from the sampling-based approach, SOS approach and the classical approach in Fig 3. For this particular choice of nominal control input, the SOS controller makes slower progress along the desired trajectory; this is because, once the robot gets closer to the safe set boundary, the safe input velocities become small. For the sampling based controller, we take $\gamma = 0.06$ and considered $\tau = 0.001$, $\tau = 0.0005$, $\tau = 0.0002$, and $\tau = 0.0001$. The sampled points along the boundary were then generated by parameterizing the boundary and constructively finding the angle that gives a new sample point τ away from the previous one. The average time required to compute the control law for the controllers is shown in Table 1.

In Fig. 4, we plot the minimum value of the Ec-CBF evaluated over all sampled points for cases in Table 1. Observe that all the values are strictly

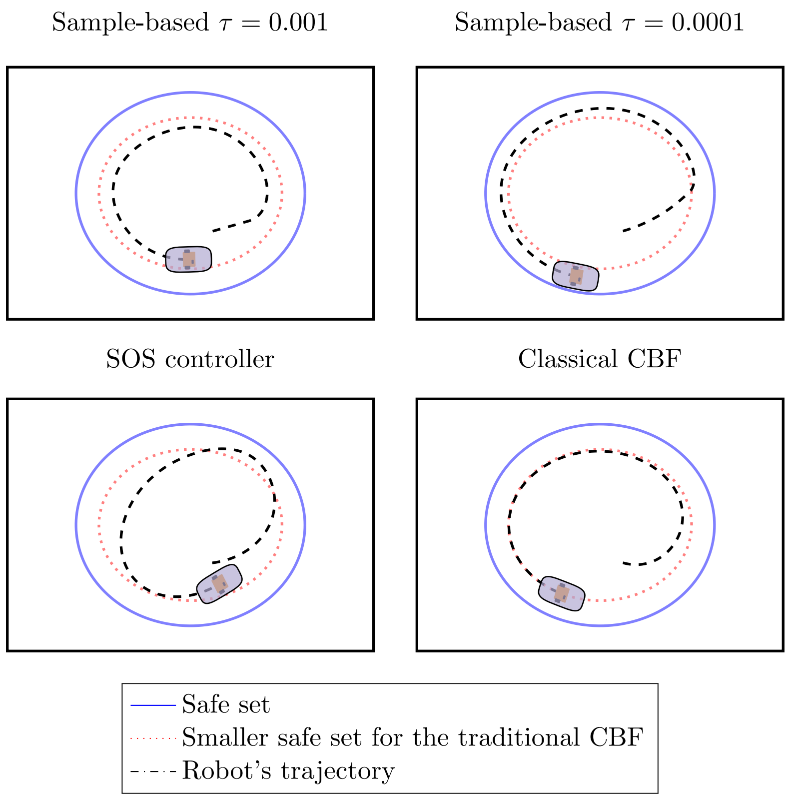


Figure 3: Trajectories for the sample-based controller with $\tau = 0.001$ and $\tau = 0.0001$, as well as the SOS controller and the traditional CBF controller. The boundary of the safe set in the workspace is shown in solid blue. The traditional CBF controller must use a smaller safe set in the statespace to accommodate the system's extent; the boundary of this smaller set is shown in dotted red. The plots demonstrate that conservatism of the sampling-based approach decreases as τ decreases and the number of samples increases. Moreover, due to the non-circular shape of the extent set, the traditional CBF controller is more conservate compared to both the sample-based controller with $\tau = 0.0001$ and the SOS controller.

Table 1: Average computation time for the controllers in the case study

τ	**No. of Samples**	**Average Run Time (in milliseconds)**
0.001	1044	18
0.0005	2088	35
0.0002	5252	83
0.0001	9436	110
SOS controller	–	2200
Classical CBF	–	3.9

positive, thus implying that none of the sampled points have violated the safe set. Also, observe that for $\tau = 0.0005$ or less, the value of $h(y)$ is lower compared to the classical CBF solution, which implies that the robot will be closer to the boundary of the safe set.

The robot's safe trajectory is influenced by the choice of τ, where a smaller τ requires more samples, as show in Table 1. In Fig. 3, we show how the trajectories for the robot differ with $\tau = 0.001$ and $\tau = 0.0001$. As expected and per the discussion in Example 1, we observe that there is larger conservatism when lower number of samples are considered. In this case, the robot does not venture close to the boundary of the safe set. However, with large number of samples, we observer that the robot gets close to the boundary. Fig. 3 also shows that both the sample based controller with $\tau = 0.0001$ and the SOS controller allows the robot to get closer to the safe set's boundary compared to the classical CBF solution.

From the above discussion, we observe that using the sampling-based controller results in a computationally efficient controller which guarantees safety, albeit it can sometimes require a conservative amount of samples. To showcase both of these facts, we implemented[3] the sampling-based technique on the Rob-

[3]Video of the experiment available at `https://youtu.be/WH99Fknxc8o`

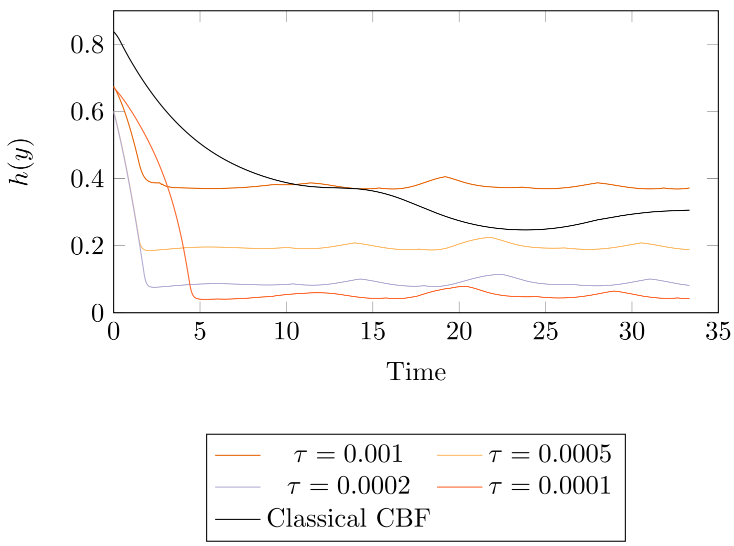
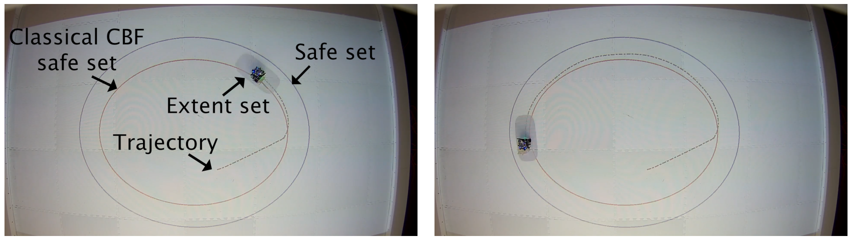


Figure 4: Minimum value of the Ec-CBF $h(y)$ evaluated for $\tau = 0.001$, $\tau = 0.0005$, $\tau = 0.0002$, and $\tau = 0.0001$ at the sampled points (i.e. for all $y \in \partial\mathcal{E}_\tau(x)$) on the boundary of the extent set. Observe that all values for each case are strictly positive, thus implying that none of the sampled points cross into the unsafe region. For comparison, the classical CBF solution is included that uses a smaller safe set. The classical solution is more conservative than almost all of the sample-based solutions.



(a) (b)

Figure 5: Overhead images from the implementation on the Robotarium test-bed. Each of the sub-figures indicate the trajectory of the robot generated over the course of the implementation. Observe that although the nominal controller tries to push the robot outside the safe set, the sampling-based controller (12) ensures that the robot state and its extent stay within the safe set.

otarium testbed. Snapshots of the experiment are shown in Fig 5. In particular, we considered a setting where the τ-value in eq. (11) was 10 times smaller than what it should be according to Theorem 3. As can be seen, the trajectory of the robot is such that safety of the system volume is guaranteed over the period of the entire experiment. It can also be seen in Fig 5, that robot is allowed to get closer to the boundary of the safe set, compared to a classical CBF solution that would have required the robot to stay inside the set bounded by the inner circle.

6. Concluding Remarks

This paper presents a barrier function-based method for ensuring the safe control of a control-affine dynamical system that incorporates its physical volume, i.e., its extent. The first proposed controller design relies on a sum-of-squares optimization program. The sum-of-squares approach is conceptually appealing and does not require knowledge of, *e.g.*, explicit bounds of the Lipschitz constants, however, sum-of-squares programs can be computationally difficult. Therefore, a sampling method is proposed as an alternative. This alternative controller is shown to retain the guarantee on safety of the system but can be more conservative than the sum-of-squares approach. Simulation and robotic implementation results are also provided. As part of future work, for systems with high relative degree, an interesting direction is to extend our framework using a methodology similar to that used to extend barrier functions to high relative degree systems [18].

References

[1] F. Blanchini, S. Miani, Set-theoretic methods in control, Springer, 2008. doi:10.1007/978-3-319-17933-9.

[2] I. Mitchell, C. J. Tomlin, Level set methods for computation in hybrid systems, in: International Workshop on Hybrid Systems: Computation and Control, Springer, 2000, pp. 310–323. doi:10.1007/3-540-46430-1_27.

[3] T. Gurriet, M. Mote, A. D. Ames, E. Feron, An online approach to active set invariance, in: 2018 IEEE Conference on Decision and Control (CDC), 2018, pp. 3592–3599. doi:10.1109/CDC.2018.8619139.

[4] A. D. Ames, J. W. Grizzle, 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. doi:10.1109/CDC.2014.7040372.

[5] L. Wang, A. D. Ames, M. Egerstedt, Safety barrier certificates for collisions-free multirobot systems, IEEE Transactions on Robotics 33 (3) (2017) 661–674. doi:10.1109/TRO.2017.2659727.

[6] M. Srinivasan, S. Coogan, M. Egerstedt, Control of multi-agent systems with finite time control barrier certificates and temporal logic, in: 2018 IEEE Conference on Decision and Control (CDC), 2018, pp. 1991–1996. doi:10.1109/CDC.2018.8619113.

[7] A. Singletary, P. Nilsson, T. Gurriet, A. D. Ames, Online active safety for robotic manipulators, in: 2019 IEEE/RSJ International Conference on

Intelligent Robots and Systems (IROS), 2019, pp. 173–178. doi:10.1109/IROS40897.2019.8968231.

[8] K. K. Hauser, Minimum constraint displacement motion planning., in: Robotics: Science and Systems, 2013.

[9] L. Wang, A. D. Ames, M. Egerstedt, Safe certificate-based maneuvers for teams of quadrotors using differential flatness, in: 2017 IEEE International Conference on Robotics and Automation (ICRA), 2017, pp. 3293–3298. doi:10.1109/ICRA.2017.7989375.

[10] A. Papachristodoulou, J. Anderson, S. P. G. Valmorbida, P. Seiler, P. A. Parrilo, SOSTOOLS: Sum of squares optimization toolbox for MATLAB, http://arxiv.org/abs/1310.4716 (2013).

[11] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, P. Tabuada, Control barrier functions: Theory and applications, in: 2019 18th European Control Conference (ECC), 2019, pp. 3420–3431. doi:10.23919/ECC.2019.8796030.

[12] X. Xu, P. Tabuada, J. W. Grizzle, A. D. Ames, Robustness of control barrier functions for safety critical control, IFAC-PapersOnLine 48 (27) (2015) 54–61.

[13] R. Konda, A. D. Ames, S. Coogan, Characterizing safety: Minimal control barrier functions from scalar comparison systems, IEEE Control Systems Letters 5 (2) (2021) 523–528. doi:10.1109/LCSYS.2020.3003887.

[14] H. M. Choset, S. Hutchinson, K. M. Lynch, G. Kantor, W. Burgard, L. E. Kavraki, S. Thrun, R. C. Arkin, Principles of robot motion: theory, algorithms, and implementation, MIT press, 2005.

[15] D. Pickem, P. Glotfelter, L. Wang, M. Mote, A. Ames, E. Feron, M. Egerstedt, The Robotarium: A remotely accessible swarm robotics research testbed, in: 2017 IEEE International Conference on Robotics and Automation (ICRA), 2017, pp. 1699–1706. doi:10.1109/ICRA.2017.7989200.

[16] H. Khalil, Nonlinear Systems, Pearson Education, Prentice Hall, 2002.

[17] L. Vandenberghe, S. Boyd, Semidefinite programming, SIAM review 38 (1) (1996) 49–95. doi:10.1137/1038003.

[18] W. Xiao, C. Belta, Control barrier functions for systems with high relative degree, in: 2019 IEEE 58th Conference on Decision and Control (CDC), 2019, pp. 474–479. doi:10.1109/CDC40024.2019.9029455.