

Integral Control Barrier Functions with Input Delay: Prediction, Feasibility, and Robustness

Adam K. Kiss, Ersin Daş, Tamas G. Molnar, and Aaron D. Ames

Abstract—Time delays in feedback control loops can cause controllers to respond too late, and with excessively large corrective actions, leading to unsafe behavior (violation of state constraints) and controller infeasibility (violation of input constraints). To address this problem, we develop a safety-critical control framework for nonlinear systems with input delay using dynamically defined (integral) controllers. Building on the concept of *Integral Control Barrier Functions* (ICBFs), we concurrently address two fundamental challenges: compensating for the effect of delays, while ensuring feasibility when state and input constraints are imposed jointly. To this end, we embed predictor feedback into the ICBF framework, making controllers operate on the predicted state. Then, utilizing ICBFs, we formulate a quadratic program for safe control design. For systems subject to simultaneous state and input constraints, we derive a closed-form feasibility condition for the resulting controller, yielding a compatible ICBF pair that guarantees forward invariance under delay. We also address robustness to prediction errors (e.g., caused by delay uncertainty) using tunable robust ICBFs. Our approach is validated on an adaptive cruise control example with actuation delay.

I. INTRODUCTION

Control Barrier Functions (CBFs) [1] provide a principled approach to safety-critical control by enabling the design of controllers that enforce the forward invariance of a safe set in the state space. This is typically implemented through safety filters, formulated as a quadratic program (QP) that minimally modifies a given nominal controller. This framework has been applied to autonomous vehicles, robotics, and multi-agent systems [2]–[5].

One of the key challenges in applying CBFs to real systems is input delay. Actuation, computation, and communication delays cause the commanded input to take effect only later. A controller unaware of this delay evaluates the CBF condition at the current state although the input takes effect later, which may cause safety violations. Predictor feedback [6] addresses this problem by forward-integrating the system over the delay interval to obtain the predicted state at which the CBF condition is evaluated [7], [8], thereby

compensating for input delay. In practice, however, the delay is often uncertain, which may induce a prediction error that acts as a disturbance. Robustness to this error can be achieved through integral quadratic constraints [9], [10], input-to-state safety [11], or robust CBFs [12], [13], while online delay estimation may further reduce conservatism [14].

The standard CBF framework [1] considers state constraints only, which limits its applicability when input constraints (such as actuator bounds) or actuator dynamics are present. This is especially important for systems with delays, because a delayed response may become more excessive with higher inputs. Integral Control Barrier Functions (ICBFs) [15] overcome input constraints by allowing the barrier function to depend on both the system state and the control input, so that input constraints and actuator dynamics can be encoded alongside the state constraint [16]. ICBFs, however, have not yet been combined with predictor-feedback formulations for input-delay systems.

Furthermore, ICBFs present another challenge. For multiple constraints (both state and input limits), the combined QP may be infeasible even if each constraint individually admits a solution. This phenomenon is well-known for CBF-based safety filters [17]: when constraints are incompatible, the QP may have no solution, or undesired equilibria may appear on the safe set boundary [18]. The compatibility (or joint feasibility) of constraints has been studied for intersections of safe sets [19], [20] and for explicit safety filter formulations [21]. Deriving a closed-form feasibility condition for ICBF-based safety filters with joint state and input constraints is still missing from the literature.

The main contributions of this paper are as follows. First, we incorporate predictor feedback into the ICBF framework so that the safety filter operates on the predicted state, compensating for the delay. Second, we derive a closed-form feasibility condition for the two-constraint QP arising from joint state and input limits, constituting a compatible ICBF pair that guarantees feasibility and forward invariance under delay. Third, we handle uncertain delay via a robust ICBF formulation based on [12], [13] that treats prediction errors as a disturbance, yielding safety guarantees that improve with prediction accuracy. We illustrate the framework on a safe adaptive cruise control example.

The rest of the paper is organized as follows. Section II revisits the ICBF framework and the two-constraint setup. Section III presents the main results: the predictor-based ICBF formulation, the feasibility condition, and tunable robustification. Section IV provides the adaptive cruise control simulation. Section V closes with conclusions.

Adam K. Kiss is with the Department of Applied Mechanics, Faculty of Mechanical Engineering, Budapest University of Technology and Economics, Budapest, Hungary, kiss_a@mm.bme.hu. Adam K. Kiss was supported by NKFIH (NKKP STARTING 153210) and the Janos Bolyai Research Scholarship of the Hungarian Academy of Sciences.

Ersin Daş is with the Department of Mechanical, Materials, and Aerospace Engineering, Illinois Institute of Technology, Chicago, IL 60616, USA, edas2@illinoisistech.edu.

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

Aaron D. Ames is with the Department of Mechanical and Civil Engineering, California Institute of Technology, Pasadena, CA 91125, USA, ames@caltech.edu.

II. BACKGROUND

We build on the framework of Control Barrier Functions (CBFs) [1], [2] and Integral CBFs [15]. We consider a general, not necessarily control-affine, nonlinear system:

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), \quad (1)$$

with continuously differentiable $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$. Throughout the paper, we omit the explicit time dependence when it is clear from the context for simplicity.

CBFs guarantee the forward invariance of the safe set

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

where the CBF $h : \mathbb{R}^n \rightarrow \mathbb{R}$ satisfies the condition

$$\sup_{\mathbf{u} \in \mathbb{R}^m} \left[\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x}) \mathbf{f}(\mathbf{x}, \mathbf{u}) + \alpha(h(\mathbf{x})) \right] \geq 0, \quad \forall \mathbf{x} \in \mathcal{C},$$

with an extended class- $\mathcal{K}$ function¹ $\alpha \in \mathcal{K}^e$. ICBFs extend this construction to dynamically defined control laws.

A. Integral Control Barrier Functions

We revisit the ICBF framework [15] for system (1). Rather than an algebraic feedback controller $\mathbf{u} = \mathbf{k}(\mathbf{x})$, we consider a *dynamically defined (integral) controller*:

$$\dot{\mathbf{u}}(t) = \phi(\mathbf{x}(t), \mathbf{u}(t)) + \mathbf{v}(t), \quad (2)$$

where $\phi : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^m$ is a locally Lipschitz nominal integral controller and $\mathbf{v} \in \mathbb{R}^m$ is an auxiliary corrective input to be determined by the safety filter. For example, with a nominal controller $\mathbf{k}_d : \mathbb{R}^n \rightarrow \mathbb{R}^m$, a common choice is [15]:

$$\phi(\mathbf{x}, \mathbf{u}) = \frac{\partial \mathbf{k}_d}{\partial \mathbf{x}}(\mathbf{x}) \mathbf{f}(\mathbf{x}, \mathbf{u}) + \frac{\alpha_\phi}{2} (\mathbf{k}_d(\mathbf{x}) - \mathbf{u}), \quad (3)$$

for $\alpha_\phi > 0$. Defining the tracking error $\mathbf{e} = \mathbf{k}_d(\mathbf{x}) - \mathbf{u}$, one obtains $\dot{\mathbf{e}} = -\frac{\alpha_\phi}{2} \mathbf{e}$ for $\mathbf{v} = \mathbf{0}$, implying that $\mathbf{u}$ converges exponentially to $\mathbf{k}_d(\mathbf{x})$ along the trajectories of (1)–(2).

We define safety using a continuously differentiable, input-dependent safety function $h : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$ acting on the augmented state $(\mathbf{x}, \mathbf{u})$. We assume that zero is a regular value of h , and consider the associated safe set:

$$\mathcal{S} := \{(\mathbf{x}, \mathbf{u}) \in \mathbb{R}^n \times \mathbb{R}^m : h(\mathbf{x}, \mathbf{u}) \geq 0\}. \quad (4)$$

Definition 1 ([15]). Function $h : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$ is an *Integral Control Barrier Function* for (1) with the controller (2) if there exists $\alpha \in \mathcal{K}^e$ such that for all $(\mathbf{x}, \mathbf{u}) \in \mathbb{R}^n \times \mathbb{R}^m$:

$$\sup_{\mathbf{v} \in \mathbb{R}^m} \left[\dot{h}(\mathbf{x}, \mathbf{u}, \mathbf{v}) + \alpha(h(\mathbf{x}, \mathbf{u})) \right] \geq 0. \quad (5)$$

Let us define:

$$\mathbf{b}(\mathbf{x}, \mathbf{u}) := \frac{\partial h}{\partial \mathbf{u}}(\mathbf{x}, \mathbf{u})^\top, \quad (6)$$

$$a(\mathbf{x}, \mathbf{u}) := -\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x}, \mathbf{u}) \mathbf{f}(\mathbf{x}, \mathbf{u}) - \frac{\partial h}{\partial \mathbf{u}}(\mathbf{x}, \mathbf{u}) \phi(\mathbf{x}, \mathbf{u}) - \alpha(h(\mathbf{x}, \mathbf{u})),$$

so that $\dot{h}(\mathbf{x}, \mathbf{u}, \mathbf{v}) + \alpha(h(\mathbf{x}, \mathbf{u})) = -a(\mathbf{x}, \mathbf{u}) + \mathbf{b}^\top(\mathbf{x}, \mathbf{u}) \mathbf{v}$. Condition (5) is equivalent to the point-wise requirement:

$$\mathbf{b}(\mathbf{x}, \mathbf{u}) = \mathbf{0} \implies a(\mathbf{x}, \mathbf{u}) \leq 0, \quad \forall (\mathbf{x}, \mathbf{u}) \in \mathbb{R}^n \times \mathbb{R}^m. \quad (7)$$

¹Function $\alpha : [0, a) \rightarrow \mathbb{R}$, $a > 0$, is of class- $\mathcal{K}$ ($\alpha \in \mathcal{K}$) if it is continuous, strictly increasing, and $\alpha(0) = 0$. Function $\alpha : (-b, a) \rightarrow \mathbb{R}$, $a, b > 0$ is of extended class- $\mathcal{K}$ ($\alpha \in \mathcal{K}^e$) if it has the same properties.

Theorem 1 ([15]). *If h is an ICBF for (1) with the controller (2), then any locally Lipschitz controller $\mathbf{v} = \mathbf{q}(\mathbf{x}, \mathbf{u})$, with $\mathbf{q} : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^m$, satisfying:*

$$\mathbf{b}^\top(\mathbf{x}, \mathbf{u}) \mathbf{v} \geq a(\mathbf{x}, \mathbf{u}) \quad (8)$$

for all $(\mathbf{x}, \mathbf{u}) \in \mathcal{S}$, renders $\mathcal{S}$ forward invariant (safe), i.e., $(\mathbf{x}(0), \mathbf{u}(0)) \in \mathcal{S} \implies (\mathbf{x}(t), \mathbf{u}(t)) \in \mathcal{S}, \forall t \geq 0$.

A minimum-norm safety filter satisfying (8) is given by

$$\mathbf{q}(\mathbf{x}, \mathbf{u}) = \underset{\mathbf{v} \in \mathbb{R}^m}{\operatorname{argmin}} \quad \|\mathbf{v}\|^2 \quad \text{s.t.} \quad \mathbf{b}^\top(\mathbf{x}, \mathbf{u}) \mathbf{v} \geq a(\mathbf{x}, \mathbf{u}). \quad (9)$$

This QP admits the explicit solution

$$\mathbf{q}(\mathbf{x}, \mathbf{u}) = \begin{cases} \frac{a(\mathbf{x}, \mathbf{u})}{\|\mathbf{b}(\mathbf{x}, \mathbf{u})\|^2} \mathbf{b}(\mathbf{x}, \mathbf{u}), & \text{if } a(\mathbf{x}, \mathbf{u}) > 0, \\ \mathbf{0}, & \text{if } a(\mathbf{x}, \mathbf{u}) \leq 0, \end{cases} \quad (10)$$

obtained via the Karush-Kuhn-Tucker (KKT) conditions.

B. State and Input Constraints

In practice, we often need to enforce two *separate* constraints simultaneously: a state constraint and an input constraint. Let the safe sets for the state and input be

$$\mathcal{S}_x := \{(\mathbf{x}, \mathbf{u}) \in \mathbb{R}^n \times \mathbb{R}^m : h_x(\mathbf{x}) \geq 0\}, \quad (11)$$

$$\mathcal{S}_u := \{(\mathbf{x}, \mathbf{u}) \in \mathbb{R}^n \times \mathbb{R}^m : h_u(\mathbf{u}) \geq 0\}, \quad (12)$$

with continuously differentiable $h_x : \mathbb{R}^n \rightarrow \mathbb{R}$, $h_u : \mathbb{R}^m \rightarrow \mathbb{R}$. Since h_x depends only on $\mathbf{x}$, it cannot serve as an ICBF. Following [15], we form the extended barrier:

$$h_e(\mathbf{x}, \mathbf{u}) := \frac{\partial h_x}{\partial \mathbf{x}}(\mathbf{x}) \mathbf{f}(\mathbf{x}, \mathbf{u}) + \alpha_x(h_x(\mathbf{x})), \quad (13)$$

with a continuously differentiable $\alpha_x \in \mathcal{K}^e$ and

$$\mathcal{S}_e := \{(\mathbf{x}, \mathbf{u}) \in \mathbb{R}^n \times \mathbb{R}^m : h_e(\mathbf{x}, \mathbf{u}) \geq 0\}. \quad (14)$$

Making $\mathcal{S}_e$ forward invariant via $\mathbf{v}$ guarantees the forward invariance of $\mathcal{S}_x \cap \mathcal{S}_e$, enforcing state constraints.

Let h_e and h_u be ICBFs. Applying (6) to h_e and h_u yields:

$$\mathbf{b}_e(\mathbf{x}, \mathbf{u}) := \frac{\partial h_e}{\partial \mathbf{u}}(\mathbf{x}, \mathbf{u})^\top, \quad (15)$$

$$a_e(\mathbf{x}, \mathbf{u}) := -\frac{\partial h_e}{\partial \mathbf{x}}(\mathbf{x}, \mathbf{u}) \mathbf{f}(\mathbf{x}, \mathbf{u}) - \frac{\partial h_e}{\partial \mathbf{u}}(\mathbf{x}, \mathbf{u}) \phi(\mathbf{x}, \mathbf{u}) - \alpha_e(h_e(\mathbf{x}, \mathbf{u})),$$

$$\mathbf{b}_u(\mathbf{x}, \mathbf{u}) := \frac{\partial h_u}{\partial \mathbf{u}}(\mathbf{u})^\top, \quad (16)$$

$$a_u(\mathbf{x}, \mathbf{u}) := -\frac{\partial h_u}{\partial \mathbf{u}}(\mathbf{u}) \phi(\mathbf{x}, \mathbf{u}) - \alpha_u(h_u(\mathbf{u})).$$

One may enforce both constraints on the safety correction $\mathbf{v}$ simultaneously, for example, by the QP:

$$\mathbf{q}(\mathbf{x}, \mathbf{u}) = \underset{\mathbf{v} \in \mathbb{R}^m}{\operatorname{argmin}} \quad \|\mathbf{v}\|^2 \quad \text{s.t.} \quad \begin{aligned} \mathbf{b}_e^\top(\mathbf{x}, \mathbf{u}) \mathbf{v} &\geq a_e(\mathbf{x}, \mathbf{u}), \\ \mathbf{b}_u^\top(\mathbf{x}, \mathbf{u}) \mathbf{v} &\geq a_u(\mathbf{x}, \mathbf{u}). \end{aligned} \quad (17)$$

Even if each constraint individually admits a feasible $\mathbf{v}$, the combined QP (17) may be infeasible when both are imposed simultaneously. This naturally raises the question: *when is the two-constraint QP guaranteed to be feasible?* We address feasibility and forward invariance in Section III-B.

III. INTEGRAL CBFs WITH INPUT DELAY

This section presents our main results: the ICBF framework for systems with input delay. We address joint state and input constraints, and robustness to uncertain prediction.

Consider the system (1) with a constant input delay $\tau > 0$:

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t-\tau)). \quad (18)$$

We assume (18) admits a unique solution $\mathbf{x}(t)$ for $t \geq 0$ given $\mathbf{x}(0)$ and a continuous input history $\mathbf{u}_t(\vartheta) := \mathbf{u}(t + \vartheta)$, $\vartheta \in [-\tau, 0)$. Due to the delay, the command $\mathbf{u}(t)$ computed at time t takes effect only at $t + \tau$, and during $[t, t + \tau]$ the system evolves under the history $\mathbf{u}_t$, which cannot be altered retroactively. Evaluating the safety condition at $\mathbf{x}(t)$ is therefore inadequate for choosing $\mathbf{u}(t)$.

Instead, we require that the physical pair $(\mathbf{x}(t), \mathbf{u}(t-\tau))$ – the current state and the delayed input that is currently acting on the system – satisfies the safety constraint:

$$h_i(\mathbf{x}(t), \mathbf{u}(t-\tau)) \geq 0, \quad (19)$$

where h_i may refer to a single constraint with h or each of the two constraints with h_e and h_u . With a slight abuse of notation, a barrier depending on a single variable is identified with its trivial extension to $\mathbf{x}, \mathbf{u}$. The delayed system must evolve inside a safe set $\mathcal{S}_i$, so that $(\mathbf{x}(t), \mathbf{u}(t-\tau)) \in \mathcal{S}_i$, with

$$\mathcal{S}_i := \{(\mathbf{x}, \mathbf{u}) \in \mathbb{R}^n \times \mathbb{R}^m : h_i(\mathbf{x}, \mathbf{u}) \geq 0\}. \quad (20)$$

We require the following regularity condition for h_i .

Assumption 1. Any nonpositive number is a regular value of h_i , that is, $[\frac{\partial h_i}{\partial \mathbf{x}}(\mathbf{x}, \mathbf{u}), \frac{\partial h_i}{\partial \mathbf{u}}(\mathbf{x}, \mathbf{u})] \neq \mathbf{0}$ for all $(\mathbf{x}, \mathbf{u}) \in \mathbb{R}^n \times \mathbb{R}^m$ satisfying $h_i(\mathbf{x}, \mathbf{u}) \leq 0$.

A. Predictor Feedback

To overcome the effect of delay, we employ predictor feedback [7]. Assuming access to $\mathbf{x}(t)$ and the stored history $\mathbf{u}_t$, the predicted state $\mathbf{x}_p(t) := \mathbf{x}(t + \tau)$ is obtained by forward-integrating (18) over the delay interval:

$$\mathbf{x}_p = \Psi(\tau, \mathbf{x}, \mathbf{u}_t) := \mathbf{x} + \int_0^\tau \mathbf{f}(\Psi(s, \mathbf{x}, \mathbf{u}_t), \mathbf{u}_t(s-\tau)) ds, \quad (21)$$

where $\mathbf{u}_t(s-\tau)$ is the known past input. A key property [6] is that $\mathbf{x}_p$ has delay-free dynamics: $\dot{\mathbf{x}}_p(t) = \mathbf{f}(\mathbf{x}_p(t), \mathbf{u}(t))$.

The integral controller is designed with the predicted state:

$$\dot{\mathbf{u}}(t) = \phi(\mathbf{x}_p(t), \mathbf{u}(t)) + \mathbf{v}(t), \quad (22)$$

where the term ϕ ensures that $\mathbf{u}$ tracks a desired nominal control law. For instance, for a static state feedback law $\mathbf{u} = \mathbf{k}_d(\mathbf{x})$, a common choice in the delay setting is:

$$\phi(\mathbf{x}_p, \mathbf{u}) = \frac{\partial \mathbf{k}_d}{\partial \mathbf{x}}(\mathbf{x}_p) \mathbf{f}(\mathbf{x}_p, \mathbf{u}) + \frac{\alpha_\phi}{2} (\mathbf{k}_d(\mathbf{x}_p) - \mathbf{u}), \quad (23)$$

for $\alpha_\phi > 0$, making $\mathbf{u}$ converge to $\mathbf{k}_d(\mathbf{x}_p)$ exponentially. Note that, because ϕ acts on the predicted state $\mathbf{x}_p$ rather than the current state $\mathbf{x}$, delay compensation is embedded in the nominal dynamics. In addition, $\mathbf{v} = \mathbf{q}(\mathbf{x}_p, \mathbf{u})$ is an auxiliary safety correction evaluated with the predicted state, which will guarantee forward invariance similar to Theorem 1. The nominal tracking objective given by ϕ is modified using $\mathbf{v}$ whenever safety intervention is necessary.

Prediction requires the delay τ and the dynamics $\mathbf{f}$ to be accurately known. However, in practice, the exact delay and dynamics are seldom known, which may lead to a *prediction error*: a mismatch between the true future state $\mathbf{x}_p$ and its estimation $\hat{\mathbf{x}}_p$. For example, if $\mathbf{f}$ is known but the controller uses a delay estimate $\hat{\tau}$, the estimated predicted state

$$\hat{\mathbf{x}}_p(t) := \Psi(\hat{\tau}, \mathbf{x}(t), \mathbf{u}_t) \quad (24)$$

differs from $\mathbf{x}_p(t) = \Psi(\tau, \mathbf{x}(t), \mathbf{u}_t)$ whenever $\hat{\tau} \neq \tau$.

While the physical system (18) is driven by the input $\mathbf{u}(t-\tau)$ and its dynamics remain undisturbed, the prediction error $\hat{\mathbf{x}}_p - \mathbf{x}_p$ affects the integral control law in (22). Namely, both the controller ϕ and the safety correction $\mathbf{q}$ are evaluated at the estimated $\hat{\mathbf{x}}_p$ instead of the true $\mathbf{x}_p$:

$$\dot{\mathbf{u}}(t) = \phi(\hat{\mathbf{x}}_p(t), \mathbf{u}(t)) + \mathbf{q}(\hat{\mathbf{x}}_p(t), \mathbf{u}(t)). \quad (25)$$

The effect of the prediction mismatch can be viewed as a matched disturbance propagated into (22):

$$\dot{\mathbf{u}}(t) = \phi(\mathbf{x}_p(t), \mathbf{u}(t)) + \mathbf{q}(\mathbf{x}_p(t), \mathbf{u}(t)) + \mathbf{d}(t), \quad (26)$$

$$\mathbf{d} := \phi(\hat{\mathbf{x}}_p, \mathbf{u}) + \mathbf{q}(\hat{\mathbf{x}}_p, \mathbf{u}) - \phi(\mathbf{x}_p, \mathbf{u}) - \mathbf{q}(\mathbf{x}_p, \mathbf{u}). \quad (27)$$

The disturbance $\mathbf{d}$ represents the effect of the prediction error on the integral controller, which may come from various prediction uncertainty sources like delay mismatch or model inaccuracies. We assume that this disturbance stays bounded.

Assumption 2. $\|\mathbf{d}(t)\| \leq \delta$ for all $t \geq 0$, for some $\delta \geq 0$.

This bounded disturbance model is motivated by the fact that prediction errors arise under state and input constraints, and accumulate over a finite horizon. Knowing a prediction error bound $\|\hat{\mathbf{x}}_p - \mathbf{x}_p\| \leq \delta_x$, the disturbance bound δ can be estimated, for example, as $\delta = (L_\phi + L_q)\delta_x$, where L_ϕ and L_q are the Lipschitz constants of ϕ and $\mathbf{q}$, respectively. When $\hat{\mathbf{x}}_p = \mathbf{x}_p$ (i.e., $\hat{\tau} = \tau$), we have $\mathbf{d} \equiv \mathbf{0}$ and $\delta = 0$.

To provide safety despite disturbances, we adopt the robust CBF framework of [12] and enforce a robust ICBF condition

$$\mathbf{b}_i^\top(\mathbf{x}_p, \mathbf{u}) \mathbf{v} \geq a_i(\mathbf{x}_p, \mathbf{u}) + r_i(\mathbf{x}_p, \mathbf{u}), \quad (28)$$

for each barrier h_i , with the additional robustness term

$$r_i(\mathbf{x}_p, \mathbf{u}) := \mu_i(h_i(\mathbf{x}_p, \mathbf{u})) \|\mathbf{b}_i(\mathbf{x}_p, \mathbf{u})\| + \sigma_i(h_i(\mathbf{x}_p, \mathbf{u})) \|\mathbf{b}_i(\mathbf{x}_p, \mathbf{u})\|^2, \quad (29)$$

where $\mu_i, \sigma_i : \mathbb{R} \rightarrow \mathbb{R}_{>0}$ are continuously differentiable and monotonously decreasing functions. A common choice is $\mu_i(h_i) = \mu_{0,i} e^{-\lambda_i h_i}$, $\sigma_i(h_i) = \sigma_{0,i} e^{-\lambda_i h_i}$ with $\mu_{0,i}, \sigma_{0,i}, \lambda_i > 0$. Using the tunability idea of [22], μ_i and σ_i concentrate robustness near the safe set boundary ($h_i \approx 0$), leaving the controller unchanged far inside the set ($h_i \gg 0$).

This finally leads to the following QP for ensuring robust safety in the presence of input delay and prediction errors:

$$\mathbf{q}(\mathbf{x}_p, \mathbf{u}) = \underset{\mathbf{v} \in \mathbb{R}^m}{\operatorname{argmin}} \quad \|\mathbf{v}\|^2 \quad (30)$$

$$\text{s.t.} \quad \mathbf{b}_i^\top(\mathbf{x}_p, \mathbf{u}) \mathbf{v} \geq a_i(\mathbf{x}_p, \mathbf{u}) + r_i(\mathbf{x}_p, \mathbf{u}),$$

where the constraint is enforced for all $i \in \{e, u\}$ in case of two constraints (with h_e, h_u), while the index i can be dropped for a single constraint (with h). Next, we establish our main theoretical results underpinning this control design.

B. Feasibility

First, we establish feasibility conditions for the proposed controller. In the case of a single constraint, we require that there exists $\alpha \in \mathcal{K}^e$ such that for all $(\mathbf{x}_p, \mathbf{u}) \in \mathbb{R}^n \times \mathbb{R}^m$:

$$\sup_{\mathbf{v} \in \mathbb{R}^m} [\dot{h}(\mathbf{x}_p, \mathbf{u}, \mathbf{v}) + \alpha(h(\mathbf{x}_p, \mathbf{u})) - r(\mathbf{x}_p, \mathbf{u})] \geq 0, \quad (31)$$

or equivalently:

$$\mathbf{b}(\mathbf{x}_p, \mathbf{u}) = \mathbf{0} \implies a(\mathbf{x}_p, \mathbf{u}) \leq 0. \quad (32)$$

Note that $r(\mathbf{x}_p, \mathbf{u}) = 0$ whenever $\mathbf{b}(\mathbf{x}_p, \mathbf{u}) = \mathbf{0}$ based on (29), hence (32) reduces to the standard ICBF condition (7). This implies that any ICBF is automatically a robust ICBF [12].

For the two-constraint case, we derive the feasibility condition based on a form of Farkas' lemma.

Lemma 1 ([19], [20]). *There exists $\mathbf{v} \in \mathbb{R}^m$ satisfying two affine constraints, $\mathbf{b}_e^\top \mathbf{v} \geq a_e$ and $\mathbf{b}_u^\top \mathbf{v} \geq a_u$, if and only if*

$$\lambda_e \mathbf{b}_e + \lambda_u \mathbf{b}_u = \mathbf{0} \implies \lambda_e a_e + \lambda_u a_u \leq 0, \quad (33)$$

for all $\lambda_e, \lambda_u \geq 0$.

Applying Lemma 1 to the two-constraint QP yields the following easy-to-check feasibility condition.

Proposition 1. *The pair (h_e, h_u) is a compatible robust ICBF pair and the QP in (30) is feasible if and only if*

$$\mathbf{b}_e = \mathbf{0} \implies a_e \leq 0, \quad (34a)$$

$$\mathbf{b}_u = \mathbf{0} \implies a_u \leq 0, \quad (34b)$$

$$\|\mathbf{b}_e\| \|\mathbf{b}_u\| + \mathbf{b}_e^\top \mathbf{b}_u = 0 \implies (a_e + r_e) \|\mathbf{b}_u\| + (a_u + r_u) \|\mathbf{b}_e\| \leq 0, \quad (34c)$$

where all quantities are evaluated at $(\mathbf{x}_p, \mathbf{u})$.

Proof. We use Lemma 1, by replacing a_i with $a_i + r_i$. We show that (33) holds for all cases where $\lambda_e \mathbf{b}_e + \lambda_u \mathbf{b}_u = \mathbf{0}$ with $\lambda_e, \lambda_u \geq 0$. If $\lambda_e = \lambda_u = 0$, (33) holds trivially. Otherwise, $\lambda_e \mathbf{b}_e + \lambda_u \mathbf{b}_u = \mathbf{0}$ means that $\mathbf{b}_e$ and $\mathbf{b}_u$ are anti-parallel or at least one of them is zero. We therefore separately analyze the cases $\mathbf{b}_e = \mathbf{0}$, $\mathbf{b}_u = \mathbf{0}$, and the anti-parallel setup corresponding to (34a), (34b), and (34c), respectively. *Case $\mathbf{b}_e = \mathbf{0}$, $\mathbf{b}_u \neq \mathbf{0}$:* $\lambda_e \mathbf{b}_e + \lambda_u \mathbf{b}_u = \mathbf{0}$ holds if $\lambda_u = 0$, so (33) requires $\lambda_e(a_e + r_e) \leq 0$ for all $\lambda_e \geq 0$, giving (34a).

Case $\mathbf{b}_e \neq \mathbf{0}$, $\mathbf{b}_u = \mathbf{0}$ is symmetric and gives (34b).

Case $\mathbf{b}_e = \mathbf{0}$, $\mathbf{b}_u = \mathbf{0}$ is covered by (34a) and (34b).

Anti-parallel case ($\|\mathbf{b}_e\| \|\mathbf{b}_u\| + \mathbf{b}_e^\top \mathbf{b}_u = 0$, while $\mathbf{b}_e \neq \mathbf{0}$, $\mathbf{b}_u \neq \mathbf{0}$): We have $\mathbf{b}_e = -c \mathbf{b}_u$ with $c = \|\mathbf{b}_e\| / \|\mathbf{b}_u\| > 0$. So $\lambda_e \mathbf{b}_e + \lambda_u \mathbf{b}_u = \mathbf{0}$ gives $\lambda_u = c \lambda_e$, and (33) yields (34c). $\square$

Conditions (34a)–(34b) coincide with the individual robust ICBF conditions (32) for h_e and h_u . Hence, they are satisfied when h_e and h_u are ICBFs. However, two ICBFs need not form a compatible ICBF pair: condition (34c) may fail when $\mathbf{b}_e$ and $\mathbf{b}_u$ are anti-parallel. Furthermore, even if compatibility holds without robustness terms, it may not hold with $r_e, r_u > 0$: unlike the single-constraint case, a compatible ICBF pair is not automatically a compatible robust ICBF pair. Condition (34c) must therefore be explicitly verified or enforced through the design of ϕ . Tuning ϕ directly shapes a_e and a_u , which can reduce the occurrence of unfavorable anti-parallel configurations and improve feasibility.

C. Robust Safety

Having established feasibility, we now state the formal safety guarantees provided by the proposed controller. Since the initial input history $\mathbf{u}(\vartheta - \tau)$ over $\vartheta \in [0, \tau]$ is already committed and cannot be altered, we make an assumption.

Assumption 3. The initial input history satisfies $(\mathbf{x}(\vartheta), \mathbf{u}(\vartheta - \tau)) \in \mathcal{S}_i$ for all $\vartheta \in [0, \tau]$.

Notably, this requirement is stronger than $(\mathbf{x}(0), \mathbf{u}(0)) \in \mathcal{S}$ which was used for the delay-free case.

Under Assumptions 1–3, we establish the safety of the time delay system (18) with the integral controller (25) in the presence of prediction errors. Specifically, we show that the set $\mathcal{S}_i$ is forward invariant if the disturbance caused by prediction errors is small (i.e., $\delta \leq \mu_i(0)$). Moreover, even if the disturbance is large (i.e., $\delta > \mu_i(0)$), we can show that trajectories remain within a neighborhood of the safe set by using the concept of input-to-state safety [23]. For each barrier h_i , we define the inflated set

$$\mathcal{S}_{i,\delta} := \{(\mathbf{x}, \mathbf{u}) \in \mathbb{R}^n \times \mathbb{R}^m : h_{i,\delta}(\mathbf{x}, \mathbf{u}) \geq 0\}, \quad (35)$$

where

$$h_{i,\delta}(\mathbf{x}, \mathbf{u}) := h_i(\mathbf{x}, \mathbf{u}) - \alpha_i^{-1} \left(-\frac{(\mu_i(0) - \delta)^2}{4 \sigma_i(h_i(\mathbf{x}, \mathbf{u}))} \right). \quad (36)$$

The expression of $h_{i,\delta}$ follows from ISSf theory by characterizing the smallest level set of h_i for which the robust barrier condition is satisfied under the disturbance bound δ . Note that the set $\mathcal{S}_{i,\delta}$ grows with δ (if $\delta > \mu_i(0)$), while δ itself grows with the prediction error $\hat{\mathbf{x}}_p - \mathbf{x}_p$. A more accurate prediction, therefore, directly reduces conservatism [11].

The following theorem establishes the forward invariance of $\mathcal{S}_i$ and $\mathcal{S}_{i,\delta}$ for both the single- and two-constraint cases.

Theorem 2. *Consider the system (18), the predicted state $\mathbf{x}_p$ in (21), its estimation $\hat{\mathbf{x}}_p$, the safe set $\mathcal{S}_i$ in (20), and the inflated set $\mathcal{S}_{i,\delta}$ in (35). Let Assumptions 1, 2, and 3 hold.*

Single-constraint case: *If the condition in (32) holds, then any locally Lipschitz controller $\mathbf{v} = \mathbf{q}(\mathbf{x}_p, \mathbf{u})$, with $\mathbf{q} : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^m$, satisfying*

$$\mathbf{b}^\top(\mathbf{x}_p, \mathbf{u}) \mathbf{v} \geq a(\mathbf{x}_p, \mathbf{u}) + r(\mathbf{x}_p, \mathbf{u}) \quad (37)$$

for all $(\mathbf{x}_p, \mathbf{u}) \in \mathcal{S}_\delta$, renders $\mathcal{S}_\delta$ forward invariant for (18)–(25). Further, if $\delta \leq \mu(0)$, then $\mathcal{S}$ is also forward invariant, i.e., $(\mathbf{x}(0), \mathbf{u}(-\tau)) \in \mathcal{S} \implies (\mathbf{x}(t), \mathbf{u}(t-\tau)) \in \mathcal{S}, \forall t \geq 0$.

Two-constraint case: *If the conditions in (34) hold, then any locally Lipschitz controller $\mathbf{v} = \mathbf{q}(\mathbf{x}_p, \mathbf{u})$, with $\mathbf{q} : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^m$, satisfying*

$$\mathbf{b}_i^\top(\mathbf{x}_p, \mathbf{u}) \mathbf{v} \geq a_i(\mathbf{x}_p, \mathbf{u}) + r_i(\mathbf{x}_p, \mathbf{u}), \quad \forall i \in \{e, u\}, \quad (38)$$

for all $(\mathbf{x}_p, \mathbf{u}) \in \mathcal{S}_{e,\delta} \cap \mathcal{S}_{u,\delta}$, renders $\mathcal{S}_{e,\delta} \cap \mathcal{S}_{u,\delta}$ forward invariant for (18)–(25). Further, if $\delta \leq \min\{\mu_e(0), \mu_u(0)\}$, then $\mathcal{S}_e \cap \mathcal{S}_u$ is also forward invariant, i.e., $(\mathbf{x}(0), \mathbf{u}(-\tau)) \in \mathcal{S}_e \cap \mathcal{S}_u \implies (\mathbf{x}(t), \mathbf{u}(t-\tau)) \in \mathcal{S}_e \cap \mathcal{S}_u, \forall t \geq 0$.

Proof. We follow [7, Appendix], [22] and give the argument for a single barrier h ; the two-barrier case follows by applying the same reasoning to h_e and h_u . Shifting time by τ , the predicted pair $(\mathbf{x}_p, \mathbf{u})$ obeys the delay-free dynamics $\dot{\mathbf{x}}_p(t) = \mathbf{f}(\mathbf{x}_p(t), \mathbf{u}(t))$, with $\dot{\mathbf{u}} = \phi(\mathbf{x}_p, \mathbf{u}) + \mathbf{q}(\mathbf{x}_p, \mathbf{u}) + \mathbf{d}$

based on (25). We prove forward invariance using Nagumo's theorem. First, we compute $\dot{h}$ along the disturbed dynamics $\dot{h}(\mathbf{x}_p, \mathbf{u}, \mathbf{q}(\mathbf{x}_p, \mathbf{u}) + \mathbf{d})$

$$= -\alpha(h(\mathbf{x}_p, \mathbf{u})) - a(\mathbf{x}_p, \mathbf{u}) + \mathbf{b}^\top(\mathbf{x}_p, \mathbf{u})(\mathbf{q}(\mathbf{x}_p, \mathbf{u}) + \mathbf{d}).$$

For ease of notation, we omit the argument of $\dot{h}$ and the arguments $(\mathbf{x}_p, \mathbf{u})$ from now on. Substituting (37), we get

$$\dot{h} \geq -\alpha(h) + r + \mathbf{b}^\top \mathbf{d}.$$

Then, we substitute r from (29), and we write $\mathbf{b}^\top \mathbf{d} \geq -\|\mathbf{b}\| \delta$ based on the Cauchy-Schwarz inequality and Assumption 2:

$$\dot{h} \geq -\alpha(h) + (\mu(h) - \delta)\|\mathbf{b}\| + \sigma(h)\|\mathbf{b}\|^2. \quad (39)$$

Further, by completing the square, it can be shown that:

$$\dot{h} \geq -\alpha(h) + (\mu(h) - \mu(0))\|\mathbf{b}\| - \frac{(\mu(0) - \delta)^2}{4\sigma(h)}. \quad (40)$$

We prove the forward invariance of $\mathcal{S}$ by considering $h = 0$. Substituting into (39) and using $\delta \leq \mu(0)$:

$$\dot{h} \geq (\mu(0) - \delta)\|\mathbf{b}\| + \sigma(0)\|\mathbf{b}\|^2 \geq 0.$$

Since zero is a regular value of h based on Assumption 1, Nagumo's theorem yields forward invariance for $\mathcal{S}$. Similarly, to show the invariance of $\mathcal{S}_\delta$, we take $h_\delta = 0$. This implies $h \leq 0$ and $\alpha(h) + \frac{(\mu(0) - \delta)^2}{4\sigma(h)} = 0$, thus (40) yields

$$\dot{h} \geq (\mu(h) - \mu(0))\|\mathbf{b}\| \geq 0,$$

where $\mu(h) \geq \mu(0)$ as μ is monotonously decreasing. Based on [22, proof of Thm. 3], $\dot{h}_\delta \geq 0 \iff \dot{h} \geq 0$, and zero is a regular value of h_δ via Assumption 1. Hence, Nagumo's theorem yields forward invariance for $\mathcal{S}_\delta$. These give safety for $t \geq \tau$, while Assumption 3 covers $t \in [0, \tau]$. $\square$

In the two-constraint case, the invariance of $\mathcal{S}_e \cap \mathcal{S}_u$ implies the invariance of $\mathcal{S}_x \cap \mathcal{S}_e \cap \mathcal{S}_u$ since $\dot{h}_x \geq -\alpha_x(h_x)$ in $\mathcal{S}_e$, enforcing state constraints. The robust case follows analogously via the ISSf interpretation associated with $\mathcal{S}_{e,\delta}$ [23].

Remark 1 (Special Cases). In case of accurate prediction ($\hat{\mathbf{x}}_p = \mathbf{x}_p$), one could set $r_i(\mathbf{x}_p, \mathbf{u}) = 0$, reducing (28) to the standard ICBF conditions evaluated at the predicted state $\mathbf{x}_p$:

$$\mathbf{b}_i^\top(\mathbf{x}_p, \mathbf{u})\mathbf{v} \geq a_i(\mathbf{x}_p, \mathbf{u}). \quad (41)$$

The corresponding minimum-norm safety filter is:

$$\mathbf{q}(\mathbf{x}_p, \mathbf{u}) = \underset{\mathbf{v} \in \mathbb{R}^m}{\operatorname{argmin}} \|\mathbf{v}\|^2 \quad (42)$$

s.t. $\mathbf{b}_i^\top(\mathbf{x}_p, \mathbf{u})\mathbf{v} \geq a_i(\mathbf{x}_p, \mathbf{u}),$

cf. (30). Additionally, in case of zero delay ($\tau = 0$), we have $\mathbf{x}_p = \mathbf{x}$, Assumption 3 holds if $(\mathbf{x}(0), \mathbf{u}(0)) \in \mathcal{S}_i$, and (41) reduces to the delay-free ICBF condition in Section II.

IV. ADAPTIVE CRUISE CONTROL WITH DELAY

We illustrate the proposed framework on an adaptive cruise control (ACC) system [1]. Consider an automated vehicle (AV) following a lead vehicle. The AV's state, $\mathbf{x} = [D \ v]^\top$, includes the distance to the lead vehicle, D , and the AV's

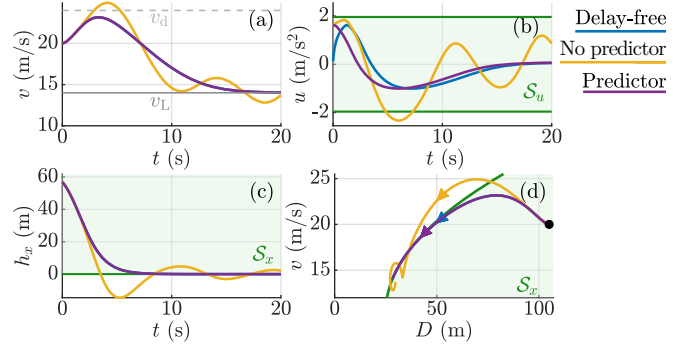


Fig. 1. Simulation of ACC, showing the effect of input delay and predictor compensation: (a) velocity; (b) acceleration; (c) CBF h_x ; (d) phase portrait. Blue: delay-free reference; yellow: naive controller unaware of input delay; purple: the proposed predictor feedback.

TABLE I. ACC simulation parameters.

Parameter	Symbol	Value	Unit
resistance	c_0	6.06×10^{-5}	m/s ²
	c_1	3.03×10^{-3}	s ⁻¹
	c_2	1.52×10^{-4}	m ⁻¹
velocity tracking gain	K_v	1	s ⁻¹
integral control gain	α_ϕ	3	s ⁻¹
R-CBF linear	$\mu_{0,e}, \mu_{0,u}$	1.0, 0.2	m/s ³ , m/s ³
R-CBF quadratic	$\sigma_{0,e}, \sigma_{0,u}$	0.1, 0.05	m/s ⁴ , s ⁻¹
R-CBF decay	λ_e, λ_u	0.05, 0.05	s/m, s ⁴ /m ²

speed, v . With actuation delay $\tau > 0$, the dynamics are

$$\dot{D}(t) = v_L(t) - v(t), \quad \dot{v}(t) = u(t - \tau) - p(v(t)), \quad (43)$$

where v_L is the lead vehicle's speed, u is the commanded acceleration, and $p(v) = c_0 + c_1 v + c_2 v^2$ is a resistance term.

Our goal is to control the AV such that it maintains a safe distance D while keeping the acceleration command u within safe bounds. We use the proposed controller in (25) and (30) with the dynamically defined controller $\phi(\mathbf{x}_p, u)$ in (23) and the nominal controller $k_d(\mathbf{x}) = K_v(v_d - v)$, which drives v towards a desired speed v_d with gain K_v . The safe following distance is encoded by the barrier

$$h_x(\mathbf{x}) = D - T_h v - \frac{(v_L - v)^2}{2 u_{\max}} - D_{\text{sf}}, \quad (44)$$

where T_h is the time headway, $u_{\max}$ is the maximum deceleration, and D_{sf} is a safe standstill distance. This is used to construct the extended barrier h_e in (13). The acceleration constraint $|u| \leq u_{\max}$ is captured by

$$h_u(u) = u_{\max}^2 - u^2. \quad (45)$$

Feasibility of the QP (30) is analyzed via Proposition 1. Conditions (34a) and (34b) are the individual ICBF conditions for h_e and h_u . Condition (34c) is nontrivial and requires attention when b_e and b_u have opposite signs, which occurs during braking ($u < 0$, so $b_u = -2u > 0$) when $b_e = -T_h + (v_L - v)/u_{\max} < 0$. In this regime, the condition (34c) must hold. Without the braking distance term $(v_L - v)^2/(2u_{\max})$ in (44), $b_e = -T_h$ would be constant and negative for all v , so opposite signs – and thus condition (34c) – would need to be checked throughout the entire braking phase with no guarantee of satisfaction. The quadratic term in (44) introduces a state-dependent contribution to b_e that reduces the region where condition (34c) is active, and encodes the physically correct stopping distance.

We present two simulation studies below using the ICBFs

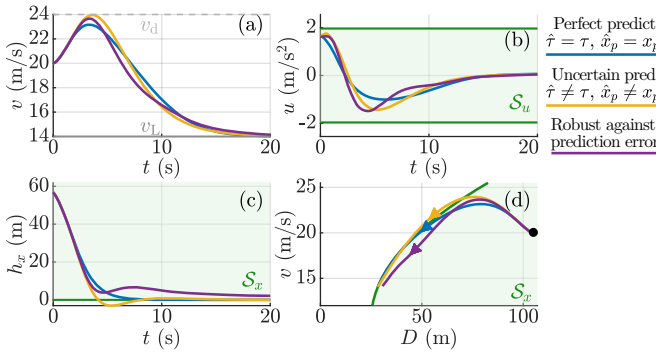


Fig. 2. Simulation of ACC, demonstrating robustness to prediction error (with 50% delay underestimation): (a) velocity; (b) acceleration; (c) CBF h_x ; (d) phase portrait. Blue: perfect prediction; yellow: uncertain prediction and no robustification; purple: uncertain prediction with robustification.

(44)–(45) and the QP (30). In the first case, we show the effect of the delay and the predictor $\hat{\mathbf{x}}_p = \Psi(\tau, \mathbf{x}, \mathbf{u}_t)$. In the second case, we highlight what happens when the delay is uncertain, $\hat{\tau} \neq \tau$, and the predictor $\hat{\mathbf{x}}_p = \Psi(\hat{\tau}, \mathbf{x}, \mathbf{u}_t)$ introduces a disturbance. The second case uses the robust margins (29) with $\mu_i(h_i) = \mu_{0,i}e^{-\lambda_i h_i}$, $\sigma_i(h_i) = \sigma_{0,i}e^{-\lambda_i h_i}$, while $\alpha_i(h) = \gamma_i h_i$ is applied in the QP (30) in each case. The simulation parameters are $u_{\max} = 1.96 \text{ m/s}^2$, $\tau = 1.2 \text{ s}$, $\hat{\tau} = 0.6 \text{ s}$, $D(0) = 105 \text{ m}$, $v(0) = 20 \text{ m/s}$, $v_L(t) \equiv 14 \text{ m/s}$, $v_d = 24 \text{ m/s}$, $\gamma_x = \gamma_e = \gamma_u = 1 \text{ s}^{-1}$, $T_h = 1.8 \text{ s}$, and $D_{sf} = 3 \text{ m}$; while the remaining parameters are in Table I.

Figure 1 shows the effect of input delay and predictor feedback. The delay-free case (blue) serves as reference with state and input constraints maintained throughout. With delay and without predictor (yellow), the ICBF filter is applied naively at the current state $\mathbf{x}$, unaware of the delay. This controller fails to anticipate the required braking and both constraints are violated. With predictor feedback (purple), the controller fully compensates for the delay: the acceleration in panel (b) is a shifted copy of the delay-free case, while the velocity and barrier in panels (a) and (c) recover the delay-free response. Thus safety is maintained per Theorem 2.

Figure 2 shows the effect of delay uncertainty, when the delay is underestimated by 50% ($\hat{\tau} = \tau/2$). With perfect prediction (blue, repeated from Fig. 1), safety is maintained exactly. With uncertain prediction and no robustification (yellow), the disturbance $\mathbf{d}$ causes the violation of the state constraint. As opposed, the proposed robust ICBF filter (purple) intervenes proactively and maintains both constraints during the simulation.

V. CONCLUSION

This paper developed an ICBF framework for the safety-critical control of nonlinear systems with input delay and joint state and input constraints. Three contributions were presented. First, predictor feedback was incorporated so that the safety filter compensates for the delay. Second, a closed-form feasibility condition was derived for the two-constraint QP arising from simultaneous state and input constraints. Third, a robust ICBF formulation was proposed to address prediction errors caused by, for example, delay uncertainty. The framework was validated on an adaptive cruise control example, where delay compensation and robustness to a

50% delay estimation error were demonstrated in simulation. Future research directions include experimental validation on embedded platforms with online delay estimation.

REFERENCES

- [1] 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, 2017.
- [2] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, “Control barrier functions: Theory and applications,” in *Eur. Control Conf.*, 2019, pp. 3420–3431.
- [3] Y. Jiang, C. Wang, B. Li, L. Song, and X. Guan, “A robust safety-critical control framework for control affine systems with applications to AUVs,” *IEEE/ASME Trans. Mechatron.*, vol. 30, no. 3, pp. 2005–2016, 2024.
- [4] Y. Xiong, D.-H. Zhai, and Y. Xia, “Robust whole-body safety-critical control for sampled-data robotic manipulators via control barrier functions,” *IEEE Trans. Autom. Sci. and Eng.*, vol. 22, pp. 16050–16061, 2025.
- [5] M. Harms, M. Jacquet, and K. Alexis, “Safe quadrotor navigation using composite control barrier functions,” in *IEEE Int. Conf. Robot. Autom.*, 2025, pp. 6343–6349.
- [6] M. Krstic, *Delay Compensation for Nonlinear, Adaptive, and PDE Systems*. Birkhäuser, 2009.
- [7] T. G. Molnar, A. K. Kiss, A. D. Ames, and G. Orosz, “Safety-critical control with input delay in dynamic environment,” *IEEE Trans. Control Syst. Technol.*, vol. 31, no. 4, pp. 1507–1520, 2023.
- [8] I. Abel, M. Janković, and M. Krstić, “Constrained control of multi-input systems with distinct input delays,” *Int. J. Robust Nonlinear Control*, vol. 34, no. 10, pp. 6659–6682, 2024.
- [9] P. Seiler, M. Jankovic, and E. Hellstrom, “Control barrier functions with unmodeled input dynamics using integral quadratic constraints,” *IEEE Control Syst. Lett.*, vol. 6, pp. 1664–1669, 2021.
- [10] Y. S. Quan, J. S. Kim, S.-H. Lee, and C. C. Chung, “Tube-based control barrier function with integral quadratic constraints for unknown input delay,” *IEEE Control Syst. Lett.*, vol. 7, pp. 1730–1735, 2023.
- [11] T. G. Molnar, A. Alan, A. K. Kiss, A. D. Ames, and G. Orosz, “Input-to-state safety with input delay in longitudinal vehicle control,” *IFAC-PapersOnLine*, vol. 55, no. 36, pp. 312–317, 2022.
- [12] R. Nanayakkara, A. D. Ames, and P. Tabuada, “Safety under state uncertainty: Robustifying control barrier functions,” *arXiv preprint arXiv:2508.17226*, 2025.
- [13] E. Daş, R. Nanayakkara, X. Tan, R. M. Bena, J. W. Burdick, P. Tabuada, and A. D. Ames, “Safe navigation under state uncertainty: Online adaptation for robust control barrier functions,” *IEEE Robot. Autom. Lett.*, vol. 11, no. 3, pp. 3398–3405, 2026.
- [14] Y. Kim, E. Daş, J. Kim, A. D. Ames, J. W. Burdick, and C. Sloth, “Minimizing conservatism in safety-critical control for input-delayed systems via adaptive delay estimation,” in *Eur. Control Conf.*, 2025, pp. 2102–2107.
- [15] A. D. Ames, G. Notomista, Y. Wardi, and M. Egerstedt, “Integral control barrier functions for dynamically defined control laws,” *IEEE Control Syst. Lett.*, vol. 5, no. 3, pp. 887–892, 2020.
- [16] E. Daş, A. D. Ames, and J. W. Burdick, “Rollover prevention for mobile robots with control barrier functions: Differentiator-based adaptation and projection-to-state safety,” *IEEE Control Syst. Lett.*, vol. 8, pp. 1727–1732, 2024.
- [17] S. S. Mousavi, X. Tan, and A. D. Ames, “From vertices to convex hulls: Certifying set-wise compatibility for CBF constraints,” *IEEE Control Syst. Lett.*, vol. 9, pp. 3011–3016, 2025.
- [18] X. Tan and D. V. Dimarogonas, “On the undesired equilibria induced by control barrier function based quadratic programs,” *Automatica*, vol. 159, p. 111359, 2024.
- [19] T. G. Molnar and A. D. Ames, “Composing control barrier functions for complex safety specifications,” *IEEE Control Syst. Lett.*, vol. 7, pp. 3615–3620, 2023.
- [20] M. H. Cohen, E. Lavretsky, and A. D. Ames, “Compatibility of multiple control barrier functions for constrained nonlinear systems,” in *IEEE Conf. Decis. Control*, 2025, pp. 771–778.
- [21] P. Mestres, S. S. Mousavi, P. Ong, L. Yang, E. Daş, J. W. Burdick, and A. D. Ames, “Explicit control barrier function-based safety filters and their resource-aware computation,” *arXiv preprint:2512.10118*, 2025.
- [22] A. Alan, A. J. Taylor, C. R. He, G. Orosz, and A. D. Ames, “Safe controller synthesis with tunable input-to-state safe control barrier functions,” *IEEE Control Syst. Lett.*, vol. 6, pp. 908–913, 2022.
- [23] S. Kolathaya and A. D. Ames, “Input-to-state safety with control barrier functions,” *IEEE Control Syst. Lett.*, vol. 3, no. 1, pp. 108–113, 2019.