Safe Backstepping with Control Barrier Functions

Andrew J. Taylor, Pio Ong, Tamas G. Molnar, and Aaron D. Ames

***Abstract*—Complex control systems are often described in a layered fashion, represented as *higher-order systems* where the inputs appear after a chain of integrators. While Control Barrier Functions (CBFs) have proven to be powerful tools for safety-critical controller design of nonlinear systems, their application to higher-order systems adds complexity to the controller synthesis process—it necessitates dynamically extending the CBF to include higher order terms, which consequently modifies the safe set in complex ways. We propose an alternative approach for addressing safety of higher-order systems through *Control Barrier Function Backstepping*. Drawing inspiration from the method of Lyapunov backstepping, we provide a constructive framework for synthesizing safety-critical controllers and CBFs for higher-order systems from a top-level dynamics safety specification and controller design. Furthermore, we integrate the proposed method with Lyapunov backstepping, allowing the tasks of stability and safety to be expressed individually but achieved jointly. We demonstrate the efficacy of this approach in simulation.**

I. Introduction

Safety is becoming an ever more prevalent design consideration in modern control systems as these systems are deployed in real-world environments. Control Barrier Functions (CBFs) have become a popular tool for constructively synthesizing controllers that endow nonlinear systems with rigorous guarantees of safety [1], [2]. Originally posed such that the input of the system directly impacted the time derivative of the CBF, recent work has sought to extend this to higher-order nonlinear systems in which multiple time derivatives are required for the input to influence the evolution of the CBF [3]–[6]. While these works allow for the safety-critical control of higher-order systems, they require verifying the feasibility of CBF conditions using the full system dynamics and change the safe set in complex ways. Alternatively, the work in [7] has explored designing CBFs for a top-level model, and using a tracking controller that addresses the full system dynamics.

As the complexity of systems increase, it is often desirable to approach the control design process with a simplified top-level model that guides design for subsystems addressing the full system dynamics. Backstepping is a well established design technique for addressing the robust stabilization of layered systems of this form, i.e., nonlinear systems with *higher-order dynamics* [8], [9]. It considers design for the top-level model and recursively designs a controller using the full system dynamics, also allowing it to address the challenge of *mixed-relative degree*, where inputs enter the system dynamics at different levels. Using backstepping to stabilize systems while meeting state constraints has been studied through lens of non-overshooting control [10], and has recently been related to CBFs [11], [12]. These works achieve safe behavior using a structured controller that yields a linear dynamic relationship between sequential states in a cascade, such that a system does not overshoot a setpoint as it stabilizes. Other work has used backstepping in the context of Lyapunov-Barrier functions to ensure state constraints are met [13]–[15]. These approaches couple ensuring safety with ensuring stability, which may impose strict structural requirements on safety constraints. To the best of our knowledge, decoupling stability and safety and exploring backstepping purely with safety constraints expressed through CBFs has not been considered.

A core challenge in combining CBF-based methods with backstepping is finding smooth controllers that ensure safety as backstepping requires the differentiation of controllers appearing higher in the integrator chain. From the conception of CBFs, they have typically been used as constraints in optimization-based controllers—either paired with CLFs [1], or filtering a desired stabilizing controller [2]—and therefore are inherently non-smooth. Additionally, typically one wishes to design controllers that are not only safe, but are also stabilizing, precluding smooth CBF controller instantiations, e.g., using Sontag's Universal formula [16]. While it may be possible to address these non-smooth challenges [17], [18], we will consider the approach in [19] for synthesizing smooth controllers meeting both CLF and CBF constraints.

The goal of this paper is to to unify backstepping with CBFs, thereby enabling safe controller design at multiple levels with varying degrees of model complexity. To this end, after a review of CBFs and Lyapunov backstepping, we begin in Section III by formulating a nonlinear controller that ensures safety of a system with a single cascade via Barrier Functions and backstepping. A consequence of this result is that we may constructively synthesize a CBF for the full cascaded system using a CBF and smooth controller designed only considering the top-level of the system, which is often easier than directly finding a CBF for the full-order system. Additionally, in Section IV, we demonstrate that this approach can be generalized to the multiple-cascade setting, and address the challenge of mixed relative-degree systems. The main result of this paper, presented in Section V, is the unification of Lyapunov and Barrier backstepping, wherein we show that by designing a controller that renders the top-level dynamics both stable and safe, we may use backstepping to achieve stability and safety of the full cascaded system. Importantly, using the techniques in [19], we are able to design a smooth top-level controller amenable to backstepping. These results are demonstrated in simulation in Section VI on multiple examples in the context of obstacle avoidance.

This research is supported in part by Ford, the National Science Foundation (CPS Award #1932091, CMMI Award #1923239), Raytheon Technologies, Aerovironment and Dow (#227027AT).

The authors are with the Department of Computing and Mathematical Sciences and the Department of Mechanical and Civil Engineering, California Institute of Technology, Pasadena, CA 91125, USA, {ajtaylor, pioong, tmolnar, ames}@caltech.edu.

II. Background

In this section we revisit Barrier Functions, Control Barrier Functions and Lyapunov backstepping as a precursor to introducing Control Barrier Function backstepping.

Consider a nonlinear control-affine system:

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

with state $\mathbf{x} \in \mathbb{R}^n$, input $\mathbf{u} \in \mathbb{R}^m$, and functions $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$ and $\mathbf{g} : \mathbb{R}^n \to \mathbb{R}^{n\times m}$ assumed to be locally Lipschitz continuous on $\mathbb{R}^n$. A locally Lipschitz continuous controller $\mathbf{k} : \mathbb{R}^n \to \mathbb{R}^m$ yields the closed loop system:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{k}(\mathbf{x}). \tag{2}$$

As the functions $\mathbf{f}$, $\mathbf{g}$, and $\mathbf{k}$ are locally Lipschitz continuous, for any initial condition $\mathbf{x}_0 \in \mathbb{R}^n$, there exists a maximal time interval $I(\mathbf{x}_0) = [0, t_{\max}(\mathbf{x}_0))$ and a unique continuously differentiable solution $\boldsymbol{\varphi} : I(\mathbf{x}_0) \to \mathbb{R}^n$ satisfying:

$$\dot{\boldsymbol{\varphi}}(t) = \mathbf{f}(\boldsymbol{\varphi}(t)) + \mathbf{g}(\boldsymbol{\varphi}(t))\mathbf{k}(\boldsymbol{\varphi}(t)), \tag{3}$$

$$\boldsymbol{\varphi}(0) = \mathbf{x}_0, \tag{4}$$

for all $t \in I(\mathbf{x}_0)$ [20].

A. Control Barrier Functions

We define the notion of safety in this context as forward invariance of a set in the state space. Specifically, suppose there exists a set $\mathcal{C} \subset \mathbb{R}^n$ defined as the 0-superlevel set of a continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$:

$$\mathcal{C} = \{\mathbf{x} \in \mathbb{R}^n \mid h(\mathbf{x}) \geq 0\}. \tag{5}$$

The set $\mathcal{C}$ is said to be *forward invariant* if for any initial condition $\mathbf{x}_0 \in \mathcal{C}$, we have $\boldsymbol{\varphi}(t) \in \mathcal{C}$ for all $t \in I(\mathbf{x}_0)$. In this case, we call the system (2) *safe* with respect to the set $\mathcal{C}$, and refer to $\mathcal{C}$ as the *safe set*.

Before defining Barrier Functions and Control Barrier Functions, we recall the following definitions. A continuous function $\alpha : [0,\infty) \to [0,\infty)$ is said to be *class* $\mathcal{K}_\infty$ ($\alpha \in \mathcal{K}_\infty$) if α is strictly monotonically increasing with $\alpha(0) = 0$ and $\lim_{r\to\infty} \alpha(r) = \infty$, and a continuous function $\alpha : \mathbb{R} \to \mathbb{R}$ is said to be *extended class* $\mathcal{K}_\infty$ ($\alpha \in \mathcal{K}^{\mathrm{e}}_\infty$) if it belongs to $\mathcal{K}_\infty$ and $\lim_{r\to-\infty} \alpha(r) = -\infty$. We now define Barrier Functions:

Definition 1 (*Barrier Function (BF)* [21])**.** Let $\mathcal{C} \subset \mathbb{R}^n$ be the 0-superlevel set of a continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$ with $\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x}) \neq \mathbf{0}$ when $h(\mathbf{x}) = 0$. The function h is a *Barrier Function* (BF) for (2) on $\mathcal{C}$ if there exists $\alpha \in \mathcal{K}^{\mathrm{e}}_\infty$ such that for all $\mathbf{x} \in \mathbb{R}^n$:

$$\underbrace{\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x})\mathbf{f}(\mathbf{x})}_{L_{\mathbf{f}}h(\mathbf{x})} + \underbrace{\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x})\mathbf{g}(\mathbf{x})}_{L_{\mathbf{g}}h(\mathbf{x})}\mathbf{k}(\mathbf{x}) \geq -\alpha(h(\mathbf{x})). \tag{6}$$

We have the following result establishing the safety of a set $\mathcal{C}$ for the closed-loop system (2) through Barrier Functions:

Theorem 1 ([21], [22])**.** *Let $\mathcal{C} \subset \mathbb{R}^n$ be the 0-superlevel set of a continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$ with $\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x}) \neq \mathbf{0}$ when $h(\mathbf{x}) = 0$. If h is a BF for (2) on $\mathcal{C}$, then the system (2) is safe with respect to the set $\mathcal{C}$.*

Control Barrier Functions provide a tool for synthesizing controllers that enforce the safety of $\mathcal{C}$:

Definition 2 (*Control Barrier Function (CBF)* [21])**.** Let $\mathcal{C} \subset \mathbb{R}^n$ be the 0-superlevel set of a continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$ with $\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x}) \neq \mathbf{0}$ when $h(\mathbf{x}) = 0$. The function h is a *Control Barrier Function* (CBF) for (1) on $\mathcal{C}$ if there exists $\alpha \in \mathcal{K}^{\mathrm{e}}_\infty$ such that for all $\mathbf{x} \in \mathbb{R}^n$:

$$\sup_{\mathbf{u}\in\mathbb{R}^m} \dot{h}(\mathbf{x},\mathbf{u}) \triangleq \sup_{\mathbf{u}\in\mathbb{R}^m} L_{\mathbf{f}}h(\mathbf{x}) + L_{\mathbf{g}}h(\mathbf{x})\mathbf{u} > -\alpha(h(\mathbf{x})). \tag{7}$$

Given a CBF h for (1) and a corresponding $\alpha \in \mathcal{K}^{\mathrm{e}}_\infty$, we define the point-wise set of control values:

$$K_{\mathrm{CBF}}(\mathbf{x}) = \left\{\mathbf{u} \in \mathbb{R}^m \;\middle|\; \dot{h}(\mathbf{x},\mathbf{u}) \geq -\alpha(h(\mathbf{x}))\right\}. \tag{8}$$

This yields the following result:

Theorem 2 ([21])**.** *Let $\mathcal{C} \subset \mathbb{R}^n$ be the 0-superlevel set of a continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$ with $\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x}) \neq \mathbf{0}$ when $h(\mathbf{x}) = 0$. If h is a CBF for (1) on $\mathcal{C}$, then the set $K_{\mathrm{CBF}}(\mathbf{x})$ is non-empty for all $\mathbf{x} \in \mathbb{R}^n$, and for any locally Lipschitz continuous controller $\mathbf{k}$ with $\mathbf{k}(\mathbf{x}) \in K_{\mathrm{CBF}}(\mathbf{x})$ for all $\mathbf{x} \in \mathbb{R}^n$, the function h is a BF for (2) on $\mathcal{C}$.*

Remark 1. The strict inequality in (7) serves two purposes. First, it ensures the set (8) is non-empty (as with a non-strict inequality in (6), the supremum may hold with equality, but there may be no input such that the supremum is attained). Second, strictness enables proving optimization-based controllers using CBFs are locally Lipschitz continuous [23].

B. Lyapunov Backstepping

Consider now a nonlinear control-affine system of the form:

$$\dot{\mathbf{x}} = \mathbf{f}_0(\mathbf{x}) + \mathbf{g}_0(\mathbf{x})\boldsymbol{\xi}, \tag{9}$$

$$\dot{\boldsymbol{\xi}} = \mathbf{f}_1(\mathbf{x},\boldsymbol{\xi}) + \mathbf{g}_1(\mathbf{x},\boldsymbol{\xi})\mathbf{u}, \tag{10}$$

with $\mathbf{x} \in \mathbb{R}^n$, $\boldsymbol{\xi} \in \mathbb{R}^p$, and $\mathbf{u} \in \mathbb{R}^m$, and functions $\mathbf{f}_0 : \mathbb{R}^n \to \mathbb{R}^n$, $\mathbf{g}_0 : \mathbb{R}^n \to \mathbb{R}^{n\times p}$, $\mathbf{f}_1 : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}^p$, and $\mathbf{g}_1 : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}^{p\times m}$ assumed to be locally Lipschitz continuous on their respective domains. This system is referred to as being in *strict-feedback form*. We further assume that $\mathbf{f}_0(\mathbf{0}) = \mathbf{f}_1(\mathbf{0},\mathbf{0}) = \mathbf{0}$ and $\mathbf{g}_1$ is pseudo-invertible on $\mathbb{R}^n \times \mathbb{R}^p$. As before, given a locally Lipschitz continuous feedback controller $\mathbf{k} : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}^m$ yielding the closed-loop system:

$$\dot{\mathbf{x}} = \mathbf{f}_0(\mathbf{x}) + \mathbf{g}_0(\mathbf{x})\boldsymbol{\xi}, \tag{11}$$

$$\dot{\boldsymbol{\xi}} = \mathbf{f}_1(\mathbf{x},\boldsymbol{\xi}) + \mathbf{g}_1(\mathbf{x},\boldsymbol{\xi})\mathbf{k}(\mathbf{x},\boldsymbol{\xi}), \tag{12}$$

for any initial condition $(\mathbf{x}_0,\boldsymbol{\xi}_0) \in \mathbb{R}^n \times \mathbb{R}^p$ there exists a maximum time interval $I((\mathbf{x}_0,\boldsymbol{\xi}_0)) \subseteq \mathbb{R}_{\geq 0}$ and a unique solution denoted by $\boldsymbol{\varphi} = (\boldsymbol{\varphi}_{\mathbf{x}}, \boldsymbol{\varphi}_{\boldsymbol{\xi}})$ satisfying (3)-(4) $\forall t \in I((\mathbf{x}_0,\boldsymbol{\xi}_0))$.

Suppose there exist a function $V_0 : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ and a function $\mathbf{k}_0(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}^p$, both twice-continuously differentiable on $\mathbb{R}^n$, and $\gamma_1, \gamma_2, \gamma_3 \in \mathcal{K}_\infty$ such that $\mathbf{k}_0(\mathbf{0}) = \mathbf{0}$ and:

$$\gamma_1(\|\mathbf{x}\|_2) \leq V_0(\mathbf{x}) \leq \gamma_2(\|\mathbf{x}\|_2), \tag{13}$$

$$\frac{\partial V_0}{\partial \mathbf{x}}(\mathbf{x})(\mathbf{f}_0(\mathbf{x}) + \mathbf{g}_0(\mathbf{x})\mathbf{k}_0(\mathbf{x})) \leq -\gamma_3(\|\mathbf{x}\|_2), \tag{14}$$

for all $\mathbf{x} \in \mathbb{R}^n$. The function $\mathbf{k}_0(\mathbf{x})$ reflects a stabilizing controller that we would implement for the system (9) if we could directly control $\boldsymbol{\xi}$. As we may only directly control $\mathbf{u}$, we must *backstep* through the state $\boldsymbol{\xi}$ to access $\mathbf{u}$. More precisely, consider a function $V : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}_{\geq 0}$ defined as:

$$V(\mathbf{x}, \boldsymbol{\xi}) = V_0(\mathbf{x}) + \frac{1}{2\mu}(\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x}))^\top(\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x})), \tag{15}$$

where $\mu \in \mathbb{R}_{>0}$. We note there exists $\gamma_1', \gamma_2' \in \mathcal{K}_\infty$ such that:

$$\gamma_1(\|\mathbf{x}\|_2) + \gamma_1'(\|\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x})\|_2) \leq V(\mathbf{x}, \boldsymbol{\xi}), \tag{16}$$

$$V(\mathbf{x}, \boldsymbol{\xi}) \leq \gamma_2(\|\mathbf{x}\|_2) + \gamma_2'(\|\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x})\|_2), \tag{17}$$

for all $\mathbf{x} \in \mathbb{R}^n$ and $\boldsymbol{\xi} \in \mathbb{R}^p$. The time derivative of V is:

$$\begin{aligned}\dot{V}(\mathbf{x}, \boldsymbol{\xi}, \mathbf{u}) = \; & \frac{\partial V_0}{\partial \mathbf{x}}(\mathbf{x})\big(\mathbf{f}_0(\mathbf{x}) + \mathbf{g}_0(\mathbf{x})\boldsymbol{\xi}\big) \\ & + \frac{1}{\mu}(\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x}))^\top \Bigg(\mathbf{f}_1(\mathbf{x}, \boldsymbol{\xi}) + \mathbf{g}_1(\mathbf{x}, \boldsymbol{\xi})\mathbf{u} \\ & - \frac{\partial \mathbf{k}_0}{\partial \mathbf{x}}(\mathbf{x})\big(\mathbf{f}_0(\mathbf{x}) + \mathbf{g}_0(\mathbf{x})\boldsymbol{\xi}\big)\Bigg).\end{aligned} \tag{18}$$

Using a locally Lipschitz continuous feedback controller $\mathbf{k} : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}^m$ defined as:

$$\begin{aligned}\mathbf{k}(\mathbf{x}, \boldsymbol{\xi}) = \mathbf{g}_1(\mathbf{x}, \boldsymbol{\xi})^\dagger \Bigg(& - \mathbf{f}_1(\mathbf{x}, \boldsymbol{\xi}) + \frac{\partial \mathbf{k}_0}{\partial \mathbf{x}}(\mathbf{x})\big(\mathbf{f}_0(\mathbf{x}) + \mathbf{g}_0(\mathbf{x})\boldsymbol{\xi}\big) \\ & - \mu\left(\frac{\partial V_0}{\partial \mathbf{x}}(\mathbf{x})\mathbf{g}_0(\mathbf{x})\right)^\top - \frac{\lambda}{2}(\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x}))\Bigg),\end{aligned} \tag{19}$$

with $\lambda \in \mathbb{R}_{\geq 0}$ yields:

$$\begin{aligned}\dot{V}(\mathbf{x}, \boldsymbol{\xi}, \mathbf{k}(\mathbf{x}, \boldsymbol{\xi})) = \; & \frac{\partial V_0}{\partial \mathbf{x}}(\mathbf{x})(\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{k}_0(\mathbf{x})) \qquad (20) \\ & - \frac{\lambda}{2\mu}(\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x}))^\top(\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x})), \\ \leq \; & -\gamma_3(\|\mathbf{x}\|_2) - \gamma_3'(\|\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x})\|_2), \qquad (21)\end{aligned}$$

for $\gamma_3' \in \mathcal{K}_\infty$ defined as $\gamma_3'(s) \triangleq \lambda/(2\mu)s^2$. Hence V is a Lyapunov function for (11)-(12), such that $I((\mathbf{x}_0, \boldsymbol{\xi}_0)) = [0, \infty)$ for all $(\mathbf{x}_0, \boldsymbol{\xi}_0) \in \mathbb{R}^n \times \mathbb{R}^p$, and $\boldsymbol{\varphi}_\mathbf{x}(t) \to \mathbf{0}$ and $\boldsymbol{\varphi}_{\boldsymbol{\xi}}(t) - \mathbf{k}_0(\boldsymbol{\varphi}_\mathbf{x}(t)) \to \mathbf{0}$ as $t \to \infty$. Furthermore, we have:

$$\inf_{\mathbf{u} \in \mathbb{R}^m} \dot{V}(\mathbf{x}, \boldsymbol{\xi}, \mathbf{u}) < -c\gamma_3(\|\mathbf{x}\|_2) - c\gamma_3'(\|\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x})\|_2), \tag{22}$$

for all $\mathbf{x} \neq \mathbf{0}$ and $\boldsymbol{\xi} \neq \mathbf{k}_0(\mathbf{x})$, where $c \in (0, 1)$, such that V is a Control Lyapunov Function (CLF) [23]. This enables a convex optimization-based controller defined as follows:

$$\begin{aligned}\mathbf{k}(\mathbf{x}, \boldsymbol{\xi}) = \; & \operatorname*{argmin}_{\mathbf{u} \in \mathbb{R}^m} \frac{1}{2}\|\mathbf{u}\|_2^2 \\ & \text{s.t. } \dot{V}(\mathbf{x}, \boldsymbol{\xi}, \mathbf{u}) \leq -c\gamma_3(\|\mathbf{x}\|_2) - c\gamma_3'(\|\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x})\|_2),\end{aligned} \tag{23}$$

that stabilizes (11)-(12) and is locally Lipschitz continuous on $(\mathbb{R}^n \times \mathbb{R}^p) \setminus \{\mathbf{0}\}$ if γ_3 is locally Lipschitz continuous [23].

III. Control Barrier Function Backstepping

In this section we explore how Control Barrier Functions can be used to achieve safety for the cascaded system in (9)-(10) when one must backstep through the state $\boldsymbol{\xi}$.

Suppose there exists a set $\mathcal{C}_0 \subset \mathbb{R}^n$ defined as the 0-superlevel set of a twice-continuously differentiable function $h_0 : \mathbb{R}^n \to \mathbb{R}$:

$$\mathcal{C}_0 = \{\mathbf{x} \in \mathbb{R}^n \mid h_0(\mathbf{x}) \geq 0\}, \tag{24}$$

that we wish to keep safe. We further assume that $\frac{\partial h_0}{\partial \mathbf{x}}(\mathbf{x}) \neq \mathbf{0}$ when $h_0(\mathbf{x}) = 0$. As the input $\mathbf{u}$ does not show up in the time derivative of h_0, we may not directly apply the Control Barrier Function methodology established in Section II. Instead, motivated by the Lyapunov setting, we take a backstepping approach using CBFs. In particular, suppose there exists a twice-continuously differentiable function $\mathbf{k}_0(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}^p$ and a function $\alpha_0 \in \mathcal{K}_\infty^e$ such that:

$$\frac{\partial h_0}{\partial \mathbf{x}}(\mathbf{x})\left(\mathbf{f}_0(\mathbf{x}) + \mathbf{g}_0(\mathbf{x})\mathbf{k}_0(\mathbf{x})\right) \geq -\alpha_0(h_0(\mathbf{x})). \tag{25}$$

As before, $\mathbf{k}_0(\mathbf{x})$ reflects a controller that renders $\mathcal{C}_0$ safe that we would implement for the system (9) if we could directly control $\boldsymbol{\xi}$. Let us consider a twice-continuously differentiable function $h : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}$ defined as:

$$h(\mathbf{x}, \boldsymbol{\xi}) = h_0(\mathbf{x}) - \frac{1}{2\mu}(\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x}))^\top(\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x})), \tag{26}$$

with $\mu \in \mathbb{R}_{>0}$. We note that instead of adding the quadratic error term as we did in (15), we have subtracted it. Let us define the set $\mathcal{C} \subset \mathbb{R}^n \times \mathbb{R}^p$ as the 0-superlevel set of the function h:

$$\mathcal{C} = \{(\mathbf{x}, \boldsymbol{\xi}) \in \mathbb{R}^n \times \mathbb{R}^p \mid h(\mathbf{x}, \boldsymbol{\xi}) \geq 0\}, \tag{27}$$

noting that $\mathcal{C} \subset \mathcal{C}_0 \times \mathbb{R}^p$. This enables the following theorem:

Theorem 3. *Let $\mathcal{C}_0$ be the 0-superlevel set of a twice-continuously differentiable function $h_0 : \mathbb{R}^n \to \mathbb{R}$ with $\frac{\partial h_0}{\partial \mathbf{x}}(\mathbf{x}) \neq \mathbf{0}$ when $h_0(\mathbf{x}) = 0$. If there exists a twice-continuously differentiable function $\mathbf{k}_0(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}^p$ and a globally Lipschitz[1] function $\alpha_0 \in \mathcal{K}_\infty^e$ such that* (25) *holds, then there exists a locally Lipschitz continuous controller $\mathbf{k} : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}^p$ such that the function h defined in* (26) *is a Barrier Function for the closed-loop system* (11)-(12) *on the set $\mathcal{C}$ defined in* (27)*. Moreover, if $(\mathbf{x}_0, \boldsymbol{\xi}_0) \in \mathcal{C}$, then $\boldsymbol{\varphi}_\mathbf{x}(t) \in \mathcal{C}_0$ for all $t \in I((\mathbf{x}_0, \boldsymbol{\xi}_0))$.*

Proof. We observe that:

$$\begin{bmatrix}\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x}, \boldsymbol{\xi}) \\ \frac{\partial h}{\partial \boldsymbol{\xi}}(\mathbf{x}, \boldsymbol{\xi})\end{bmatrix} = \begin{bmatrix}\frac{\partial h_0}{\partial \mathbf{x}}(\mathbf{x}) + \frac{1}{\mu}(\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x}))^\top \frac{\partial \mathbf{k}_0}{\partial \mathbf{x}}(\mathbf{x}) \\ -\frac{1}{\mu}(\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x}))^\top\end{bmatrix}, \tag{28}$$

from which we may conclude that if $\frac{\partial h}{\partial \boldsymbol{\xi}}(\mathbf{x}, \boldsymbol{\xi}) = \mathbf{0}$ and $h(\mathbf{x}, \boldsymbol{\xi}) = 0$, we must have $h_0(\mathbf{x}) = 0$, and thus $\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x}, \boldsymbol{\xi}) =$

[1] We note this assumption permits linear extended class $\mathcal{K}$ functions, i.e, $\alpha_0(r) = kr$ for some $k \in \mathbb{R}_{>0}$, which are often used in practice

$\frac{\partial h_0}{\partial \mathbf{x}}(\mathbf{x}) \neq \mathbf{0}$ by assumption. Furthermore, taking the time derivative of h yields:

$$\begin{aligned}\dot{h}(\mathbf{x},\boldsymbol{\xi},\mathbf{u}) = & \frac{\partial h_0}{\partial \mathbf{x}}(\mathbf{x})\big(\mathbf{f}_0(\mathbf{x}) + \mathbf{g}_0(\mathbf{x})\boldsymbol{\xi}\big) \\ & - \frac{1}{\mu}(\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x}))^\top \bigg(\mathbf{f}_1(\mathbf{x},\boldsymbol{\xi}) + \mathbf{g}_1(\mathbf{x},\boldsymbol{\xi})\mathbf{u} \\ & \qquad - \frac{\partial \mathbf{k}_0}{\partial \mathbf{x}}(\mathbf{x})\big(\mathbf{f}_0(\mathbf{x}) + \mathbf{g}_0(\mathbf{x})\boldsymbol{\xi}\big)\bigg).\end{aligned} \tag{29}$$

Using a locally Lipschitz continuous feedback controller $\mathbf{k} : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}^m$ defined as:

$$\begin{aligned}\mathbf{k}(\mathbf{x},\boldsymbol{\xi}) = \mathbf{g}_1(\mathbf{x},\boldsymbol{\xi})^\dagger \bigg(& - \mathbf{f}_1(\mathbf{x},\boldsymbol{\xi}) + \frac{\partial \mathbf{k}_0}{\partial \mathbf{x}}(\mathbf{x})\big(\mathbf{f}_0(\mathbf{x}) + \mathbf{g}_0(\mathbf{x})\boldsymbol{\xi}\big) \\ & + \mu\left(\frac{\partial h_0}{\partial \mathbf{x}}(\mathbf{x})\mathbf{g}_0(\mathbf{x})\right)^\top - \frac{\lambda}{2}(\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x}))\bigg),\end{aligned} \tag{30}$$

with $\lambda \in \mathbb{R}_{\geq 0}$ yields:

$$\begin{aligned}\dot{h}(\mathbf{x},\boldsymbol{\xi},\mathbf{k}(\mathbf{x},\boldsymbol{\xi})) = & \frac{\partial h_0}{\partial \mathbf{x}}(\mathbf{x})(\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{k}_0(\mathbf{x})) && (31) \\ & + \frac{\lambda}{2\mu}(\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x}))^\top(\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x})), \\ \geq & -\alpha_0(h_0(\mathbf{x})) + \frac{\lambda}{2\mu}\|\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x})\|_2^2. && (32)\end{aligned}$$

Let L be the Lipschitz constant of α_0. Choosing $\lambda \geq L$, we have that:

$$\dot{h}(\mathbf{x},\boldsymbol{\xi},\mathbf{k}(\mathbf{x},\boldsymbol{\xi})) \geq -\alpha_0(h_0(\mathbf{x})) + \frac{L}{2\mu}\|\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x})\|_2^2, \tag{33}$$

and the global Lipschitz property of α_0 yields that:

$$\left|\alpha_0\left(h_0(\mathbf{x}) - \frac{1}{2\mu}\|\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x})\|^2\right) - \alpha_0(h_0(\mathbf{x}))\right| \leq \frac{L}{2\mu}\|\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x})\|_2^2. \tag{34}$$

Noting the definition of (26), we may rearrange (34) to yield:

$$\alpha_0(h(\mathbf{x},\boldsymbol{\xi})) \geq \alpha_0(h_0(\mathbf{x})) - \frac{L}{2\mu}\|\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x})\|_2^2. \tag{35}$$

Negating both sides of this expression and combining with (33) allows us to conclude that:

$$\dot{h}(\mathbf{x},\boldsymbol{\xi},\mathbf{k}(\mathbf{x},\boldsymbol{\xi})) \geq -\alpha_0(h(\mathbf{x},\boldsymbol{\xi})). \tag{36}$$

Thus, h is a BF for the closed-loop system (11)-(12) on the set $\mathcal{C}$. Hence, by Theorem 1 we may conclude the set $\mathcal{C}$ is safe, i.e., $(\mathbf{x}_0,\boldsymbol{\xi}_0) \in \mathcal{C} \implies \boldsymbol{\varphi}(t) \in \mathcal{C} \implies \boldsymbol{\varphi}_{\mathbf{x}}(t) \in \mathcal{C}_0$ for all $t \in I((\mathbf{x}_0,\boldsymbol{\xi}_0))$. □

Remark 2. The preceding result establishes the safety of the set $\mathcal{C}$, rather than the set $\mathcal{C}_0$. We do not necessarily have that $\mathbf{x}_0 \in \mathcal{C}_0$ implies $\boldsymbol{\varphi}_{\mathbf{x}}(t) \in \mathcal{C}_0$ for all $t \in I((\mathbf{x}_0,\boldsymbol{\xi}_0))$. The further requirement on the initial condition $\boldsymbol{\xi}_0$ is expected, and appears in other results studying safety for higher-order systems [3]–[5].

We now make the following observation. Suppose that $h_0(\mathbf{x}^*) = 0$, $\boldsymbol{\xi} = \mathbf{k}_0(\mathbf{x}^*)$ and:

$$\frac{\partial h_0}{\partial \mathbf{x}}(\mathbf{x}^*)(\mathbf{f}_0(\mathbf{x}^*) + \mathbf{g}_0(\mathbf{x}^*)\mathbf{k}_0(\mathbf{x}^*)) = 0, \tag{37}$$

for some $\mathbf{x}^* \in \mathcal{C}_0$. Then, we have that:

$$\sup_{\mathbf{u}\in\mathbb{R}^m} \dot{h}(\mathbf{x}^*,\mathbf{k}_0(\mathbf{x}^*),\mathbf{u}) = 0 = -\alpha(h(\mathbf{x}^*,\mathbf{k}_0(\mathbf{x}^*))), \tag{38}$$

for any $\alpha \in \mathcal{K}_\infty^e$. Thus, we do not have that there exists an extended class $\mathcal{K}_\infty$ function α such that the strict inequality in (7) is met, and hence we may not conclude that h is a CBF for the system (9)-(10) on $\mathcal{C}$. The primary reason that h is not a CBF lies in the fact that when $\boldsymbol{\xi} = \mathbf{k}_0(\mathbf{x}^*)$, the input does not have an effect on the time derivative of h. In this situation, the evolution of h is entirely dependent on the design of the controller $\mathbf{k}_0$. Suppose that instead of (25), we have that:

$$\frac{\partial h_0}{\partial \mathbf{x}}(\mathbf{x})\left(\mathbf{f}_0(\mathbf{x}) + \mathbf{g}_0(\mathbf{x})\mathbf{k}_0(\mathbf{x})\right) > -\alpha_0(h_0(\mathbf{x})). \tag{39}$$

Considering any $\mathbf{x} \in \mathbb{R}^n$ now, if $\boldsymbol{\xi} = \mathbf{k}_0(\mathbf{x})$, we have that:

$$\dot{h}(\mathbf{x},\mathbf{k}_0(\mathbf{x}),\mathbf{u}) > -\alpha_0(h_0(\mathbf{x})) = -\alpha_0(h(\mathbf{x},\mathbf{k}_0(\mathbf{x}))), \tag{40}$$

for all $\mathbf{u} \in \mathbb{R}^m$. Noting that if $\boldsymbol{\xi} \neq \mathbf{k}_0(\mathbf{x})$, $\dot{h}$ can be made arbitrarily large through input, we may conclude that:

$$\sup_{\mathbf{u}\in\mathbb{R}^m} \dot{h}(\mathbf{x},\boldsymbol{\xi},\mathbf{u}) > -\alpha_0(h(\mathbf{x},\boldsymbol{\xi})). \tag{41}$$

This is summarized in the following theorem:

Theorem 4. *Let $\mathcal{C}_0$ be the 0-superlevel set of a twice-continuously differentiable function $h_0 : \mathbb{R}^n \to \mathbb{R}$ with $\frac{\partial h_0}{\partial \mathbf{x}}(\mathbf{x}) \neq \mathbf{0}$ when $h_0(\mathbf{x}) = 0$. If there exists a twice-continuously differentiable function $\mathbf{k}_0(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}^p$ and a function $\alpha_0 \in \mathcal{K}_\infty^e$ such that* (39) *holds, then the function h defined in* (26) *is a Control Barrier Function for the system* (9)-(10) *on the set $\mathcal{C}$ defined in* (27).

Theorem 4 does not explicitly require the assumption of global Lipschitz continuity on α_0, which was needed to achieve (36) when using the particular controller (30). As CBFs are typically used in the context of control synthesis (beyond purely verification), we notice that (41) implies that:

$$\sup_{\mathbf{u}\in\mathbb{R}^m} \dot{h}(\mathbf{x},\boldsymbol{\xi},\mathbf{u}) > -\alpha_1(h(\mathbf{x},\boldsymbol{\xi})), \tag{42}$$

for any $\alpha_1 \in \mathcal{K}_\infty^e$ such that $\alpha_1(s) \geq \alpha_0(s)$ for all $s \in \mathbb{R}$. Thus we may view α_1 as an design parameter we may specify. For any such locally Lipschitz[2] $\alpha_1 \in \mathcal{K}_\infty^e$ and any locally Lipschitz continuous $\mathbf{k}_{\mathrm{d}} : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}^m$, we can synthesize an optimization-based controller:

$$\begin{aligned}\mathbf{k}(\mathbf{x},\boldsymbol{\xi}) = \underset{\mathbf{u}\in\mathbb{R}^m}{\operatorname{argmin}} \ & \frac{1}{2}\|\mathbf{u} - \mathbf{k}_{\mathrm{d}}(\mathbf{x},\boldsymbol{\xi})\|_2^2 \\ \text{s.t. } & \dot{h}(\mathbf{x},\boldsymbol{\xi},\mathbf{u}) \geq -\alpha_1(h(\mathbf{x},\boldsymbol{\xi})),\end{aligned} \tag{43}$$

that is locally Lipschitz continuous on $\mathbb{R}^n \times \mathbb{R}^p$ [23] and renders h a BF for (11)-(12) on $\mathcal{C}$.

[2]Though it is not necessary for α_0 to be locally Lipschitz continuous to imply the existence of such an α_1, it is a sufficient condition.

IV. Multi-Step CBF Backstepping

In this section we extend the preceding CBF backstepping approach to higher-order mixed-relative degree systems via a recursive design process typical of backstepping.

Consider the nonlinear system[3] in strict feedback form:

$$\dot{\boldsymbol{\xi}}_0 = \mathbf{f}_0(\boldsymbol{\xi}_0) + \mathbf{g}_{0,\boldsymbol{\xi}}(\boldsymbol{\xi}_0)\boldsymbol{\xi}_1 + \mathbf{g}_{0,\mathbf{u}}(\boldsymbol{\xi}_0)\mathbf{u}_0, \tag{44a}$$

$$\dot{\boldsymbol{\xi}}_1 = \mathbf{f}_1(\boldsymbol{\xi}_0,\boldsymbol{\xi}_1) + \mathbf{g}_{1,\boldsymbol{\xi}}(\boldsymbol{\xi}_0,\boldsymbol{\xi}_1)\boldsymbol{\xi}_2 + \mathbf{g}_{1,\mathbf{u}}(\boldsymbol{\xi}_0,\boldsymbol{\xi}_1)\mathbf{u}_1, \tag{44b}$$

$$\vdots$$

$$\dot{\boldsymbol{\xi}}_r = \mathbf{f}_r(\boldsymbol{\xi}_0,\boldsymbol{\xi}_1,\boldsymbol{\xi}_2,\dots\boldsymbol{\xi}_r) + \mathbf{g}_r(\boldsymbol{\xi}_0,\boldsymbol{\xi}_1,\dots,\boldsymbol{\xi}_r)\mathbf{u}_r, \tag{44c}$$

with states $\boldsymbol{\xi}_i \in \mathbb{R}^{p_i}$ and inputs $\mathbf{u}_i \in \mathbb{R}^{m_i}$ for $i = 0,\dots,r$. The functions $\mathbf{f}_i$, $\mathbf{g}_{i,\mathbf{u}}$ for $i = 0,\dots,r$ and $\mathbf{g}_{i,\boldsymbol{\xi}}$ for $i = 0,\dots,r-1$ are assumed to be smooth on their respective domains. We further assume that the functions $\mathbf{g}_i = (\mathbf{g}_{i,\boldsymbol{\xi}}, \mathbf{g}_{i,\mathbf{u}})$ for $i = 1,\dots,r-1$ and the function $\mathbf{g}_r$ are pseudo-invertible on their respective domains. Let us denote $q_i \triangleq \sum_{j=0}^{i} p_j$, $M_r \triangleq \sum_{j=0}^{r} m_j$, and $\mathbf{z}_i \triangleq (\boldsymbol{\xi}_0,\boldsymbol{\xi}_1,\dots,\boldsymbol{\xi}_i) \in \mathbb{R}^{q_i}$ for $i = 0,\dots,r$. We seek to construct a controller $\mathbf{k} : \mathbb{R}^{q_r} \to \mathbb{R}^{M_r}$ such that setting $\mathbf{u} = (\mathbf{u}_0,\dots,\mathbf{u}_r) = \mathbf{k}(\mathbf{z}_r)$ achieves safety.

Suppose the set $\mathcal{C}_0$ is defined as the 0-superlevel set of a smooth function $h_0 : \mathbb{R}^{q_0} \to \mathbb{R}$ as in (24), with $\frac{\partial h_0}{\partial \boldsymbol{\xi}_0}(\mathbf{z}_0) \neq \mathbf{0}$ when $h_0(\mathbf{z}_0) = 0$. Let smooth functions $\mathbf{k}_{0,\boldsymbol{\xi}} : \mathbb{R}^{q_0} \to \mathbb{R}^{p_1}$ and $\mathbf{k}_{0,\mathbf{u}} : \mathbb{R}^{q_0} \to \mathbb{R}^{m_0}$, and a globally Lipschitz continuous function $\alpha_0 \in \mathcal{K}_\infty^e$ with Lipschitz constant L satisfy:

$$\frac{\partial h_0}{\partial \boldsymbol{\xi}_0}(\mathbf{z}_0)\big(\mathbf{f}_0(\mathbf{z}_0) + \mathbf{g}_{0,\boldsymbol{\xi}}(\mathbf{z}_0)\mathbf{k}_{0,\boldsymbol{\xi}}(\mathbf{z}_0) + \mathbf{g}_{0,\mathbf{u}}(\mathbf{z}_0)\mathbf{k}_{0,\mathbf{u}}(\mathbf{z}_0)\big) \geq -\alpha_0(h_0(\mathbf{z}_0)), \tag{45}$$

for all $\mathbf{z}_0 \in \mathbb{R}^{q_0}$. Consider smooth functions (to be defined) $\mathbf{k}_{i,\boldsymbol{\xi}} : \mathbb{R}^{q_i} \to \mathbb{R}^{p_{i+1}}$ for $i = 1,\dots,r-1$ and $\mathbf{k}_{i,\mathbf{u}} : \mathbb{R}^{q_i} \to \mathbb{R}^{m_i}$ for $i = 1,\dots,r$, and define the smooth function $h : \mathbb{R}^{q_r} \to \mathbb{R}$:

$$h(\mathbf{z}_r) = h_0(\mathbf{z}_0) - \sum_{i=1}^{r} \frac{1}{2\mu_i}\|\boldsymbol{\xi}_i - \mathbf{k}_{i-1,\boldsymbol{\xi}}(\mathbf{z}_{i-1})\|_2^2, \tag{46}$$

with $\mu_i \in \mathbb{R}_{>0}$ for $i = 1,\dots,r$. Define the set $\mathcal{C} \subset \mathbb{R}^{q_r}$ as:

$$\mathcal{C} = \{\mathbf{z}_r \in \mathbb{R}^{q_r} \mid h(\mathbf{z}_r) \geq 0\}, \tag{47}$$

noting that $\mathcal{C} \subseteq \mathcal{C}_0 \times \mathbb{R}^{p_1} \times \dots \times \mathbb{R}^{p_r}$. Given this construction, we have the following result:

Theorem 5. *Let $\mathcal{C}_0$ be the 0-superlevel set of smooth function $h_0 : \mathbb{R}^{p_0} \to \mathbb{R}$ with $\frac{\partial h_0}{\partial \boldsymbol{\xi}_0}(\mathbf{z}_0) \neq \mathbf{0}$ when $h_0(\mathbf{z}_0) = 0$. If there exist smooth functions $\mathbf{k}_{0,\boldsymbol{\xi}} : \mathbb{R}^{p_0} \to \mathbb{R}^{p_1}$ and $\mathbf{k}_{0,\mathbf{u}} : \mathbb{R}^{p_0} \to \mathbb{R}^{m_0}$ and a globally Lipschitz function $\alpha_0 \in \mathcal{K}_\infty^e$ such that* (45) *holds, then there exists a smooth controller $\mathbf{k} : \mathbb{R}^{q_r} \to \mathbb{R}^{M_r}$ and functions $\mathbf{k}_{i,\boldsymbol{\xi}} : \mathbb{R}^{q_i} \to \mathbb{R}^{p_{i+1}}$ for $i = 1,\dots,r-1$ such that the function $h : \mathbb{R}^{q_r} \to \mathbb{R}$ defined in* (46) *is a Barrier Function for the closed-loop system* (44) *on the set $\mathcal{C}$ defined in* (47). *Moreover, if the initial condition $\mathbf{z}_{r,0} \in \mathcal{C}$, then $\boldsymbol{\varphi}_{\boldsymbol{\xi}_0}(t) \in \mathcal{C}_0$ for all $t \in I(\mathbf{z}_{r,0})$.*

[3] We do not notate a closed-loop system, but assume it is understood that when we refer to this system as closed-loop, it is operating under a controller.

Proof. We observe that:

$$\frac{\partial h}{\partial \boldsymbol{\xi}_0}(\mathbf{z}_r) = \frac{\partial h_0}{\partial \boldsymbol{\xi}_0}(\mathbf{z}_0) + \sum_{j=1}^{r} \frac{1}{\mu_j}(\boldsymbol{\xi}_j - \mathbf{k}_{j-1,\boldsymbol{\xi}}(\mathbf{z}_{j-1}))^\top \frac{\partial \mathbf{k}_{j-1,\boldsymbol{\xi}}}{\partial \boldsymbol{\xi}_0}(\mathbf{z}_{j-1}), \tag{48}$$

and for $i \in \{1,\dots,r\}$, we have that:

$$\frac{\partial h}{\partial \boldsymbol{\xi}_i}(\mathbf{z}_r) = -\frac{1}{\mu_i}(\boldsymbol{\xi}_i - \mathbf{k}_{i-1,\boldsymbol{\xi}}(\mathbf{z}_{i-1}))^\top + \sum_{j=i+1}^{r} \frac{1}{\mu_j}(\boldsymbol{\xi}_j - \mathbf{k}_{j-1,\boldsymbol{\xi}}(\mathbf{z}_{j-1}))^\top \frac{\partial \mathbf{k}_{j-1,\boldsymbol{\xi}}}{\partial \boldsymbol{\xi}_i}(\mathbf{z}_{j-1}). \tag{49}$$

We can see recursively (backwards) that if $\frac{\partial h}{\partial \boldsymbol{\xi}_i}(\mathbf{z}_r) = \mathbf{0}$ for $i = 1,\dots,r$, then we must have $\boldsymbol{\xi}_i = \mathbf{k}_{i-1,\boldsymbol{\xi}}(\mathbf{z}_{i-1})$ for $i = 1,\dots,r$, and thus $h(\mathbf{z}_r) = h_0(\boldsymbol{\xi}_0)$ and $\frac{\partial h}{\partial \boldsymbol{\xi}_0}(\mathbf{z}_r) = \frac{\partial h_0}{\partial \boldsymbol{\xi}_0}(\mathbf{z}_0)$. As $\frac{\partial h_0}{\partial \boldsymbol{\xi}_0}(\mathbf{z}_0) \neq \mathbf{0}$ when $h_0(\mathbf{z}_0) = 0$, we have that $\frac{\partial h}{\partial \boldsymbol{\xi}_0}(\mathbf{z}_r) \neq \mathbf{0}$ when $h(\mathbf{z}_r) = 0$, such that $\frac{\partial h}{\partial \mathbf{z}_r}(\mathbf{z}_r) \neq \mathbf{0}$ when $h(\mathbf{z}_r) = 0$.

Using $\mathbf{k}_{0,\boldsymbol{\xi}}$ and $\mathbf{k}_{0,\mathbf{u}}$, we define the smooth functions:

$$\begin{bmatrix}\mathbf{k}_{1,\boldsymbol{\xi}}(\mathbf{z}_1)\\ \mathbf{k}_{1,\mathbf{u}}(\mathbf{z}_1)\end{bmatrix} = \mathbf{g}_1(\mathbf{z}_1)^\dagger\bigg(-\mathbf{f}_1(\mathbf{z}_1) + \mu_0\Big(\frac{\partial h_0}{\partial \boldsymbol{\xi}_0}(\mathbf{z}_0)\mathbf{g}_{0,\boldsymbol{\xi}}(\mathbf{z}_0)\Big)^\top + \frac{\partial \mathbf{k}_{0,\boldsymbol{\xi}}}{\partial \boldsymbol{\xi}_0}(\mathbf{z}_0)\big(\mathbf{f}_0(\mathbf{z}_0) + \mathbf{g}_{0,\boldsymbol{\xi}}(\mathbf{z}_0)\boldsymbol{\xi}_1 + \mathbf{g}_{0,\mathbf{u}}(\mathbf{z}_0)\mathbf{k}_{0,\mathbf{u}}(\mathbf{z}_0)\big) - \frac{\lambda_1}{2}(\boldsymbol{\xi}_1 - \mathbf{k}_{0,\boldsymbol{\xi}}(\mathbf{z}_0))\bigg). \tag{50}$$

For $i = 2,\dots,r-1$, we recursively define the smooth functions:

$$\begin{bmatrix}\mathbf{k}_{i,\boldsymbol{\xi}}(\mathbf{z}_i)\\ \mathbf{k}_{i,\mathbf{u}}(\mathbf{z}_i)\end{bmatrix} = \mathbf{g}_i(\mathbf{z}_i)^\dagger\bigg(-\mathbf{f}_i(\mathbf{z}_i) - \mu_i\mathbf{g}_{i-1,\boldsymbol{\xi}}(\mathbf{z}_{i-1})^\top(\boldsymbol{\xi}_{i-1} - \mathbf{k}_{i-2,\boldsymbol{\xi}}(\mathbf{z}_{i-2})) + \sum_{j=0}^{i-1}\frac{\partial \mathbf{k}_{i-1,\boldsymbol{\xi}}}{\partial \boldsymbol{\xi}_j}(\mathbf{z}_{i-1})\big(\mathbf{f}_j(\mathbf{z}_j) + \mathbf{g}_{j,\boldsymbol{\xi}}(\mathbf{z}_j)\boldsymbol{\xi}_{j+1} + \mathbf{g}_{j,\mathbf{u}}(\mathbf{z}_j)\mathbf{k}_{j,\mathbf{u}}(\mathbf{z}_j)\big) - \frac{\lambda_i}{2}(\boldsymbol{\xi}_i - \mathbf{k}_{i-1,\boldsymbol{\xi}}(\mathbf{z}_{i-1}))\bigg), \tag{51}$$

and lastly define the smooth function:

$$\mathbf{k}_{r,\mathbf{u}}(\mathbf{z}_i) = \mathbf{g}_r(\mathbf{z}_r)^\dagger\bigg(-\mathbf{f}_r(\mathbf{z}_r) - \mu_r\mathbf{g}_{r-1,\boldsymbol{\xi}}(\mathbf{z}_{r-1})^\top(\boldsymbol{\xi}_{r-1} - \mathbf{k}_{r-2,\boldsymbol{\xi}}(\mathbf{z}_{r-2})) + \sum_{j=0}^{r-1}\frac{\partial \mathbf{k}_{r-1,\boldsymbol{\xi}}}{\partial \boldsymbol{\xi}_j}(\mathbf{z}_{r-1})\big(\mathbf{f}_j(\mathbf{z}_j) + \mathbf{g}_{j,\boldsymbol{\xi}}(\mathbf{z}_j)\boldsymbol{\xi}_{j+1} + \mathbf{g}_{j,\mathbf{u}}(\mathbf{z}_j)\mathbf{k}_{j,\mathbf{u}}(\mathbf{z}_j)\big) - \frac{\lambda_r}{2}(\boldsymbol{\xi}_r - \mathbf{k}_{r-1,\boldsymbol{\xi}}(\mathbf{z}_{r-1}))\bigg), \tag{52}$$

Letting the controller $\mathbf{k} : \mathbb{R}^{q_r} \to \mathbb{R}^{M_r}$ be defined as:

$$\mathbf{k}(\mathbf{z}_r) = \begin{bmatrix}\mathbf{k}_{0,\mathbf{u}}(\mathbf{z}_0)^\top & \cdots & \mathbf{k}_{r,\mathbf{u}}(\mathbf{z}_r)^\top\end{bmatrix}^\top, \tag{53}$$

a sequence of (laborious) calculations yields:

$$\dot{h}(\mathbf{z}_r, \mathbf{k}(\mathbf{z}_r)) \geq -\alpha_0(h_0(\mathbf{z}_0)) + \sum_{i=1}^{r}\frac{\lambda_i}{2\mu_i}\|\boldsymbol{\xi}_i - \mathbf{k}_{i-1,\boldsymbol{\xi}}(\mathbf{z}_{i-1})\|_2^2.$$

Choosing $\lambda_i \geq L$ for $i = 1, \dots, r$ and following the same argument as in (31)-(36), we arrive at:

$$\dot{h}(\mathbf{z}_r, \mathbf{k}(\mathbf{z}_r)) \geq -\alpha_0(h(\mathbf{z}_r)). \tag{54}$$

Thus, h is a BF for the closed-loop system (44) on the set $\mathcal{C}$. Hence, by Theorem 1 we may conclude the set $\mathcal{C}$ is safe, i.e., $\mathbf{z}_{r,0} \in \mathcal{C} \implies \boldsymbol{\varphi}(t) \in \mathcal{C} \implies \boldsymbol{\varphi}_{\boldsymbol{\xi}_0}(t) \in \mathcal{C}_0$. □

If instead of (45) we suppose that:

$$\frac{\partial h_0}{\partial \boldsymbol{\xi}_0}(\mathbf{z}_0)\big(\mathbf{f}_0(\mathbf{z}_0) + \mathbf{g}_{0,\boldsymbol{\xi}}(\mathbf{z}_0)\mathbf{k}_{0,\boldsymbol{\xi}}(\mathbf{z}_0) + \mathbf{g}_{0,\mathbf{u}}\mathbf{k}_{0,\mathbf{u}}(\mathbf{z}_0)\big) > -\alpha_0(h_0(\mathbf{z}_0)), \tag{55}$$

we have the following result:

Theorem 6. *Let $\mathcal{C}_0$ be the 0-superlevel set of a smooth function $h_0 : \mathbb{R}^{q_0} \to \mathbb{R}$ with $\frac{\partial h_0}{\partial \boldsymbol{\xi}_0}(\mathbf{z}_0) \neq \mathbf{0}$ when $h_0(\mathbf{z}_0) = 0$. If there exist smooth functions $\mathbf{k}_{0,\boldsymbol{\xi}} : \mathbb{R}^{p_0} \to \mathbb{R}^{p_1}$ and $\mathbf{k}_{0,\mathbf{u}} : \mathbb{R}^{p_0} \to \mathbb{R}^{m_0}$ and a globally Lipschitz continuous function $\alpha_0 \in \mathcal{K}_\infty^e$ such that* (55) *holds, then the function h defined in* (46) *is a Control Barrier Function for the system* (44) *on the set $\mathcal{C}$ defined in* (47).

Consequently, for any locally Lipschitz $\alpha_1 \in \mathcal{K}_\infty^e$ such that $\alpha_1(s) \geq \alpha_0(s)$ for all $s \in \mathbb{R}$ and any locally Lipschitz continuous $\mathbf{k}_{\rm d} : \mathbb{R}^{q_r} \to \mathbb{R}^{M_r}$, we can synthesize a controller:

$$\begin{aligned}\mathbf{k}(\mathbf{z}_r) = \underset{\mathbf{u} \in \mathbb{R}^{M_r}}{\operatorname{argmin}}\ & \frac{1}{2}\|\mathbf{u} - \mathbf{k}_{\rm d}(\mathbf{z}_r)\|_2^2 \\ \text{s.t. } & \dot{h}(\mathbf{z}_r, \mathbf{u}_0, \dots, \mathbf{u}_r) \geq -\alpha_1(h(\mathbf{z}_r)),\end{aligned} \tag{56}$$

that is locally Lipschitz continuous on $\mathbb{R}^{q_r}$ [23] and renders h a BF for (44) on $\mathcal{C}$.

V. Joint CLF and CBF Backstepping

In this section we use joint Lyapunov and CBF backstepping to achieve both stability and safety of a cascaded system. For simplicity, let us consider the system (9)-(10). Suppose there exists functions $V_0 : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$, $h_0 : \mathbb{R}^n \to \mathbb{R}$ and $\mathbf{k}_0 : \mathbb{R}^n \to \mathbb{R}^p$ with $\mathbf{k}_0(\mathbf{0}) = \mathbf{0}$, all twice-continuously differentiable, and functions $\gamma_1, \gamma_2, \gamma_3 \in \mathcal{K}_\infty$ and a globally Lipschitz continuous function $\alpha_0 \in \mathcal{K}_\infty^e$ such that (13)-(14) and (25) are satisfied. Furthermore, let us define the set $\mathcal{C}_0 \subset \mathbb{R}^n$ as in (24). As before, we wish to stabilize the state to the origin while ensuring it remains in the set $\mathcal{C}_0$. Let us construct twice-continuously differentiable functions $V : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}_{\geq 0}$ and $h : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}$ as:

$$V(\mathbf{x}, \boldsymbol{\xi}) = V_0(\mathbf{x}) + \frac{1}{2\mu_V}(\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x}))^\top(\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x})), \tag{57}$$

$$h(\mathbf{x}, \boldsymbol{\xi}) = h_0(\mathbf{x}) - \frac{1}{2\mu_h}(\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x}))^\top(\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x})), \tag{58}$$

with $\mu_V, \mu_h \in \mathbb{R}_{>0}$. The time derivatives for V and h are given in (18) and (29), using their respective values μ_V and μ_h. We express them compactly here as:

$$\dot{V}(\mathbf{x}, \boldsymbol{\xi}, \mathbf{u}) = b_{V,1}(\mathbf{x}, \boldsymbol{\xi}) + \frac{1}{\mu_V}\mathbf{a}_1(\mathbf{x}, \boldsymbol{\xi})^\top \mathbf{u} \tag{59}$$

$$\dot{h}(\mathbf{x}, \boldsymbol{\xi}, \mathbf{u}) = b_{h,1}(\mathbf{x}, \boldsymbol{\xi}) - \frac{1}{\mu_h}\mathbf{a}_1(\mathbf{x}, \boldsymbol{\xi})^\top \mathbf{u}, \tag{60}$$

for functions $b_{V,1}, b_{h,1} : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}$ and $\mathbf{a}_1 : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}^m$. As we saw in the individual backstepping cases, it was possible to design (different) controllers such that the bounds on the derivatives in (20) and (31) were met. This implies that:

$$\begin{aligned}\inf_{\mathbf{u} \in \mathbb{R}^m} b_{V,1}(\mathbf{x}, \boldsymbol{\xi}) + \frac{1}{\mu_V}\mathbf{a}_1(\mathbf{x}, \boldsymbol{\xi})^\top \mathbf{u} \\ \leq -\gamma_3(\|\mathbf{x}\|) - \gamma_3'(\|\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x})\|_2),\end{aligned} \tag{61}$$

$$\begin{aligned}\inf_{\mathbf{u} \in \mathbb{R}^m} -b_{h,1}(\mathbf{x}, \boldsymbol{\xi}) + \frac{1}{\mu_h}\mathbf{a}_1(\mathbf{x}, \boldsymbol{\xi})^\top \mathbf{u} \\ \leq \alpha_0(h_0(\mathbf{x})) - \frac{\lambda}{2\mu_h}\|\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x})\|_2^2,\end{aligned} \tag{62}$$

We can rewrite these two inequality constraints as:

$$\mathbf{a}_1(\mathbf{x}, \boldsymbol{\xi})^\top \mathbf{u} \leq c_{V,1}(\mathbf{x}, \boldsymbol{\xi}), \tag{63}$$

$$\mathbf{a}_1(\mathbf{x}, \boldsymbol{\xi})^\top \mathbf{u} \leq c_{h,1}(\mathbf{x}, \boldsymbol{\xi}), \tag{64}$$

for functions $c_{V,1}, c_{h,1} : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}$. A key observation is that these constraints are mutually satisfiable, i.e, if we design a controller $\mathbf{k} : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}^m$ such that:

$$\mathbf{a}_1(\mathbf{x}, \boldsymbol{\xi})^\top \mathbf{k}(\mathbf{x}, \boldsymbol{\xi}) \leq \min\{c_{V,1}(\mathbf{x}, \boldsymbol{\xi}), c_{h,1}(\mathbf{x}, \boldsymbol{\xi})\}, \tag{65}$$

for all $(\mathbf{x}, \boldsymbol{\xi}) \in \mathbb{R}^n \times \mathbb{R}^p$, then both (63) and (64) are met. Thus under this controller, V is a Lyapunov function and h is a Barrier Function on $\mathcal{C}$ for the closed-loop system (11)-(12), such that we may conclude both stability and safety. An optimization-based controller achieving this is defined as:

$$\begin{aligned}\mathbf{k}(\mathbf{x}, \boldsymbol{\xi}) = \underset{\mathbf{u} \in \mathbb{R}^m}{\operatorname{argmin}}\ & \frac{1}{2}\|\mathbf{u}\|_2^2 \\ \text{s.t. } & \mathbf{a}_1(\mathbf{x}, \boldsymbol{\xi})^\top \mathbf{u} \leq \min\{c_{V,1}(\mathbf{x}, \boldsymbol{\xi}), c_{h,1}(\mathbf{x}, \boldsymbol{\xi})\}.\end{aligned} \tag{66}$$

The intuition behind the joint feasibility of these constraints is that the controller $\mathbf{k}_0$ has been designed to provide both stability and safety, and we are using the input $\mathbf{u}$ to drive $\boldsymbol{\xi}$ to $\mathbf{k}_0(\mathbf{x})$, thus benefiting both stability and safety. The challenge is then to design a continuously differentiable controller $\mathbf{k}_0$ satisfying both (14) and (25). To accomplish this, we will use the techniques presented in [19]. We note that designing smooth stabilizing controllers via Lyapunov functions often faces challenges at the origin [16]. With a cascaded system, we may encounter the origin of the top-level state without the entire state being at the origin. Thus, in this work we slightly relax (14) to ensure smoothness, in which case we achieve practical stability as opposed to asymptotic stability.

Suppose that we are given a smooth desired controller $\mathbf{k}_{0,\rm d} : \mathbb{R}^n \to \mathbb{R}^p$ we wish to implement at the top-level, that is not necessarily stable nor safe. Consider the top-level constraints:

$$\begin{aligned}\frac{\partial V_0}{\partial \mathbf{x}}(\mathbf{x})(\mathbf{f}_0(\mathbf{x}) + \mathbf{g}_0(\mathbf{x})(\mathbf{k}_{0,\rm d}(\mathbf{x}) + \mathbf{v})) \leq & -\gamma_3(\|\mathbf{x}\|_2) \\ & + \delta\psi(\|\mathbf{x}\|_2),\end{aligned}$$

$$\frac{\partial h_0}{\partial \mathbf{x}}(\mathbf{x})\,(\mathbf{f}_0(\mathbf{x}) + \mathbf{g}_0(\mathbf{x})(\mathbf{k}_{0,\rm d}(\mathbf{x}) + \mathbf{v})) \geq -\alpha_0(h_0(\mathbf{x})).$$

with $\delta \in \mathbb{R}_{>0}$ and $\psi : \mathbb{R} \to \mathbb{R}_{\geq 0}$ a bump function defined as:

$$\psi(s) = \begin{cases} \exp\left(-\frac{1}{\epsilon^2 - s^2}\right), & s \in (-\epsilon, \epsilon), \\ 0, & \text{otherwise}, \end{cases} \tag{67}$$

with $\epsilon \in \mathbb{R}_{>0}$. We can rewrite these constraints as:

$$\mathbf{a}_{V,0}(\mathbf{x})^\top \mathbf{v} + b_{V,0}(\mathbf{x}) \leq 0, \tag{68}$$
$$\mathbf{a}_{h,0}(\mathbf{x})^\top \mathbf{v} + b_{h,0}(\mathbf{x}) \leq 0, \tag{69}$$

for functions $\mathbf{a}_{V,0}, \mathbf{a}_{h,0} : \mathbb{R}^n \to \mathbb{R}^p$ and $b_{V,0}, b_{h,0} : \mathbb{R}^n \to \mathbb{R}$. Assuming V_0 is a CLF and h_0 is a CBF on $\mathcal{C}_0$ for (9) implies the set-valued functions $\mathcal{U}_V, \mathcal{U}_h : \mathbb{R}^n \to \mathcal{P}(\mathbb{R}^p)$ defined as:

$$\mathcal{U}_i(\mathbf{x}) = \left\{ \mathbf{v} \in \mathbb{R}^p \mid \mathbf{a}_{i,0}(\mathbf{x})^\top \mathbf{v} + b_{i,0}(\mathbf{x}) \leq 0 \right\}, \tag{70}$$

with $i \in \{V, h\}$ satisfy $\mathcal{U}_i(\mathbf{x}) \neq \{\emptyset\}$ for all $\mathbf{x} \in \mathbb{R}^n$. Moreover, for simplicity let us assume that $\mathcal{U}_V(\mathbf{x}) \cap \mathcal{U}_h(\mathbf{x}) \neq \{\emptyset\}$ for all $\mathbf{x} \in \mathbb{R}^n$, such that there exists a $\mathbf{v}$ that satisfies both (68) and (69) simultaneously. We note that if this is not possible, this construction can be done relaxing stability and enforcing safety as is common with combined CLF-CBF methods [2].

For a set $\mathcal{U} \subseteq \mathbb{R}^p$, define the Gaussian weighted centroid function $\boldsymbol{\mu} : \mathbb{R}^n \to \mathbb{R}^p$ as:

$$\boldsymbol{\mu}(\mathbf{x}; \mathcal{U}) \triangleq \frac{\int_{\mathcal{U}} \mathbf{v} \phi(\mathbf{x}, \mathbf{v}) d\mathbf{v}}{\int_{\mathcal{U}} \phi(\mathbf{x}, \mathbf{v}) d\mathbf{v}}, \tag{71}$$

where $\phi : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}_{\geq 0}$ is defined as:

$$\phi(\mathbf{x}, \mathbf{v}) = \frac{1}{\sqrt{2\pi}} \mathrm{e}^{-\|\mathbf{v}\|_2^2/(2\sigma(\mathbf{x}))}, \tag{72}$$

with a smooth function $\sigma : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$. As in [19], we may synthesize a controller:

$$\mathbf{k}_0(\mathbf{x}) = \mathbf{k}_{0,\mathrm{d}}(\mathbf{x}) + \zeta(\rho(\mathbf{x}))(\boldsymbol{\mu}(\mathbf{x}; \mathcal{U}_V) + \boldsymbol{\mu}(\mathbf{x}; \mathcal{U}_h)) + (1 - \zeta(\rho(\mathbf{x})))\boldsymbol{\mu}(\mathbf{x}; \mathcal{U}_V \cap \mathcal{U}_h), \tag{73}$$

where $\zeta : \mathbb{R} \to [0, 1]$ is a smooth partition of unity function with $\zeta(s) = 0$ for $s \leq 0$ and $\zeta(s) = 1$ for $s \geq 1$, and:

$$\rho(\mathbf{x}) = \frac{\mathbf{a}_{V,0}(\mathbf{x})^\top \mathbf{a}_{h,0}(\mathbf{x})}{\|\mathbf{a}_{V,0}(\mathbf{x})\|_2 \|\mathbf{a}_{h,0}(\mathbf{x})\|_2}, \tag{74}$$

encodes the angle between $\mathbf{a}_{V,0}$ and $\mathbf{a}_{h,0}$. The Gaussian weighted centroid functions in (73) have closed-form solutions [24], [25]. The controller in (73) respects both constraints, i.e., $(\mathbf{k}_0(\mathbf{x}) - \mathbf{k}_{0,\mathrm{d}}(\mathbf{x})) \in \mathcal{U}_V(\mathbf{x}) \cap \mathcal{U}_h(\mathbf{x})$. In addition, $\mathbf{k}_0$ is smooth if the functions $\mathbf{a}_{V,0}, \mathbf{a}_{h,0}, b_{V,0}$ and $b_{h,0}$ are smooth.

VI. Simulation

We now demonstrate CBF backstepping with two examples.

Example 1. Consider the planar double integrator system:

$$\dot{\mathbf{x}} = \boldsymbol{\xi}, \quad \dot{\boldsymbol{\xi}} = \mathbf{u}, \tag{75}$$

with $\mathbf{x}, \boldsymbol{\xi}, \mathbf{u} \in \mathbb{R}^2$. We intend to control the system to a goal position $\mathbf{x}_\mathrm{g} \in \mathbb{R}^2$ (such that $\lim_{t\to\infty} \boldsymbol{\varphi}_\mathbf{x}(t) = \mathbf{x}_\mathrm{g}$) while avoiding an obstacle centered at $\mathbf{x}_\mathrm{O} \in \mathbb{R}^2$ with radius $R_\mathrm{O} \in \mathbb{R}_{>0}$. Collision-free behavior is captured by the safe set $\mathcal{C}_0$ with:

$$h_0(\mathbf{x}) = \frac{1}{2}\left(\|\mathbf{x} - \mathbf{x}_\mathrm{O}\|_2^2 - R_\mathrm{O}^2 \right), \tag{76}$$

that satisfies $h_0(\mathbf{x}) = 0 \implies \frac{\partial h_0}{\partial \mathbf{x}}(\mathbf{x}) = (\mathbf{x} - \mathbf{x}_\mathrm{O})^\top \neq \mathbf{0}$. To reach the goal $\mathbf{x}_\mathrm{g}$, we rely on the desired smooth controller $\mathbf{k}_{0,\mathrm{d}}(\mathbf{x}) = -K_\mathrm{p}(\mathbf{x} - \mathbf{x}_\mathrm{g})$ which is used to define $\mathbf{k}_0$ through the smooth safety filter in (73). This is used to define h as in (26), which used with the desired controller $\mathbf{k}_\mathrm{d}(\mathbf{x}, \boldsymbol{\xi}) = -K_\mathrm{v}(\boldsymbol{\xi} - \mathbf{k}_0(\mathbf{x}))$ in the quadratic-program safety filter (43).

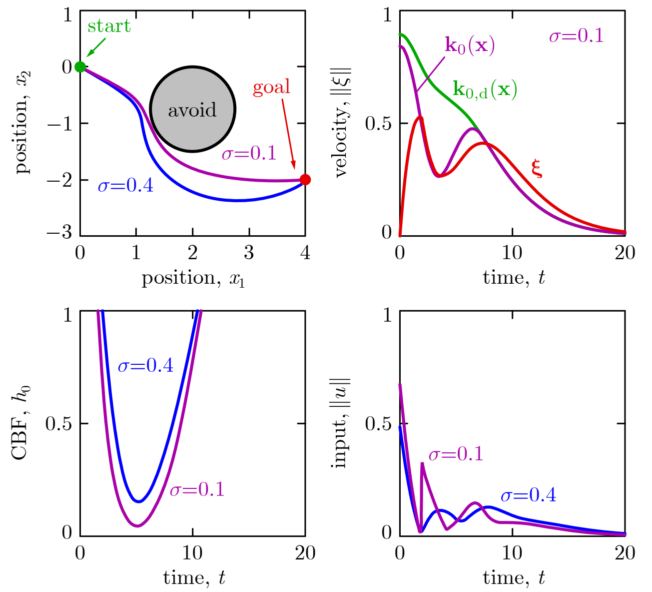

Fig. 1. Obstacle avoidance with double integrator model via backstepping. The system successfully avoids the obstacle and reaches the goal, while the conservatism of the route can be tuned by the smoothing parameter.

The closed-loop system is simulated in Fig. 1 for $K_\mathrm{p} = 0.2$, $K_\mathrm{v} = 0.8$, $\mu = 1$, $\alpha_0(s) = \alpha_1(s) = s$, $\sigma \equiv 0.1$ (purple) and $\sigma \equiv 0.4$ (blue). The system safely reaches the goal without colliding with the obstacle. As the smoothing parameter σ is increased, the system takes a more conservative route farther from the obstacle. This reduces the peak in the control input.

Example 2. Consider the planar unicycle model:

$$\dot{x} = v \cos\psi, \quad \dot{y} = v \sin\psi, \quad \dot{\psi} = \omega. \tag{77}$$

where $x, y, \psi, v, \omega \in \mathbb{R}$. This system can be written as:

$$\dot{\mathbf{x}} = \boldsymbol{\xi} u_0 \triangleq \mathbf{w}, \quad \dot{\boldsymbol{\xi}} = \begin{bmatrix} -\xi_2 & \xi_1 \end{bmatrix}^\top u_1, \tag{78}$$

with $\mathbf{x} = \begin{bmatrix} x & y \end{bmatrix}^\top$ and $\boldsymbol{\xi} = \begin{bmatrix} \cos\psi & \sin\psi \end{bmatrix}^\top$. Our goal is obstacle avoidance like in Example 1, via the CBF (76).

The unicycle model is in the form of (44) except for an additional nonlinearity: the product of the heading direction $\boldsymbol{\xi}$ and the speed u_0 that gives the velocity vector $\mathbf{w} = \boldsymbol{\xi} u_0$. With some care, this nonlinearity can be handled as follows. First, notice that (78) is affine in both $\mathbf{w}$ and u_0. Thus, a safe value $\mathbf{k}_0(\mathbf{x})$ for the velocity $\mathbf{w}$ can be designed such that it satisfies (25), which is the same as $\mathbf{k}_0(\mathbf{x})$ in Example 1. We convert the safe velocity $\mathbf{k}_0(\mathbf{x})$ into a safe heading direction $\mathbf{k}_{0,\boldsymbol{\xi}}(\mathbf{x}) = \mathbf{k}_0(\mathbf{x}) / \|\mathbf{k}_0(\mathbf{x})\|_2$ and safe speed $k_{0,u}(\mathbf{x}) = \|\mathbf{k}_0(\mathbf{x})\|_2$ by restricting to $\mathbf{k}_0(\mathbf{x}) \neq \mathbf{0}$. Then, $\mathbf{k}_{0,\boldsymbol{\xi}}(\mathbf{x})$ is incorporated into the composite barrier function h

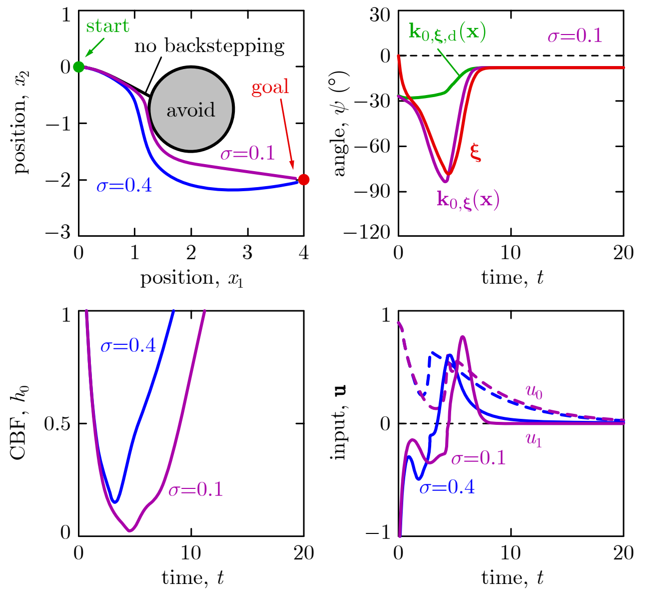

Fig. 2. Obstacle avoidance with unicycle model via backstepping. Remarkably, the unicycle is able to drive around the obstacle, while a standard safety filter without backstepping makes the unicycle stop in front of the obstacle.

in (46). By denoting the safe heading angle as $\psi_0(\mathbf{x})$, i.e., by writing $\mathbf{k}_{0,\boldsymbol{\xi}}(\mathbf{x}) = \begin{bmatrix}\cos\psi_0(\mathbf{x}) & \sin\psi_0(\mathbf{x})\end{bmatrix}^\top$, we get:

$$h(\mathbf{x},\boldsymbol{\xi}) = h_0(\mathbf{x}) - \frac{1}{\mu}\big(1-\cos(\psi-\psi_0(\mathbf{x}))\big), \qquad (79)$$

that gives penalty to heading in unsafe directions. Then, we synthesize the controller $\mathbf{u} = \begin{bmatrix}u_0 & u_1\end{bmatrix}^\top = \mathbf{k}(\mathbf{x},\boldsymbol{\xi})$ via backstepping based on (56), where we use the desired controller $\mathbf{k}_\mathrm{d}(\mathbf{x},\boldsymbol{\xi}) = \begin{bmatrix}K_\mathrm{p}\|\mathbf{x}-\mathbf{x}_\mathrm{g}\|_2 & -K_\psi\big(\sin\psi - \sin\psi_0(\mathbf{x})\big)\end{bmatrix}^\top$.

The behavior of the closed-loop system is shown by simulation results in Fig. 2 for $K_\mathrm{p}=0.2$, $K_\psi=3$, $\mu=1$, $\alpha_0(s)=\alpha_1(s)=s$, $\sigma\equiv 0.1$ (purple) and $\sigma\equiv 0.4$ (blue). Again, safety is guaranteed and more conservative smoothing makes the unicycle take a longer route. We remark that safety could also be enforced without backstepping, by relying on the input u_0 (speed) only. Then, the input u_1 (angular velocity) would not be constrained and could be chosen freely. This would result in the unicycle stopping in front of the obstacle and not reaching the goal (see black trajectory). As opposed, backstepping synthesizes a barrier function h such that inputs at all levels are utilized for safety. Such barrier synthesis is nontrivial, and backstepping provides a systematic solution.

VII. Conclusion

In conclusion, we have proposed a novel approach for using backstepping with Control Barrier Functions to design safety-critical controllers for nonlinear systems. Moreover, we unified this approach with Control Lyapunov Functions to achieve both stability and safety. Future work includes considering alternative methods for the smooth design of top-level controllers that are stabilizing and safe, and exploring the robustness to parameter uncertainty seen with backstepping.

References

[1] A. Ames, J. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs with application to adaptive cruise control," in *Conference on Decision & Control (CDC)*. IEEE, 2014, pp. 6271–6278.

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

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

[4] W. Xiao and C. Belta, "Control barrier functions for systems with high relative degree," in *Conference on Decision & Control (CDC)*. IEEE, 2019, pp. 474–479.

[5] ——, "High order control barrier functions," *Transactions on Automatic Control*, 2021.

[6] J. Breeden and D. Panagou, "High relative degree control barrier functions under input constraints," *arXiv preprint arXiv:2106.10345*, 2021.

[7] T. G. Molnar, R. K. Cosner, A. W. Singletary, W. Ubellacker, and A. D. Ames, "Model-free safety-critical control for robotic systems," *Robotics and Automation Letters*, vol. 7, no. 2, pp. 944–951, 2021.

[8] R. A. Freeman and P. V. Kokotović, "Backstepping design of robust controllers for a class of nonlinear systems," in *Nonlinear Control Systems Design 1992*. Elsevier, 1993, pp. 431–436.

[9] R. Sepulchre, M. Jankovic, and P. V. Kokotovic, *Constructive nonlinear control*. Springer Science & Business Media, 2012.

[10] M. Krstic and M. Bement, "Nonovershooting control of strict-feedback nonlinear systems," *Transactions on Automatic Control*, vol. 51, no. 12, pp. 1938–1943, 2006.

[11] I. Abel, D. Steeves, M. Krstic, and M. Jankovic, "Prescribed-time safety design for a chain of integrators," *arXiv preprint arXiv:2201.09447*, 2022.

[12] S. Koga and M. Krstic, "Safe PDE backstepping QP control with high relative degree CBFs: Stefan model with actuator dynamics," *arXiv preprint arXiv:2111.01187*, 2021.

[13] K. B. Ngo, R. Mahony, and Z.-P. Jiang, "Integrator backstepping using barrier functions for systems with multiple state constraints," in *Conference on Decision and Control (CDC)*. IEEE, 2005, pp. 8306–8312.

[14] G. Xia, J. Xue, C. Sun, and B. Zhao, "Backstepping control using barrier Lyapunov function for dynamic positioning control system with passive observer," *Mathematical Problems in Engineering*, vol. 2019, 2019.

[15] M. Fu, L. Dong, Y. Xu, and C. Wang, "Barrier Lyapunov function-based backstepping control for acv safety trajectory tracking," in *Global Oceans 2020: Singapore–US Gulf Coast*. IEEE, 2020, pp. 1–7.

[16] E. D. Sontag, "A 'universal' construction of Artstein's theorem on nonlinear stabilization," *Systems & Control Letters*, vol. 13, no. 2, pp. 117–123, 1989.

[17] H. G. Tanner and K. J. Kyriakopoulos, "Backstepping for nonsmooth systems," *Automatica*, vol. 39, no. 7, pp. 1259–1265, 2003.

[18] P. Glotfelter, J. Cortés, and M. Egerstedt, "Nonsmooth barrier functions with applications to multi-robot systems," *Control Systems Letters*, vol. 1, no. 2, pp. 310–315, 2017.

[19] P. Ong and J. Cortés, "Universal formula for smooth safe stabilization," in *Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 2373–2378.

[20] L. Perko, *Differential equations and dynamical systems*. Springer Science & Business Media, 2013, vol. 7.

[21] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *Transactions on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, 2017.

[22] R. Konda, A. D. Ames, and S. Coogan, "Characterizing safety: Minimal control barrier functions from scalar comparison systems," *Control Systems Letters*, vol. 5, no. 2, pp. 523–528, 2020.

[23] M. Jankovic, "Robust control barrier functions for constrained stabilization of nonlinear systems," *Automatica*, vol. 96, pp. 359–367, 2018.

[24] G. M. Tallis, "The moment generating function of the truncated multi-normal distribution," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 23, no. 1, pp. 223–229, 1961.

[25] ——, "Plane truncation in normal populations," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 27, no. 2, pp. 301–307, 1965.