Adaptive Control Barrier Functions

Wei Xiao, *Member, IEEE*, Calin Belta, *Fellow, IEEE*, and Christos G. Cassandras, *Life Fellow, IEEE*

***Abstract*—It has been shown that optimizing quadratic costs while stabilizing affine control systems to desired (sets of) states subject to state and control constraints can be reduced to a sequence of quadratic programs (QPs) by using control barrier functions (CBFs) and control Lyapunov functions (CLFs). In this article, we introduce adaptive CBFs (aCBFs) that can accommodate time-varying control bounds and noise in the system dynamics while also guaranteeing the feasibility of the QPs if the original quadratic cost optimization problem itself is feasible, which is a challenging problem in current approaches. We propose two different types of aCBFs: parameter-adaptive CBF (PACBF) and relaxation-adaptive CBF (RACBF). Central to aCBFs is the introduction of appropriate time-varying functions to modify the definition of a common CBF. These time-varying functions are treated as high-order CBFs with their own auxiliary dynamics, which are stabilized by CLFs. We demonstrate the advantages of using aCBFs over the existing CBF techniques by applying both the PACBF-based method and the RACBF-based method to a cruise control problem with time-varying road conditions and noise in the system dynamics, and compare their relative performance.**

***Index Terms*—Control barrier function (CBF), Lyapunov methods, optimal control, safety-critical control.**

I. Introduction

Barrier functions (BFs) are Lyapunov-like functions [1], [2], whose use can be traced back to optimization problems [3]. More recently, they have been employed to prove set invariance [4]–[6] and for multiobjective control [7]. In [1], it was proved that if a BF for a given set satisfies Lyapunov-like conditions, then the set is forward invariant. A less restrictive form of a BF, which is allowed to grow when far away from the boundary of the set, was proposed in [8]. Another approach that allows a BF to become zero (the safe set boundary) was proposed in [9] and [10]. This simpler form has also been considered in time-varying cases and applied to enforce signal temporal logic formulas as hard constraints [10]. Control BFs (CBFs) are extensions of BFs for control systems, and are used to map a constraint defined over system states onto a constraint on the control input. Recently, it has been shown that, to stabilize an affine control system while optimizing a quadratic cost and satisfying state and control constraints, CBFs can be combined with control Lyapunov functions (CLFs) [11]–[14] to form quadratic programs (QPs) [8], [9], [15] that are solved in real time.

The CBFs from [8] and [9] work for constraints that have relative degree one with respect to the system dynamics. A backstepping approach was introduced in [16] to address higher relative degree constraints, and it was shown to work for relative degree two. A CBF method for position-based constraints with relative degree two was also proposed in [17]. A more general form [18], which works for arbitrarily high relative degree constraints, employs input–output linearization and finds a pole placement controller with negative poles to stabilize the CBF to zero. Thus, this is an exponential CBF. The high-order CBF (HOCBF) proposed in [19] is simpler and more general than the exponential CBF [18] in the sense that it only needs to recursively define a sequence of CBFs. Adaptive CBFs (aCBFs) have been proposed in [20] for systems with parameter uncertainties, and a less conservative Robust aCBF (RaCBF) [21] that is combined with a data-driven method is proposed to achieve adaptive safety. Machine learning techniques have also been applied to achieve adaptive safety for systems with parameter uncertainties [22], [23]. However, the associated QPs can easily be infeasible when both state constraints (enforced by HOCBFs) and tight control bounds are involved. Most works [10], [18], [20]–[23] to date have not considered control bounds; moreover, safety constraints are still not guaranteed when the QP becomes infeasible. Although the constant penalties used in the definition of the HOCBF can help to improve feasibility [19], this may not work under *time-varying* control bounds and noisy dynamics. In addition, the HOCBF method is conservative in the sense that the satisfaction of the HOCBF constraint is only sufficient for the original constraint to hold, which can adversely affect system performance.

To address the aforementioned problem feasibility and conservativeness issues under time-varying control bounds and noisy dynamics (as opposed to system parameter uncertainties), in this article, we propose *Adaptive* CBFs (aCBFs). Specifically, we first propose a parameter-adaptive CBF (PACBF) by introducing penalty functions in the definition of an HOCBF, and define auxiliary dynamics for these penalty functions that are themselves HOCBFs (such that they are guaranteed to be

Manuscript received December 14, 2020; revised February 24, 2021; accepted April 17, 2021. Date of publication April 21, 2021; date of current version April 26, 2022. This work was supported in part by the NSF under Grants ECCS-1931600, DMS-1664644, CNS-1645681, IIS-1723995, and IIS-2024606, in part by the ARPAE under Grant DE-AR0001282 and by its NEXTCAR Program under Grant DE-AR0000796, in part by the AFOSR under Grant FA9550-19-1-0158, in part by the MathWorks, and in part by the NPRP under Grant 12S-0228-190177 from the Qatar National Research Fund (a member of the Qatar Foundation). Recommended by Associate Editor Z. Li. *(Corresponding author: Wei Xiao.)*

The authors are with the Division of Systems Engineering and Center for Information and Systems Engineering, Boston University, Brookline, MA 02446 USA (e-mail: xiaowei@bu.edu; cbelta@bu.edu; cgc@bu.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TAC.2021.3074895.

Digital Object Identifier 10.1109/TAC.2021.3074895

nonnegative) and are stabilized by CLFs. This way, the PACBF constraint is relaxed by the penalty functions through the control inputs of the auxiliary penalty function dynamics while the forward invariance property of the HOCBF method is still guaranteed. Since the PACBF constraint is relaxed through the penalty functions, we show that its satisfaction is a necessary and sufficient condition for the satisfaction of the original constraint, which leads to improvements in the performance of the system. In addition, we propose an alternative type of aCBF termed relaxation-adaptive CBF (RACBF) by introducing a relaxation variable to the original constraint, and treat it similarly to the PACBF so that the relaxation is guaranteed to be nonnegative. In this way, the original constraint is also guaranteed to be satisfied. Both PACBF and RACBF provide adaptivity for the CBFs, but in different ways. The safety guarantee and adaptivity of aCBFs are formally shown in the article.

The use of the CBF method can significantly simplify the derivation of controls that satisfy all the constraints in an optimal control problem, especially for high relative degree constraints and nonlinear systems. The aCBF method we propose in this article can address the feasibility and conservativeness issues of the CBF method mentioned before. Thus, we can also formulate optimal control problems with constraints given by aCBFs (either PACBFs or RACBFs) and CLFs. We validate the adaptivity of the proposed aCBF on an adaptive cruise control (ACC) problem with different and time-varying control bounds (e.g., on different road surfaces and with tires slipping), as well as with noisy dynamics. The results clearly demonstrate the advantages of the proposed aCBF. We also compare the performances of the CBF-based, PACBF-based, and RACBF-based methods.

The remainder of the article is organized as follows. In Section II, we provide background and preliminary results on HOCBFs and CLFs, and then introduce the PACBF in Section III. The RACBF is presented in Section IV. The ACC problem is formulated and solved in Section V, followed by simulations in Section VI. We conclude with final remarks and directions for future work in Section VII.

II. Preliminaries

Definition 1 (Class $\mathcal{K}$ function [24]): A continuous function $\alpha : [0, a) \to [0, \infty), a > 0$ is said to belong to class $\mathcal{K}$ if it is strictly increasing and $\alpha(0) = 0$.

Consider an affine control system of the form

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

where $\boldsymbol{x} \in \mathbb{R}^n$, $f : \mathbb{R}^n \to \mathbb{R}^n$ and $g : \mathbb{R}^n \to \mathbb{R}^{n\times q}$ are locally Lipschitz, and $\boldsymbol{u} \in U \subset \mathbb{R}^q$ is the control constraint set defined as

$$U := \{\boldsymbol{u} \in \mathbb{R}^q : \boldsymbol{u}_{\min} \leq \boldsymbol{u} \leq \boldsymbol{u}_{\max}\} \tag{2}$$

with $\boldsymbol{u}_{\min}, \boldsymbol{u}_{\max} \in \mathbb{R}^q$ and the inequalities are interpreted componentwise. There is usually some noise in the dynamics (1), and such noise could enter either additively or multiplicatively in $f(\boldsymbol{x})$ and/or $g(\boldsymbol{x})$. We will show in the sequel how to deal with the presence of noise.

Definition 2: A set $C \subset \mathbb{R}^n$ is forward invariant for system (1) if its solutions starting at any $\boldsymbol{x}(0) \in C$ satisfy $\boldsymbol{x}(t) \in C$, $\forall t \geq 0$.

Definition 3 (Relative degree): The relative degree of a (sufficiently many times) differentiable function $b : \mathbb{R}^n \to \mathbb{R}$ with respect to system (1) is the number of times it needs to be differentiated along its dynamics until the control $\boldsymbol{u}$ explicitly shows in the corresponding derivative.

In this article, since function b is used to define a constraint $b(\boldsymbol{x}) \geq 0$, we will also refer to the relative degree of b as the relative degree of the constraint. For a constraint $b(\boldsymbol{x}) \geq 0$ with relative degree m, $b : \mathbb{R}^n \to \mathbb{R}$, and $\psi_0(\boldsymbol{x}) := b(\boldsymbol{x})$, we define a sequence of functions $\psi_i : \mathbb{R}^n \to \mathbb{R}, i \in \{1, \ldots, m\}$

$$\psi_i(\boldsymbol{x}) := \dot{\psi}_{i-1}(\boldsymbol{x}) + \alpha_i(\psi_{i-1}(\boldsymbol{x})), \quad i \in \{1, \ldots, m\} \tag{3}$$

where $\alpha_i(\cdot), i \in \{1, \ldots, m\}$ denotes a $(m - i)$th order differentiable class $\mathcal{K}$ function.

We further define a sequence of sets $C_i, i \in \{1, \ldots, m\}$ associated with (3) in the form

$$C_i := \{\boldsymbol{x} \in \mathbb{R}^n : \psi_{i-1}(\boldsymbol{x}) \geq 0\}, \quad i \in \{1, \ldots, m\}. \tag{4}$$

Definition 4 (High-Order Control BF [19]): Let $C_1, \ldots, C_m$ be defined by (4) and $\psi_1(\boldsymbol{x}), \ldots, \psi_m(\boldsymbol{x})$ be defined by (3). A function $b : \mathbb{R}^n \to \mathbb{R}$ is an HOCBF of relative degree m for system (1) if there exist $(m - i)th$ order differentiable class $\mathcal{K}$ functions $\alpha_i, i \in \{1, \ldots, m - 1\}$ and a class $\mathcal{K}$ function α_m such that

$$\begin{aligned} \sup_{\boldsymbol{u}\in U}[L_f^m b(\boldsymbol{x}) + [L_g L_f^{m-1} b(\boldsymbol{x})]\boldsymbol{u} + S(b(\boldsymbol{x})) \\ + \alpha_m(\psi_{m-1}(\boldsymbol{x}))] \geq 0 \end{aligned} \tag{5}$$

for all $\boldsymbol{x} \in C_1 \cap, \ldots, \cap C_m$. In (5), L_f^m (L_g) denotes Lie derivatives along f (g) m (one) times, and

$$S(b(\boldsymbol{x})) = \sum_{i=1}^{m-1} L_f^i(\alpha_{m-i} \circ \psi_{m-i-1})(\boldsymbol{x})$$

where $\circ$ denotes the composition of functions. Furthermore, $b(\boldsymbol{x})$ is such that $L_g L_f^{m-1} b(\boldsymbol{x}) \neq 0$ on the boundary of the set $C_1 \cap, \ldots, \cap C_m$.

The HOCBF is a general form of the relative degree one CBF [8]–[10] (setting $m = 1$ reduces the HOCBF to the common CBF form in [8]–[10]), and it is also a more general form of the exponential CBF [18]. Note that we can define $\alpha_i(\cdot), i \in \{1, \ldots, m\}$ in Definition 4 to be extended class $\mathcal{K}$ functions ($\alpha : [-a, a] \to [-\infty, \infty]$ as in Definition 1) to ensure robustness of an HOCBF to perturbations [25]; this is due to the fact that the HOCBF constraint becomes a Lyapunov-like condition with extended class $\mathcal{K}$ functions. However, the use of extended class $\mathcal{K}$ functions cannot ensure a constraint to be satisfied if it is initially violated. Thus, in this article, we only focus on the use of class $\mathcal{K}$ functions.

Theorem 1 (see [19]): Given an HOCBF $b(\boldsymbol{x})$ from Definition 4 with the associated sets $C_1, \ldots, C_m$ defined by (4), if $\boldsymbol{x}(0) \in C_1 \cap, \ldots, \cap C_m$, then any Lipschitz continuous controller $\boldsymbol{u}(t)$ that satisfies (5), $\forall t \geq 0$ renders $C_1 \cap, \ldots, \cap C_m$ forward invariant for system (1).

Definition 5 (Control Lyapunov function [14]): A continuously differentiable function $V : \mathbb{R}^n \to \mathbb{R}$ is an exponentially stabilizing CLF for system (1) if there exist constants $c_1 > 0, c_2 > 0, c_3 > 0$ such that for $\forall \boldsymbol{x} \in \mathbb{R}^n$, $c_1||\boldsymbol{x}||^2 \leq V(\boldsymbol{x}) \leq c_2||\boldsymbol{x}||^2$

$$\inf_{u \in U}[L_f V(\boldsymbol{x}) + L_g V(\boldsymbol{x})\boldsymbol{u} + c_3 V(\boldsymbol{x})] \leq 0. \tag{6}$$

Many existing works [8], [18], [26] combine CBFs for systems with relative degree one with quadratic costs to form optimization problems. An explicit solution to such problems can be obtained based on some assumptions [25]. Alternatively, we can discretize time and an optimization problem with constraints given by the CBFs [inequalities of the form (5)] is solved at each time step. The intersampling effect in this approach is considered in [26]. If convergence to a state is desired, then a CLF constraint of the form (6) is added, as in [8] and [26]. Note that these constraints are linear in control since the state value is fixed at the beginning of the interval; therefore, each optimization problem is a QP. The optimal control obtained by solving each QP is applied at the current time step and held constant for the whole interval. The state is updated using dynamics (1), and the procedure is repeated. Replacing CBFs by HOCBFs allows us to handle constraints with arbitrary relative degree [19]. This method works conditioned on the fact that the QP at every time step is feasible. However, this is not guaranteed, in particular under tight control bounds. In this article, we show how the QP feasibility can be guaranteed by using adaptive CBFs.

III. Parameter-aCBFs

We introduce the PACBF in this section. We begin with a simple example to motivate the need for PACBFs and to illustrate the main ideas.

A. Example: Simplified ACC

Consider the simplified ACC (SACC) problem with the ego (controlled) vehicle dynamics in the form

$$\begin{bmatrix} \dot{v}(t) \\ \dot{z}(t) \end{bmatrix} = \begin{bmatrix} 0 \\ v_p - v(t) \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u(t) \tag{7}$$

where $z(t)$ denotes the distance between the preceding and the ego vehicle, $v_p > 0, v(t)$ denote the velocities of the preceding and ego vehicles along the lane (the velocity of the preceding vehicle is assumed constant), respectively, and $u(t)$ is the control of the ego vehicle, subject to the constraints

$$u_{\text{min}} \leq u(t) \leq u_{\text{max}} \quad \forall t \geq 0 \tag{8}$$

where $u_{\text{min}} < 0$ and $u_{\text{max}} > 0$ are the minimum and maximum control input, respectively.

We require that the distance $z(t)$ between the ego vehicle and its immediately preceding vehicle be greater than $l_p > 0$ (the coordinates of the ego vehicle and the preceding vehicle, respectively, are measured from the same origin), i.e.,

$$z(t) \geq l_p \quad \forall t \geq 0. \tag{9}$$

Let $\boldsymbol{x}(t) := (v(t), z(t))$ and $b(\boldsymbol{x}(t)) := z(t) - l_p$. The relative degree of $b(\boldsymbol{x}(t))$ is $m = 2$, so we choose an HOCBF following Definition 4 by defining $\psi_0(\boldsymbol{x}(t)) := b(\boldsymbol{x}(t))$, $\alpha_1(\psi_0(\boldsymbol{x}(t))) := \psi_0(\boldsymbol{x}(t))$ and $\alpha_2(\psi_1(\boldsymbol{x}(t))) := \psi_1(\boldsymbol{x}(t))$. Then, we seek a control for the ego vehicle such that the constraint (9) is satisfied. The control $u(t)$ should satisfy (5), which in this case is

$$\begin{aligned} &\underbrace{0}_{L_f^2 b(\boldsymbol{x}(t))} + \underbrace{-1}_{L_g L_f b(\boldsymbol{x}(t))} \times u(t) + \underbrace{v_p - v(t)}_{S(b(\boldsymbol{x}(t)))} \\ &\quad + \underbrace{v_p - v(t) + z(t) - l_p}_{\alpha_2(\psi_1(\boldsymbol{x}(t)))} \geq 0. \end{aligned} \tag{10}$$

Suppose we wish to minimize $\int_0^T u^2(t)dt$. Then, we can use the QP-based method [8] outlined at the end of the last section to solve this SACC problem. However, the HOCBF constraint (10) can easily conflict with $u_{\text{min}} \leq u(t)$ in (8) when the two vehicles get close to each other, as shown in [19]. When this happens, the QP will be infeasible. We can use the penalty method from the work in [19] to improve the QP feasibility, i.e., we define $\psi_1(\boldsymbol{x}(t)) = \dot{b}(\boldsymbol{x}(t)) + pb(\boldsymbol{x}(t)), \psi_2(\boldsymbol{x}(t)) = \dot{\psi}_1(\boldsymbol{x}(t)) + p\psi_1(\boldsymbol{x}(t)), p > 0$ in (3). Based on (10), the control $u(t)$ should then satisfy

$$\begin{aligned} &\underbrace{0}_{L_f^2 b(\boldsymbol{x}(t))} + \underbrace{-1}_{L_g L_f b(\boldsymbol{x}(t))} \times u(t) + \underbrace{p(v_p - v(t))}_{S(b(\boldsymbol{x}(t)))} \\ &\quad + \underbrace{p(v_p - v(t)) + p^2(z(t) - l_p)}_{p\alpha_2(\psi_1(\boldsymbol{x}(t)))} \geq 0. \end{aligned} \tag{11}$$

Given u_{min}, we can find a small enough value for p such that (11) will not conflict with u_{min} in (8), i.e., the QP is always feasible. However, in practice the value of u_{min} is not a constant; it depends on weather conditions, different road surfaces, etc. Therefore, a proper choice of p for specific conditions ensuring that the QP is feasible is not easy to make when the environment changes. Moreover, the assumption of a constant speed v_p for the front vehicle is too strong, and noise in the vehicle dynamics may also make the QP infeasible. This motivates us to define an aCBF that works for time-varying control bounds and noisy dynamics, i.e., it theoretically ensures that the QP is always feasible.

B. Parameter aCBF

The key idea in converting a regular CBF into an adaptive one is to include the penalty terms as shown in (11) and, then, replace them by time-varying functions with suitable properties as detailed next. Starting with a relative degree m function $b : \mathbb{R}^n \to \mathbb{R}$, let $\psi_0(\boldsymbol{x}) := b(\boldsymbol{x})$. Then, instead of using a constant penalty $p_i > 0, i \in \{1, \ldots, m\}$ for each class $\mathcal{K}$ function $\alpha_i(\cdot)$ in the definition of an HOCBF [19], we define a time-varying *penalty function* $p_i(t) \geq 0$, $i \in \{1, \ldots, m\}$, and use it as a multiplicative factor for each class $\mathcal{K}$ function $\alpha_i(\cdot)$. Let $\boldsymbol{p}(t) := (p_1(t), \ldots, p_m(t))$. Similar to (3), we define a sequence of functions $\psi_i : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}, i \in \{1, \dots, m\}$ in the form

$$\begin{aligned} \psi_1(\boldsymbol{x}, \boldsymbol{p}(t)) &:= \dot{\psi}_0(\boldsymbol{x}) + p_1(t)\alpha_1(\psi_0(\boldsymbol{x})) \\ \psi_i(\boldsymbol{x}, \boldsymbol{p}(t)) &:= \dot{\psi}_{i-1}(\boldsymbol{x}, \boldsymbol{p}(t)) + p_i(t)\alpha_i(\psi_{i-1}(\boldsymbol{x}, \boldsymbol{p}(t))), \\ & \qquad i \in \{2, \dots, m\} \end{aligned} \tag{12}$$

where $\alpha_i(\cdot), i \in \{1, \dots, m-1\}$ is a $(m-i)$th-order differentiable class $\mathcal{K}$ function, and $\alpha_m(\cdot)$ is a class $\mathcal{K}$ function.

We further define a sequence of sets $C_i, i \in \{1, \dots, m\}$ associated with (12) in the form

$$\begin{aligned} C_1 &:= \{\boldsymbol{x} \in \mathbb{R}^n : \psi_0(\boldsymbol{x}) \geq 0\} \\ C_i &:= \{(\boldsymbol{x}, \boldsymbol{p}(t)) \in \mathbb{R}^n \times \mathbb{R}^m : \psi_{i-1}(\boldsymbol{x}, \boldsymbol{p}(t)) \geq 0\}. \end{aligned} \tag{13}$$

The remaining question is how to choose $p_i(t), i \in \{1, \dots, m\}$. We require that $p_i(t) \geq 0, \forall i \in \{1, \dots, m-1\}$; therefore, we define each $p_i(t)$ to be an HOCBF, similar to the definition of $b(\boldsymbol{x}) \geq 0$ in Definition 4. Just like $b(\boldsymbol{x})$ is associated with the dynamic system (1), we need to introduce an *auxiliary dynamic system* for $p_i(t)$. Moreover, as in Definition 4, each penalty function $p_i(t), i \in \{1, \dots, m-1\}$ will be differentiated $m - i$ times while $p_m(t)$ is not differentiated. Thus, we start by defining $\boldsymbol{\pi}_i(t) := (\pi_{i,1}(t), \pi_{i,2}(t), \dots, \pi_{i,m-i}(t)) \in \mathbb{R}^{m-i}, i \in \{1, \dots, m-2\}$, where $\pi_{i,j} \in \mathbb{R},\ j \in \{1, \dots, m-i\}$ are the auxiliary state variables. Next, we define $\boldsymbol{\pi}_{m-1}(t) = p_{m-1}(t) \in \mathbb{R}$, which needs to be differentiated only once. Finally, we set $p_m(t) \geq 0$ as some function to be determined. Let $\pi_{i,1}(t) = p_i(t)$ in (12). We define input–output linearizable auxiliary dynamics for each p_i (we, henceforth, omit the time variable t for simplicity) through the auxiliary state $\boldsymbol{\pi}_i$ in the form

$$\begin{aligned} \dot{\boldsymbol{\pi}}_i &= F_i(\boldsymbol{\pi}_i) + G_i(\boldsymbol{\pi}_i)\nu_i, \quad i \in \{1, \dots, m-1\} \\ y_i &= p_i \end{aligned} \tag{14}$$

where y_i denotes the output, $F_i : \mathbb{R}^{m-i} \to \mathbb{R}^{m-i}, G_i : \mathbb{R}^{m-i} \to \mathbb{R}^{m-i}$, and $\nu_i \in \mathbb{R}$ denotes the control input for the auxiliary dynamics (14). The exact forms of F_i, G_i do not greatly affect the system performance as they are mainly used to guarantee the nonnegative property of p_i shown later. For simplicity, we usually adopt linear forms. For example, we define $\dot{p}_{m-2} = \pi_{m-2,2}$, $\dot{\pi}_{m-2,2} = \nu_{m-2}$ since we need to differentiate p_{m-2} twice as in Definition 4, and define $\dot{p}_{m-1} = \nu_{m-1}$ since we need to differentiate p_{m-1} once. We can initialize $\boldsymbol{\pi}_i(0)$ to any vector as long as $p_i(0) > 0$.

An alternative way of viewing (14) is by defining a set of additional state variables, which cause the dynamic system (1) to be augmented. In particular, let $\boldsymbol{\Pi} := (\boldsymbol{\pi}_1, \dots, \boldsymbol{\pi}_{m-1}), \boldsymbol{\nu} := (\nu_1, \dots, \nu_{m-1})$, where $\nu_i, i \in \{1, \dots, m-1\}$ are the controls in the auxiliary dynamics (14). In order to properly define the PACBF, we augment system (1) with the auxiliary dynamics (14) in the form

$$\begin{bmatrix} \dot{\boldsymbol{x}} \\ \dot{\boldsymbol{\Pi}} \end{bmatrix} = \underbrace{\begin{bmatrix} f(\boldsymbol{x}) \\ \boldsymbol{F}_0(\boldsymbol{\Pi}) \end{bmatrix}}_{F(\boldsymbol{x}, \boldsymbol{\Pi})} + \underbrace{\begin{bmatrix} g(\boldsymbol{x}) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{G}_0(\boldsymbol{\Pi}) \end{bmatrix}}_{G(\boldsymbol{x}, \boldsymbol{\Pi})} \begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{\nu} \end{bmatrix} \tag{15}$$

where $\boldsymbol{F}_0(\boldsymbol{\Pi}) = (F_1(\boldsymbol{\pi}_1), \dots, F_{m-1}(\boldsymbol{\pi}_{m-1}))$ and $\boldsymbol{G}_0(\boldsymbol{\Pi})$ is a matrix composed by $G_i(\boldsymbol{\pi}_i), i \in \{1, \dots, m-1\}$ with dimension $\frac{m(m-1)}{2} \times (m-1)$. $F : \mathbb{R}^{n + \frac{m(m-1)}{2}} \to \mathbb{R}^{n + \frac{m(m-1)}{2}}, G : \mathbb{R}^{n + \frac{m(m-1)}{2}} \to \mathbb{R}^{(n + \frac{m(m-1)}{2}) \times (q+m-1)}$.

Since p_i is an HOCBF with relative degree $m - i$ for (14), similar to (5), we define a constraint set $U_{cbf}(\boldsymbol{\Pi})$ for $\boldsymbol{\nu}$

$$\begin{aligned} U_{cbf}(\boldsymbol{\Pi}) = \{\boldsymbol{\nu} \in \mathbb{R}^{m-1} : L_{F_i}^{m-i} p_i + [L_{G_i} L_{F_i}^{m-i-1} p_i]\nu_i + S(p_i) \\ + \alpha_{m-i}(\psi_{i,m-i-1}(p_i)) \geq 0 \quad \forall i \in \{1, 2, \dots, m-1\}\} \end{aligned} \tag{16}$$

where $\psi_{i,m-i-1}(\cdot)$ is defined similar to (3).

Definition 6: Let $C_i, i \in \{1, \dots, m\}$ be defined by (13), $\psi_i(\boldsymbol{x}, \boldsymbol{p}), i \in \{1, \dots, m\}$ be defined by (12), and the auxiliary dynamics be defined by (14). A function $b : \mathbb{R}^n \to \mathbb{R}$ is a PACBF with relative degree m for (1) if every $p_i, i \in \{1, \dots, m-1\}$ is an HOCBF with relative degree $m - i$ for the auxiliary dynamics (14), and there exist $(m-i)th$-order differentiable class $\mathcal{K}$ functions $\alpha_i, i \in \{1, \dots, m-1\}$, and a class $\mathcal{K}$ function α_m such that

$$\begin{aligned} \sup_{\boldsymbol{u} \in U, \boldsymbol{\nu} \in U_{cbf}} & [L_f^m b(\boldsymbol{x}) + [L_g L_f^{m-1} b(\boldsymbol{x})]\boldsymbol{u} \\ & + \sum_{i=1}^{m-1} \alpha_i(\psi_{i-1})[L_{G_i} L_{F_i}^{m-i-1} p_i]\nu_i + R(b(\boldsymbol{x}), \boldsymbol{p}) \\ & + \sum_{i=1}^{m-1} [L_{F_i}^{m-i} p_i]\alpha_i(\psi_{i-1}) + p_m \alpha_m(\psi_{m-1})] \geq 0 \end{aligned} \tag{17}$$

for all $\boldsymbol{x} \in C_1, (\boldsymbol{x}, \boldsymbol{p}) \in C_2 \cap, \dots, \cap C_m$, and all $p_m \geq 0$. In (17), $R(b(\boldsymbol{x}), \boldsymbol{p})$ denotes the remaining Lie derivative terms of $b(\boldsymbol{x})$ (or $\boldsymbol{p}$) along f (or F_i,$i \in \{1, \dots, m-1\}$) with degree less than m (or $m - i$), similar to the form in (5).

In (17), the explicit terms $R(b(\boldsymbol{x}), \boldsymbol{p})$ are omitted for simplicity, but examples can be found in the revisited SACC example later in this section or in Section V. The complex PACBF constraint (17) can be simplified [similar to (5)] if we just consider the augmented dynamics (15) in the form

$$\begin{aligned} \sup_{\boldsymbol{u} \in U, \boldsymbol{\nu} \in U_{cbf}} & [L_F^m b(\boldsymbol{x}) + [L_G L_F^{m-1} b(\boldsymbol{x})]\boldsymbol{u} \\ & + S(b(\boldsymbol{x}), \boldsymbol{p}, \boldsymbol{\nu}) + \alpha_m(\psi_{m-1}(\boldsymbol{x}, \boldsymbol{p}))] \geq 0 \end{aligned} \tag{18}$$

where $S(b(\boldsymbol{x}), \boldsymbol{p}, \boldsymbol{\nu})$ is linear in $\boldsymbol{\nu}$ and contains $\alpha_i(\cdot), i \in \{1, \dots, m-1\}$ [similar to the form in (5)]. Note that $\alpha_i(\cdot)$ in the last equation denote general class $\mathcal{K}$ functions, and they can be different from the ones in (17).

Given a PACBF $b(\boldsymbol{x})$, we consider all control values $(\boldsymbol{u}, \boldsymbol{\nu}) \in U \times U_{cbf}(\boldsymbol{\Pi})$ that satisfy

$$\begin{aligned} K_{acbf}(\boldsymbol{x}, \boldsymbol{\Pi}) = \{(\boldsymbol{u}, \boldsymbol{\nu}) \in U \times U_{cbf}(\boldsymbol{\Pi}) : [L_g L_f^{m-1} b(\boldsymbol{x})]\boldsymbol{u} \\ + \sum_{i=1}^{m-1} \alpha_i(\psi_{i-1})[L_{G_i} L_{F_i}^{m-i-1} p_i]\nu_i + \sum_{i=1}^{m-1} [L_{F_i}^{m-i} p_i]\alpha_i(\psi_{i-1}) \\ + L_f^m b(\boldsymbol{x}) + R(b(\boldsymbol{x}), \boldsymbol{p}) + p_m \alpha_m(\psi_{m-1}) \geq 0\}. \end{aligned} \tag{19}$$

Theorem 2: Given a PACBF $b(\boldsymbol{x})$ from Definition 6 with the associated sets $C_1, C_2, \dots, C_m$ defined by (13), if $\boldsymbol{x}(0) \in C_1$

and $(\boldsymbol{x}(0), \boldsymbol{p}(0)) \in C_2 \cap, \dots, \cap C_m$, then any Lipschitz continuous controller $(\boldsymbol{u}(t), \boldsymbol{\nu}(t)) \in K_{acbf}(\boldsymbol{x}(t), \boldsymbol{\Pi}(t)), \forall t \geq 0$ renders the set C_1 forward invariant for system (1) and $C_2 \cap \cdots \cap C_m$ forward invariant for systems (1) and (14), respectively.

Proof: If $b(\boldsymbol{x})$ is a PACBF, then $p_m(t) \geq 0, \forall t \geq 0$. Constraint (17) is the Lie derivative form of $\dot{\psi}_{m-1}(\boldsymbol{x}, \boldsymbol{p}) + p_m \alpha_m(\psi_{m-1}(\boldsymbol{x}, \boldsymbol{p})) \geq 0$. If $p_m(t) > 0, \forall t \geq 0$, it follows from Theorem 1 that $\psi_{m-1}(\boldsymbol{x}(t), \boldsymbol{p}(t)) \geq 0, \forall t \geq 0$. If $p_m(t) = 0$, then $\dot{\psi}_{m-1}(\boldsymbol{x}, \boldsymbol{p}) \geq 0$. Since $(\boldsymbol{x}, \boldsymbol{p}) \in C_m$ (i.e., $\psi_{m-1}(\boldsymbol{x}, \boldsymbol{p}) \geq 0$ is initially satisfied), we have $\psi_{m-1}(\boldsymbol{x}(t), \boldsymbol{p}(t)) \geq 0, \forall t \geq 0$. Because $p_i, \forall i \in \{1, \dots, m-1\}$ is an HOCBF for the auxiliary dynamics (14), it follows from Theorem 1 that $p_i(t) \geq 0, \forall t \geq 0, \forall i \in \{1, \dots, m-1\}$. Then, we can recursively prove that $\psi_i(\boldsymbol{x}(t), \boldsymbol{p}(t)) \geq 0, \forall t \geq 0, \forall i \in \{2, \dots, m-2\}$ similar to the case $i = m-1$, and eventually prove that $\psi_0(\boldsymbol{x}(t)) \geq 0, \forall t \geq 0$, i.e., $b(\boldsymbol{x}(t)) \geq 0, \forall t \geq 0$. Therefore, the sets C_1 and $C_2 \cap, \dots, \cap C_m$ are forward invariant. ■

Remark 1 (Adaptivity of PACBFs): In the PACBF constraint (17), the control $\boldsymbol{u}$ of system (1) depends on the controls $\nu_i, \forall i \in \{1, \dots, m-1\}$ of the auxiliary dynamics (14). The control ν_i is only constrained by the HOCBF constraint in (16) since we require that p_i is an HOCBF, and there are no control bounds on ν_i. Therefore, we partially relax the constraints on the control input of system (1) in the PACBF constraint (17) by allowing the penalty function $p_i(t), \forall i \in \{1, \dots, m\}$ to change through $\boldsymbol{\nu}$. However, the forward invariance of the set C_1 is still guaranteed, i.e., the original constraint $b(\boldsymbol{x}) \geq 0$ is guaranteed to be satisfied. This is how a PACBF provides "adaptivity." Note that we may not need to define a penalty function p_i for every class $\mathcal{K}$ function $\alpha_i(\cdot)$ in (12); we can instead define penalty functions for only some of them.

Adaptivity to Changing Control Bounds and Noisy Dynamics: In the HOCBF method, the QPs may be infeasible in the presence of both control limitations (2) and the HOCBF constraint (5). There are two reasons for the problem to become infeasible: (i) the control limitations (2) are too tight or they are time varying such that the HOCBF constraint (5) will conflict with (2) after it becomes active; (ii) the dynamics (1) are not accurately modeled, there may be uncertain variables, etc. In case (ii), the HOCBF constraint (5) might also conflict with (2) when both of them become active. This is because the state variables also show up in the HOCBF constraint (5); thus, the noisy dynamics can easily (and randomly) change the HOCBF constraint (5) through the (noisy) state variables such that (5) may conflict with the control limitations when they are also active. However, the QP feasibility is not only improved, but in fact guaranteed as will be shown in Theorem 4. This is because in the PACBF method, the control $\boldsymbol{u}$ in the PACBF constraint (17) is relaxed by $\nu_i, \forall i \in \{1, \dots, m-1\}$, as discussed in Remark 1.

Theorem 3: Given an PACBF $b(\boldsymbol{x})$ from Definition 6 with the associated sets $C_1, \dots, C_m$ defined by (13), if $b(\boldsymbol{x}(0)) > 0$, then the satisfaction of the PACBF constraint (17) is a necessary and sufficient condition for the satisfaction of the original constraint $b(\boldsymbol{x}) > 0$.

Proof: If $b(\boldsymbol{x}(0)) > 0$, it follows from Theorem 2 that we can always choose proper class $\mathcal{K}$ functions (linear, quadratic, etc.) such that $\psi_0(\boldsymbol{x}(t)) = b(\boldsymbol{x}) > 0$ and $\psi_i(\boldsymbol{x}(t), \boldsymbol{p}(t)) > 0, i \in \{1, \dots, m-1\}, \forall t \geq 0$. Thus, the satisfaction of the PACBF constraint (17) is a sufficient condition for the satisfaction of the original constraint $b(\boldsymbol{x}) > 0$.

If $b(\boldsymbol{x}(t)) > 0$, there exists a penalty function $p_1(t) \geq 0$ (since $p_1(t)$ is an HOCBF) such that $\dot{b}(\boldsymbol{x}) > -p_1(t)\alpha_1(b(\boldsymbol{x}))$ for any $\dot{b}(\boldsymbol{x})$ with respect to dynamics (1) (because $\alpha_1(b(\boldsymbol{x})) > 0$). With $\psi_0(\boldsymbol{x}) = b(\boldsymbol{x})$ in (12), we have $\dot{\psi}_0(\boldsymbol{x}) + p_1(t)\alpha_1(\psi_0(\boldsymbol{x})) > 0$ (i.e., $\psi_1(\boldsymbol{x}, \boldsymbol{p}) > 0$). The ith derivative of $b(\boldsymbol{x})$ shows in $\psi_i, i \in \{2, \dots, m-1\}$, and we can also prove similarly that there exists a penalty function $p_i(t) \geq 0$ (since $p_i(t)$ is an HOCBF) such that $\psi_i(\boldsymbol{x}, \boldsymbol{p}) > 0, i \in \{2, \dots, m-1\}$ in a recursive way. Eventually, there exists $p_m(t) \geq 0$ such that $\dot{\psi}_{m-1}(\boldsymbol{x}, \boldsymbol{p}) + p_m(t)\alpha_m(\psi_{m-1}(\boldsymbol{x}, \boldsymbol{p})) \geq 0$ (i.e., $\psi_m(\boldsymbol{x}, \boldsymbol{p}) \geq 0$). Since $\psi_m(\boldsymbol{x}, \boldsymbol{p}) \geq 0$ is equivalent to the satisfaction of the PACBF constraint (17), it follows that the satisfaction of the PACBF constraint (17) is a necessary condition for the satisfaction of the original constraint $b(\boldsymbol{x}(t)) > 0$. ■

Note that we exclude $b(\boldsymbol{x}) = 0$ in Theorem 3. The satisfaction of the PACBF constraint (17) is equivalent to the satisfaction of $b(\boldsymbol{x}) \geq 0$; hence, the system performance is not reduced by the mapping of a constraint from state to control (this is how the standard CBFs work). In essence, we have added some extra degrees of freedom to our ability to satisfy the constraints of the original optimal control problem by augmenting system (1)–(15). These degrees of freedom come from $\boldsymbol{\Pi}$, which is controlled by $\boldsymbol{\nu}$. Therefore, if the PACBF is a legitimate CBF for the augmented system, it guarantees the satisfaction of the original constraint and the QP feasibility. The use of a PACBF provides flexibility in constraint satisfaction in a dynamic (time varying) way at the expense of dealing with additional state variables $\boldsymbol{\Pi}$.

Example Revisited: For the SACC problem introduced in Section III-A, we define a PACBF with $m = 2$ for (9). We still choose $\alpha_1(b(\boldsymbol{x}(t))) = b(\boldsymbol{x}(t))$ and $\alpha_2(\psi_1(\boldsymbol{x}(t))) = \psi_1(\boldsymbol{x}(t))$ in Definition 6. Suppose we only consider a penalty function $p_1(t)$ on the class $\mathcal{K}$ function $\alpha_1(\cdot)$ and define linear dynamics for p_1 in the simple form $\dot{p}_1 = \nu_1$ [note that $\pi_1 = p_1$ in (14)]. In order for $b(\boldsymbol{x}(t)) := z(t) - l_p$ to be a PACBF for (7), a control input $u(t)$ should satisfy

$$\underbrace{0}_{L_f^2 b(\boldsymbol{x}(t))} + \underbrace{-1}_{L_g L_f b(\boldsymbol{x}(t))} \times u(t) + \underbrace{(z(t) - l_p)}_{[L_{G_1} p_1(t)]\alpha_1(b(\boldsymbol{x}(t)))} \times \nu_1(t) + \underbrace{p_1(t)(v_p - v(t))}_{R(b(\boldsymbol{x}(t)), \boldsymbol{p}(t))} + \underbrace{v_p - v(t) + p_1(t)(z(t) - l_p)}_{\alpha_2(\psi_1(\boldsymbol{x}(t), \boldsymbol{p}(t)))} \geq 0. \tag{20}$$

Comparing the previous equation with (11), we replace the constant p in (11) with $p_1(t)$, and thus, the control $u(t)$ is relaxed by $\nu_1(t)$. This can actually guarantee the problem feasibility as will be shown in Theorem 4.

Since $u(t)$ depends on ν_1 that has no bounds, the control input $u(t)$ in the above PACBF constraint is relaxed. Thus, this constraint is adaptive to the change of the control bound $u_{\min}$ in (8) and uncertainties in v_p and $x_p(t)$ from the front vehicle. Note that $p_1(t)$ should be an HOCBF for the auxiliary dynamics $\dot{p}_1 = \nu_1$. The control input ν_1 is subject to the corresponding HOCBF constraint such that $p_1(t) \geq 0, \forall t \geq 0$ is satisfied.

C. Optimal Control With PACBFs

Consider an optimal control problem for system (1)

$$\min_{\boldsymbol{u}(t)} \int_0^T \mathcal{C}(||\boldsymbol{u}(t)||)dt \tag{21}$$

where $||\cdot||$ denotes the two-norm of a vector, $\mathcal{C}(\cdot)$ is a strictly increasing function of its argument, and $T > 0$.

Suppose system (1) is not accurately modeled and may also have noisy states [e.g., as an additive term to (1)]. In addition, system (1) has time-varying control bounds defined as $\boldsymbol{u}_{\min}(t), \boldsymbol{u}_{\max}(t) \in \mathbb{R}^q$, where we assume that $\boldsymbol{u}_{\min}(t), \boldsymbol{u}_{\max}(t)$ are Lipschitz continuous

$$U(t) := \{\boldsymbol{u} \in \mathbb{R}^q : \boldsymbol{u}_{\min}(t) \leq \boldsymbol{u} \leq \boldsymbol{u}_{\max}(t)\}. \tag{22}$$

Assume a (safety) constraint $b(\boldsymbol{x}) \geq 0$ with relative degree m has to be satisfied by system (1). We use the PACBF method to guarantee $b(\boldsymbol{x}) \geq 0$ so that $\boldsymbol{u}$ should satisfy the PACBF constraint (17). Moreover, each $\nu_i, i \in \{1, \ldots, m-1\}$ is constrained by the HOCBF constraint (16) corresponding to the constraint $p_i(t) \geq 0$ for the auxiliary dynamics (14).

Note that the control $\boldsymbol{\nu}$ from the auxiliary dynamics is only subject to the HOCBF constraints defined in (16). In particular, for each $\nu_i, i \in \{1, \ldots, m-1\}$ this constraint is one-sided, i.e.,

$$\nu_i \geq \frac{-L_{F_i}^{m-i} p_i - S(p_i) - \alpha_{m-i}(\psi_{i,m-i-1}(p_i))}{L_{G_i} L_{F_i}^{m-i-1} p_i} \tag{23}$$

if $L_{G_i} L_{F_i}^{m-i-1} p_i > 0$. The adaptivity of a PACBF depends on the auxiliary dynamics (14) through ν_i. If ν_i changes too fast, it can affect the smoothness of the control $\boldsymbol{u}$ obtained through solving the associated QPs, which may adversely affect the performance of system (1).

If we add control bounds on ν_i, the problem feasibility may be decreased (i.e., the adaptivity of a PACBF is weakened). If we add a quadratic penalty term ν_i^2 into the cost function, then p_i may maintain a large value, which contradicts the penalty method from [19] (i.e., we wish to have a small enough value of p_i to improve the problem feasibility). Therefore, in order to decrease p_i when it is large, we seek to minimize ν_i and stabilize each $p_i(t)$ to a small enough value $p_i^* > 0$ (for example, as recommended by the penalty method from [19] or by the optimal penalties learned in [27]). We choose smaller p_i^* if $\alpha_i(\cdot)$ is a high-order (e.g., polynomial) function as its value is larger and requires more penalization. The choice of p_i^* can provide conditions such that the problem feasibility is guaranteed, as shown in Theorem 4.

Assuming the auxiliary dynamics (14) are input–output linearized (otherwise, we perform input–output linearization), we can use either the tracking control from [24] or the CLF to stabilize $p_i(t)$, i.e., if $m = 1$, we minimize $(p_1 - p_1^*)^2$, if $m = 2$, we define a CLF $V_1(p_1) := (p_1 - p_1^*)^2$ as in Definition 5, and if $m > 2$, we find a desired state feedback form $\hat{p}_{i,m-i}$ for $p_{i,m-i}$

$$\hat{p}_{i,m-i} = \begin{cases} -k_1(p_i - p_i^*), & i = m-2 \\ -k_1(p_i - p_i^*) - k_2 p_{i,2} - \ldots, -k_{m-i-1} p_{i,m-i-1}, & \\ & i < m-2 \end{cases} \tag{24}$$

where $k_1 > 0, \ldots, k_{m-i-1} > 0$. In the last equation, if $i = m-1$, we can directly define a CLF $V_i(\boldsymbol{\pi}_i) := (p_i - p_i^*)^2$ as in Definition 5.

Then, we can define a CLF $V_i(\boldsymbol{\pi}_i) := (p_{i,m-i} - \hat{p}_{i,m-i})^2, i \in \{1, \ldots, m-1\}$ (the relative degree of $V_i(\boldsymbol{\pi}_i)$ is one) to stabilize each p_i with $c_2 = \epsilon > 0$ in Definition 5, so that any control input ν_i should satisfy

$$L_{F_i} V_i(\boldsymbol{\pi}_i) + L_{G_i} V_i(\boldsymbol{\pi}_i)\nu_i + \epsilon V_i(\boldsymbol{\pi}_i) \leq \delta_i \tag{25}$$

where δ_i is a relaxation variable that we seek to minimize.

In all cases, we may also want to stabilize p_m, which is not differentiated, by minimizing $(p_m - p_m^*)^2$. Therefore, letting $\boldsymbol{\delta} := (\delta_1, \delta_2, \ldots, \delta_{m-1})$, we can reformulate the cost (21) to incorporate the PACBF as follows:

$$\min_{\boldsymbol{u}(t), \boldsymbol{\nu}(t), \boldsymbol{\delta}(t), p_m(t)} \int_0^T \left[\mathcal{C}(||\boldsymbol{u}(t)||) + \sum_{i=1}^{m-1} W_i \nu_i(t) + \sum_{i=1}^{m-1} P_i \delta_i^2(t) + Q(p_m(t) - p_m^*)^2 \right] dt \tag{26}$$

subject to (1), (14), (17), and (22), the HOCBF constraint in (16) for each $p_i \geq 0, i \in \{1, \ldots, m-1\}$, $p_m(t) \geq 0$, and the CLF constraint (25). In (26), $W_i > 0, P_i > 0, i \in \{1, \ldots, m-1\}$, and $Q \geq 0$. Then, we can use the QP-based approach introduced at the end of Section II to solve (26).

Finding the weights in (26) may not be a simple task, as the ranges of the associated controllable variables may vary widely. Therefore, we normalize each term and form a convex combination

$$\min_{\boldsymbol{u}(t), \boldsymbol{\nu}(t), \boldsymbol{\delta}(t), p_m(t)} \int_0^T \left[c_0 \frac{\mathcal{C}(||\boldsymbol{u}(t)||)}{u_{\lim}} + \sum_{i=1}^{m-1} W_i \frac{\nu_i(t)}{\nu_{i,\max}} + \sum_{i=1}^{m-1} P_i \frac{\delta_i^2(t)}{\delta_{i,\max}^2} + Q \frac{(p_m(t) - p_m^*)^2}{p_{m,\max}^2} \right] dt \tag{27}$$

subject to the same constraints as (26), where $c_0 + \sum_{i=1}^{m-1} W_i + \sum_{i=1}^{m-1} P_i + Q = 1$. The aforementioned upper bounds are chosen so that $u_{\lim} = \sup_{\boldsymbol{u} \in U} \mathcal{C}(||\boldsymbol{u}(t)||)$, $\nu_{i,\max} > 0$, $\delta_{i,\max} > 0$, $p_{m,\max} > 0$. Since $\nu_i(t), \delta_i(t), p_m(t)$ do not have any natural upper bounds, it is hard to determine $\nu_{i,\max}, \delta_{i,\max}, p_{m,\max}$. However, we can exploit the fact that $\nu_{i,\max}, \delta_{i,\max}, p_{m,\max}$ are always desired to have small values and set these to be the largest acceptable values, depending on the problem of interest.

Complexity: The time complexity of each QP is polynomial in the dimension $d > 0$ of decision variables. In the HOCBF-based QP, $d = q$, where q is the dimension of the control $\boldsymbol{u}$. However, in (27), $d = q + 2\text{ m} - 1$. We increase the adaptivity of the CBF method at the expense of more computation time, but the PACBF-based QP is still fast enough, as also seen in Section VI.

Due to the pointwise solving method of the QP (27) introduced at the end of Section II, we can claim that an optimal control problem is "intrinsically" infeasible starting at some time instant when the safety constraint $b(\boldsymbol{x}) \geq 0$ conflicts with the control bounds (22) no matter how we choose the control law for system (1). The PACBF constraint (17) is active when $(\boldsymbol{u}, \boldsymbol{\nu})$ makes both (17) and (16) equalities. If the PACBF constraint (17) becomes active earlier, the QP (27) is easier to be feasible as system (1) has a longer time horizon to adjust its state under control bounds (22). Let $t_f \geq 0$ denote the last time that the problem (21), subject to (22) and $b(\boldsymbol{x}) \geq 0$, is feasible. In the following theorem, we provide conditions such that the feasibility of the QP (27) is guaranteed.

Theorem 4: Suppose $\boldsymbol{x}(0)$ is not on the boundary of C_1. If the PACBF constraint (17) is active before t_f, then the QP (27) feasibility is guaranteed.

Proof: Since $\boldsymbol{x}(0)$ is not on the boundary of C_1, we have by Theorem 3 that the satisfaction of the PACBF constraint (17) is a necessary and sufficient condition for the satisfaction of $b(\boldsymbol{x}) \geq 0$, thus, $b(\boldsymbol{x}) \geq 0 \Leftrightarrow$ (17), and the mapping of a constraint from the state onto the control will not limit the control that system (1) can take with respect to the original problem (21), subject to (22) and $b(\boldsymbol{x}) \geq 0$. As the problem (21), subject to (22) and $b(\boldsymbol{x}) \geq 0$, is feasible, there exists a control law such that the problem is feasible after the PACBF constraint (17) becomes active; hence, the QP (27) feasibility is guaranteed. ■

Remark 2: In order to apply Theorem 4, we may try to find $\bar{p}_i > 0, i \in \{1, \dots, m\}$ with $p_i^* = \bar{p}_i$ in (25) such that system (1) should take $\boldsymbol{u}(t) = \boldsymbol{u}_{\max}(t)$ or $\boldsymbol{u}_{\min}(t)$ (or mixture of maximum and minimum controls), $\forall t \geq t_a$ to make the QP (27) feasible, where $t_a \geq 0$ denotes the time that the PACBF constraint (17) first becomes active. Then, any $p_i^* \leq \bar{p}_i, \forall i \in \{1, \dots, m\}$ can make the QP (27) feasible with the PACBF method, as smaller penalties make the PACBF constraint become active earlier [19]. When noise is involved and it is bounded, we can apply the worst case noise (i.e., its bound) to system (1) and, as above, find $\bar{p}_i > 0$ with $p_i^* = \bar{p}_i$ in (25) such that system (1) should take $\boldsymbol{u}(t) = \boldsymbol{u}_{\max}(t)$ or $\boldsymbol{u}_{\min}(t)$ (or mixture of maximum and minimum controls), $\forall t \geq t_a$ to make the QP (27) feasible. Then, any $p_i^* \leq \bar{p}_i$ would work when the noise is within the bound.

IV. Relaxation-aCBFs

In this section, we introduce the RACBF. The RACBF works similarly as the PACBF, but tries to obtain adaptivity through a relaxation variable from the original constraint instead of introducing penalty functions to the HOCBF in Definition 4.

A. Relaxation-aCBF

Recall that in PACBFs, we define $\psi_0(\boldsymbol{x}) = b(\boldsymbol{x})$ for a relative degree m function $b(\boldsymbol{x})$, where $b : \mathbb{R}^n \to \mathbb{R}$, and introduce multiplicative penalty functions to all class $\mathcal{K}$ functions in (12) to obtain adaptivity. As an alternative, we may also relax ψ_0 in the form

$$\psi_0(\boldsymbol{x}, r(t)) := b(\boldsymbol{x}) - r(t) \tag{28}$$

where $r(t) \geq 0$ is a relaxation that plays a similar role as penalty functions in a PACBF to obtain adaptivity.

We require that $r(t) \geq 0, \forall t \geq 0$; therefore, we define $r(t)$ to be an HOCBF, similar to the definition of $b(\boldsymbol{x}) \geq 0$ in Definition 4. Just like $b(\boldsymbol{x})$ is associated with the dynamic system (1), we need to introduce an *auxiliary dynamic system* for $r(t)$. Moreover, as in Definition 4, the relaxation $r(t)$ will be differentiated m times. Thus, we define $\boldsymbol{R}(t) := (r_1(t), r_2(t), \dots, r_m(t)) \in \mathbb{R}^m$, where $r_1(t) \equiv r(t)$ and $r_j \in \mathbb{R}$, $j \in \{2, \dots, m\}$ are the auxiliary state variables for which we define input–output linearizable auxiliary dynamics for $r(t)$ (we, henceforth, omit the time variable t for simplicity) in the form

$$\begin{aligned} \dot{\boldsymbol{R}} &= f_0(\boldsymbol{R}) + g_0(\boldsymbol{R})\nu \\ y &= r \end{aligned} \tag{29}$$

where y denotes the output, $f_0 : \mathbb{R}^m \to \mathbb{R}^m$, $g_0 : \mathbb{R}^m \to \mathbb{R}^m$, and $\nu \in \mathbb{R}$ denotes the control input for the auxiliary dynamics (29). The exact forms of f_0, g_0 may be defined based on a specific application. For simplicity, we usually adopt a linear form. We can initialize $\boldsymbol{r}(0)$ to any real number vector as long as $r(0) > 0$.

In order to properly define the RACBF, we augment system (1) with the auxiliary dynamics (29) in the form

$$\begin{bmatrix} \dot{\boldsymbol{x}} \\ \dot{\boldsymbol{R}} \end{bmatrix} = \underbrace{\begin{bmatrix} f(\boldsymbol{x}) \\ f_0(\boldsymbol{R}) \end{bmatrix}}_{F(\boldsymbol{x},\boldsymbol{R})} + \underbrace{\begin{bmatrix} g(\boldsymbol{x}) & 0 \\ \boldsymbol{0} & g_0(\boldsymbol{R}) \end{bmatrix}}_{G(\boldsymbol{x},\boldsymbol{R})} \begin{bmatrix} \boldsymbol{u} \\ \nu \end{bmatrix} \tag{30}$$

where $F : \mathbb{R}^{n+m} \to \mathbb{R}^{n+m}, G : \mathbb{R}^{n+m} \to \mathbb{R}^{(n+m)\times(q+1)}$.

Since r is an HOCBF with relative degree m for (29), similar to (5), we define a constraint set $U_0(\boldsymbol{R})$ for ν

$$\begin{aligned} U_0(\boldsymbol{R}) = \{\nu \in \mathbb{R} : L_{f_0}^m r + [L_{g_0} L_{f_0}^{m-1} r]\nu + S(r) \\ + \alpha_m(\psi_{m-1}(r)) \geq 0\} \end{aligned} \tag{31}$$

where $\psi_{m-1}(r)$ is defined similar to (3).

With the auxiliary dynamics (29), we have $\psi_0(\boldsymbol{x}, \boldsymbol{R}) = b(\boldsymbol{x}) - r$. We define a sequence of functions $\psi_i : \mathbb{R}^{n+m} \to \mathbb{R}, i \in \{1, \dots, m\}$ in the form

$$\psi_i(\boldsymbol{x}, \boldsymbol{R}) := \dot{\psi}_{i-1}(\boldsymbol{x}, \boldsymbol{R}) + \alpha_i(\psi_{i-1}(\boldsymbol{x}, \boldsymbol{R})), i \in \{1, \dots, m\} \tag{32}$$

where $\alpha_i(\cdot), i \in \{1, \dots, m\}$ denotes a $(m-i)$th-order differentiable class $\mathcal{K}$ function.

We further define a sequence of sets $C_i, i \in \{1, \dots, m\}$ associated with (32) in the form

$$C_i := \{(\boldsymbol{x}, \boldsymbol{R}) \in \mathbb{R}^{n+m} : \psi_{i-1}(\boldsymbol{x}, \boldsymbol{R}) \geq 0\}, i \in \{1, \dots, m\}. \tag{33}$$

Definition 7: Let $C_i, i \in \{1, \dots, m\}$ be defined by (33), $\psi_i(\boldsymbol{x}, \boldsymbol{R}), i \in \{1, \dots, m\}$ be defined by (32) with $\psi_0(\boldsymbol{x}, \boldsymbol{R}) := b(\boldsymbol{x}) - r(t)$, and the auxiliary dynamics be defined by (29). A function $b : \mathbb{R}^n \to \mathbb{R}$ is an RACBF with relative degree m for (30) if r is an HOCBF with relative degree m for the auxiliary dynamics (29), and there exist $(m-i)th, i \in \{1, \dots, m-1\}$, order differentiable class $\mathcal{K}$ functions α_i, and a class $\mathcal{K}$ function α_m such that

$$\sup_{\boldsymbol{u}\in U,\nu\in U_0} [L_F^m \psi_0(\boldsymbol{x},\boldsymbol{R}) + L_G L_F^{m-1}\psi_0(\boldsymbol{x},\boldsymbol{R}) \begin{bmatrix} \boldsymbol{u} \\ \nu \end{bmatrix} + S(\psi_0(\boldsymbol{x},\boldsymbol{R})) + \alpha_m(\psi_{m-1}(\boldsymbol{x},\boldsymbol{R}))] \geq 0 \tag{34}$$

for all $(\boldsymbol{x},\boldsymbol{R}) \in C_1\cap,\ldots,\cap C_m$. In (34), $S(\psi_0(\boldsymbol{x},\boldsymbol{R}))$ denotes the remaining Lie derivative terms of $\psi_0(\boldsymbol{x},\boldsymbol{R})$ along F with degree less than m.

Theorem 5: Given an RACBF $b(\boldsymbol{x})$ from Definition 7 with the associated sets $C_1,\ldots,C_m$ defined by (33), if $(\boldsymbol{x}(0),\boldsymbol{R}(0) \in C_1\cap,\ldots,\cap C_m$, then any Lipschitz continuous controller $(\boldsymbol{u}(t),\nu(t))$ that satisfies (34), $\forall t\geq 0$ renders $C_1\cap,\ldots,\cap C_m$ forward invariant for system (30), and renders $b(\boldsymbol{x}(t))\geq 0, \forall t\geq 0$ for system (1).

Proof: It follows from Theorem 1 that the set $C_1\cap,\ldots,\cap C_m$ is forward invariant for system (30). Since $r(t)$ is an HOCBF, we also have $r(t)\geq 0,\forall t\geq 0$. Since C_1 is forward invariant, we have $\psi_0(\boldsymbol{x}(t),\boldsymbol{R}(t)) = b(\boldsymbol{x}(t)) - r(t)\geq 0, \forall t\geq 0$. Therefore, $b(\boldsymbol{x}(t))\geq r(t)\geq 0,\forall t\geq 0$. ■

Remark 3 (Adaptivity of RACBF): The control $\boldsymbol{u}$ of system (1) depends on the control ν of the auxiliary system (29) in the RACBF constraint (34). Similar to the PACBF, the control ν is only constrained by the HOCBF constraint in (31) since we require that r is an HOCBF. Therefore, we also relax the constraints on the control input of system (1) in the RACBF by allowing the relaxation r to be time varying. The satisfaction of the original constraint $b(\boldsymbol{x})\geq 0$ is still guaranteed. This is how an RACBF provides "adaptivity." This adaptivity can guarantee problem feasibility under time-varying control bounds and noisy dynamics if some additional conditions are satisfied, as will be discussed in the next section.

B. Optimal Control With RACBF

Consider an optimal control problem for system (1) with the cost defined as (21), and the time-varying control bounds defined as (22).

Assume a (safety) constraint $b(\boldsymbol{x})\geq 0$ with relative degree m has to be satisfied by system (1). We use the RACBF method to guarantee $b(\boldsymbol{x})\geq 0$ so that $\boldsymbol{u}$ should satisfy the RACBF constraint (34). Moreover, each ν is constrained by the HOCBF constraint (31) corresponding to the constraint $r(t)\geq 0$ for the auxiliary dynamics (29).

However, if $\boldsymbol{R}(t) = \boldsymbol{0}, \forall t\in[0,T]$, we have $\nu = 0$ following from (29) and (31). Then, an RACBF loses its adaptivity as the control $\boldsymbol{u}$ is not relaxed by ν in the RACBF constraint (34). Similar to the penalty function $p_i(t)$ of a PACBF, we also use a CLF to stabilize $r(t)$ to some desirable value $r^* > 0$. Assuming the auxiliary dynamics (14) are input–output linearized (otherwise, we perform input–output linearization), we can use either the tracking control from [24] or the CLF to stabilize $r(t)$, i.e., if $m = 1$, we define a CLF $V(\boldsymbol{r}) := (r - r^*)^2$ as in Definition 5, and if $m\geq 2$, we find a desired state feedback form $\hat{r}_m$ for r_m [note that $r_1\equiv r$ in (29)]

$$\hat{r}_m = \begin{cases} -k_1(r-r^*), & m=2 \\ -k_1(r-r^*)-k_2r_2-\ldots,-k_{m-1}r_{m-1}, & m>2 \end{cases} \tag{35}$$

where $k_1 > 0,\ldots,k_{m-1} > 0$.

Remark 4: The desired value r^* should be chosen such that the problem itself is feasible when the RACBF constraint (34) first becomes active. We may find some $\bar{r} > 0$ such that the problem is feasible under the worst case conditions (e.g., the maximum approaching speed of a vehicle toward another vehicle) and worst case noise (i.e., its bound). Then, any $r^*\geq\bar{r}$ would work when the noise is within the bound. The exact choice of r^* depends on the particular application. For the ACC problem, we should choose r^* to be no less than the minimum braking distance as in [8].

Then, we can define a CLF $V(\boldsymbol{R}) := (r_m - \hat{r}_m)^2$ [the relative degree of $V(\boldsymbol{R})$ is one] to stabilize r with $c_2 = \epsilon > 0$ in Definition 5, so that any control input ν should satisfy

$$L_{f_0}V(\boldsymbol{R}) + L_{g_0}V(\boldsymbol{R})\nu + \epsilon V(\boldsymbol{R}) \leq \delta_r \tag{36}$$

where δ_r is a relaxation variable that we want to minimize.

Therefore, we can reformulate the cost (21) to incorporate the RACBF as follows:

$$\min_{\boldsymbol{u}(t),\nu(t),\delta_r(t)} \int_0^T \mathcal{C}(||\boldsymbol{u}(t)||) + P_r\nu^2(t) + P_r\delta_r^2(t)dt \tag{37}$$

subject to (1), (22), (29), (34), the HOCBF constraint in (31) for $r\geq 0$, and the CLF constraint (36). In (37), $P_r > 0$. We can, then, use the QP-based approach introduced at the end of Section II to solve (37). We can also normalize each term in (37) and form a convex combination as in (27).

Complexity: In (37), the time complexity is still polynomial in the dimension d of the decision variables, where $d = q + 2$. Note that this is smaller than $q + 2m - 1$ in the PACBF method (the relative degree m does not affect the complexity).

Remark 5 (Forward invariance of RACBF with noise): In order to make sure that $b(\boldsymbol{x})\geq 0$ is guaranteed under noisy dynamics, we can make $r\geq r_a > 0$ instead of $r\geq 0$, i.e., define $r - r_a$ as an HOCBF. The value of r_a depends on the magnitude of the noise level.

Comparison Between PACBF and RACBF: The PACBF achieves adaptivity by using penalty functions in its definition, and this can alleviate the conservativeness of the CBF method, whereas the RACBF achieves adaptivity through a relaxation variable. The definition of an RACBF is close to an HOCBF in Definition 4 such that its conservativeness still exists, whereas a PACBF is not conservative following Theorem 3. Therefore, a PACBF has better adaptivity than an RACBF. However, as already stated, RACBF has better time complexity than PACBF. The tradeoff between adaptivity and complexity depends on the available computational resources—PACBF is preferable if computational resources are available. We can also combine these two alternative approaches by simultaneously adding a relaxation variable as in (28) and multiplying (partially) penalty functions as in (12).

V. ACC Problem Formulation

In this section, we consider the ACC problem, which is a more realistic version of the SACC problem introduced in Section III-A.

Vehicle Dynamics: Instead of the simple dynamics in (7), we consider more accurate vehicle dynamics in the form

$$\underbrace{\begin{bmatrix}\dot{v}(t)\\ \dot{z}(t)\end{bmatrix}}_{\dot{\boldsymbol{x}}(t)} = \underbrace{\begin{bmatrix}-\frac{1}{M}F_r(v(t))\\ v_p - v(t)\end{bmatrix}}_{f(\boldsymbol{x}(t))} + \underbrace{\begin{bmatrix}\frac{1}{M}\\ 0\end{bmatrix}}_{g(\boldsymbol{x}(t))} u(t) \tag{38}$$

where M denotes the mass of the controlled vehicle and $F_r(v(t))$ denotes the resistance force, which is expressed [24] as: $F_r(v(t)) = f_0\text{sgn}(v(t)) + f_1 v(t) + f_2 v^2(t)$, where $f_0 > 0, f_1 > 0$, and $f_2 > 0$ are scalars determined empirically. The first term in $F_r(v(t))$ denotes the Coulomb friction force, the second term denotes the viscous friction force, and the last term denotes the aerodynamic drag.

Vehicle limitations consist of vehicle constraints on speed and acceleration

$$\begin{aligned} v_{\min} \le v(t) \le v_{\max} \quad \forall t \in [0, t_f] \\ -c_d(t)Mg \le u(t) \le c_a(t)Mg \quad \forall t \in [0, t_f] \end{aligned} \tag{39}$$

where $v_{\max} > 0$ and $v_{\min} \ge 0$ denote the maximum and minimum allowed speeds, respectively, while $c_d(t) > 0$ and $c_a(t) > 0$ are deceleration and acceleration coefficients, respectively, and g is the gravity constant.

Safety constraint consists of the safety requirement (9).

Objective 1 (Desired speed): The controlled vehicle should achieve a desired speed $v_d > 0$.

Objective 2 (Minimum acceleration): The controlled vehicle should minimize

$$\min_{u(t)} \int_0^T \left(\frac{u(t) - F_r(v(t))}{M}\right)^2 dt. \tag{40}$$

Problem 1: Determine control laws to achieve Objectives 1 and 2 subject to the vehicle limitations (39) and safety constraint (9) for the vehicle governed by dynamics (38).

To solve Problem 1, we will use an aCBF (PACBF or RACBF) to implement constraint (9) and use HOCBFs to impose constraints (39) on the control input. We will also use a CLF as in Definition 5 to achieve Objective 1. We seek to achieve Objective 2 through a minimization problem with the cost in (40). We adopt the QP-based method introduced in [8]. The relative degree of (9) is 2, so we define an aCBF with $m = 2$ for it.

To achieve Objective 1, we use a CLF to stabilize $v(t)$ to v_d and relax the corresponding constraint (6) to make it a soft one [14]. Consider a Lyapunov function $V_{\text{acc}}(\boldsymbol{x}(t)) := (v(t) - v_d)^2$, with $c_1 = c_2 = 1$ and $c_3 = \epsilon > 0$ in Definition 5. Any control input $u(t)$ should satisfy for all $t \in [0, t_f]$ (the explicit form is omitted but can be found in [19])

$$L_f V_{\text{acc}}(\boldsymbol{x}(t)) + \epsilon V_{\text{acc}}(\boldsymbol{x}(t)) + L_g V_{\text{acc}}(\boldsymbol{x}(t))u(t) \le \delta_{\text{acc}}(t). \tag{41}$$

Here, $\delta_{\text{acc}}(t)$ is a relaxation variable that makes (41) a soft constraint.

To impose the *vehicle limitations*, note that the relative degrees of the speed limit constraints are 1. Therefore, we use HOCBFs with $m = 1$ to map these constraints from the speed $v(t)$ to the control input $u(t)$. Let $b_1(\boldsymbol{x}(t)) := v_{\max} - v(t)$, $b_2(\boldsymbol{x}(t)) := v(t) - v_{\min}$ and choose $\alpha_1(b_i) = b_i, i \in \{1, 2\}$ in Definition 4 for both HOCBFs. Then, any control $u(t)$ should satisfy (the explicit form is omitted but can be found in [19])

$$\begin{aligned} L_f b_1(\boldsymbol{x}(t)) + L_g b_1(\boldsymbol{x}(t))u(t) + b_1(\boldsymbol{x}(t)) \ge 0 \\ L_f b_2(\boldsymbol{x}(t)) + L_g b_2(\boldsymbol{x}(t))u(t) + b_2(\boldsymbol{x}(t)) \ge 0. \end{aligned} \tag{42}$$

Since the control limitations are already constraints on the control input, we obviously do not need HOCBFs for them.

Safety Constraint With PACBF: Since the HOCBF constraint for (9) can easily conflict with the control limitations in (39), we use an aCBF with $m = 2$, the relative degree of (9). Letting $b(\boldsymbol{x}(t)) := z(t) - l_p$, we define $\psi_1(\cdot), \psi_2(\cdot)$

$$\begin{aligned} \psi_1(\boldsymbol{x}(t), \boldsymbol{p}(t)) &:= \dot{b}(\boldsymbol{x}(t)) + p_1(t)b^2(\boldsymbol{x}(t)) \\ \psi_2(\boldsymbol{x}(t), \boldsymbol{p}(t)) &:= \dot{\psi}_1(\boldsymbol{x}(t), \boldsymbol{p}(t)) + p_2(t)\psi_1(\boldsymbol{x}(t), \boldsymbol{p}(t)). \end{aligned} \tag{43}$$

Note that in the above equations, we define $\alpha_1(\cdot)$ as a quadratic function. We define high-order class $\mathcal{K}$ functions (such as high-order polynomials) in order to make the original constraint $b(\boldsymbol{x}) \ge 0$ not be violated by the noise, as shown in [19].

We then define the auxiliary dynamics (14) for $p_1(t)$ and adopt the form

$$\dot{p}_1(t) = \nu_1(t). \tag{44}$$

There are no derivatives involved for $p_2(t)$, so we just set $p_2(t) \ge 0, \forall t \ge 0$ to be determined. Combining the dynamics (38) and (44), with (43), any control input $u(t)$ should satisfy the aCBF constraint (17), which in this case is

$$\begin{aligned} &\underbrace{\frac{F_r(v(t))}{M}}_{L_f^2 b(\boldsymbol{x}(t))} + \underbrace{\frac{-1}{M}}_{L_g L_f b(\boldsymbol{x}(t))} u(t) + \underbrace{b^2(\boldsymbol{x}(t))}_{L_{G_1} p_1(t) \alpha_2(\psi_1)} \nu_1(t) \\ &+ 2p_1(t)b(\boldsymbol{x}(t))L_f b(\boldsymbol{x}(t)) + p_2(t)\psi_1(\boldsymbol{x}(t), \boldsymbol{p}(t)) \ge 0. \end{aligned} \tag{45}$$

Since $p_1(t)$ has to be an HOCBF and its relative degree is 1 for (44), any control input $\nu_1(t)$ should satisfy the HOCBF constraint (16), which in this case is (taking α_1 as a linear function in Definition 4)

$$\underbrace{0}_{L_{F_1} p_1(t)} + \underbrace{1}_{L_{G_1} p_1(t)} \nu_1(t) + p_1(t) \ge 0 \tag{46}$$

with $p_1(0) > 0$. Finally, we wish to stabilize $p_1(t)$ to a desired $p_1^* > 0$ (usually a small number), and define a CLF $V_1(p_1(t)) := (p_1(t) - p_1^*)^2$ with $c_1 = c_2 = 1$ and $c_3 = \epsilon > 0$ in Definition 5. Any control input should satisfy

$$\underbrace{0}_{L_{F_1} V_1(p_1(t))} + \underbrace{2(p_1(t) - p_1^*)}_{L_{G_1} V_1(p_1(t))} \nu_1(t) + \epsilon V_1(p_1(t)) \le \delta_1(t). \tag{47}$$

Note that we also wish to minimize $(p_2(t) - p_2^*)^2$, where $p_2^* > 0$.

Safety Constraint With RACBF: We can also use an RACBF with $m = 2$ to implement the safety constraint (9). Letting $b(\boldsymbol{x}(t)) := z(t) - l_p$ and $\psi_0(\boldsymbol{x}(t), \boldsymbol{R}(t)) = z(t) - l_p - r(t)$, we define $\psi_1(\cdot), \psi_2(\cdot)$

$$\begin{aligned} \psi_1(\boldsymbol{x}(t), \boldsymbol{R}(t)) &:= \dot{\psi}_0(\boldsymbol{x}(t), \boldsymbol{R}(t)) + k_1 \psi_0^2(\boldsymbol{x}(t), \boldsymbol{R}(t)) \\ \psi_2(\boldsymbol{x}(t), \boldsymbol{R}(t)) &:= \dot{\psi}_1(\boldsymbol{x}(t), \boldsymbol{R}(t)) + k_2 \psi_1(\boldsymbol{x}(t), \boldsymbol{R}(t)) \end{aligned} \tag{48}$$

where $k_1 > 0, k_2 > 0$ are two constants instead of the penalty functions in (43). We then define the auxiliary dynamics (29) for $r(t)$ and adopt the form $\dot{r}(t) = r_2(t), \dot{r}_2(t) = \nu(t)$. We augment the dynamics (38) in the form

$$\underbrace{\begin{bmatrix} \dot{v}(t) \\ \dot{z}(t) \\ \dot{r}(t) \\ \dot{r}_2(t) \end{bmatrix}}_{(\dot{\boldsymbol{x}}(t), \dot{\boldsymbol{R}}(t))} = \underbrace{\begin{bmatrix} -\frac{1}{M}F_r(v(t)) \\ v_p - v(t) \\ r_2(t) \\ 0 \end{bmatrix}}_{F(\boldsymbol{x}(t), \boldsymbol{R}(t))} + \underbrace{\begin{bmatrix} \frac{1}{M} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}}_{G(\boldsymbol{x}(t), \boldsymbol{R}(t))} \begin{bmatrix} u \\ \nu \end{bmatrix}. \tag{49}$$

Combining dynamics (49) with (48), any control input $(u(t), \nu(t))$ should satisfy the RACBF constraint (34), which in this case is

$$\underbrace{\frac{F_r(v(t))}{M}}_{L_F^2\psi_0(\boldsymbol{x}, \boldsymbol{R})} + \underbrace{[\frac{-1}{M}, -1]}_{L_G L_F \psi_0(\boldsymbol{x}, \boldsymbol{R})} \begin{bmatrix} u \\ \nu \end{bmatrix} + k_2\psi_1(\boldsymbol{x}, \boldsymbol{R}) \tag{50}$$
$$+ 2k_1\psi_0(\boldsymbol{x}, \boldsymbol{R}) L_F\psi_0(\boldsymbol{x}, \boldsymbol{R}) \geq 0.$$

Following Remark 5, let $b_r(\boldsymbol{R}(t)) = r(t) - r_a$, where $r_a > 0$. $b_r(\boldsymbol{R}(t))$ has to be an HOCBF whose relative degree is 2 for the auxiliary dynamics. Taking α_1, α_2 as linear functions in Definition 4, any control input $\nu(t)$ should satisfy the HOCBF constraint (31), which in this case is

$$\underbrace{0}_{L_{f_0}^2 b_r(\boldsymbol{R}(t))} + \underbrace{1}_{L_{g_0} L_{f_0} b_r(\boldsymbol{R}(t))} \nu(t) + 2L_{f_0} b_r(\boldsymbol{R}(t)) + b_r(\boldsymbol{R}(t)) \geq 0 \tag{51}$$

with $r(0) > 0$. Finally, we also wish to stabilize $r(t)$ to a desired $r^* > 0$, and define a CLF $V(\boldsymbol{R}(t)) := (r_2(t) + k_1(r(t) - r^*))^2$ with $k_1 > 0$, $c_1 = c_2 = 1$ and $c_3 = \epsilon > 0$ in Definition 5. Any control input should satisfy

$$2(r_2(t) + k_1(r(t) - r^*))(\nu(t) + k_1 r_2(t)) + \epsilon V(\boldsymbol{R}(t)) \leq \delta_r(t). \tag{52}$$

ACC Problem Reformulation: As explained at the end of Section II, we partition the time interval $[0, T]$ into a set of equal time intervals $\{[0, \Delta t), [\Delta t, 2\Delta t), \ldots\}$, where $\Delta t > 0$. In each interval $[k\Delta t, (k+1)\Delta t)$ $(k = 0, 1, 2, \ldots)$, we assume the control is constant (i.e., the overall control will be piecewise constant), and reformulate Problem 1 as a sequence of QPs. Specifically, with the PACBF method, at $t = k\Delta t$ $(k = 0, 1, 2, \ldots)$, we solve

$$\boldsymbol{u}^*(t) = \arg\min_{\boldsymbol{u}(t)} \frac{1}{2}\boldsymbol{u}(t)^T H \boldsymbol{u}(t) + F^T \boldsymbol{u}(t) \tag{53}$$

$$\boldsymbol{u}(t) = \begin{bmatrix} u(t) \\ \delta_{\text{acc}}(t) \\ \nu_1(t) \\ \delta_1(t) \\ p_2(t) \end{bmatrix}, F = \begin{bmatrix} \frac{-2F_r(v(t))}{M^2} \\ 0 \\ W_1 \\ 0 \\ -2Qp_2^* \end{bmatrix}$$

$$H = \begin{bmatrix} \frac{2}{M^2} & 0 & 0 & 0 & 0 \\ 0 & 2p_{\text{acc}} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2P_1 & 0 \\ 0 & 0 & 0 & 0 & 2Q \end{bmatrix}$$

where $p_{\text{acc}} > 0, W_1 > 0, P_1 > 0, Q \geq 0$. We also assume F is a constant vector in each interval. The cost (53) is subject to (38), (41), (42), (44)–(47), and the control bounds in (39). The explicit form is omitted for simplicity, but can be found in [28]. After solving each QP, we update (38) with $u^*(t)$, update (44) with $\nu_1^*(t)$, and update $p_2(t)$ with $p_2^*(t), \forall t \in (k\Delta t, (k+1)\Delta t)$.

For the RACBF method, at $t = k\Delta t$ $(k = 0, 1, 2, \ldots)$, we solve

$$\boldsymbol{u}^*(t) = \arg\min_{\boldsymbol{u}(t)} \frac{1}{2}\boldsymbol{u}(t)^T H \boldsymbol{u}(t) + F^T \boldsymbol{u}(t) \tag{54}$$

$$\boldsymbol{u}(t) = \begin{bmatrix} u(t) \\ \delta_{\text{acc}}(t) \\ \nu(t) \\ \delta_r(t) \end{bmatrix}, F = \begin{bmatrix} \frac{-2F_r(v(t))}{M^2} \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$H = \begin{bmatrix} \frac{2}{M^2} & 0 & 0 & 0 \\ 0 & 2p_{\text{acc}} & 0 & 0 \\ 0 & 0 & 2P_r & 0 \\ 0 & 0 & 0 & 2P_r \end{bmatrix}$$

where $P_r > 0$, subject to (49), (41), (42), (50)–(52) and the control bounds in (39). After solving each QP, we update (49) with $u^*(t), \nu^*(t), \forall t \in (k\Delta t, (k+1)\Delta t)$.

VI. Implementation and Results

In this section, we show how our proposed aCBFs can address the conservativeness of standard CBFs [2], [25] and consider time-varying bounds. We consider the ACC problem (Problem 1) under noisy dynamics and time-varying control bounds (due to tire slipping, different road surfaces, etc.), and implement the aCBF approaches for solving Problem 1 in MATLAB to illustrate the adaptive properties described in Sections III and IV. We used quadprog to solve the QPs and ode45 to integrate the dynamics. All the computation was performed on a Intel(R) Core(TM) i7-8700 CPU @ 3.2 GHz×2.

The parameters are $v(0) = 20$ m/s, $z(0) = 100$ m, $v_p = 13.89$ m/s,$v_d = 24$ m/s, $M = 1650$ kg, $g = 9.81$ m/s^2, $f_0 = 0.1$ N, $f_1 = 5\text{N} \cdot \text{s/m}, f_2 = 0.25\text{N} \cdot \text{s/m}^2$, $l_p = 10$ m, $v_{\max} = 30$ m/s, $v_{\min} = 0$ m/s,$\Delta t = 0.1$ s, $\epsilon = 10$, $c_a(t) = 0.4$. If we apply the HOCBF approach to implement the safety constraint (9) with *fixed* $p_1(t) = 0.1, p_2(t) = 1$, and $c_d(t) = 0.4$, the QPs will be infeasible after the corresponding HOCBF constraint becomes active. Therefore, we need an aCBF to implement this safety constraint, as shown next.

A. Implementation With PACBFs

As motivated in (27), we first normalize each term in the cost (53) by dividing its maximum value: $u_{\text{lim}} = \frac{(c_a Mg)^2}{M^2} = (c_a g)^2, \nu_{1,\max} = \delta_{1,\max}^2 = \delta_{acc,\max}^2 = p_{1,\max}^2 = u_{\text{lim}}$, respectively, where $\delta_{\text{acc},\max}$ denotes the maximum value of δ_{acc}, and set $p_{\text{acc}} = c_0 = e^{-12}, W_1 = 2e^{-12}$, $P_1 = Q = 0.5$.

1) Adaptivity to the Changing Control Bound $-c_d(t)Mg$:

We first study what happens when we change the lower control bound $-c_d(t)Mg$. In each simulated trajectory, we set the lower

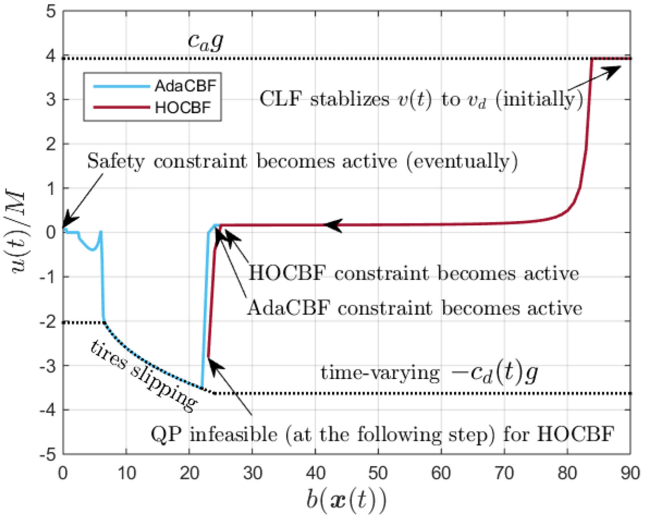


Fig. 1. Control input $u(t)$ variation as $b(\boldsymbol{x}(t)) \to 0$ for HOCBF and PACBF for linearly decreasing $c_d(t)$ (0.37 → 0.2) after the PACBF (or HOCBF) constraint (45) becomes active. The arrow denotes the changing trend for $b(\boldsymbol{x}(t))$ that captures the safety constraint (9) (distance between vehicles) with respect to time.

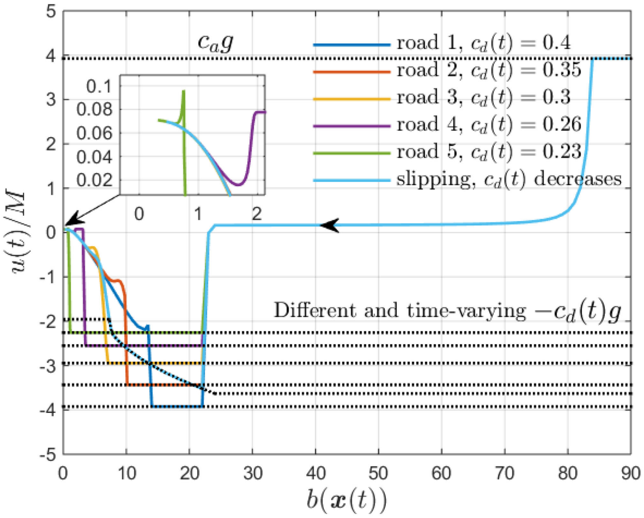


Fig. 2. Control input $u(t)$ variations as $b(\boldsymbol{x}(t)) \to 0$ under different and time-varying control lower bounds. The arrow denotes the changing trend for $b(\boldsymbol{x}(t))$ with respect to time. $b(\boldsymbol{x}) \geq 0$ implies the *forward invariance* of $C_1 := \{\boldsymbol{x} : b(\boldsymbol{x}) \geq 0\}$.

control bound coefficient $c_d(t)$ to a different constant or to be time-varying (e.g., linearly decreasing $c_d(t)$). In this case, we set $T = 30$ s, $p_1(0) = p_1^* = 0.1, p_2^* = 1$. We first present a case study of linearly decreasing $c_d(t)$ representing, for example, tires slipping, as shown in Fig. 1. When we decrease $c_d(t)$ (weaken the braking capability of the vehicle) after the HOCBF constraint becomes active, the QPs can easily become infeasible in the HOCBF method, as the red curve shows in Fig. 1: Reading the red curve from right to left as the value of $b(\boldsymbol{x})$ becomes small as time goes by, the speed is initially stabilized to v_d until the HOCBF constraint becomes active. It is important to note that at some point a QP becomes infeasible. However, using the PACBF method, the QPs are always feasible (blue curve in Fig. 1), demonstrating the adaptivity of the PACBF to the time-varying control bound (wheels slipping).

The computational time of the QP at each time step for both the HOCBF and PACBF methods is less than 0.01 s. Note that there is a control overshot when $b(\boldsymbol{x})$ is small; this can be alleviated by increasing the weight on the control $u(t)$ or by decreasing the weights P_1, Q after the control constraint becomes inactive, as seen by the blue curve in Fig. 2.

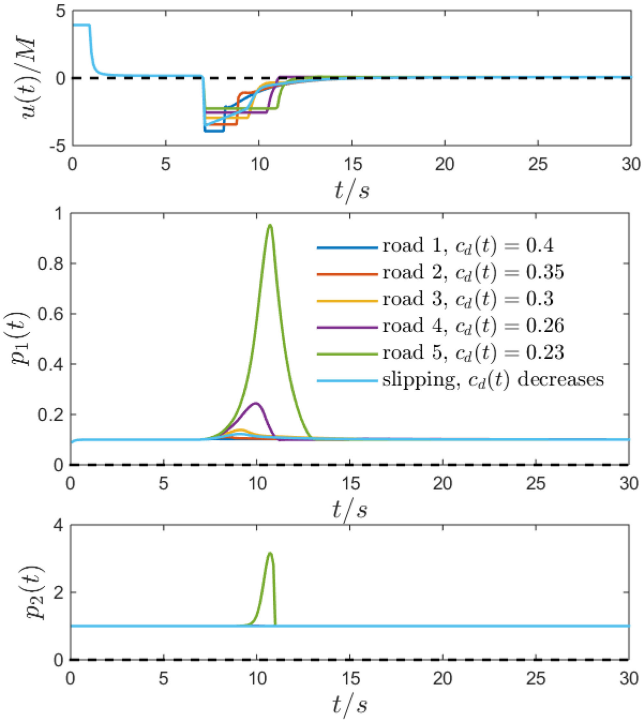


Fig. 3. Penalty functions $p_1(t), p_2(t)$, and control input $u(t)$ profiles under different and time-varying control lower bounds. The change in the values of the penalty functions $p_1(t), p_2(t)$ demonstrates the adaptivity of the PACBF to changes in the control bound (or tight control bound).

The simulated trajectories for different (constant) $c_d(t)$ values (e.g., as the vehicle encounters *different road surfaces*) are shown in Figs. 2 and 3. As shown in Fig. 2, when $c_d(t) = 0.4$, the QPs exhibit good feasibility. This induces only a small change in the penalty variable $p_1(t)$ and no change in $p_2(t)$, as shown in Fig. 3. As we decrease $c_d(t)$ (i.e., limit the braking capability of the vehicle), the variation in the penalty $p_1(t)$ becomes large after the PACBF constraint (45) becomes active. When $c_d(t) = 0.23$, the vehicle needs to brake with $u(t) = -c_d(t)Mg$ almost all the way to the safety constraint (9) becoming active, as the green curves show in Figs. 2 and 3. On the other hand, the penalty functions $p_1(t)$, $p_2(t)$ both change to a large value, as shown in Fig. 3. If we further decrease $c_d(t)$, the safety constraint (9) will be violated. The change in the values of the penalty functions $p_1(t), p_2(t)$ demonstrates the adaptivity of the PACBF to changes in the control bound. The penalty method [19] has shown that smaller penalties are needed to improve QP feasibility before the HOCBF constraint becomes active, but the PACBF shows that we may actually want to increase the value of the penalties after the PACBF constraint becomes active, as

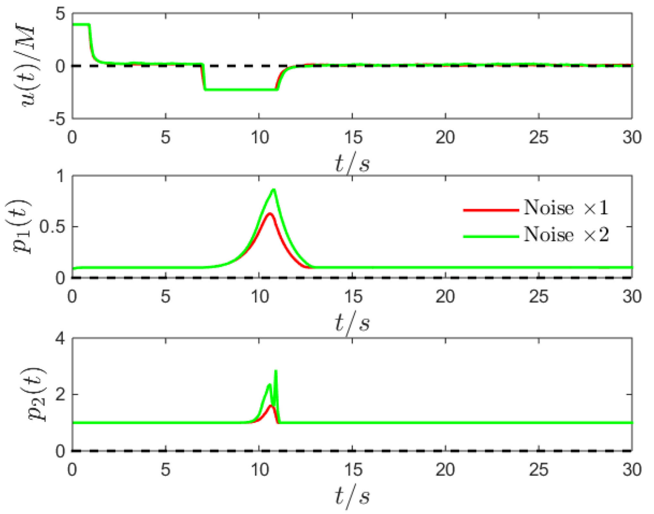


Fig. 4. Penalty functions $p_1(t), p_2(t)$ and control input $u(t)$ profiles under different noise levels. The change in the values of the penalty functions $p_1(t), p_2(t)$ demonstrates the adaptivity of the PACBF to the control bound and noise.

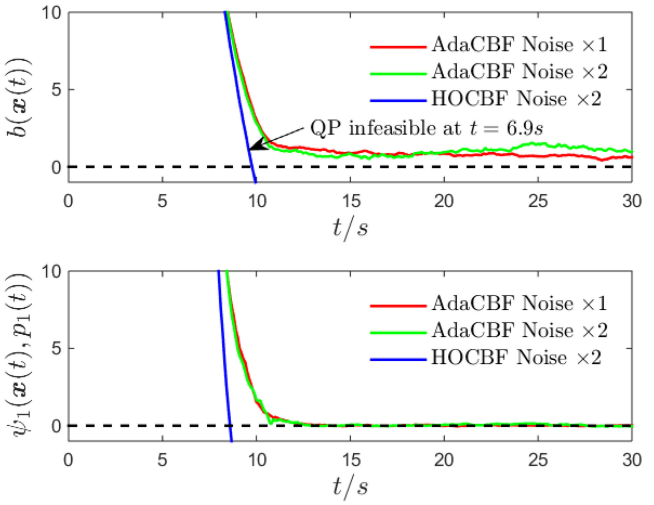


Fig. 5. Profiles of $b(\boldsymbol{x}), \psi_1(\boldsymbol{x}, p_1)$ under different noise levels for PACBF and HOCBF, $b(\boldsymbol{x}) \geq 0, \psi_1(\boldsymbol{x}, p_1) \geq 0$ imply the *forward invariance* of C_1 and C_2.

the last plot in Fig. 3 demonstrates (a similar phenomenon is observed in a different example in [28]).

2) Adaptivity to Noisy Dynamics: Suppose we add two noise terms $w_1(t), w_2(t)$ to the speed and acceleration in (38), respectively, where $w_1(t), w_2(t)$ denote two random processes defined in an appropriate probability space. In the simulated system, $w_1(t), w_2(t)$ randomly take values in $[-2\text{ m/s}, 2\text{ m/s}]$ and $[-0.45\text{ m/s}^2, 0.45\text{ m/s}^2]$ with equal probability at time t, respectively. We fix the value of $c_d(t)$ to 0.23 in (39) and set $T = 30$ s, $p_1(0) = p_1^* = 0.1, p_2^* = 1$. The simulation results under different noise levels are shown in Figs. 4 and 5, the noise is based on $[-2\text{ m/s}, 2\text{ m/s}]$ and $[-0.45\text{ m/s}^2, 0.45\text{ m/s}^2]$ for $w_1(t), w_2(t)$, respectively.

When the control constraint is active, i.e., $u(t) = -c_d(t)Mg$, it can easily conflict with the HOCBF constraint (if we apply the HOCBF method) that is subjected to noise, which may make the safety constraint (9) violated as the blue line shows in Fig. 5. However, the PACBF constraint is relaxed by the penalty functions $p_1(t)$ [through $\nu_1(t)$] and $p_2(t)$; hence, it is adaptive to different noise levels and can enable the feasibility of the QPs, as seen in Fig. 4.

The forward invariance of $C_1 := \{\boldsymbol{x} : b(\boldsymbol{x}) \geq 0\}$ and $C_2 = \{(\boldsymbol{x}, p_1) : \psi_1(\boldsymbol{x}, p_1) \geq 0\}$ is illustrated in Fig. 5. Note that $\psi_1(\boldsymbol{x}, p_1)$ might be temporarily negative due to noise during simulation, but will be positive again soon after. This is due to the definition of $\psi_2 := \dot{\psi}_1 + p_2(t)\psi_1$ in (43), in which $\alpha_2(\cdot)$ is defined as an extended class $\mathcal{K}$ function (linear function), as shown in [25]. When $\psi_1 < 0$, the PACBF constraint ensures $\dot{\psi}_1 + p_2(t)\psi_1 \geq 0$, thus, $\dot{\psi}_1 \geq -p_2(t)\psi_1 > 0$ since $p_2(t) > 0$. Therefore, ψ_1 will be increasing and eventually becomes positive. In this article, we have considered high-order polynomial class $\mathcal{K}$ functions to make ψ_i stay away from zero [19] such that $b(\boldsymbol{x}) \geq 0$ is guaranteed in the presence of noise. The forward invariance guarantee can also be achieved by considering the noise bounds in the PACBF constraint. Note that we can also define $\alpha_2(\cdot)$ as a quadratic function in the definition of the PACBF in (43) to make $\psi_1(\boldsymbol{x}, p_1)$ also stay away from 0 in Fig. 5, and define $\alpha_1(\cdot)$ as a high-order polynomial function to make the PACBF $b(\boldsymbol{x})$ stay further away to 0, so that it can be adaptive (in the sense of both QP feasibility and forward invariance) to higher noise levels.

3) Leveraging the Cost and the Stabilization: As shown in Fig. 2, we have to tune the weights P_1, Q to avoid the control overshooting when $b(\boldsymbol{x})$ is small, i.e., the constraint close to being violated. This is due to the huge relative value between c_0 and P_1, and the cost (21) is not actually minimized in a proper way. Here, we provide a simple approach to leverage the cost and the stabilization. We put a small weight on the relaxation δ_i in the CLF constraint (25), i.e., we replace (25) by

$$L_{F_i}V_i(\boldsymbol{\pi}_i) + L_{G_i}V_i(\boldsymbol{\pi}_i)\nu_i + \epsilon V_i(\boldsymbol{\pi}_i) \leq \eta\delta_i \tag{55}$$

where $\eta > 0$ is a small number. Thus, we are *limiting the relaxation* of stabilizing p_i to p_i^* in (55) through η.

Then, we use the CLF form in (55) to leverage the cost and the stabilization. We normalize each term in the cost (53) as above, and set $p_{\text{acc}} = c_0 = P_1 = e^{-5}, W_1 = 2e^{-5}, Q = 1$, and $\eta = e^{-5}$ with the CLF constraint (55). The simulation results are shown in Fig. 6. With the CLF constraint (55), we can significantly decrease the relative values of P_1, Q with respect to other weights while avoiding control overshooting.

B. Implementation With RACBFs

In this section, we study the adaptivity of RACBFs. We first normalize each term in the cost (54) by dividing its maximum value: $u_{\lim} = \nu_{\max}^2 = \delta_{r,\max}^2 = \delta_{acc,\max}^2 = (c_a g)^2$, respectively, where $\nu_{\max}$ denotes the maximum value of ν, and set $c_0 = p_{\text{acc}} = e^{-4}, P_r = 0.5, r_a = 1$ m, $\boldsymbol{r}(0) = (3, 0)$.

We also change the lower control bound $-c_d(t)Mg$ in this section. In each simulated trajectory, we set the lower control bound coefficient $c_d(t)$ to a different constant or to be time-varying (e.g., linearly decreasing $c_d(t)$). In this case, we set $T = 30$ s,

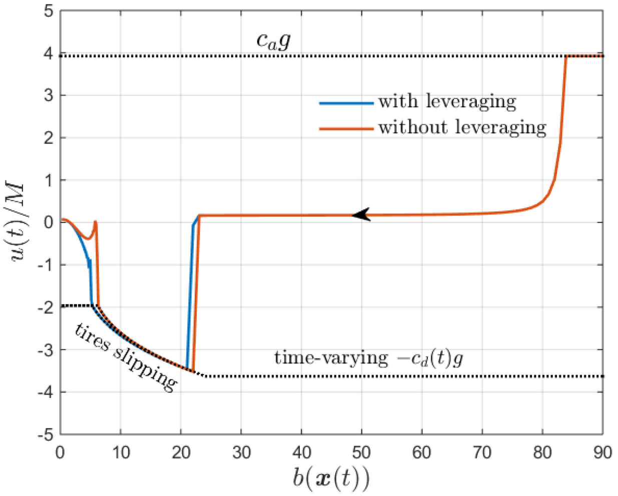


Fig. 6. Control input $u(t)$ variation as $b(\boldsymbol{x}(t)) \rightarrow 0$ for PACBF with or without leveraging for linearly decreasing $c_d(t)$ (0.37 → 0.2) after the PACBF constraint (45) becomes active.

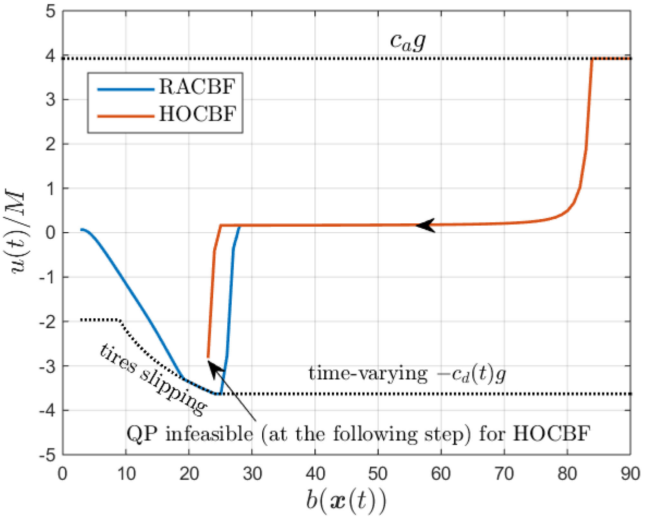


Fig. 7. Control input $u(t)$ variation as $b(\boldsymbol{x}(t)) \rightarrow 0$ for HOCBF and RACBF for linearly decreasing $c_d(t)$ (0.37 → 0.2) after the RACBF (or HOCBF) constraint (45) becomes active.

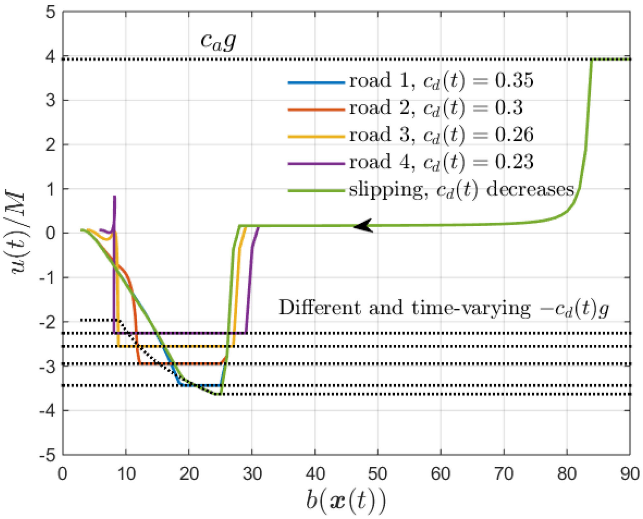


Fig. 8. Control input $u(t)$ variations as $b(\boldsymbol{x}(t)) \rightarrow 0$ under different and time-varying control lower bounds for RACBFs.

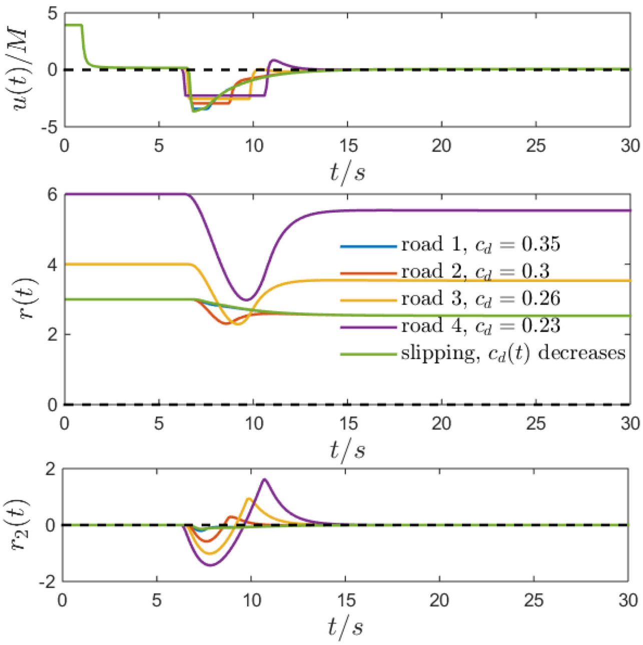


Fig. 9. Relaxation $r(t), r_2(t)$ and control input $u(t)$ profiles under different and time-varying control lower bounds. The change in the values of the relaxation $r(t), r_2(t)$ demonstrates the adaptivity of the RACBF to changes in the control bound (or tight control bound).

$p_1 = 0.1,\ p_2 = 1$. We first present a case study of linearly decreasing $c_d(t)$ representing, for example, tires slipping, as shown in Fig. 7. When we decrease $c_d(t)$ (weaken the braking capability of the vehicle) after the HOCBF constraint becomes active, the QPs can easily become infeasible in the HOCBF method, as the brown curve shows in Fig. 7. It is important to note that at some point a QP becomes infeasible. However, using the RACBF method, the QPs are always feasible (blue curve in Fig. 7), demonstrating the adaptivity of the RACBF to the time-varying control bound (wheels slipping).

The computational time at each time step for the RACBF method is less than 0.01 s in MATLAB (Intel(R) Core(TM) i7-8700 CPU @ 3.2 GHz×2). The simulated trajectories for different (constant) $c_d(t)$ values (e.g., as the vehicle encounters *different road surfaces*) are shown in Figs. 8 and 9. Similar to the PACBF, the RACBF can also be adaptive to the time-varying control bounds. The adaptivity of RACBF to noise is shown in the next section.

C. Comparison Between PACBF and RACBF

In this section, we compare the PACBF and RACBF methods. We first normalize the two costs (53) and (54) as in the previous two sections, and set $c_0 = p_{\text{acc}} = 0.005, P_r = 0.495, r_a = 1$ m for (54) and set $p_{\text{acc}} = c_0 = e^{-12}, W_1 = 2e^{-12}, P_1 = Q = 0.5$ for (53). The lower control bound coefficient is $c_d(t) = 0.23$. $p_1^* = 0.1, p_2^* = 1$ for the PACBF, and $k_1 = 0.1, k_2 = 1$ for the RACBF. In the simulated system, $w_1(t), w_2(t)$ randomly take

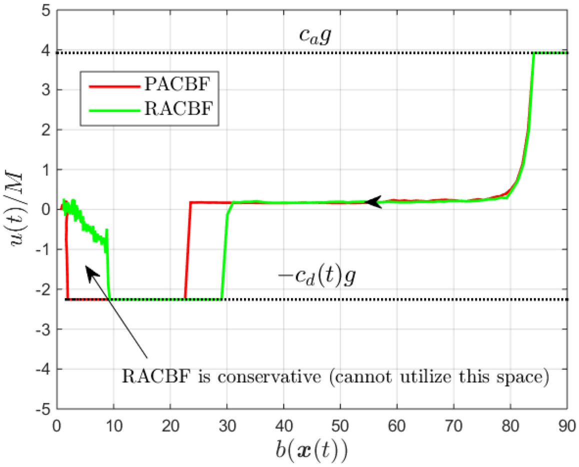


Fig. 10. Adaptivity comparison between the PACBF and RACBF methods.

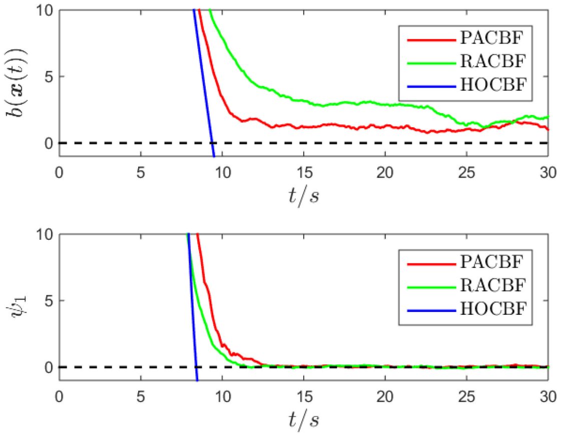


Fig. 11. Profiles of $b(\boldsymbol{x}), \psi_1(\boldsymbol{x}, p_1)$ under noise for PACBF, RACBF, and HOCBF, $b(\boldsymbol{x}) \geq 0$ implies the *forward invariance* of $C_1 = \{\boldsymbol{x} : b(\boldsymbol{x}) \geq 0\}$.

values in $[-4\text{ m/s}, 4\text{ m/s}]$ and $[-0.9\text{ m/s}^2, 0.9\text{ m/s}^2]$ with equal probability at time t, respectively. The simulation results are shown in Figs. 10 and 11.

It follows from Fig. 10 that the RACBF method is still conservative such that it cannot utilize the space marked in the figure to obtain the adaptivity. This is due to the fact that the RACBF method obtains adaptivity by the relaxation $r(t)$, and this will make the RACBF constraint become active earlier than the PACBF method, as shown in Fig. 10. The PACBF is not conservative and, thus, is more adaptive than the RACBF. However, the complexity of a PACBF increases faster than the one of an RACBF as the relative degree m of the constraint becomes larger. It follows from Fig. 11 that the satisfaction of $b(\boldsymbol{x}) \geq 0$ under noise is guaranteed with both the PACBF and RACBF methods. The QPs are always feasible.

VII. Conclusion

We introduced aCBFs (AdaCBFs) that can accommodate time-varying control bounds and noisy dynamics while also addressing the QP feasibility problem, which arises when using standard CBFs. The proposed AdaCBFs can also alleviate the conservativeness of the CBF method such that the satisfaction of the AdaCBF constraint is a necessary and sufficient condition for the satisfaction of the original constraint, and, therefore, improve system performance. We have also shown that AdaCBFs inherit the forward invariance properties of standard CBFs. The advantages of AdaCBFs over the existing CBF techniques are demonstrated by applying them to an ACC problem. In the future, we will apply the AdaCBF method to more complex problems and systems.

Acknowledgment

The statements made herein are solely the responsibility of the authors.

References

[1] K. P. Tee, S. S. Ge, and E. H. Tay, “Barrier Lyapunov functions for the control of output-constrained nonlinear systems,” *Automatica*, vol. 45, no. 4, pp. 918–927, 2009.

[2] P. Wieland and F. Allgower, “Constructive safety using control barrier functions,” in *7th IFAC Symposium on Nonlinear Control Systems.*, vol. 40, no. 12, pp. 462–467, 2007.

[3] S. P. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge Univ. Press, 2004.

[4] J. P. Aubin, *Viability Theory*. Berlin, Germany: Springer, 2009.

[5] S. Prajna, A. Jadbabaie, and G. J. Pappas, “A framework for worst-case and stochastic safety verification using barrier certificates,” *IEEE Trans. Autom. Control*, vol. 52, no. 8, pp. 1415–1428, Aug. 2007.

[6] R. Wisniewski and C. Sloth, “Converse barrier certificate theorem,” in *Proc. 52nd IEEE Conf. Decis. Control*, 2013, pp. 4713–4718.

[7] D. Panagou, D. M. Stipanovic, and P. G. Voulgaris, “Multi-objective control for multi-agent systems using Lyapunov-like barrier functions,” in *Proc. 52nd IEEE Conf. Decis. Control*, Dec. 2013, pp. 1478–1483.

[8] A. D. Ames, J. W. Grizzle, and P. Tabuada, “Control barrier function based quadratic programs with application to adaptive cruise control,” in *Proc. 53rd IEEE Conf. Decis. Control*, 2014, pp. 6271–6278.

[9] P. Glotfelter, J. Cortes, and M. Egerstedt, “Nonsmooth barrier functions with applications to multi-robot systems,” *IEEE Control Syst. Lett.*, vol. 1, no. 2, pp. 310–315, Oct. 2017.

[10] L. Lindemann and D. V. Dimarogonas, “Control barrier functions for signal temporal logic tasks,” *IEEE Control Syst. Lett.*, vol. 3, no. 1, pp. 96–101, Jan. 2019.

[11] E. Sontag, “A Lyapunov-like stabilization of asymptotic controllability,” *SIAM J. Control Optim.*, vol. 21, no. 3, pp. 462–471, 1983.

[12] Z. Artstein, “Stabilization with relaxed controls,” *Nonlinear Anal.: Theory, Methods, Appl.*, vol. 7, no. 11, pp. 1163–1173, 1983.

[13] R. A. Freeman and P. V. Kokotovic, *Robust Nonlinear Control Design*. Cambridge, MA, USA: Birkhauser, 1996.

[14] A. D. Ames, K. Galloway, and J. W. Grizzle, “Control Lyapunov functions and hybrid zero dynamics,” in *Proc. 51st IEEE Conf. Decis. Control*, 2012, pp. 6837–6842.

[15] K. Galloway, K.Sreenath, A. D. Ames, and J. Grizzle, “Torque saturation in bipedal robotic walking through control Lyapunov function based quadratic programs,” *IEEE Access*, vol. 3, pp. 323–332, 2015.

[16] S. C. Hsu, X. Xu, and A. D. Ames, “Control barrier function based quadratic programs with application to bipedal robotic walking,” in *Proc. Amer. Control Conf.*, 2015, pp. 4542–4548.

[17] G. Wu and K.Sreenath, “Safety-critical and constrained geometric control synthesis using control Lyapunov and control barrier functions for systems evolving on manifolds,” in *Proc. Amer. Control Conf.*, 2015, pp. 2038–2044.

[18] Q. Nguyen and K.Sreenath, “Exponential control barrier functions for enforcing high relative-degree safety-critical constraints,” in *Proc. Amer. Control Conf.*, 2016, pp. 322–328.

[19] W. Xiao and C. Belta, “Control barrier functions for systems with high relative degree,” in *Proc. 58th IEEE Conf. Decis. Control*, 2019, pp. 474–479.

[20] A. J. Taylor and A. D. Ames, “Adaptive safety with control barrier functions,” in *Proc. Amer. Control Conf.*, 2020, pp. 1399–1405.

[21] B. T. Lopez, J. J. E. Slotine, and J. P. How, “Robust adaptive control barrier functions: An adaptive and data-driven approach to safety,” *IEEE Control Syst. Lett.*, vol. 5, no. 3, pp. 1031–1036, Jul. 2021.

[22] D. D. Fan, J. Nguyen, R. Thakker, N. Alatur, A. A. Agha-mohammadi, and E. A. Theodorou, “Bayesian learning-based adaptive control for safety critical systems,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 4093–4099.

[23] M. J. Khojasteh, V. Dhiman, M. Franceschetti, and N. Atanasov, “Probabilistic safety constraints for learned high relative degree system dynamics,” in *Proc. Learn. Dyn. Control*, 2020, pp. 781–792.

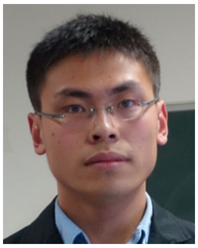
[24] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 2002.

[25] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, “Control barrier function based quadratic programs for safety critical systems,” *IEEE Trans. Autom. Control*, vol. 62, no. 8, pp. 3861–3876, Aug. 2017.

[26] G. Yang, C. Belta, and R. Tron, “Self-triggered control for safety critical systems using control barrier functions,” in *Proc. Amer. Control Conf.*, 2019, pp. 4454–4459.

[27] W. Xiao, C. Belta, and C. G. Cassandras, “Feasibility guided learning for constrained optimal control problems,” in *Proc. 59th IEEE Conf. Decis. Control*, 2020, pp. 1896–1901.

[28] W. Xiao, C. Belta, and C. G. Cassandras, “Adaptive control barrier functions for safety-critical systems,” 2020, *arXiv:2002.04577*.



Wei Xiao (Student Member, IEEE) received the B.Sc. degree from the University of Science and Technology, Beijing, China, in 2013, and the M.Sc. degree from the Institute of Automation, Chinese Academy of Sciences, Beijing, in 2016. He is currently working toward the Ph.D. degree in systems engineering with Boston University, Brookline, MA, USA.

His research interests include Lyapunov methods, machine learning, and formal methods, with particular emphasis on robotics and autonomous driving.

Mr. Xiao was a recipient of an Outstanding Student Paper Award at the 2020 IEEE Conference on Decision and Control.

Calin Belta (Fellow, IEEE) received the B.Sc. and M.Sc. degrees from the Technical University of Iasi, Iasi, Romania, in 1995 and 1997, respectively, and the M.Sc. and Ph.D. degrees from the University of Pennsylvania, Philadelphia, PA, USA, in 2001 and 2003.

He is currently a Professor with the Department of Mechanical Engineering, Boston University (BU), Brookline, MA, USA, where he holds the Tegan Family Distinguished Faculty Fellowship. He is also the Director of BU Robotics Lab, Brookline, and of the Center for Autonomous and Robotic Systems, Boston University, MA, United States. His research interests include dynamics and control theory, with particular emphasis on hybrid and cyber-physical systems, formal synthesis and verification, and applications in robotics and systems biology.

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

Christos G. Cassandras (Life Fellow, IEEE) received the B.S. degree from Yale University, New Haven, CT, in 1977, the M.S.E.E. degree from Stanford University, Stanford, CA, in 1978 and the S.M. and Ph.D. degrees from Harvard University, Cambridge, MA, in 1979 and 1982, respectively.

He is currently a Distinguished Professor of Engineering with Boston University, Brookline, MA, USA. He is also the Head of the Division of Systems Engineering, a Professor of Electrical and Computer Engineering, and a co-founder of the Center for Information and Systems Engineering, Boston University. From 1982 to 1984, he was with ITP Boston, Inc., Cambridge, where he worked on the design of automated manufacturing systems. From 1984 to 1996, he was a Faculty Member with the Department of Electrical and Computer Engineering, University of Massachusetts Amherst, MA, USA. In addition to his academic activities, he has worked extensively with industrial organizations on various systems integration projects and the development of decision support software. He has most recently collaborated with The MathWorks, Inc., in the development of the discrete event and hybrid system simulator SimEvents. He has authored/coauthored 6 books, and more than 450 refereed papers in the areas of discrete event and hybrid systems, cooperative control, stochastic optimization, and computer simulation, with applications to computer and sensor networks, manufacturing systems, and transportation systems.

Dr. Cassandras has guest-edited several technical journal issues and currently serves on several journal editorial boards, including Editor of *Automatica*. He was the Editor-in-Chief for the IEEE TRANSACTIONS ON AUTOMATIC CONTROL from 1998 to 2009 and was also the Editor for Technical Notes and Correspondence and an Associate Editor. He was the 2012 President of the IEEE Control Systems Society (CSS). He was also the Vice-President for publications and on the Board of Governors of the CSS, as well as on several IEEE committees, and has chaired several conferences. He has been a plenary/keynote speaker at numerous international conferences, including the 2017 IFAC World Congress, the American Control Conference in 2001, and the IEEE Conference on Decision and Control in 2002 and 2016, and has also been an IEEE Distinguished Lecturer. He is the recipient of several awards, including the 2011 IEEE Control Systems Technology Award, the 2006 Distinguished Member Award of the IEEE Control Systems Society, the 1999 Harold Chestnut Prize (IFAC Best Control Engineering Textbook) for *Discrete Event Systems: Modeling and Performance Analysis*, a 2011 Prize and a 2014 Prize for the IBM/IEEE Smarter Planet Challenge competition, the 2014 Engineering Distinguished Scholar Award at Boston University, several honorary professorships, a 1991 Lilly Fellowship, and a 2012 Kern Fellowship. He is a member of Phi Beta Kappa and Tau Beta Pi. He is also a Fellow of the IFAC.