2026 American Control Conference (ACC)
May 27-29, 2026. New Orleans, Louisiana, USA

Lag-Compensating Control Barrier Functions for Feasible Safety-critical Control

Yuchen Chen, Tamas G. Molnar, and Gábor Orosz

***Abstract*— First-order lags are commonly used in control system design to approximate system delays, which can significantly compromise system safety. To maintain safety under lags, this paper introduces the concept of *lag-compensating control barrier function* (LCCBF). First, we propose a lag-eliminating state transformation that removes first-order lag dynamics from linear systems. Then, we use this transformation to construct an LCCBF from the CBF designed for the lag-free system. We prove that controllers synthesized via the LCCBF ensure safety under actuator lags, while preserving feasibility for the same input bounds that are used to synthesize controllers via the original CBF. Finally, we simulate a double integrator with actuator limits to validate that the LCCBF maintains both safety and feasibility under lags.**

I. Introduction

Ensuring safety is a fundamental requirement when controlling dynamical systems in safety-critical environments. In recent years, the development of safety-critical control has received significant attention. Methods such as model predictive control [1], reachability analysis [2] and control barrier functions (CBFs) [3] have been leveraged to develop safety-critical controllers. Among these methods, CBFs have gained wide popularity because of their compatibility with existing control frameworks and computational efficiency, enabling real-world deployment. CBFs have been successfully applied in diverse domains, including connected automated vehicles [4], [5], bipeds [6], and multi-robot systems [7].

First-order lags are often used in control design to capture delayed responses from actuators [8]. However, such lags can degrade system safety, as late control actions may not be sufficient to enforce safety constraints. To maintain safety for lagged systems, the CBF designed for the lag-free system needs to be modified. This can be achieved, for example, using high-order CBFs (HOCBFs) or backstepping CBFs (BSCBFs) [9], [10], [11], [12]. However, HOCBFs may be invalid and fail to guarantee safety [13], [14], while BSCBFs often require nontrivial design processes and may lead to overly conservative behavior [12], [15]. Moreover, even if the original safety-critical controller satisfies actuator limits, neither the HOCBF nor the BSCBF design guarantees feasibility under the same actuator limits for lagged systems [16].

Yuchen Chen and Gábor Orosz are with the Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI 48109, USA, ethanch@umich.edu, orosz@umich.edu.

Tamas Molnar is with the Department of Mechanical Engineering, Wichita State University, Wichita, KS 67260, USA, tamas.molnar@wichita.edu.

Gábor Orosz is also with the Department of Civil and Environmental Engineering, University of Michigan, Ann Arbor, MI 48109, USA.

To address these limitations, inspired by delay-compensating CBFs [17], we propose the concept of *lag-compensating control barrier function* (LCCBF). Specifically, we introduce a lag-eliminating state transformation for linear systems with first-order input lags and derive the LCCBF by applying this transformation to the original CBF defined for the lag-free system. We prove that the LCCBF inherits the safety guarantees of the original CBF, thereby ensuring safety for the lagged system. We also show that the LCCBF preserves the feasibility of the original CBF, enabling the synthesis of safety-critical controllers that remain feasible under the same input bounds. A double integrator example illustrates how the proposed approach maintains safety while guaranteeing feasibility.

The rest of the paper is organized as follows. Section II reviews the CBF framework. Section III introduces the lag-eliminating state transformation. Section IV proposes the novel LCCBF that guarantees safety and preserves feasibility under input lag. Section V validates the LCCBF on a double integrator example. Section VI concludes the results.

II. Background on control barrier functions

In this section, we review the background on control barrier functions, which forms the basis for constructing LCCBFs. Consider the control-affine system

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

with state $\mathbf{x} \in \mathbb{R}^n$, input $\mathbf{u} \in \mathbb{R}^m$, and locally Lipschitz continuous functions $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$ and $\mathbf{g} : \mathbb{R}^n \to \mathbb{R}^{n \times m}$. For a locally Lipschitz continuous controller $\mathbf{k} : \mathbb{R}^n \to \mathbb{R}^m$ with $\mathbf{u} = \mathbf{k}(\mathbf{x})$, we obtain the closed-loop system

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

To characterize safety, we define the *safe set*

$$\mathcal{S} = \{\mathbf{x} \in \mathbb{R}^n : h(\mathbf{x}) \geq 0\}, \tag{3}$$

where $h : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable function. System (2) is considered safe w.r.t. $\mathcal{S}$ if $\mathcal{S}$ is forward invariant, i.e., every system state trajectory starting from $\mathcal{S}$ always stays within $\mathcal{S}$. To guarantee safety, CBFs [3] were proposed to synthesize safety-critical controllers for (1).

Definition 1 **([3]).** Function h is a *control barrier function* for (1) on $\mathcal{S}$ if there exists $\alpha \in \mathcal{K}^{\mathrm{e}}$ such that for all $\mathbf{x} \in \mathcal{S}$:

$$\sup_{\mathbf{u} \in \mathbb{R}^m} \dot{h}(\mathbf{x}, \mathbf{u}) > -\alpha\big(h(\mathbf{x})\big), \tag{4}$$

where

$$\dot{h}(\mathbf{x},\mathbf{u}) = \underbrace{\nabla h(\mathbf{x})\mathbf{f}(\mathbf{x})}_{L_{\mathbf{f}}h(\mathbf{x})} + \underbrace{\nabla h(\mathbf{x})\mathbf{g}(\mathbf{x})}_{L_{\mathbf{g}}h(\mathbf{x})}\mathbf{u}, \tag{5}$$

and $L_{\mathbf{f}}h(\mathbf{x})$ and $L_{\mathbf{g}}h(\mathbf{x})$ denote the Lie derivatives.

Remark 1. A continuous function $\alpha : \mathbb{R} \to \mathbb{R}$ belongs to the extended class-$\mathcal{K}$ ($\alpha \in \mathcal{K}^{\mathrm{e}}$) if it is strictly increasing and satisfies $\alpha(0) = 0$. A common choice of such a function is a linear function such as $\alpha(r) = \gamma r$ with $\gamma > 0$.

Lemma 1 ([18]). A continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$ is a valid CBF for (1) on $\mathcal{S}$ if and only if

$$L_{\mathbf{g}}h(\mathbf{x}) = \mathbf{0} \implies L_{\mathbf{f}}h(\mathbf{x}) > -\alpha\big(h(\mathbf{x})\big), \quad \forall \mathbf{x} \in \mathcal{S}. \tag{6}$$

Theorem 1 ([3]). *If h is a CBF for (1) on $\mathcal{S}$, then any locally Lipschitz continuous controller $\mathbf{k}(\mathbf{x})$ that satisfies*

$$\dot{h}\big(\mathbf{x},\mathbf{k}(\mathbf{x})\big) \geq -\alpha\big(h(\mathbf{x})\big), \tag{7}$$

for all $\mathbf{x} \in \mathcal{S}$ renders (2) safe w.r.t. $\mathcal{S}$.

Condition (7) can be incorporated as a constraint in optimization problems to synthesize safety-critical controllers. One can design a so-called safety filter that minimally alters a desired (but not inherently safe) controller $\mathbf{k}_{\mathrm{d}} : \mathbb{R}^n \to \mathbb{R}^m$ to a safe one by solving the quadratic program

$$\begin{aligned} \mathbf{k}(\mathbf{x}) = \underset{\mathbf{u}\in\mathbb{R}^m}{\operatorname{argmin}} \quad & \|\mathbf{u} - \mathbf{k}_{\mathrm{d}}(\mathbf{x})\|^2 \\ \text{s.t.} \quad & \dot{h}(\mathbf{x},\mathbf{u}) \geq -\alpha\big(h(\mathbf{x})\big). \end{aligned} \tag{8}$$

Remark 2 ([18]). The strict inequality in (4), (6) ensures Lipschitz continuity for CBF-based controllers when $L_{\mathbf{g}}h(\mathbf{x}) = \mathbf{0}$. If the controller is already Lipschitz continuous, (4), (6) can be relaxed to non-strict inequality as in (7).

The analytical solution of (8) was given in [19] as

$$\begin{aligned} \mathbf{k}(\mathbf{x}) &= \begin{cases} \mathbf{k}_{\mathrm{d}}(\mathbf{x}) + \max\{0, \zeta(\mathbf{x})\} \frac{L_{\mathbf{g}}h(\mathbf{x})^\top}{\|L_{\mathbf{g}}h(\mathbf{x})\|^2}, & \text{if } L_{\mathbf{g}}h(\mathbf{x}) \neq \mathbf{0}, \\ \mathbf{k}_{\mathrm{d}}(\mathbf{x}), & \text{if } L_{\mathbf{g}}h(\mathbf{x}) = \mathbf{0}, \end{cases} \\ \zeta(\mathbf{x}) &= -L_{\mathbf{f}}h(\mathbf{x}) - L_{\mathbf{g}}h(\mathbf{x})\mathbf{k}_{\mathrm{d}}(\mathbf{x}) - \alpha\big(h(\mathbf{x})\big). \end{aligned} \tag{9}$$

Note that if $L_{\mathbf{g}}h(\mathbf{x}) \equiv \mathbf{0}$ for all $\mathbf{x} \in \mathcal{S}$, which will be the case when a CBF designed for a lag-free system is directly applied to a lagged system, the resulting controller satisfies $\mathbf{k}(\mathbf{x}) = \mathbf{k}_{\mathrm{d}}(\mathbf{x})$, cf. (9). In this case, h is no longer a valid CBF. To resolve this, one popular approach is to construct a high-order CBF (HOCBF) [9]:

$$h_{\mathrm{e}}(\mathbf{x}) = L_{\mathbf{f}}h(\mathbf{x}) + \alpha\big(h(\mathbf{x})\big), \tag{10}$$

such that $L_{\mathbf{g}}h_{\mathrm{e}}(\mathbf{x}) \not\equiv \mathbf{0}$ with the *extended safe set*

$$\mathcal{S}_{\mathrm{e}} = \{\mathbf{x} \in \mathbb{R}^n : h_{\mathrm{e}}(\mathbf{x}) \geq 0\}. \tag{11}$$

Corollary 1 ([9]). *If $L_{\mathbf{g}}h(\mathbf{x}) \equiv \mathbf{0}$ and h_{e} in (10) is a CBF for (1) on $\mathcal{S}_{\mathrm{e}}$ with $\alpha_{\mathrm{e}} \in \mathcal{K}^{\mathrm{e}}$, then any locally Lipschitz continuous controller $\mathbf{k}(\mathbf{x})$ that satisfies:*

$$\dot{h}_{\mathrm{e}}\big(\mathbf{x},\mathbf{k}(\mathbf{x})\big) \geq -\alpha_{\mathrm{e}}\big(h_{\mathrm{e}}(\mathbf{x})\big), \tag{12}$$

for all $\mathbf{x} \in \mathcal{S}_{\mathrm{e}}$ renders (2) safe w.r.t. $\mathcal{S} \cap \mathcal{S}_{\mathrm{e}}$.

However, establishing that the derived h_{e} is a valid CBF can be challenging [13]. Moreover, under actuator limits, safe controllers synthesized from h_{e} may be infeasible even when those generated by h are feasible [16]. To address these for systems with first-order input lag, we introduce the LCCBF. We prove its validity and the feasibility of its safety-critical controllers, provided that the original CBF for the lag-free system is valid and its corresponding controller is feasible. The HOCBF is used as a baseline for comparison.

III. Lag-eliminating state transformation

In this section, we consider a linear time-invariant (LTI) system subject to first-order lag input dynamics. We establish that such systems can be equivalently represented as lag-free LTI systems through an appropriate state transformation.

First, we introduce a lag-free LTI system:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \tag{13}$$

with state $\mathbf{x} \in \mathbb{R}^n$, input $\mathbf{u} \in \mathbb{R}^m$, and dynamics governed by the nonzero matrices $\mathbf{A} \in \mathbb{R}^{n\times n}$ and $\mathbf{B} \in \mathbb{R}^{n\times m}$.

Next, we augment (13) with a first-order input lag $\xi > 0$:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{w}, \\ \dot{\mathbf{w}} &= \tfrac{1}{\xi}(\mathbf{u} - \mathbf{w}). \end{aligned} \tag{14}$$

Here $\mathbf{w} \in \mathbb{R}^m$ denotes the lagged input, which is also treated as an additional state governed by the first-order lag dynamics, i.e., the second equation in (14). Then, we define the lag-eliminating state transformation $T : \mathbb{R}^{n+m} \to \mathbb{R}^n$ as

$$\hat{\mathbf{x}} \; = \; T(\mathbf{x},\mathbf{w}) = \mathbf{x} + \xi\dot{\mathbf{x}} \; = \; \mathbf{x} + \xi(\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{w}). \tag{15}$$

Note that $\hat{\mathbf{x}}$ can be viewed as a prediction of the future state, similar to predictors for time delay systems [17], [20].

Theorem 2. *For the first-order lag system (14), the transformed state $\hat{\mathbf{x}}$ in (15) is governed by the lag-free system*

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}\hat{\mathbf{x}} + \mathbf{B}\mathbf{u}. \tag{16}$$

Proof. Differentiating (15) and substituting (14) yields

$$\begin{aligned} \dot{\hat{\mathbf{x}}} &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{w} + \xi\bigg(\mathbf{A}(\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{w}) + \mathbf{B}\frac{1}{\xi}(\mathbf{u} - \mathbf{w})\bigg) \\ &= \mathbf{A}\big(\mathbf{x} + \xi(\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{w})\big) + \mathbf{B}\mathbf{u} = \mathbf{A}\hat{\mathbf{x}} + \mathbf{B}\mathbf{u}, \end{aligned} \tag{17}$$

which gives the lag-free system (16). □

IV. Lag-compensating control barrier functions

In this section, we establish the concept of lag-compensating safety-critical control based on the lag-eliminating state transformation above. This allows us to construct a lag-compensating control barrier function (LCCBF) from a CBF defined for the lag-free system. We prove that the LCCBF is a valid CBF for the first-order lag system. Moreover, under a directional concavity condition on the original CBF, the LCCBF guarantees safety on a subset of the original safe set. Finally, we show that safe controllers synthesized with the LCCBF preserve the feasibility of those derived from the original CBF, thereby facilitating feasible and lag-aware safety-critical control.

First, we express the lag-free LTI system (13) in the control affine form (1) with $\mathbf{f}(\mathbf{x}) = \mathbf{A}\mathbf{x}$ and $\mathbf{g}(\mathbf{x}) = \mathbf{B}$. Let $h : \mathbb{R}^n \to \mathbb{R}$ be a CBF for (13) with safe set $\mathcal{S}$ in (3). By Lemma 1, it follows that for all $\mathbf{x} \in \mathcal{S}$, we have

$$\begin{aligned} L_{\mathbf{g}}h(\mathbf{x}) =& \nabla h(\mathbf{x})\mathbf{B} = \mathbf{0} \implies \\ & L_{\mathbf{f}}h(\mathbf{x}) = \nabla h(\mathbf{x})\mathbf{A}\mathbf{x} > -\alpha\big(h(\mathbf{x})\big). \end{aligned} \tag{18}$$

This guarantees the existence of a controller $\mathbf{k}$ satisfying (7), which ensures safety w.r.t. $\mathcal{S}$ for (13) based on Theorem 1.

Next, we write (14) in the control affine form as

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\mathbf{w}} \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{w} \\ -\frac{1}{\xi}\mathbf{w} \end{bmatrix}}_{\mathbf{F}(\mathbf{x},\mathbf{w})} + \underbrace{\begin{bmatrix} \mathbf{0} \\ \frac{1}{\xi}\mathbf{I} \end{bmatrix}}_{\mathbf{G}(\mathbf{x},\mathbf{w})} \mathbf{u}, \tag{19}$$

where $\mathbf{0}$ and $\mathbf{I}$ are zero and identity matrices of appropriate size. We lift h into the state space of the first-order lag system (14), i.e., $(\mathbf{x}, \mathbf{w})$, and define the lifted function $\tilde{h} : \mathbb{R}^{n+m} \to \mathbb{R}$, $\tilde{h}(\mathbf{x}, \mathbf{w}) \triangleq h(\mathbf{x})$, with the set $\tilde{\mathcal{S}} = \{(\mathbf{x}, \mathbf{w}) \in \mathbb{R}^{n+m} : \tilde{h}(\mathbf{x}, \mathbf{w}) \geq 0\} = \mathcal{S} \times \mathbb{R}^m$. This gives

$$L_{\mathbf{G}}\tilde{h}(\mathbf{x}, \mathbf{w}) = \mathbf{0}, \quad L_{\mathbf{F}}\tilde{h}(\mathbf{x}, \mathbf{w}) = \nabla h(\mathbf{x})(\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{w}). \tag{20}$$

Because $L_{\mathbf{F}}\tilde{h}$ differs from $L_{\mathbf{f}}h$ in (18), $\tilde{h}$ may violate Lemma 1, rendering it invalid as a CBF.

To address this, we introduce the lag-compensating CBF (LCCBF) $h_{\rm lc} : \mathbb{R}^{n+m} \to \mathbb{R}$ by applying the lag-eliminating transformation (15) to the original CBF h.

Definition 2. Given a CBF h for the lag-free system (13) and the transformation (15), the function

$$h_{\rm lc}(\mathbf{x}, \mathbf{w}) \triangleq h\big(T(\mathbf{x}, \mathbf{w})\big) = h(\hat{\mathbf{x}}), \tag{21}$$

is a *lag-compensating CBF* candidate for the lagged system (14) on the set $\mathcal{S}_{\rm lc} = \{(\mathbf{x}, \mathbf{w}) \in \mathbb{R}^{n+m} : h_{\rm lc}(\mathbf{x}, \mathbf{w}) \geq 0\}$.

Given a lag-free safety-critical controller $\mathbf{k}$ satisfying (7), we define the lag-compensating safety-critical controller

$$\mathbf{k}_{\rm lc}(\mathbf{x}, \mathbf{w}) \triangleq \mathbf{k}\big(T(\mathbf{x}, \mathbf{w})\big) = \mathbf{k}(\hat{\mathbf{x}}). \tag{22}$$

The following theorem shows that $h_{\rm lc}$ is a valid CBF, and $\mathbf{k}_{\rm lc}$ enforces the lag-compensating safety constraint

$$\dot{h}_{\rm lc}\big(\mathbf{x}, \mathbf{w}, \mathbf{k}_{\rm lc}(\mathbf{x}, \mathbf{w})\big) \geq -\alpha\big(h_{\rm lc}(\mathbf{x}, \mathbf{w})\big), \tag{23}$$

for all $(\mathbf{x}, \mathbf{w}) \in \mathcal{S}_{\rm lc}$, thereby ensuring safety of the lagged system (14) w.r.t. $\mathcal{S}_{\rm lc}$ according to Theorem 1.

Theorem 3. *If h is a CBF for (13) on $\mathcal{S}$, then $h_{\rm lc}$ in (21) is a CBF for (14) on $\mathcal{S}_{\rm lc}$. Furthermore, if $\mathbf{k}$ satisfies (7) for all $\mathbf{x} \in \mathcal{S}$, $\mathbf{k}_{\rm lc}$ in (22) ensures the safety of (14) w.r.t. $\mathcal{S}_{\rm lc}$.*

Proof. We first show that $h_{\rm lc}$ is a valid CBF for (14) on $\mathcal{S}_{\rm lc}$ by verifying Lemma 1. Since h is a valid CBF for (13), and $\hat{\mathbf{x}}$ follows the same lag-free dynamics as in (16), then for all $\hat{\mathbf{x}} \in \mathcal{S}$ we have

$$\begin{aligned} L_{\mathbf{g}}h(\hat{\mathbf{x}}) =& \nabla h(\hat{\mathbf{x}})\mathbf{B} = \mathbf{0} \implies \\ & L_{\mathbf{f}}h(\hat{\mathbf{x}}) = \nabla h(\hat{\mathbf{x}})\mathbf{A}\hat{\mathbf{x}} > -\alpha\big(h(\hat{\mathbf{x}})\big), \end{aligned} \tag{24}$$

cf. (18). Note that $\hat{\mathbf{x}} \in \mathcal{S} \iff (\mathbf{x}, \mathbf{w}) \in \mathcal{S}_{\rm lc}$. From (15), (21), we calculate the gradient of $h_{\rm lc}$ using the chain rule:

$$\nabla h_{\rm lc}(\mathbf{x}, \mathbf{w}) = \begin{bmatrix} \nabla h(\hat{\mathbf{x}})(\mathbf{I} + \xi\mathbf{A}) & \nabla h(\hat{\mathbf{x}})\xi\mathbf{B} \end{bmatrix}. \tag{25}$$

Then we calculate the Lie derivatives of $h_{\rm lc}$ w.r.t. (19):

$$\begin{aligned} L_{\mathbf{F}}h_{\rm lc}(\mathbf{x}, \mathbf{w}) &= \nabla h_{\rm lc}(\mathbf{x}, \mathbf{w})\mathbf{F}(\mathbf{x}, \mathbf{w}) \\ &= \nabla h(\hat{\mathbf{x}})\mathbf{A}\hat{\mathbf{x}} = L_{\mathbf{f}}h(\hat{\mathbf{x}}), \\ L_{\mathbf{G}}h_{\rm lc}(\mathbf{x}, \mathbf{w}) &= \nabla h_{\rm lc}(\mathbf{x}, \mathbf{w})\mathbf{G}(\mathbf{x}, \mathbf{w}) \\ &= \nabla h(\hat{\mathbf{x}})\mathbf{B} = L_{\mathbf{g}}h(\hat{\mathbf{x}}). \end{aligned} \tag{26}$$

Therefore, (24) and (26) give

$$L_{\mathbf{G}}h_{\rm lc}(\mathbf{x}, \mathbf{w}) = \mathbf{0} \implies L_{\mathbf{F}}h_{\rm lc}(\mathbf{x}, \mathbf{w}) > -\alpha\big(h_{\rm lc}(\mathbf{x}, \mathbf{w})\big), \tag{27}$$

for all $(\mathbf{x}, \mathbf{w}) \in \mathcal{S}_{\rm lc}$. Thus, Lemma 1 holds and $h_{\rm lc}$ is a valid CBF for (14) on $\mathcal{S}_{\rm lc}$.

Next, we prove that $\mathbf{k}_{\rm lc}$ ensures the safety of (14) w.r.t. $\mathcal{S}_{\rm lc}$. For all $\hat{\mathbf{x}} \in \mathcal{S}$, the safe controller $\mathbf{k}$ satisfies that

$$\dot{h}\big(\hat{\mathbf{x}}, \mathbf{k}(\hat{\mathbf{x}})\big) = L_{\mathbf{f}}h(\hat{\mathbf{x}}) + L_{\mathbf{g}}h(\hat{\mathbf{x}})\mathbf{k}(\hat{\mathbf{x}}) \geq -\alpha\big(h(\hat{\mathbf{x}})\big); \tag{28}$$

cf. (7). Substituting (21), (22), (26) into (28) yields (23) for all $(\mathbf{x}, \mathbf{w}) \in \mathcal{S}_{\rm lc}$, which implies that $\mathbf{k}_{\rm lc}$ renders (14) safe w.r.t. $\mathcal{S}_{\rm lc}$ according to Theorem 1, completing the proof. □

Remark 3. If $\mathbf{k}$ is Lipschitz continuous, $\mathbf{k}_{\rm lc}$ is also Lipschitz continuous from (15), (22). Moreover, (24) can be relaxed to a nonstrict inequality as discussed in Remark 2, which leads to a nonstrict inequality in (27).

Theorem 3 shows that, unlike HOCBF, the LCCBF $h_{\rm lc}$ is indeed a valid CBF and its corresponding safe controller $\mathbf{k}_{\rm lc}$ establishes safety for the lagged system (14). However, safety is only established on the new set $\mathcal{S}_{\rm lc}$, while the original safe set $\mathcal{S}$ encodes the original safety requirements that should also be considered. To remedy this, the following theorem states that, if the original CBF h satisfies a directional concavity condition, $\mathbf{k}_{\rm lc}$ also ensures safety w.r.t. both sets, i.e., makes $\mathcal{S}_{\rm lc} \cap \tilde{\mathcal{S}}$ forward invariant.

Theorem 4. *Consider the lagged system (14) with LCCBF $h_{\rm lc}$ in (21) and controller $\mathbf{k}_{\rm lc}$ in (22). If the original CBF h satisfies the directional concavity condition*

$$\dot{\mathbf{x}}^\top \nabla^2 h(\mathbf{x} + s\dot{\mathbf{x}})\dot{\mathbf{x}} \leq 0, \tag{29}$$

for all $s \in [0, \xi]$ and $(\mathbf{x}, \mathbf{w}) \in \mathcal{S}_{\rm lc} \cap \tilde{\mathcal{S}}$, then $\mathbf{k}_{\rm lc}$ renders (14) safe w.r.t. $\mathcal{S}_{\rm lc} \cap \tilde{\mathcal{S}}$.

Proof. Since Theorem 3 guarantees the safety of (14) on $\mathcal{S}_{\rm lc}$ using $\mathbf{k}_{\rm lc}$, Theorem 4 holds if there exists $\alpha \in \mathcal{K}^{\rm e}$ such that

$$\dot{\tilde{h}}\big(\mathbf{x}, \mathbf{w}\big) \geq -\alpha\big(\tilde{h}(\mathbf{x}, \mathbf{w})\big), \quad \forall (\mathbf{x}, \mathbf{w}) \in \mathcal{S}_{\rm lc} \cap \tilde{\mathcal{S}}, \tag{30}$$

based on Theorem 1. Using the second-order Taylor expansion with the mean-value theorem applied to the LCCBF (21), there exists $s \in [0, \xi]$ such that

$$\begin{aligned} h_{\rm lc}(\mathbf{x}, \mathbf{w}) &= h\big(T(\mathbf{x}, \mathbf{w})\big) = h(\mathbf{x} + \xi\dot{\mathbf{x}}) \\ &= h(\mathbf{x}) + \xi\nabla h(\mathbf{x})\dot{\mathbf{x}} + \tfrac{\xi^2}{2}\dot{\mathbf{x}}^\top \nabla^2 h(\mathbf{x} + s\dot{\mathbf{x}})\dot{\mathbf{x}} \\ &\leq h(\mathbf{x}) + \xi\nabla h(\mathbf{x})\dot{\mathbf{x}}, \end{aligned} \tag{31}$$

for all $(\mathbf{x},\mathbf{w}) \in \mathcal{S}_{\rm lc} \cap \tilde{\mathcal{S}}$ based on (29). From (14), (20), the lifted function $\tilde{h}$ satisfies

$$\dot{\tilde{h}}\big(\mathbf{x},\mathbf{w}\big) = L_{\mathbf{F}}\tilde{h}(\mathbf{x},\mathbf{w}) = \nabla h(\mathbf{x})\dot{\mathbf{x}}. \tag{32}$$

Given that $\tilde{h}(\mathbf{x},\mathbf{w}) = h(\mathbf{x})$, substituting (32) into (31) yields

$$\dot{\tilde{h}}\big(\mathbf{x},\mathbf{w}\big) \geq \frac{1}{\xi}\big(h_{\rm lc}(\mathbf{x},\mathbf{w}) - \tilde{h}(\mathbf{x},\mathbf{w})\big). \tag{33}$$

Since $h_{\rm lc}(\mathbf{x},\mathbf{w}) \geq 0$ for all $(\mathbf{x},\mathbf{w}) \in \mathcal{S}_{\rm lc} \cap \tilde{\mathcal{S}}$, (33) implies (30) with $\alpha(r) = r/\xi$, thus completing the proof. □

Note that the safety-critical controllers discussed above do not explicitly account for input limits and may therefore yield safe but infeasible control inputs. However, if the original safety-critical controller $\mathbf{k}$ satisfies an input constraint for the lag-free system, then, unlike the HOCBF approach, the lag-compensating controller $\mathbf{k}_{\rm lc}$ in (22) also satisfies the same input bounds for the lagged system. This is formalized in the following corollary.

Corollary 2. *Let $\mathcal{U} \subset \mathbb{R}^m$ denote a nonempty admissible input set. If $\mathbf{k}$ is a feasible safety-critical controller for the lag-free system* (13)*, i.e., $\mathbf{k}(\mathbf{x}) \in \mathcal{U}$ and $\mathbf{k}(\mathbf{x})$ satisfies* (7) *for all $\mathbf{x} \in \mathcal{S}$, then $\mathbf{k}_{\rm lc}$ in* (22) *is a feasible safety-critical controller for the lagged system* (14)*, i.e., $\mathbf{k}_{\rm lc}(\mathbf{x},\mathbf{w}) \in \mathcal{U}$ and $\mathbf{k}_{\rm lc}(\mathbf{x},\mathbf{w})$ satisfies* (23) *for all $(\mathbf{x},\mathbf{w}) \in \mathcal{S}_{\rm lc}$.*

Proof. Replacing $\mathbf{x}$ with $\hat{\mathbf{x}}$ yields that $\mathbf{k}(\hat{\mathbf{x}}) \in \mathcal{U}$ and (28) hold for all $\hat{\mathbf{x}} \in \mathcal{S}$. Because $\mathbf{k}_{\rm lc}(\mathbf{x},\mathbf{w}) = \mathbf{k}(\hat{\mathbf{x}})$ based on (22) and $\hat{\mathbf{x}} \in \mathcal{S} \iff (\mathbf{x},\mathbf{w}) \in \mathcal{S}_{\rm lc}$, we have $\mathbf{k}_{\rm lc}(\mathbf{x},\mathbf{w}) \in \mathcal{U}$ for all $(\mathbf{x},\mathbf{w}) \in \mathcal{S}_{\rm lc}$. Furthermore, (28) implies that (23) holds for all $(\mathbf{x},\mathbf{w}) \in \mathcal{S}_{\rm lc}$ as discussed in the proof of Theorem 3, which completes the proof. □

Corollary 2 shows that the feasibility of safety-critical controllers is preserved when extending from lag-free to lagged systems. This is because, unlike HOCBF, the LCCBF explicitly incorporates the lag ξ and compensates for its adverse effects by constructing a more conservative safe set, while keeping the safe input constraints unchanged, thereby preserving feasibility. This property is further demonstrated in the following example.

V. Example: double integrator

In this section, we illustrate the proposed LCCBF approach on a double integrator model with first-order input lag. Specifically, we show that if an original CBF for the double integrator can synthesize a safe and feasible controller, then the corresponding LCCBF also guarantees safety and feasibility for the double integrator with first-order lag. A comparison with the HOCBF approach is also provided.

First, consider the lag-free double integrator as:

$$\dot{d} = v, \quad \dot{v} = u, \tag{34}$$

Here d and v denote position and velocity, and $u \in \mathcal{U}$ is the control input where $\mathcal{U} = \{u \in \mathbb{R} : u_{\min} \leq u \leq u_{\max}\}$. Rewriting (34) in the control-affine form (1) yields the state $\mathbf{x} = [d \quad v]^\top \in \mathbb{R}^2$ with dynamics $\mathbf{f}(\mathbf{x}) = [v \quad 0]^\top$ and $\mathbf{g}(\mathbf{x}) = [0 \quad 1]^\top$.

We define the CBF for (34) as

$$h(\mathbf{x}) = -d - \frac{v^2}{2\mu}, \tag{35}$$

with parameter $\mu > 0$ and safe set $\mathcal{S}$ given in (3). Note that h is globally concave as $\nabla^2 h(\mathbf{x}) \preceq 0$ for all $\mathbf{x} \in \mathbb{R}^2$ and the directional concavity condition (29) is satisfied.

We design a safety filter as in (9) to filter out unsafe behavior from the desired controller $k_{\rm d}$, which is chosen as a state-feedback law with stable gain $\mathbf{K}$ given in Table I:

$$k_{\rm d}(\mathbf{x}) = {\rm sat}\big(-\mathbf{K}\mathbf{x}\big), \tag{36}$$

where the saturation function

$${\rm sat}(u) = \max\left\{u_{\min}, \min\{u, u_{\max}\}\right\}, \tag{37}$$

enforces actuator limits $u \in \mathcal{U}$ for all $\mathbf{x} \in \mathbb{R}^2$.

For scalar input u, as shown in [5], the safety-critical controller k in (9) simplifies to

$$k(\mathbf{x}) = \begin{cases} \min\left\{k_{\rm d}(\mathbf{x}), k_{\rm s}(\mathbf{x})\right\}, & \text{if } L_{\mathbf{g}}h(\mathbf{x}) < 0\,, \\ k_{\rm d}(\mathbf{x}), & \text{if } L_{\mathbf{g}}h(\mathbf{x}) = 0\,, \\ \max\left\{k_{\rm d}(\mathbf{x}), k_{\rm s}(\mathbf{x})\right\}, & \text{if } L_{\mathbf{g}}h(\mathbf{x}) > 0\,, \end{cases} \tag{38}$$

with

$$k_{\rm s}(\mathbf{x}) = -\frac{L_{\mathbf{f}}h(\mathbf{x}) + \alpha\big(h(\mathbf{x})\big)}{L_{\mathbf{g}}h(\mathbf{x})} = -\mu + \frac{\mu\gamma h(\mathbf{x})}{v}, \tag{39}$$

where (34), (35) yield the Lie derivatives $L_{\mathbf{f}}h(\mathbf{x}) = -v$ and $L_{\mathbf{g}}h(\mathbf{x}) = -v/\mu$, and we choose $\alpha(r) = \gamma r$ with $\gamma > 0$.

Theorem 5. *The safe controller k in (38), (39) is feasible, i.e., satisfies the input constraint $k(\mathbf{x}) \in \mathcal{U}$ for all $\mathbf{x} \in \mathcal{S}$, if*

$$u_{\min} \leq -\mu \leq u_{\max}. \tag{40}$$

Proof. Consider first $L_{\mathbf{g}}h(\mathbf{x}) = 0 \iff v = 0$ in (38). From (36), (37), we obtain $k(\mathbf{x}) = k_{\rm d}(\mathbf{x}) \in [u_{\min}, u_{\max}]$.

For $L_{\mathbf{g}}h(\mathbf{x}) < 0 \iff v > 0$, $k(\mathbf{x}) = \min\{k_{\rm d}(\mathbf{x}), k_{\rm s}(\mathbf{x})\}$, which implies $k(\mathbf{x}) \leq k_{\rm d}(\mathbf{x}) \leq u_{\max}$. Since $h(\mathbf{x}) \geq 0$ for all $\mathbf{x} \in \mathcal{S}$ and $\mu > 0$, (39) and (40) yield $k_{\rm s}(\mathbf{x}) \geq -\mu \geq u_{\min}$. As $k_{\rm d}(\mathbf{x}) \geq u_{\min}$, it follows that $k(\mathbf{x}) \geq u_{\min}$, so $k(\mathbf{x}) \in \mathcal{U}$ holds for all $\mathbf{x} \in \mathcal{S}$ with $v > 0$.

For $L_{\mathbf{g}}h(\mathbf{x}) > 0 \iff v < 0$, $k(\mathbf{x}) = \max\{k_{\rm d}(\mathbf{x}), k_{\rm s}(\mathbf{x})\}$, which gives $k(\mathbf{x}) \geq k_{\rm d}(\mathbf{x}) \geq u_{\min}$. From (39) and (40), we have $k_{\rm s}(\mathbf{x}) \leq -\mu \leq u_{\max}$ for all $\mathbf{x} \in \mathcal{S}$. Since $k_{\rm d}(\mathbf{x}) \leq u_{\max}$, it follows that $k(\mathbf{x}) \leq u_{\max}$. Therefore, $k(\mathbf{x}) \in \mathcal{U}$ holds for all $\mathbf{x} \in \mathcal{S}$ with $v < 0$.

In all cases $k(\mathbf{x}) \in \mathcal{U}$ holds, proving the theorem. □

Theorem 5 shows that, using the CBF h in (35), one can synthesize a feasible safety-critical controller for the lag-free double integrator. We now introduce a first-order input lag $\xi > 0$ to (34), leading to the dynamics

$$\dot{d} = v, \quad \dot{v} = a, \quad \dot{a} = \tfrac{1}{\xi}(u - a), \tag{41}$$

where $a \in \mathbb{R}$ denotes the lagged input, which can be interpreted as the actual acceleration. Writing (41) in the control-affine form (1) yields $\mathbf{F}(\mathbf{x}, a) = [v \quad a \quad -a/\xi]^\top$ and $\mathbf{G}(\mathbf{x}, a) = [0 \quad 0 \quad 1/\xi]^\top$. Since $L_{\mathbf{G}}\tilde{h}(\mathbf{x}, a) \equiv 0$, h in

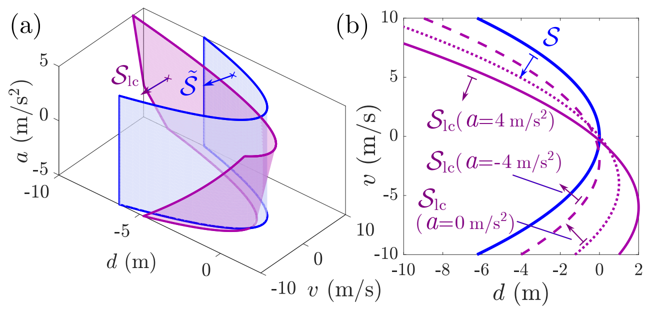

Fig. 1. Safe sets $\mathcal{S}$, $\tilde{\mathcal{S}} = \mathcal{S} \times \mathbb{R}$ and $\mathcal{S}_{\rm lc}$, defined by the CBF h in (35) and the LCCBF $h_{\rm lc}$ in (43) with lag $\xi = 0.5$ s. Panel (a) show the sets in the 3-dimensional state space while panel (b) depicts 2-dimensional sections for various values of a. Arrows indicate the safe side of the set boundaries.

(35) is not a valid CBF for (41). To address this, we apply the lag-eliminating state transformation given in (15):

$$\hat{\mathbf{x}} = \mathbf{x} + \xi \dot{\mathbf{x}} = \begin{bmatrix} \hat{d} \\ \hat{v} \end{bmatrix} = \begin{bmatrix} d + \xi v \\ v + \xi a \end{bmatrix}, \tag{42}$$

and substitute it into h, which yields the LCCBF

$$h_{\rm lc}(\mathbf{x}, a) = h(\hat{\mathbf{x}}) = -d - \xi v - \frac{(v + \xi a)^2}{2\mu}, \tag{43}$$

with the safe set $\mathcal{S}_{\rm lc} = \{(\mathbf{x}, a) \in \mathbb{R}^3 : h_{\rm lc}(\mathbf{x}, a) \geq 0\}$.

Fig. 1 illustrates the sets $\mathcal{S}$, $\tilde{\mathcal{S}} = \mathcal{S} \times \mathbb{R}$ (blue) and $\mathcal{S}_{\rm lc}$ (purple), given by (35) and (43), in both the (d, v, a) and (d, v) spaces. For $v > 0$, a larger acceleration a causes $\mathcal{S}_{\rm lc}$ to shrink, yielding a smaller safe region than $\mathcal{S}$. This conservatism considers the negative effect of lag, as lagged responses under large accelerations reduce safety margins. Moreover, as $h_{\rm lc}$ depends on the lag ξ, larger lag values yield more conservative safe sets. Conversely, for $v < 0$, increasing a expands $\mathcal{S}_{\rm lc}$ beyond $\mathcal{S}$. Importantly, since h is globally concave, Theorem 4 ensures the safety of $\mathcal{S}_{\rm lc} \cap \tilde{\mathcal{S}}$ under the lag-compensating controller $k_{\rm lc}$ derived below.

From (41), (43), we compute the Lie derivatives:

$$\begin{aligned} L_{\mathbf{F}} h_{\rm lc}(\mathbf{x}, a) &= -v - \xi a = -\hat{v} = L_{\mathbf{f}} h(\hat{\mathbf{x}}), \\ L_{\mathbf{G}} h_{\rm lc}(\mathbf{x}, a) &= -\frac{v + \xi a}{\mu} = -\frac{\hat{v}}{\mu} = L_{\mathbf{g}} h(\hat{\mathbf{x}}); \end{aligned} \tag{44}$$

cf. (26). From (9), we obtain the lag-compensating safety-critical controller

$$k_{\rm lc}(\mathbf{x}, a) = \begin{cases} \min\left\{k_{\rm d}(\mathbf{x}), k_{\rm s}^{\rm lc}(\mathbf{x}, a)\right\}, & \text{if } v > -\xi a, \\ k_{\rm d}(\mathbf{x}), & \text{if } v = -\xi a, \\ \max\left\{k_{\rm d}(\mathbf{x}), k_{\rm s}^{\rm lc}(\mathbf{x}, a)\right\}, & \text{if } v < -\xi a, \end{cases} \tag{45}$$

with

$$k_{\rm s}^{\rm lc}(\mathbf{x}, a) = -\mu + \frac{\mu \gamma h_{\rm lc}(\mathbf{x}, a)}{v + \xi a}; \tag{46}$$

cf. (38), (39). Note that Corollary 2 and Theorem 5 ensure that $k_{\rm lc}$ is a feasible safety-critical controller, satisfying $k_{\rm lc}(\mathbf{x}, a) \in \mathcal{U}$ on $\mathcal{S}_{\rm lc}$, if condition (40) holds.

We also construct the HOCBF for (41) using (10), (35):

$$h_{\rm e}(\mathbf{x}, a) = -v - \frac{va}{\mu} - \gamma_{\rm e}\left(d + \frac{v^2}{2\mu}\right), \tag{47}$$

with the set $\mathcal{S}_{\rm e}$ defined in (11), where $\alpha_{\rm e}(r) = \gamma_{\rm e} r$ with

TABLE I
PARAMETERS OF THE SIMULATIONS

Variable	Symbol	Value	Unit
Actuator limits	$u_{\min}, u_{\max}$	$-8, 8$	m/s^2
First-order lag	ξ	0.5	s
CBF parameter	μ	8	m/s^2
Nominal controller gain	$\mathbf{K}$	$(0.005, 0.15)$	1/s
Class-$\mathcal{K}$ parameters	$\gamma, \gamma_{\rm e}$	5, 5	1/s

$\gamma_{\rm e} > 0$. From (9), the HOCBF safety filter takes the form

$$k_{\rm e}(\mathbf{x}, a) = \begin{cases} \min\left\{k_{\rm d}(\mathbf{x}), k_{\rm s}^{\rm e}(\mathbf{x}, a)\right\}, & \text{if } v > 0, \\ k_{\rm d}(\mathbf{x}), & \text{if } v = 0, \\ \max\left\{k_{\rm d}(\mathbf{x}), k_{\rm s}^{\rm e}(\mathbf{x}, a)\right\}, & \text{if } v < 0, \end{cases} \tag{48}$$

where

$$k_{\rm s}^{\rm e}(\mathbf{x}, a) = -\gamma_{\rm e}\xi\mu + (1 - \gamma_{\rm e}\xi)a + \frac{\xi\mu\gamma h_{\rm e}(\mathbf{x}, a) - \xi a(\mu + a)}{v}. \tag{49}$$

However, from (41), (47), we have $L_{\mathbf{G}} h_{\rm e}(\mathbf{x}, a) = 0$ when $v = 0$. In this case, $L_{\mathbf{F}} h_{\rm e}(\mathbf{x}, a) = -a - a^2/\mu$, which becomes arbitrarily negative for sufficiently large $a > 0$, violating Lemma 1. Hence, $h_{\rm e}$ in (47) is not a valid CBF, and $k_{\rm e}$ in (48) may fail to ensure safety, as also observed in [14].

We simulate the lag-free double integrator (34) with the lag-free safety filter k in (38), and the lagged system (41) with the lag-compensating controller $k_{\rm lc}$ in (45); see Fig. 2(a,d,e) and Fig. 2(b,f,g), respectively. For comparison, the HOCBF controller $k_{\rm e}$ in (48) is also simulated for the same lagged system; see Fig. 2(c,h,i). All simulations use the parameters in Table I and start at $a = 0$ m/s^2.

In Fig. 2(a-c), all safety-critical controllers successfully guarantee safety, as all gray and purple trajectories remain within the original safe set $\mathcal{S}$ (green), while the desired controller (36) violates safety, as shown by the red trajectories. The LCCBF generates trajectories (gray) similar to those of the original CBF, while the HOCBF yields significantly different flows. Furthermore, the LCCBF safety filter (45) intervenes earlier than both the original CBF (38) and the HOCBF (48); see the purple trajectories in Fig. 2(b) compared to Fig. 2(a,c). This earlier deceleration from LCCBF accounts for the negative impacts of the lag on safety and ensures safety under the same input bounds as the lag-free original CBF, see the purple curves in Fig. 2(d,f).

Conversely, the safe set induced by the HOCBF in (47) does not compensate for the lag magnitude ξ. As a result, its safety filter (48) violates actuator limits even if its original CBF yields feasible input, see the purple curve in Fig. 2(h). Importantly, although safety is maintained in this simulation, the HOCBF is invalid and can lead to safety violations in some extreme cases, e.g., when system starts at $d = 0$ m, $v = 0$ m/s and $a > 0$ m/s^2, whereas the LCCBF provides provable safety for all time, as indicated by Theorem 3 and the nonnegative LCCBF values in Fig. 2(g). Overall, the proposed LCCBF not only maintains safety under lag but also preserves feasibility under the same conditions as the lag-free original CBF, which may not be guaranteed by the HOCBF-based approach.

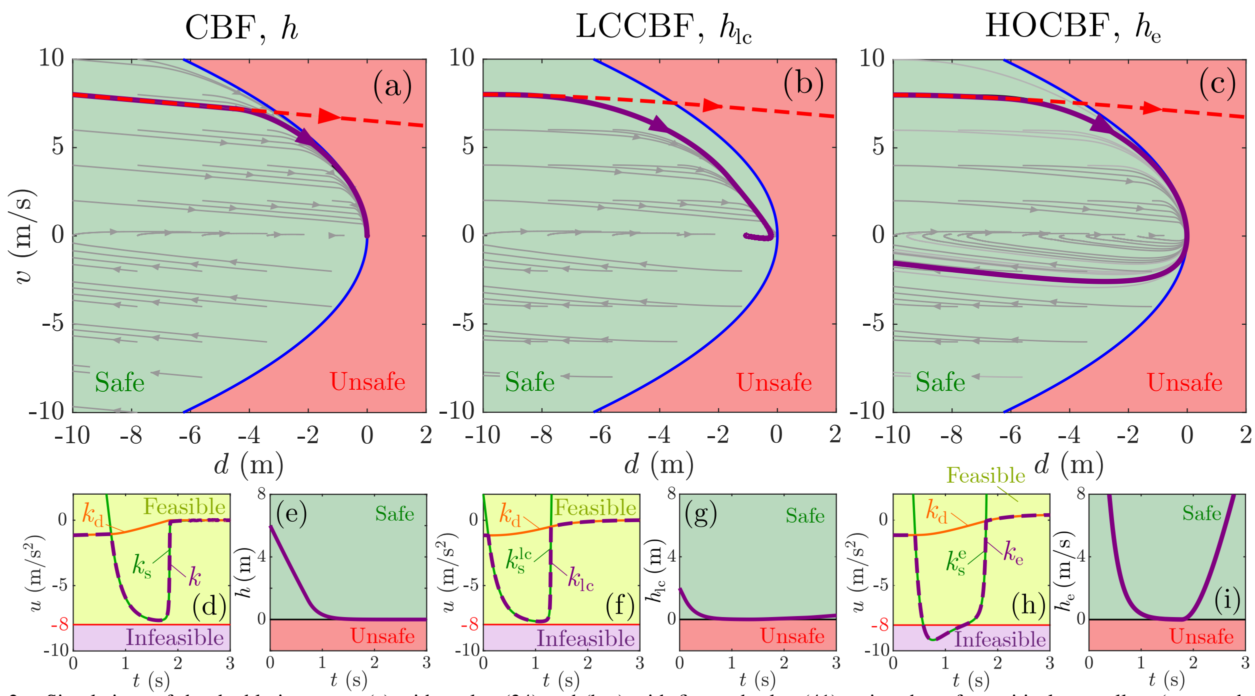

Fig. 2. Simulations of the double integrator (a) without lag (34) and (b-c) with first-order lag (41), using the safety-critical controllers (gray and purple) given by (38), (45), (48), respectively, and the desired controller (red) given by (36). Blue lines indicate the boundary of $\mathcal{S}$ given by (35). (d,f,h) Control inputs for the purple trajectories in panels (a-c), including desired controllers (orange), safe controllers (green) and safety filter inputs (purple). (e,g,i) CBF values h, h_{lc}, and h_{e} for the purple trajectories in panels (a-c).

VI. Conclusion

In this paper, we established a lag-eliminating state transformation to construct lag-compensating control barrier function (LCCBF) for linear systems with first-order input lag. The LCCBF ensures safety while preserving feasibility under the same input bounds as the original CBF of the lag-free system. With an additional directional concavity condition on the original CBF, safety guarantees extend to the nominal safe sets. We also demonstrated by simulations that our approach maintains safety and feasibility for lagged systems, outperforming high-order CBFs. In future work, we plan to extend this framework to nonlinear control-affine systems.

References

[1] C. Massera Filho, M. H. Terra, and D. F. Wolf, "Safe optimization of highway traffic with robust model predictive control-based cooperative adaptive cruise control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 11, pp. 3193–3203, 2017.

[2] M. Chen, S. L. Herbert, H. Hu, Y. Pu, J. F. Fisac, S. Bansal, S. Han, and C. J. Tomlin, "FaSTrack: A modular framework for real-time motion planning and guaranteed safe tracking," *IEEE Transactions on Automatic Control*, vol. 66, no. 12, pp. 5861–5876, 2021.

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

[4] Y. Chen, T. G. Molnar, and G. Orosz, "Safety-critical connected cruise control: Leveraging connectivity for safe and efficient longitudinal control of automated vehicles," in *27th IEEE International Conference on Intelligent Transportation Systems*, 2024, pp. 110–115.

[5] T. G. Molnar, G. Orosz, and A. D. Ames, "On the safety of connected cruise control: Analysis and synthesis with control barrier functions," in *62nd IEEE Conference on Decision and Control*, 2023, pp. 1106–1111.

[6] R. M. Bena, G. Bahati, B. Werner, R. K. Cosner, L. Yang, and A. D. Ames, "Geometry-aware predictive safety filters on humanoids: From Poisson safety functions to CBF constrained MPC," in *IEEE-RAS 24th International Conference on Humanoid Robots*, 2025, pp. 1–8.

[7] D. R. Agrawal and D. Panagou, "Online safety under multiple constraints and input bounds using gatekeeper: Theory and applications," *IEEE Control Systems Letters*, vol. 9, pp. 2309–2314, 2025.

[8] G. Orosz and T. G. Molnár, *Dynamics and Control of Connected Vehicles*. Springer, 2025.

[9] W. Xiao and C. Belta, "Control barrier functions for systems with high relative degree," in *58th IEEE Conference on Decision and Control*, 2019, pp. 474–479.

[10] A. J. Taylor, P. Ong, T. G. Molnar, and A. D. Ames, "Safe backstepping with control barrier functions," in *61st IEEE Conference on Decision and Control*, 2022, pp. 5775–5782.

[11] Y. Chen, G. Orosz, and T. G. Molnar, "Safe and stable connected cruise control for connected automated vehicles with response lag," *arXiv preprint arXiv:2409.06884*, 2024.

[12] Y. Chen, T. G. Molnar, and G. Orosz, "Feasible safe connected cruise control with backstepping control barrier functions," in *9th IEEE Conference on Control Technology and Applications*, 2025, pp. 590–595.

[13] X. Tan, W. S. Cortez, and D. V. Dimarogonas, "High-order barrier functions: Robustness, safety, and performance-critical control," *IEEE Transactions on Automatic Control*, vol. 67, no. 6, pp. 3021–3028, 2022.

[14] Y. Chen, I. Vörös, T. G. Molnar, and G. Orosz, "Safe lane-keeping with lag-compensating control barrier functions," in *IEEE Intelligent Vehicles Symposium*, 2026, accepted.

[15] M. H. Cohen, T. G. Molnar, and A. D. Ames, "Safety-critical control for autonomous systems: Control barrier functions via reduced-order models," *Annual Reviews in Control*, vol. 57, p. 100947, 2024.

[16] M. Cohen and C. Belta, *Adaptive and Learning-Based Control of Safety-Critical Systems*. Springer, 2023.

[17] T. G. Molnar, A. K. Kiss, A. D. Ames, and G. Orosz, "Safety-critical control with input delay in dynamic environment," *IEEE Transactions on Control Systems Technology*, vol. 31, no. 4, pp. 1507–1520, 2023.

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

[19] A. Alan, A. J. Taylor, C. R. He, A. D. Ames, and G. Orosz, "Control barrier functions and input-to-state safety with application to automated vehicles," *IEEE Transactions on Control Systems Technology*, vol. 31, no. 6, pp. 2744–2759, 2023.

[20] N. Bekiaris-Liberis and M. Krstic, *Nonlinear Control Under Nonconstant Delays*. SIAM, 2013.