Constrained control of multi-input systems with distinct input delays

Imoleayo Abel1 | Mrdjan Janković2 | Miroslav Krstić1

1Department of Mechanical and Aerospace Engineering, University of California, San Diego, California, USA
2(Retired) Research and Advanced Engineering, Ford Motor Company, Dearborn, Michigan, USA

Correspondence
Imoleayo Abel, Department of Mechanical and Aerospace Engineering, University of California San Diego, La Jolla, CA 92093-0411, USA. Email: abelimoleayo@gmail.com

Funding information
National Science Foundation, Grant/Award Number: ECCS-2151525; Office of Naval Research, Grant/Award Number: N00014-23-1-2376

Abstract
We consider the problem of enforcing safety in multi-input systems with distinct input delays via the use of Control Barrier Functions (CBFs). For systems with input delay(s), a popular approach for enforcing safety is the combination of a CBF designed for the delay-free system with state-predictors that compensate the input delays. Typically, this comes with the assumption that the system does not violate safety constraints before all input delays have been compensated and the system is fully under control. In this paper, we introduce two control approaches that enforce safety before all input delays have been compensated, whenever it is possible to do so. We do this by utilizing a robust CBF formulation that treats longer-delayed inputs as known disturbances when determining control effort for shorter delayed inputs. This formulation ensures that, whenever possible, a subset of input channels with shorter delays will be utilized for keeping the system in the admissible safe set before longer input delays have been compensated. The effectiveness of our approaches is demonstrated with two numerical examples.

KEYWORDS
constrained control, delay systems, Robust control barrier functions, safety-critical control

1 | INTRODUCTION

1.1 Background and motivation

The task of enforcing the safe operation of nonlinear systems is often framed as a constrained control problem, where the output of a system is constrained to be within a desired set. One increasingly popular approach for this problem is the use of what is now referred to as Control Barrier Functions (CBFs)1–6 and in the preceding times was referred to as “non-overshooting control”.7 Like Lyapunov functions for stabilization, CBFs provide sufficient conditions on system input for enforcing forward invariance of a safe-set, and they have been used in various applications ranging from safe navigation8,9 to infectious disease control.10,11

As interest in CBFs have developed over time, several extensions have emerged that render CBFs applicable to systems of different forms. For example, CBFs have been used for the safe operation of hybrid12 and discrete-time13,14 systems; robust CBF formulations15,16 have also been developed for enforcing safety in the presence of disturbances,17 and several
studies (see References 10-11,18-24) have explored the usage of CBFs for systems with delays. In particular, in the work of Prajna and Jadbabaie18 and the work of Orosz and Ames,22 Lyapunov-Krasovskii-like functionals were developed for verifying safety in systems with state delays; and in the works of Ames et al.10 and Molnár et al.,11 the problem of safety in the context of infectious disease control with a single measurement delay was considered. Similarly, in the works of Molnár et al.24 and Singletary et al.23 (sampled-data systems) the safety of systems with a single input delay was considered, and in the work of Janković,19 the author leverages results from the works of Krstić25,26 to develop controllers for safely stabilizing multi-input linear systems with same-length delays across input channels. This result was later extended in our previous work21 to linear systems with distinct delays across input channels using state-predictor designs from the works of Tsubakino and Krstić27 and Bekiaris-Liberis and Krstić.28 The result in our work21 was the first to consider safety for multi-input systems with distinct input delays. However, just like all the aforementioned results for systems with input delays, safety-guarantees are made only after the longest input delay has been compensated; that is, after the longest input delay time has elapsed. In essence, there is an assumption that the initial conditions of the inputs are such that the system remains safe until the designed control inputs have “kicked in” at all input channels. While this assumption is reasonable, it is limiting, especially in the case where the delay in some input channels is significantly longer than the delay in other input channels. In this case, the longest input delays would need to have been compensated before safety guarantees can be made. Ideally, it would be beneficial to be able to guarantee safety as soon as the delays in some shorter-delayed input channels have been compensated.

While there are several examples of real-world multi-input systems with distinct input delays, the underlying motivation for this work is collision avoidance in autonomous vehicles with different actuation delays in the steering and braking inputs. Inspired by human driving, where it is often possible—sometimes necessary—to avoid unforeseen collisions using only the more-responsive steering input, we desire a control approach that uses the shortest-delayed actuators to avoid collision whenever possible, especially in scenarios where there is not enough time to compensate the delays in longer-delayed actuators. The continued emphasis on “whenever possible” is because it will sometimes be impossible to enforce safety constraints without depending on all input channels, including longer-delayed ones. This will manifest as infeasibility of the CBF-based QP problems designed under the assumption that some—not all—inputs are available for safety enforcement; and we will address methods for accommodating this challenge.

1.2 Contributions

In this paper, we present results for the safety problem in control-affine nonlinear systems with input delays. Relative to the works of Janković19 and Tamas et al.24 which study linear and nonlinear systems (respectively) with a single input delay, we present results for nonlinear systems with multiple, distinct input delays. Specifically, we introduce a predictor-based approach for systems with distinct input delays that enforces safety before all input delays have been compensated – whenever it is possible to do so. We do this by treating inputs from longer-delayed input channels as known disturbances when determining inputs for shorter-delayed input channels. In particular, we consider the following two cases

1. First, we present results for the case where there exists a strict subset of shorter-delayed input channels sufficient for enforcing safety constraint everywhere in the safe set. When such subset of input channels exists, they are preferentially used for enforcing safety everywhere in the safe set, therefore allowing safety guarantees to be made as soon as the longest input delay in that subset of inputs has been compensated.

2. Secondly, we present results for the general case where a strict subset of input channels sufficient for enforcing safety constraint everywhere in the safe set may not exist. In this case, we introduce a control approach that ensures that the subset of input channels with already-compensated delays exert their “best” effort to enforce safety constraints whether or not this effort is sufficient. When it is not, the control approach allows the already delay-compensated inputs to minimize the violation of safety constraints until enough input delays have been compensated and system safety can be enforced.

This paper is a journal version of the conference papers20,21 whose contents are restricted to linear systems, and the special case of nonlinear systems where a strict subset of control inputs sufficient for enforcing safety everywhere in the safe set exists. The additional contributions are in Section 3.3 which constitutes an additional nine pages.
1.3 | Organization and notation

The remainder of this paper is organized as follows. Section 2 contains the problem description and an overview of some preliminaries. In Section 3, we discuss our control design methodologies and prove the results of this paper with examples illustrating the effectiveness of the proposed designs. We end with a brief conclusion in Section 4.

Notation: For differentiable function $h(x)$ and vector field $f(x)$, the notation $\mathcal{L}_h(x)$ denotes the Lie derivative of $h(x)$ with respect to $f(x)$ and is defined as $\frac{dh}{dx}(f)$. For the matrix-valued function $G : \mathbb{R}^n \to \mathbb{R}^{nxr}$ with columns $g_i \in \mathbb{R}^n$, we define $\mathcal{L}_G h(x) = \left[ \frac{\partial h}{\partial x} g_1(x), \ldots, \frac{\partial h}{\partial x} g_r(x) \right]$. A function $f(x)$ is said to be Lipschitz continuous at $x_0$ if there exists constants $L$ and $\epsilon > 0$ (that may depend on $x_0$) such that $\|f(x) - f(y)\| \leq L\|x - y\|$ wherever $\|x - x_0\| \leq \epsilon$ and $\|y - x_0\| \leq \epsilon$. A function $\alpha : (-b, a) \to (-\infty, \infty)$ for some $a, b > 0$ belongs to extended class $\mathcal{K}$ if it is continuous, strictly increasing, and $\alpha(0) = 0$. We shall also assume that any extended class $\mathcal{K}$ function used in control laws is Lipschitz continuous. For vector $u \in \mathbb{R}^r$, and constants $1 \leq j \leq k \leq r$, we define $u_{\cdot\cdot j:k} = [u_j, u_{j+1}, \ldots, u_{k-1}, u_k]^\top$; and for matrix $G \in \mathbb{R}^{nxr}$ with columns $g_i \in \mathbb{R}^n$ and constants $1 \leq j \leq k \leq r$, we define $G_{\cdot\cdot j:k} = [g_j, g_{j+1}, \ldots, g_{k-1}, g_k]$.

2 | PROBLEM DESCRIPTION AND PRELIMINARIES

2.1 | Problem description

We consider systems of the following form:

$$\dot{x}(t) = f(x(t)) + \sum_{i=1}^{r} g_i(x(t))u_i(t - D_i), \quad (1)$$

where $x \in \mathbb{R}^n$ is the state, and $u_i \in \mathbb{R}$, $i = 1, \ldots, r$ are inputs with possibly distinct delays $D_i$ satisfying (without loss of generality) $0 \leq D_1 \leq \ldots \leq D_r$, with initial conditions $u_i(\theta) = 0$ for $\theta \in [-D_i, 0)$. The vector fields $g_1(x), \ldots, g_r(x), f(x) : \mathbb{R}^n \to \mathbb{R}^n$ are assumed to be locally Lipschitz continuous. We further assume that the delay-free system

$$\dot{x}(t) = f(x(t)) + G(x(t))u(t) \quad (2)$$

with $G(x) := [g_1(x), \ldots, g_r(x)]$ and $u := [u_1, \ldots, u_r]^\top$ is forward complete with respect to the input $u$; that is, for every initial condition $x(0) = x_0$ and locally bounded input signals $u_i(t), i = 1, \ldots, r$, the corresponding solution of (2) is well defined for all $t \geq 0$. The importance of this assumption is that the plant in (1) does not exhibit finite escape before all feedback control inputs arrive.

The goal of this paper is to design controls $u_i(t), i = 1, \ldots, r$ for the system (1) that enforce the forward invariance of an admissible set

$$C = \{x \in \mathbb{R}^n \mid h(x) \geq 0\} \quad (3)$$

classified by a continuously differentiable scalar-valued function $h(x)$. That is, we aim to design controls $u_i(t), i = 1, \ldots, r$ that enforce $h(x(t)) \geq 0$ for all $t \geq 0$ provided that $h(x(0)) \geq 0$. We will refer to the set $C$ as the “safe set”, and we shall use the phrase “enforcing safety” to mean keeping the state $x(t)$ inside the set $C$.

Remark 1. We emphasize that the sole objective of our design is to render the set $C$ forward invariant; therefore, we assume that any other nominal objective (e.g., stabilization, trajectory tracking) is achieved by a baseline controller that will be overridden whenever it is at risk of destroying the forward invariance of $C$. In fact, it is crucial that one not commit an error in logic and confine safety with stability, or to regard stability as a requirement for safety—as happens too often to readers and reviewers new to CBFs, regretfully. The user’s objective may well be instability—but subject to some constraints. Any example of an objective of growing resources, with no upper limit but with a lower limit (bankruptcy) will do. In fact, there are examples where not only is instability desired, but finite-escape instability is preferable over exponential instability. In addition, we note that irrespective of input delays, it would sometimes be impossible to enforce set invariance without the system state growing unbounded. For example, it was shown in the work of Saberi that in
linear systems with a non-minimum phase function \( h(x) \) as output, there are initial states from which keeping the set \( C = \{ x | h(x) \geq 0 \} \) forward invariant necessarily lead to the system state growing unbounded. As such, we shall assume that the system (1) and the CBF \( h(x) \) are such that the set \( C \) can be kept forward invariant without finite escape—an assumption that applies in the operational domain of most mechanical systems.

Now, due to the presence of input delays, the system evolves uncontrolled in the first \( D_1 \) time units and we therefore make the following assumptions.

**Assumption 1.** The initial state \( x(0) \) and initial input histories \( u_i(\theta), \theta \in [-D_i, 0) \) for \( i = 1, \ldots, r \) are such that \( x(t) \in C \) for all \( t \in [0, D_1] \).

This assumption ensures that the system does not exit the safe set before any feedback controller has arrived at the plant. Now, because the input signals \( u_i(t) \) “kick in” at different times due to the distinct input delays, we desire a control approach that keeps the state inside of the set \( C \) using as few input channels as possible, whenever it is possible to do so. This is of particular interest in the case that the input delays are of significantly different lengths.

## 2.2 Control barrier functions

For systems with no input delays, the constrained control problem is often solved with the use of Control Barrier Functions (CBFs) defined as follows.

**Definition 1** (Control Barrier Function (CBF)). A continuously differentiable function \( h(x) \) is a Control Barrier Function (CBF) for the delay-free system (2) with respect to the admissible set \( C = \{ x \in \mathbb{R}^n | h(x) \geq 0 \} \) if there exists a Lipschitz continuous, extended class \( \mathcal{K}^\infty \) function \( a \) such that, for all \( x \in \mathbb{R}^n \),

\[
\mathcal{L}_C h(x) = 0 \Rightarrow \mathcal{L}_f h(x) + a(h(x)) > 0.
\]

It was shown in corollary 2 of the work of Ames et al.\(^2\) that if \( h(x) \) is a CBF for (2), then the controller \( u(x(t)) \) where \( u : \mathbb{R}^n \to \mathbb{R}^r \) is any locally Lipschitz feedback law satisfying

\[
\mathcal{L}_f h(x) + \mathcal{L}_C h(x) u(x) + a(h(x)) \geq 0
\]

for every \( x \in \mathbb{R}^n \), renders the set \( C \) forward invariant. Moreover, it was shown in proposition 4 of Xu et al.\(^17\) that the set \( C \) is also rendered asymptotically stable in the case that the system stays outside of the admissible set \( C \). The inequality (5) is called the barrier constraint, and is equivalent to \( \dot{h}(x) + a(h(x)) \geq 0 \) along the solution of (2) with \( u(t) = u(x(t)) \).

When safe set invariance is to be combined with other control objectives like stabilization or trajectory tracking, a common approach is to first design a baseline controller \( u_0 : \mathbb{R}^n \to \mathbb{R}^r \) achieving the desired control objective, and then treat safe-set invariance as an add-on objective enforced by a supervisory controller. In essence, the baseline controller is modified by the ‘smallest’ (in the Euclidean norm sense) additional control \( \overline{u} \) that allows the barrier constraint (5) to be satisfied. More concretely, to render a Lipschitz continuous baseline control \( u_0 \) safe, it is overridden with \( u = u_0 + \overline{u} \) where \( \overline{u} \) is the solution of the following quadratic programming (QP) problem:

\[
\overline{u} = \arg \min_{v \in \mathbb{R}^r} \| v \|^2 \quad \text{subject to} \quad a(x) + b_1(x)(u_0 + v) \geq 0
\]

where

\[
a(x) = \mathcal{L}_f h(x) + a(h(x)), \quad b_1(x) = \mathcal{L}_C h(x).
\]

The feasibility of the QP problem (6) follows from the definition of a CBF. Using the Karush-Kuhn-Tucker (KKT) optimality conditions for the solution of (6), we get the following explicit solution for \( \overline{u} \):

\[
\overline{u}(x, u_0) = \begin{cases} 0 & \text{if } \mu(x, u_0) \geq 0 \\ -\frac{\mu(x, u_0)}{b_1(x)b_1(x)^	op} b_1(x) & \text{otherwise} \end{cases}
\]

where \( \mu(x, u_0) = a(x) + b_1(x)(u_0 + \overline{u}) \).

This approach that keep the state inside of the set \( \mathbb{R}^n \) = \{ \} is called the barrier constraint, and is equivalent to \( \dot{u} \) being satisfied. More concretely, to render a Lipschitz continuous baseline control \( u_0 \) safe, it is overridden with \( u = u_0 + \overline{u} \) where \( \overline{u} \) is the solution of the following quadratic programming (QP) problem:

\[
\overline{u} = \arg \min_{v \in \mathbb{R}^r} \| v \|^2 \quad \text{subject to} \quad a(x) + b_1(x)(u_0 + v) \geq 0
\]

where

\[
a(x) = \mathcal{L}_f h(x) + a(h(x)), \quad b_1(x) = \mathcal{L}_C h(x).
\]

The feasibility of the QP problem (6) follows from the definition of a CBF. Using the Karush-Kuhn-Tucker (KKT) optimality conditions for the solution of (6), we get the following explicit solution for \( \overline{u} \):

\[
\overline{u}(x, u_0) = \begin{cases} 0 & \text{if } \mu(x, u_0) \geq 0 \\ -\frac{\mu(x, u_0)}{b_1(x)b_1(x)^	op} b_1(x) & \text{otherwise} \end{cases}
\]
where

\[ \mu(x, u_0) = a(x) + b_1(x)u_0. \quad (10) \]

The controller

\[ u(x) = u_0(x) + \overline{u}(x, u_0(x)) \quad (11) \]

for any Lipschitz continuous baseline controller \( u_0(x) \) and \( \overline{u} \) given in (9) enforces the forward invariance of the set \( C \) in (3) for the delay-free system (2) as established in corollary 2 of Ames et al.\(^2\)

**Assumption 2.** The baseline control laws \( u_0(x) \) considered in all subsequent designs are assumed to be Lipschitz continuous.

**Remark 2.** It is important to note, that while the safety feedback/override (9) is “pointwise optimal” (pointwise in state space location and in time), such feedback is not optimal over a time horizon. In fact, ensuring that the control applied at present time is the closest possible to the nominal input at present time may well force the future control to have to deviate much from the future nominal controls to ensure safety. This issue is recognized in Reference 30 and addressed through a variety of “inverse optimal” redesigns. These redesigns present parameterized families of controllers that all ensure that constraints are not violated and that, additionally, (1) a payoff function that rewards safety is maximized, and (2) a cost function that penalizes the deviation of the control applied from the nominal input is minimized. This methodology can be extended to systems with input delays but this is beyond the scope of the present paper.

### 2.3 Robust control barrier functions

Here, we briefly describe the robust CBF formulation in the work of Janković\(^{15}\) as it applies to the zeroing-type CBF utilized in this paper. Consider the delay-free system (2), but with an external bounded disturbance \( \omega(t) \)

\[ \dot{x}(t) = f(x) + G(x)u + Z(x)\omega, \quad (12) \]

where the disturbance \( \omega(t) \in \mathbb{R}^n \) satisfies \( 0 \leq \| \omega(t) \| \leq \omega_{\text{max}} \) for all \( t \), and \( Z : \mathbb{R}^n \to \mathbb{R}^{n \times n} \) is locally Lipschitz.

**Definition 2** (Robust-CBF). A continuously differentiable function \( h(x) \) is a robust-CBF (RCBF) for the system (12) with respect to the admissible set \( C = \{ x \in \mathbb{R}^n \mid h(x) \geq 0 \} \) if there exists a Lipschitz continuous, extended class \( \mathcal{K} \) function \( a \) such that, for all \( x \in \mathbb{R}^n \),

\[ \mathcal{L}_f h(x) = 0 \Rightarrow \mathcal{L}_f h(x) - \| \mathcal{L}_Z h(x) \| \omega_{\text{max}} + a(h(x)) > 0. \quad (13) \]

The \(-\| \mathcal{L}_Z h(x) \| \omega_{\text{max}} \) term in (13) represents the worst case impact of the disturbance \( \omega \) on \( \dot{h} = \mathcal{L}_f h + \mathcal{L}_G h u + \mathcal{L}_Z h \omega \), therefore the condition in (13) necessitates that an RCBF satisfies \( \dot{h}(x) + a(h(x)) \geq 0 \) in the presence of the worst-case disturbance even when the input has no effect on \( h(x) \).

As shown in the work of Janković,\(^{15}\) with an RCBF available, the task of enforcing forward invariance of \( C \) in the presence of disturbance signal \( \omega(t) \) boils down to choosing for every \( x \in \mathbb{R}^n \), control input \( u \) from the set of controls satisfying

\[ \mathcal{L}_f h(x) + \mathcal{L}_G h(x) u - \| \mathcal{L}_Z h(x) \| \omega_{\text{max}} + a(h(x)) \geq 0. \quad (14) \]

Here, the input \( u \) will achieve \( \dot{h} + a(h) \geq 0 \) in the presence of the worst case disturbance, thereby guaranteeing safe-set invariance. In the case that an accurate measurement \( \hat{\omega}(t) \) of the disturbance exists or the disturbance is known \textit{a priori}, the robust barrier constraint (14) can be relaxed from accounting for the worst case impact of the disturbance to accounting for the specific impact of the disturbance to safety as follows

\[ \mathcal{L}_f h(x) + \mathcal{L}_G h(x) u + \mathcal{L}_Z h(x) \hat{\omega} + a(h(x)) \geq 0 \quad (15) \]
in which case the input $u$ counteracts the actual impact of $\omega$ on $\dot{h}$ when $L_G h(x) \neq 0$ as opposed to accounting for the worst case impact at all times. For the rest of this paper, we shall utilize the RCBF formulation that assumes an accurate measurement of the disturbance exists. In fact, in the designs we present in this paper, the “disturbance” will be known definitively.

As in the disturbance-free case, rendering a baseline control $u_0$ robustly safe translates to overriding $u_0$ with the closest control $u = u_0 + \overline{u}$ that allows the robust barrier constraint (15) to be satisfied, where $\overline{u}$ is the solution of the QP problem

$$\overline{u} = \arg \min_{v \in \mathbb{R}^r} \|v\|^2 \quad \text{subject to}$$

$$a(x) + b_1(x)(u_0 + v) + b_2(x)\dot{\omega} \geq 0,$$

with $a(x), b_1(x)$ as defined in (7), (8), and

$$b_2(x) = L_z h(x).$$

The feasibility of this QP problem follows from the definition of an RCBF, and has the explicit solution

$$\overline{u}(x, u_0, \dot{\omega}) = \begin{cases} 0 & \text{if } \mu(x, u_0, \dot{\omega}) \geq 0 \\ -\frac{\mu(x, u_0, \dot{\omega})}{b_1(x)} b_1(x)^\top & \text{otherwise} \end{cases}$$

(18)

where

$$\mu(x, u_0, \dot{\omega}) = a(x) + b_1(x)u_0 + b_2(x)\dot{\omega}. \quad (19)$$

The controller

$$u(x) = u_0(x) + \overline{u}(x, u_0(x), \dot{\omega}) \quad (20)$$

for any baseline controller $u_0(x)$ and with $\overline{u}$ given in (18) enforces the forward invariance of the set $C$ in (3) for the delay-free system (2) with a bounded external disturbance as shown in Theorem 2 of the work of Janković.\textsuperscript{15} We note that while the results in the work of Janković\textsuperscript{15} were stated for the CLF-RCBF QP problem of simultaneous stabilization and safety, its adaptation in this paper replaces the use of a Lyapunov function with the baseline control which the designer is allowed to design using any desired approach.

### 2.4 Exact predictor feedback for systems with distinct input delays

Now we return to the original disturbance-free system (1) with distinct input delays. A predictor-feedback approach for stabilization was developed in the work of Bekiaris-Liberis and Krstić\textsuperscript{28} which combines state predictors with a globally asymptotically stabilizing feedback law $u = \kappa(x)$ designed for the delay-free system (2). Specifically, the approach involves designing individual state predictors $p_i(t)$, $i = 1, \ldots, r$ corresponding to each of the $r$ input channels with $p_i(t)$ being the $D_i$-time units ahead prediction of $x(t)$ that is, $p_i(t) = x(t + D_i)$ for all $i = 1, \ldots, r$. These predictor states are then used as feedback to get the following control law

$$u_i(t) = \kappa_i(p_i(t)) \quad (21)$$

where $\kappa_i$ is the $i$-th component of $\kappa$. Defining $p_0(t) = x(t)$, $D_0 = 0$, and $D_{ji} = D_j - D_i$, the predictor state equations are given recursively as:

$$p_i(t) = p_{i-1}(t) + \int_{t-D_{ji}}^{t} \left[ f(p_i(s)) + \sum_{j=1}^{i-1} g_j(p_i(s)) \kappa_j(p_i(s)) + \sum_{j=i}^{m} g_j(p_i(s)) u_j(s - D_{ji}) \right] ds \quad (22)$$
with initial conditions
\[
p_i(\theta) = p_{i-1}(0) + \int_{-D_{i-1}}^{\theta} \left[ f(p_i(s)) + \sum_{j=1}^{i-1} g_j(p_i(s))\kappa_j(p_i(s)) + \sum_{j=i+1}^{m} g_j(p_i(s))u_j(s - D_{j-i}) \right] ds, \quad \text{for} \quad -D_{i-1} \leq \theta \leq 0 \tag{23}
\]

It was shown in the work of Bekiaris-Liberis and Krstić that the predictor (22) satisfies
\[
p_i(t) = x(t + D_i) \quad \forall \ i = 1, \ldots, r. \tag{24}
\]

The main idea of the predictor-feedback approach is that the controllers (21) satisfy \( u_i(t - D_i) = \kappa_i(x(t)) \) leading to the closed-loop dynamics of (1), (21) being \( \dot{x}(t) = f(x) + \sum_{j=1}^{r} g_j(x)\kappa_j(x(t)) \) which is delay-free. Therefore, a feedback controller designed for the delay-free system (2) achieves the same closed loop behavior for the system (1) under predictor feedback. Importantly, since the feedback control signals \( \kappa_i(p_i(t)) \), \( i = 1, \ldots, r \) kick in at different times, we need to make the additional assumption that the system does not exhibit finite escape during time \( D_1 \leq t < D_r \) when only some, not all of the feedback controllers have kicked in. Specifically, we make the following additional assumption

**Assumption 3.** The systems \( \dot{x} = f(x) + \sum_{j=1}^{r} g_j(x)\kappa_j(x) + \sum_{j=i+1}^{r} g_j(x)\theta_j \) for all \( i = 1, \ldots, r - 1 \) are forward complete with respect to locally bounded input signal \( [\theta_{i+1}, \ldots, \theta_r]^{\top} \in \mathbb{R}^{r-i} \).

**Remark 3.** While the predictor Equations (22), (23) are implicitly defined and may not have explicit solutions, they can be implemented numerically and the implications of such numerical implementation are an area of research that is beyond the scope of this work; see the work of Karafyllis and Krstić for details. We shall assume in this paper that state measurements and numerical predictor implementations are accurate.

### 3 | CONTROL DESIGN AND SAFETY ANALYSIS

#### 3.1 | Safety after longest input delay compensation

Guided by the predictor-feedback approach in the work of Bekiaris-Liberis and Krstić, a first-attempt at designing safe controllers for the system (1) with distinct input delays is to design a safe CBF-based controller for the delay-free system (2), and then use the predictor (22) for each \( i = 1, \ldots, r \) to compensate the input delays. Specifically, using the CBF-based approach in Section 2.2, we have the following safe-guarding controller for the delay-free (DF) system (2)

\[
u = \kappa_{DF}(x) = u_0(x) + \bar{u}(x, u_0(x)) \tag{25}
\]

where \( u_0(x) \) is any safety-agnostic baseline controller for the delay-free system (2), and \( \bar{u} \) is the CBF-based modification defined in (9). It follows from corollary 2 in the work of Ames et al. that the controller (25) enforces the forward invariance of the safe set \( C \) for the delay-free system (2) and any baseline controller \( u_0 \). Now, with a safe controller for the delay-free system available, we can enforce safety for the original system (1) with input delays using the control law

\[
u_i(t) = \kappa^DF_i(p_i(t)), \quad t \geq 0, \quad i = 1, \ldots, r \tag{26}
\]

where \( p_i(t) \) is as defined in (22) with \( \kappa_j \) replaced with \( \kappa^DF_j \).

**Theorem 1.** Let \( h(x) \) be a CBF for the delay free system (2), and consider the closed loop system of (1) and controller (26), where \( \kappa^DF \) is the feedback law in (25), (9), and \( p_i(t) \) are state predictors given in (22), (23). For all \( t \geq D_r \), we have

\[\dot{h}(x(t)) + a(h(x(t))) \geq 0.\tag{27}\]

Moreover, if \( x(D_r) \in C \), then the set \( C \) is forward invariant.
Proof. Since \( p_i(t) = x(t + D_i) \), the controller (26) satisfies
\[
    u_i(t - D_i) = k_i^{DF}(x(t)), \quad t \geq D_i, \quad i = 1, \ldots, r.
\] (28)

Thus, for all \( t \geq D_r \), the closed loop system of (1), (26) becomes
\[
    \dot{x}(t) = f(x(t)) + G(x(t))k^{DF}(x(t))
\] (29)

which has no input delays, and since \( k^{DF} \) is the CBF QP based controller in (11), it follows by definition that for all \( t \geq D_r \), \( \dot{h}(x(t)) + a(h(x(t))) \geq 0 \), and therefore \( h(x(D_r)) \geq 0 \) implies \( h(x(t)) \geq 0 \) for all \( t \geq D_r \). \( \blacksquare \)

Remark 4. The result of Theorem 1 was stated for times \( t \geq D_r \), because only after time \( t = D_r \) does the input vector \([u_1(t - D_1), \ldots, u_r(t - D_r)]^T\) become equivalent to \( k^{DF}(x(t)) \). Therefore, the enforcement of the barrier constraint \( h + a(h) \geq 0 \) for the system (1) under control law (26) can only be guaranteed for times \( t \geq D_r \) when all the input delays have been compensated and the closed loop system exhibits nominal delay-free behavior. In fact, it is possible that the system exits the safe-set between time \( t = D_1 \) and \( t = D_r \), even though some of the input delays have been compensated. This is because the controller (26) is not guaranteed to achieve \( h + a(h) \geq 0 \) until time \( t = D_r \), even though the delays in some shorter-delayed input channels would have been compensated, and these input channels may be sufficient for enforcing the barrier constraint.

To allow comparison with subsequent control approaches we shall present, the controller (26) can be written as
\[
    u_i(t) = u_0(p_i(t)) + \bar{u}_i(p_i(t), u_0(p_i(t)))
\] (30)

where \( u_0 : \mathbb{R}^n \to \mathbb{R}^r \) is any baseline controller and \( \bar{u}(p_i(t), u_0(p_i(t))) \) is the solution of the QP problem
\[
    \bar{u} = \arg \min_{v \in \mathbb{R}^r} \|v\|^2 \quad \text{subject to}
    \begin{align*}
        & \mathcal{L}_f h(p_i(t)) + a(h(p_i(t))) + \mathcal{L}_G h(p_i(t))(u_0(p_i(t)) + v) \geq 0
    \end{align*}
\] (31)

which is given as
\[
    \bar{u}_i(p_i(t), u_0(p_i(t))) = \begin{cases} 
        0, & \mu_i \geq 0 \\
        -\mu_i \frac{\mathcal{L}_f h(p_i)(u_0)}{\mathcal{L}_G h(p_i)(u_0)}, & \text{otherwise}
    \end{cases}
\] (32)

where
\[
    \mu_i = \mathcal{L}_f h(p_i(t)) + a(h(p_i(t))) + \mathcal{L}_G h(p_i(t))u_0(p_i(t))
\] (33)

We note here that the QP problem (31) is defined for each of the \( r \) input channels, and produces an input vector \( \bar{u} \in \mathbb{R}^r \) whose \( i \)-th component is used as the input modification \( \bar{u}_i(p_i(t), u_0(p_i(t))) \) as in (32). In the controller (30), (32), there is an implicit assumption that at every time \( t \geq D_r \), the predicted states \( p_i(t - D_i) \) for all input channels \( i = 1, \ldots, r \) are equal and consistent with the actual state \( x(t) \). The same assumption is made about the predicted values of the barrier function and its Lie derivatives. An implication of this assumption is that the controller (30) can only respond to perturbations in \( h(x(t)) \) whose effects do not render the system unsafe in under \( D_r \) units of time. One practical example of this limitation is the following: consider an autonomous robot with distinct longitudinal and lateral input delays ranging from \( D_{min} \) to \( D_{max} \) navigating an obstacle-free path under controller (30), (32). If an obstacle is introduced sufficiently close to the robot so that a collision occurs in \( D_{min} < t < D_{max} \) units of time, the predictions made by longer delayed input channels about the current status of the system will become outdated, and the controller may not enforce safety because it assumes that the predictions made for all control channels match the current reality. To formalize this notion of prediction consistency, we introduce the following definition

**Definition 3.** For the system (1) and state predictors (22), a function \( s(x) \) is said to be \( i \)-consistent at time \( t \geq D_i \) if, for all \( j = 1, \ldots, i \)
\[
    s(x(t)) = s(p_i(t - D_i)).
\] (34)
A function being \( i \)-consistent at time \( t \) means that the actual value of the function at time \( t \) is equivalent to the values predicted for the first \( i \) input channels. When a function is \( r \)-consistent at time \( t \), we say it is fully consistent; if not, we say it is partially consistent.

Using this definition, the assumption made for the controller (30), (32) is that the functions \( I(x(t)) = x(t), h(x(t)), \mathcal{L}_j h(x(t)) \) and \( \mathcal{L}_{G_h} h(x(t)) \) are fully consistent at all times \( t \geq 0 \). Notice however that by definition these functions cannot become \( r \)-consistent until time \( t = D_i \), and when the \( r \)-consistency assumption does not hold at a particular time \( t \), barrier constraint satisfaction is not guaranteed at that time. We note here that a function \( s(x(t)) \) can also become \( i < r \) consistent due to several factors including but not limited to prediction error, or an unforeseen perturbation of the function \( s(x(t)) \). Ideally, it would be preferred to have a controller that guarantees barrier constraint enforcement when predictions are \( i \)-consistent for some \( i < r \). One approach for doing this is presented next.

### 3.2 Safety before longest input delay compensation using fixed number of input channels

To enable barrier constraint satisfaction when predictions are \( i < r \) consistent, we utilize the robust RCBF-based QP formulation in Section 2.3 in the following way. When solving for the \( i \)-th input \( u_i(t) \), we compute the state prediction \( p_i(t) \) and compute the baseline control \( u_0(p_i(t)) \in \mathbb{R}^r \) as before, but for safety, we assume only the first \( i \) input channels are available for enforcing the barrier constraint. The idea here is that when determining input \( u_i(t) \) that would reach the plant at time \( t + D_i \), all other inputs from channels \( j = i + 1, \ldots, r \) that would reach the plant at the same time \( t + D_i \) have already been determined, and can no longer be modified. In addition to assuming only the first \( i \) inputs are available for barrier constraint enforcement, we will treat these already determined inputs \( u_j(t - (D_j - D_i)) \) for \( j = i + 1, \ldots, r \) as known disturbances entering the system at time \( t + D_i \) and include them in the barrier constraint inequality as is the case with the robust QP problem (16). In essence, when determining input \( u_i(t) \), for each \( i = 1, \ldots, r \), the modification of the baseline control \( u_0(p_i(t)) \) is chosen as the \( i \)-th component of the solution of the following “robust” QP problem defined for each \( i \).

\[
\overline{u} = \arg \min_{v \in \mathbb{R}^r} ||v||^2 \quad \text{subject to} \\
\mathcal{L}_j h(p_i(t)) + \alpha(h(p_i(t))) + \sum_{j=1}^{i} \mathcal{L}_{g_j} h(p_i(t)) (u_0(p_i(t)) + v_j) + \sum_{j=i+1}^{r} \mathcal{L}_{g_j} h(p_i(t)) u_j(t - D_j) \geq 0 \tag{35}
\]

where \( D_{j2} = D_j - D_i \), and \( \omega_j \) for \( j > i \) are already-determined inputs from longer-delayed channels arriving at the same time \( t + D_i \) as \( u_i(t) \). While it may appear computationally expensive to solve different QP problems at every time \( t \), the availability of explicit solutions via (18) mitigates this challenge. Now, the idea behind QP problem (35) is to mimic the robust QP problem (16) and treat the already determined inputs as known disturbances. This way, the first \( i \) input channels alone can be used for enforcing safety at time \( t + D_i \); especially in the case where delays in longer input channels have not been compensated.

Observe that for \( i = r \), the QP problem (35) is equivalent to (31) and it is feasible by virtue of \( h \) being a CBF for the system when all input channels are used for barrier constraint enforcement. However, for \( i < r \), the feasibility of (35) is not guaranteed because \( \mathcal{L}_{g_i} h, \ldots, \mathcal{L}_{g_r} h \) could potentially all be 0 and \( h \) is not an RCBF for the system with only the first \( i \) inputs available for safety enforcement. In other words, when \( \mathcal{L}_{G_{l_i}} h(p_i) = 0 \), (35) can be infeasible since the additional control \( v \) has no impact on the inequality constraint. This infeasibility is a reflection of the inability of the first \( i \) inputs alone to impact \( h(p_i(t)) \). To mitigate this challenge, we introduce the set

\[
\Phi = \left\{ i \in \{1, \ldots, r-1\} \mid \mathcal{L}_{G_{l_i}} h(x) \neq 0 \ \forall x \in \mathbb{R}^n \right\} \cup \{r\} \tag{36}
\]

and define

\[
\varphi = \min \Phi. \tag{37}
\]

Here, \( \varphi \) is the minimum number of input channels sufficient for enforcing the barrier constraint everywhere in \( \mathbb{R}^n \). Notice that for all \( i \geq \varphi \), the QP problem (35) will be feasible because \( \mathcal{L}_{G_{l_{\varphi}}} h \neq 0 \Rightarrow \mathcal{L}_{G_{l_i}} h \neq 0 \). However for \( i < \varphi \), (35) may be infeasible. To prevent infeasibility for \( i < \varphi \), we will assume when solving for \( u_i(t) \), \( i < \varphi \) that the first \( \varphi \) input channels
are available for barrier constraint enforcement, and use the \(i\)-th component of the resulting control modification. Thus, the QP (35) for each \(i = 1, \ldots, r\) becomes

\[
\overline{u} = \arg \min_{v \in \mathbb{R}^{d_r}} \|v\|^2 \quad \text{subject to}
\]

\[
\mathcal{L}_j h(p_i(t)) + \alpha(h(p_i(t))) + \sum_{j=1}^s \mathcal{L}_{g_j} h(p_i(t)) \left( u_0(p_i(t)) + v_j \right) + \sum_{j=s+1}^r \mathcal{L}_{g_j} h(p_i(t)) u_j(t-D_{ij}) \geq 0
\]

where

\[
s = \max\{\phi, i\}. \tag{39}\]

This leads to the controller

\[
u_i(t) = u_0(p_i(t)) + \overline{u}_i(p_i(t), u_0, (p_i(t), \omega_{s+1:r}) \tag{40}\]

for all \(t \geq 0\), where

\[
\omega_{s+1:r} = \begin{bmatrix} u_{s+1}(t - (D_{s+1} - D_s)) \\ \vdots \\ u_r(t - (D_r - D_s)) \end{bmatrix} \in \mathbb{R}^{r-s}, \tag{41}\]

and

\[
\overline{u}_i = \begin{cases} 0, & \mu_s \geq 0 \\ -\mu_s \frac{\mathcal{L}_{g_i} h(p_i)}{\mathcal{L}_{g_i} h(p_i)} \end{cases}, \quad \text{otherwise} \tag{42}\]

with

\[
\mu_s = \mathcal{L}_j h(p_i(t)) + \alpha(h(p_i(t))) + \sum_{j=1}^s \mathcal{L}_{g_j} h(p_i(t)) u_0(p_i(t)) + \sum_{j=s+1}^r \mathcal{L}_{g_j} h(p_i(t)) \omega_j \tag{43}\]

Notice the similarity of (40)–(43) to the RCBF controller in (20), (19), (18). If \(\phi < r\), then the predictions of the state and barrier function (and its Lie derivatives) only need be \(\phi\)-consistent since the shorter-delayed inputs \(u_1, \ldots, u_{\phi}\) are sufficient for barrier constraint enforcement. Furthermore, since the baseline controller \(u_0(x)\) is assumed to be Lipschitz continuous (Assumption 2), if \(s\) satisfies (39), then the QP (38) is always feasible because the LHS of (38) is fully controllable from the first \(\phi\) inputs. Therefore, the resulting control modification \(\overline{u}\) is Lipschitz continuous. In particular, the denominator of (42) is never zero.

**Theorem 2.** Let \(I : \mathbb{R}^n \to \mathbb{R}^n\) be the identity function \(I(x) = x\), and let \(h(x)\) be a CBF for the delay-free system (2) with \(\phi\) as defined in (36) and (37). Consider the closed-loop system of (1) and the controller (40)–(42). If \(I(x), h(x), \mathcal{L}_j h(x)\) and \(\mathcal{L}_{g_i} h(x)\) are \(\phi\)-consistent at time \(t > D_{\phi}\), then

\[
\mathcal{L}_j h(x(t)) + \alpha(h(x(t))) \geq 0. \tag{44}\]

**Proof.** If \(\phi = r\), the result follow directly from Theorem 1. Now, consider the case where \(\phi < r\). By the \(\phi\)-consistency of \(I(x(t)), h(x), \mathcal{L}_j h(x)\) and \(\mathcal{L}_{g_i} h(x)\), we have that \(x(t) = p_i(t - D_i), h(x(t)) = h(p_i(t - D_i)), \mathcal{L}_j h(x(t)) = \mathcal{L}_j h(p_i(t - D_i))\) and \(\mathcal{L}_{g_i} h(x(t)) = \mathcal{L}_{g_i} h(p_i(t - D_i))\) for all \(i = 1, \ldots, \phi\), and therefore the controller (40) for \(i = 1, \ldots, \phi\) satisfies

\[
u_i(t-D_i) = u_0(x(t)) + \overline{u}_i(x(t), u_0, (x(t), \omega_{\phi+1:r}) \tag{45}\]
which can be re-written in vector form as

\[
\begin{bmatrix}
    u_1(t - D_1) \\
    \vdots \\
    u_p(t - D_p)
\end{bmatrix}
= u_0(x(t)) + \overline{u}_{1;\varphi}(x(t), u_0(x(t)), \omega_{\varphi+1};\varphi).
\]  

(46)

Thus, the system evolution at time \( t \) satisfies

\[
\dot{x}(t) = f(x(t)) + G_1;\varphi(x(t))u_1;\varphi + G_{\varphi+1};\varphi(x(t))\omega_{\varphi+1};\varphi.
\]  

(47)

By the definition of \( \varphi \), we have that the CBF \( h(x) \) is an RCBF for the system (47) with respect to the “disturbance” \( \omega_{\varphi+1};\varphi \), and since \( \overline{u}_{1;\varphi} \) solves the robust QP problem (38), it follows that

\[
h(x(t)) + \alpha(h(x(t))) \geq 0.
\]

\[\square \]

Remark 5. The result of Theorem 2, that is, the satisfaction of the barrier constraint \( h(x(t)) + \alpha(h(x(t))) \geq 0 \) holds for all \( t \geq D_\varphi \) where predictions are \( \varphi \)-consistent. If the state starts inside of the admissible set \( C \) at some time \( t^* \geq D_\varphi \) and \( \varphi \)-consistency holds for all \( t \geq t^* \), then the forward invariance of the set \( C \) from time \( t^* \) follows.

Example 1. Consider the following nonlinear system adapted from the example in the work of Janković\textsuperscript{15}

\[
\begin{align*}
\dot{x}_1(t) &= -x_1(t) - x_2(t) + u_1(t - D_1) \\
\dot{x}_2(t) &= x_1^2(t) + x_2(t)u_2(t - D_2)
\end{align*}
\]  

(48)

with \( D_1 = 0.2, D_2 = 1, u_1(\theta) = 0 \) for \( \theta \in [-0.2, 0) \), and \( u_2(\theta) = 0 \) for \( \theta \in [-1, 0) \). The control objective is to stabilize the system to the origin while keeping the set

\[
C = \{ x : h(x) = x_2^2 - x_1 + 1 \geq 0 \}
\]  

(49)

forward invariant.

First, we verify that when \( D_1 = D_2 = 0 \), the candidate barrier function \( h(x) \) in (49) is indeed a CBF for (48). Specifically, we verify that for some class \( \mathcal{K} \) function \( \alpha(h) \), \( \mathcal{L}_f h(x) = 0 \Rightarrow \mathcal{L}_f h(x) + \alpha(h(x)) \geq 0 \). From (48) and (49) we have

\[
\mathcal{L}_f h(x) = x_1 + x_2 + 2x_1^2 x_2,
\]  

(50)

\[
\mathcal{L}_g h(x) = \begin{bmatrix}
-1, & 2x_2^2
\end{bmatrix},
\]  

(51)

and it follows that \( \mathcal{L}_g h(x) \neq 0 \) for all \( x \in \mathbb{R}^n \) which implies that \( h(x) \) is indeed a CBF for (48) with no delays. For control design, we choose \( \alpha(h) = h \). Next, we determine the set \( \Phi \) and its minimum \( \varphi \) as defined in (36) and (37). For the system (48) and control barrier function (49), we have

\[
\Phi = \{ 1 \} \cup \{ 2 \}, \Rightarrow \varphi = 1.
\]  

(52)

Thus, we only need one-consistent predictions to enforce safe set invariance and safety can be enforced earlier at time \( t = D_1 \) before both input delays have been fully compensated at time \( t = D_2 \).

For the baseline controller \( u_0(x) \), we use a standard control Lyapunov function (CLF) based controller to stabilize the system to origin. Specifically, we use the Lyapunov function

\[
V(x) = \frac{1}{4} x_1^4 + \frac{1}{2} x_2^2
\]  

(53)
and baseline control

\[
u_0(x) = \begin{cases} 
0, & \mathcal{L}_g V = 0 \\
-\frac{(\mathcal{L}_V V^\top) \mathcal{L}_g V^\top}{\mathcal{L}_g V (\mathcal{L}_g V^\top)}, & \mathcal{L}_g V \neq 0
\end{cases}
\]  

which achieves

\[
\dot{V} \leq -||x||^2 \quad \text{for all } x \in \mathbb{R}^2 \setminus \{0\}
\]  

when the barrier constraint is inactive.

We run numerical simulations for two separate cases: the first using control law (30), and the second using the new control law (40). To capture the significance of the new approach, we choose initial conditions close enough to the boundary of the admissible set so that the drift of the system causes a constraint violation in \(D_1 \leq t \leq D_2\) units of time. That way, only the delay in the first input channel would have been compensated by the time the state trajectory exits the admissible set. This is reflected in Figure 1 where the system evolves uncontrolled for 0.2 time units due to the input delays, and arrives very close to the boundary of the admissible set.

At time \(t = 0.2\), the delay in the first input channel has been compensated and the designed control \(u_1\) begins to impact the system. As shown in Figure 2, under control law (30), (32), the system violates the barrier constraint because the control inputs were designed under the assumption that both inputs are always available for enforcing safety and have consistent predictions of state, which is not the case in the time interval \(0.2 \leq t < 1\). However, under the new control law (40), (42) where at all times \(t \geq 0\), we incorporate the impact of the already-determined input signal \(u_2(t - 0.8)\) when determining \(u_1(t)\), the system remains in the admissible set even though only the delay in first input channel has been compensated. The control signals under both control approaches are shown in Figure 3.

For numerical simulation we use the forward Euler method with \(\Delta t = 0.01\) for propagating system dynamics forward in time. For predictor implementation, at every time \(t\), we propagate the dynamics for \(D_1\) time units using the already-available input histories \(u_1[t - D_1, t - \Delta t]\) and \(u_2[t - D_2, t - (D_2 - D_1) - \Delta t]\) to determine the predicted state \(p_1(t)\), from which we solve for \(u_1(t)\) using the baseline feedback law and the safety modification. Next, we propagate the system dynamics from state \(p_1(t)\) forward one time-step with input vector \([u_1(t), u_2(t - (D_2 - D_1))]^\top\) to arrive at \(p_1(t + \Delta t)\) from
FIGURE 2  Left: System response for $0 \leq t \leq 1$ when only the shorter-delayed designed input $u_1$ has been delay-compensated. Under controller (40), (42) (solid blue) the system remains in the admissible set using only $u_1$, while for controller (30), (32) (dashed blue), the system exits the admissible set. Right: With both controllers, the system eventually gets stabilized to the origin once the constraint has been cleared.

FIGURE 3  Control signals $u_1(t)$ and $u_2(t)$ under controllers (30), (32) (dashed blue) and (40), (42) (solid blue).

which we compute $u_1(t + \Delta t)$. This process is repeated for $D_2 - D_1$ time steps to arrive at state $p_1(t + D_2 - D_1) = p_2(t)$ which is then used for computing the input $u_2(t)$.

3.3 Safety before longest input delay compensation using variable number of input channels

While the previous approach relaxes the need for $r$-consistent predictions for safety to only needing $\varphi$-consistent predictions, it is only beneficial when $\varphi$ is strictly less than $r$ i.e. when there exists a strict subset of input channels sufficient to impact $h$ everywhere in $\mathbb{R}^n$. When there is not a strict subset of input channels, that is, when $\varphi = r$, the controller (40)–(42) would be equivalent to controller (30)–(32) and fully consistent predictions would be required to guarantee safety. Ideally, we want the smallest subset of input channels to be utilized whenever they can enforce constraint adherence, even if they cannot be used everywhere in the safe set. To this effect, we introduce a control approach that uses a modified variant of the QP problem (35).
Consider the following QP problem denoted $\text{QP}_i^j$ for input channel $i \in \{1, \ldots, r\}$ at time $t > 0$:

$$\text{QP}_i^j : \quad \bar{u} = \arg \min_{\forall j \in \{i\} \cup \{0\}} \|v\|^2 + m\|\sigma\|^2 \quad \text{subject to}$$

$$\gamma_j \left( \sum_{j=1}^{i} \mathcal{L}_j h + a(h) + \sum_{j=i+1}^{r} \mathcal{L}_j h \left( v_j + \alpha_j \right) \right) + \sum_{j=1}^{i} \mathcal{L}_j h \left( v_j + \alpha_j \right) + \sum_{j=i+1}^{r} \mathcal{L}_j h \left( \omega_j + \sigma_j \right) \geq 0 \tag{56}$$

where $\sigma_i$ are slack variables and all functions of state $h$, $\mathcal{L}_j h$, $\mathcal{L}_j h$, and $u_0$, with suppressed arguments are evaluated at $p_i(t)$ defined in (22), $\omega_j = u_j(t - D_{j,i})$, $m \geq 1$ is a design parameter, and

$$\gamma_j(s) = \begin{cases} 
\gamma s & \text{if } s \leq 0 \\
\sigma & \text{if } s > 0 
\end{cases} \tag{57}$$

for some $\gamma \geq 1$ (another design parameter).

The notation $\text{QP}_i^j$ indicates that this QP problem is “solved” for each $i \in \{1, \ldots, r\}$ at all times $t \geq 0$, and will be useful for clarity in the analysis that follow. As previously mentioned, an explicit solution for (56) exists and no online optimization is required. The QP problem (56) attempts to find the control modification $\bar{u} \in \mathbb{R}^l$ for the first $i$ input channels that allows the barrier function to satisfy the inequality in (56) at time $t + D_i$. Notice that the inequality in (56) is not exactly the barrier constraint $\dot{h}(p_i(t)) + a(h(p_i(t))) \geq 0$ since it includes the additional term $\mathcal{L}_j h(p_i(t))\sigma$ and the function $\gamma_j$. The terms $\alpha_{i+1}, \ldots, \alpha_r$ in (56) are fictitious control inputs included to make $\text{QP}_i^j$ always feasible, while the terms $\sigma_1, \ldots, \sigma_r$ are included to ensure that when all predictions are accurate, the resulting $\bar{u} \in \mathbb{R}^l$ from $\text{QP}_i^j$, is equivalent to the first $j$ components of $\mathbb{R}^l$ from $\text{QP}_i^{j-D_{j,i}}$. This equivalence will be shown later in Lemma 2. Since the terms $\sigma_1, \ldots, \sigma_r$ are fictitious, we include the penalty $m \geq 1$ in the cost function of the QP problem. Lastly, the function $\gamma_j$ is used to overcome the impact of the fictitious control inputs by exaggerating the negative effect of the baseline control on $h(p_i(t))$. This “$\gamma m$” approach is similar to and motivated by the $\gamma m$ CLF-CBF QP problem in the work of Janković.15 Subsequent analysis will reveal that a suitable choice of $\gamma$ is $\frac{m+1}{m}$. For a fixed $m$, the larger the scalar $\gamma$ is, the more conservative the resulting controller gets.

The QP problem (56) has solution

$$\bar{u}_j = \begin{cases} 
0, & \text{if } \zeta_j(t) \geq 0 \\
-\frac{\zeta_j(t)}{\xi_j(t)} \mathcal{L}_j h(p_i(t)), & \text{otherwise} 
\end{cases} \tag{58}$$

for $j = 1, \ldots, i$ and

$$\sigma_j = \begin{cases} 
0, & \text{if } \zeta_j(t) \geq 0 \\
-\frac{1}{m} \frac{\zeta_j(t)}{\xi_j(t)} \mathcal{L}_j h(p_i(t)), & \text{otherwise} 
\end{cases} \tag{59}$$

for $j = 1, \ldots, r$ where

$$\zeta_j(t) = \gamma_j(z_j(t)) + \sum_{k=i+1}^{r} \mathcal{L}_g h(p_i(t))\omega_k(t - D_{j,k}) \tag{60}$$

$$\xi_j(t) = \| \mathcal{L}_g h(p_i(t)) \|^2 + \frac{1}{m} \| \mathcal{L}_g h(p_i(t)) \|^2 \tag{61}$$

and

$$z_j(t) := \mathcal{L}_j h(p_i(t)) + a(h(p_i(t))) + \sum_{j=1}^{i} \mathcal{L}_j h(p_i(t))u_0(p_i(t)) \tag{62}$$

The proposed control law is given as

$$u_i(t) = u_0(p_i(t)) + \bar{u}_j, \quad t \geq 0, \; i = 1, \ldots, r \tag{63}$$
For the rest of this paper, we will assume for the sake of clarity and without loss of generality that the input delays are all distinct that is, \(0 \leq D_1 < D_2 < \cdots < D_r\). With this assumption, we have \(k = i\) in (64). The theorems and proofs that follow apply with minor modifications when this assumption does not hold. In addition, since the baseline controller \(u_0\) is assumed to be Lipschitz continuous, the control modification \(\bar{u}_{\text{QP}}\) is also Lipschitz since the QP (56) is always feasible. The feasibility of (56) is guaranteed due to the presence of the fictitious inputs \(\sigma_j\) acting as slack variables. That said, the Lipschitz constant for \(\bar{u}_{\text{QP}}\) could be potentially large (order \(m\)) due to the penalty on \(\sigma\) in (56). For further discussion on the Lipschitz continuity of \(\gamma m\) QP-based controllers, we refer the reader to the work of Janković.\(^{15}\)

**Proposition 1.** If trajectories of the system remain in a compact set \(\mathcal{Y} \subset \mathbb{R}^n\), then the control input \([u_1(t), \ldots, u_r(t)]^T\) in (63) is contained in a compact set \(U'(\mathcal{Y}) \subset \mathbb{R}^r\) for all \(t \geq 0\).

**Proof.** This follows from the Lipschitz continuity of \(u_0\) and \(\bar{u}_i, i = 1, \ldots, r\). \(\blacksquare\)

**Theorem 3.** For the closed loop system of (1) and controller (63), (58), suppose trajectories of the system stay inside of a compact subset \(\mathcal{Y} \subset \mathbb{R}^n\) where the parameter \(m \geq 1\) satisfies

\[
m \geq \frac{1}{\varepsilon \delta^2} \max_{x \in \mathbb{R}^n} \left\{ -\|L_G h(y)\|^2 (L_f h(y) + a(h(y))) + L_G h(y) \nu + \varepsilon \right\},
\]

for some \(\delta > 0, \varepsilon > 0\), then the \(i\)-consistency of the identity function \(I(x)\), CBF \(h(x)\), and Lie derivatives \(L_f h(x)\) and \(L_G h(x)\) at time \(t > D_i\) for some \(i \in \{1, \ldots, r\}\) implies that

\[
\|L_{G_{i}}, h(x(t))\| > \delta \Rightarrow h(x(t)) + a(h(x(t))) > -\varepsilon.
\]

for all \(x(t) \in \mathcal{Y}\). Moreover, if \(I(x), h(x), L_f h(x)\) and \(L_G h(x)\) are \(r\)-consistent at \(t > D_r\) and \(\gamma \geq \frac{m+1}{m}\), then

\[
h(x(t)) + a(h(x(t))) \geq 0
\]

for all \(x(t) \in \mathcal{Y}\).

**Remark 6.** The second part of Theorem 3 states that after time \(t > D_r\), whenever all input delays are fully compensated as reflected by the \(r\)-consistency of \(I(x)\) and \(h(x)\) (and its Lie derivatives), the controller (63), (58) achieves the standard barrier constraint inequality (67) irrespective of the choice of \(m \geq 1\), as long as \(\gamma\) is chosen to satisfy \(\gamma \geq \frac{m+1}{m}\). In fact, since \(\frac{m+1}{m} \leq 2\), choosing \(\gamma \geq 2\) guarantees the satisfaction of (67) everywhere in \(\mathbb{R}^n\). The first part of the theorem however states that at any given time \(t > D_i\), if predictions are only \(i\)-consistent, then the first \(i\) input channels would enforce the barrier constraint as best as possible, with a possible violation dependent on the magnitude of \(\|L_{G_{i}}, h(x(t))\|\), provided that trajectories remain inside of some compact set \(\mathcal{Y}\) where the inequality (66) holds. Notice that the inverse relationship of the parameter \(m\) and \(c \delta^2\) in (65) suggests the choice of a sufficiently large penalty \(m\) to keep \(\varepsilon\) small for a fixed \(\delta\). In addition, for a fixed choice of parameter \(m\), the magnitude \(\varepsilon\) of the violation of the barrier constraint in (66) decreases as \(\delta\) – the lower bound on \(\|L_{G_{i}}, h(x(t))\|\) – increases.

**Remark 7.** The maximum on the RHS of (65) exists because it is taken over continuous functions over compact sets \(\mathcal{Y} \subset \mathbb{R}^n\) and \(U' \subset \mathbb{R}^r\). In addition, if one chooses a sufficiently large \(m\), then there exists a non-empty compact set \(\mathcal{Y}\) inside of which the inequality holds.

The proof of Theorem 3 is included in the Appendix.

**Theorem 4.** For all \(t^* \geq D_1\), if \(h(x(t^*)) \geq 0\) and if the conditions of Theorem 3 are satisfied for all \(t \in [t^*, T]\), then

\[
h(x(t)) > -a^{-1}(-\varepsilon), \quad \text{for all } t \in [t^*, T]
\]
Proof of Theorem 4. From Theorem 3, we have for all \( t \in [t^*, T] \)

\[
\dot{h}(x(t)) + a(h(x(t))) > -\varepsilon.
\]  

We apply the comparison lemma from the work of Khalil\(^{32}\) on the initial value problem

\[
\dot{y}(t) = -a(y(t)) - \varepsilon, \quad y(t^*) = h(x(t^*)).
\]

Since \( y(t^*) > 0 \), we have that \( y \) decreases monotonically until \( \dot{y} = 0 \). Therefore, it follows that

\[
y(t) \geq a^{-1}(-\varepsilon)
\]

By the comparison lemma, we have that

\[
h(x(t)) > a^{-1}(-\varepsilon)
\]

Example 2. Consider the following kinematic bicycle model of a vehicle for a circular robot (the host)

\[
\begin{align*}
\dot{x}_1 &= x_4(t) \cos(x_3(t)) \\
\dot{x}_2 &= x_4(t) \sin(x_3(t)) \\
\dot{x}_3 &= \frac{x_4}{L} u_1(t - D_1), \quad \delta(t) = \arctan(u_1(t)) \\
\dot{x}_4 &= u_2(t - D_2)
\end{align*}
\]

where \((x_1, x_2) \in \mathbb{R}^2\) is the position of the host, \(x_3 \in [0, 2\pi)\) is its orientation, \(x_4 \in \mathbb{R}\) is the longitudinal velocity, and \(L = 6\) is the distance between its front and rear axles. The input \(u_1\) is the tangent of the steering wheel angle \(\delta\), and \(u_2\) is the acceleration/deceleration input. The input channels have delays \(D_1\) and \(D_2\) respectively with \(0 < D_1 < D_2\) and have initial values

\[
\begin{align*}
u_1(\theta) &= 0, \quad \text{for} \quad \theta \in [-D_1, 0), \\
u_2(\theta) &= 0, \quad \text{for} \quad \theta \in [-D_2, 0).
\end{align*}
\]

The control objective is to keep the host traveling at a reference velocity \(\bar{x}_4\) along a prescribed path in the \((x_1, x_2)\)-plane while avoiding obstacles introduced in its path.

For collision avoidance, we use the following candidate barrier function

\[
h(x) = (x_1 - x_1^{obs})^2 + (x_2 - x_2^{obs})^2 - r^2
\]

where \((x_1^{obs}, x_2^{obs})\) is the position of the obstacle and \(r\) is the sum of the radii of the host and the obstacle. Keeping \(h(x) \geq 0\) therefore ensures that a collision does not occur. Note that the candidate barrier function \((76)\) has a relative degree of 2 from the inputs and as such the inputs do not appear in the barrier constraint \(\dot{h} + a(h) \geq 0\). To circumvent this, we use the barrier constraint formulation with relative degree higher than one, introduced in the non-overshooting control work by Krstic and Bement,\(^7\) and independently discovered a decade later by Xu\(^{13}\) and Sreenath and Nguyen.\(^{34}\) Given that the CBF \((76)\) is of relative degree two, we get

\[
\dot{h} + l_1\dot{h} + l_0h \geq 0,
\]

\[
L_f^2 h(x) + L_G L_f h(x) u + l_1 L_f h(x) + l_0 h(x) \geq 0
\]

where \(l_0\) and \(l_1\) are constants chosen so that the roots of the polynomial \(s^2 + l_1s + l_0 = 0\) are negative real. The idea being that this allows \(h(x)\) to decrease towards zero but not oscillate around zero as the case would be for a polynomial with complex roots. The consequences of using this higher relative degree barrier constraint are the following
1. In addition to ensuring forward invariance of the set \( C = \{ x \mid h(x) \geq 0 \} \), satisfying the barrier constraint (78) also ensures \( h(x) \geq \frac{1}{|\lambda|_{\text{max}}} \hat{h}(x) \) where \( |\lambda|_{\text{max}} \) is the magnitude of the more negative root of the characteristic equation \( s^2 + l_1 s + l_0 = 0 \). Therefore, satisfying (78) guarantees the forward invariance of

\[
C_2 = \left\{ x \mid h(x) \geq 0, \quad h(x) + \frac{1}{|\lambda|_{\text{max}}} \hat{h}(x) \geq 0 \right\} \subseteq C
\]  

(79)

With the barrier constraint (78), the results of this paper apply with the following substitutions made in the constraint of all QP problems considered:

\[
\mathcal{L}_{f}^2 h + l_1 \mathcal{L}_f h + l_0 h \to \mathcal{L}_f h + a(h), \quad \text{and} \quad \mathcal{L}_G \mathcal{L}_f h \to \mathcal{L}_G h. 
\]

(80)

Next, we verify that the candidate barrier function (76) is indeed a CBF for (2) that is, we want to show that

\[
\mathcal{L}_G \mathcal{L}_f h = 0 \Rightarrow \mathcal{L}_f^2 h + l_1 \mathcal{L}_f h + l_0 h \geq 0.
\]

(81)

We have

\[
\mathcal{L}_f h(x) = 2(x_1 - x_1^{\text{obs}})x_4 \cos(x_1) + 2(x_2 - x_2^{\text{obs}})x_4 \sin(x_1)
\]

(82)

\[
\mathcal{L}_{f}^2 h(x) = 2x_4^2
\]

(83)

\[
\mathcal{L}_G \mathcal{L}_f h(x) = \begin{bmatrix}
-\frac{2}{L}(x_1 - x_1^{\text{obs}})x_4^2 \sin(x_1) + \frac{2}{L}(x_2 - x_2^{\text{obs}})x_4^2 \cos(x_1) \\
2(x_1 - x_1^{\text{obs}}) \cos(x_1) + 2(x_2 - x_2^{\text{obs}}) \sin(x_1)
\end{bmatrix}^T
\]

(84)

It follows from (82) and (84) that \( \mathcal{L}_f h(x) = x_4 \mathcal{L}_G \mathcal{L}_f h(x) \), and therefore \( \mathcal{L}_G \mathcal{L}_f h = 0 \Rightarrow \mathcal{L}_f h = 0 \Rightarrow \mathcal{L}_f^2 h + l_1 \mathcal{L}_f h + l_0 h = 2x_4^2 + l_0 h \geq 0 \) for any \( l_0 > 0 \), and thus (76) is indeed a control barrier function for (73) with no delays.

For the baseline control \( u_0 \), we linearize the dynamics about a desired configuration \([x_1^{\text{des}}, x_2^{\text{des}}, x_3^{\text{des}}, x_4^{\text{des}}]^T\) along the the prescribed path to be followed, and then we design an LQR controller to stabilize the host to this configuration. Once the baseline control input is computed, it is applied unmodified if no obstacle is present in the host's path. If an obstacle is present, we modify the baseline control using two predictor-based safety approaches for comparison. In the first case, we use the controller \((63), (58)\) where the longer delayed acceleration input \( u_2 \) is considered unavailable for safety-enforcement when determining the shorter delayed steering input \( u_1 \). In the second case, we use controller \((30), (32)\) where two-consistency is implicitly assumed and both inputs are always considered available for safety enforcement. Notice that for this example, \( \phi = 2 \) (as defined in (37)) since \( \mathcal{L}_G \mathcal{L}_f h(x) = 0 \) when \( x_4 = 0 \). This is supported by intuition because the steering input alone cannot affect the distance of a circular agent to a circular obstacle if the velocity \( x_4 \) is zero. Thus, the controller \((40), (42)\) is equivalent to \((30), (32)\) and is therefore not considered separately. The following parameter values are used for the simulation: \( D_1 = 0.2, D_2 = 1.5, L = r = 6, l_0 = l_1 = 4, m = 50, \gamma = \frac{m+1}{m}, \bar{x}_4 = 3, \) and \([x_1^{\text{obs}}, x_2^{\text{obs}}] = (-3, 0.5)\). The numerical simulations and predictor implementation was done similarly as in Example 1 with \( \Delta t = 0.01 \).

Figure 4 shows the initial configuration of the host and the reference path to be followed. The host travels at a constant velocity for the first 2.75 time units before an obstacle gets introduced in its path. For the purpose of safety enforcement, the time \( t = 2.75 \) is the “initial time”, and any steering input \( u_1 \) applied at this time with knowledge of the obstacle will not arrive at the plant until time \( t = 2.75 + D_1 = 2.95 \). Similarly for the acceleration braking input \( u_2 \), any new control signal issued will not arrive at the plant until \( t = 2.75 + D_2 = 4.25 \). Thus, for the purpose of safety, the delay is not fully compensated until time \( t = 4.25 \) and only becomes partially compensated at time \( t = 2.95 \).

As shown in the Figure 5, under controller \((63), (58)\), the host clears the obstacle without collision using the shorter-delayed control input preferentially. Under controller \((30), (32)\) however, a collision occurs as shown in Figure 6. For both controllers, we include a plot of the input signals in Figure 9, and a plot of barrier function \( h \), and the function \( h + \frac{1}{|\lambda|_{\text{max}}} \hat{h} \) (cf., (79)) in Figures 7 and 8.
FIGURE 4  Left: initial configuration of the host (solid blue) with heading (green) along prescribed path (dotted blue). Center: Host accelerates to reference velocity. Since path is obstacle-free, baseline control is applied unmodified. Right: Obstacle (solid red) introduced in path of the host.

FIGURE 5  Obstacle avoidance under controller (63), (58). Here, the host does not collide with obstacle (see barrier function plot in Figure 8).

FIGURE 6  Obstacle avoidance under controller (30), (32) where collision occurs (see barrier function plot in Figure 8).

\[ h(x(t)) + \frac{1}{\lambda_{max}} \dot{h}(x(t)) \]

FIGURE 7  CBF value under controller (63), (58) (blue) and controller (30), (32) (red). The CBF \( h(x) \) is defined from the moment the obstacle appears at time \( t = 2.75 \). See Figure 8 for corresponding scaled-up plots emphasizing function behavior in the neighborhood of zero.
CONCLUSION

We studied the problem of enforcing safety in multi-input nonlinear systems with distinct input delays using predictor feedback. In particular, we presented two approaches that preferentially use shorter-delayed inputs for enforcing safety before all input delays have been compensated. Compared to a naive combination of state-predictors and a nominal safe-guarding controller for the delay free system where safety-guarantees cannot be made until after the longest input delay has been compensated, we showed that the two introduced methods enforce safety before the longest input delay has been compensated whenever it is possible to do so. We included illustrative examples to demonstrate the performance of the two introduced control methods. One possible direction for future work is the study of safety in the presence of distinct input delays that are non-constant and/or unknown a priori, a problem that is relevant in many practical systems.

ACKNOWLEDGMENTS

This work was supported by the National Science Foundation (Grant no. ECCS-2151525) and Office of Naval Research (Grant no. N00014-23-1-2376).

CONFLICT OF INTEREST STATEMENT

All authors declare that they have no conflicts of interest.
DATA AVAILABILITY STATEMENT
Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

ORCID
Imoleayo Abel https://orcid.org/0000-0003-0018-982X

REFERENCES
5. Tee KP, Ge SS, Tay EH. Barrier Lyapunov functions for the control of output-constrained nonlinear systems. Automatica. 2009;45:918-927.
APPENDIX

The proof of Theorem 3 is comprised of the following Lemmas.

Lemma 1. Let $\bar{u} \in \mathbb{R}^b$ be the unique solution of

$$
\bar{u} = \arg \min_{w \in \mathbb{R}^b} ||w||^2 \quad \text{subject to} \quad \beta^T w \geq c
$$

for some $\beta \in \mathbb{R}^b$, $\beta \neq 0$ and $c \in \mathbb{R}$, then the vector $\bar{v} \in \mathbb{R}^a$, $a < b$ that uniquely solves

$$
\bar{v} = \arg \min_{w \in \mathbb{R}^a} ||w||^2 \quad \text{subject to} \quad \beta_1^T \cdot \bar{v} + \beta_{a+1}^T \cdot \bar{u}_{a+1:b} \geq c
$$

satisfies

$$
\bar{v} = \bar{u}_{1:a}
$$

Proof. The proof follows from Bellman’s principle of optimality. 35

Lemma 2. If the identity function $I(x) = x$, $\text{CBF } h(x)$, and the Lie derivatives $\mathcal{L}_f h(x)$ and $\mathcal{L}_\gamma h(x)$ are $i$-consistent at time $t$, then for all $j = 1, \ldots, i$,

$$
\bar{u}_j^{Q^{i-1}j} = \bar{u}_j^{Q^{i-1}j}
$$

Proof. Define

$$
\phi := \left[ \frac{\sigma_1}{\sqrt{m}}, \ldots, \frac{\sigma_r}{\sqrt{m}}, \bar{u}_1, \ldots, \bar{u}_i \right]^T,
$$

$$
\nu := \left[ \frac{\sigma_1}{\sqrt{m}}, \ldots, \frac{\sigma_r}{\sqrt{m}}, \bar{u}_1, \ldots, \bar{u}_j \right]^T,
$$

and let

$$
c = -\gamma_f \left( \mathcal{L}_f h + \alpha(h) + \sum_{k=1}^{i} \mathcal{L}_{\gamma_k} h u_0 \right) - \sum_{k=i+1}^{r} \mathcal{L}_{\gamma_k} h \omega_k,
$$

$$
\beta = \left[ \sqrt{m} \mathcal{L}_{\gamma_1} h, \ldots, \sqrt{m} \mathcal{L}_{\gamma_i} h, \mathcal{L}_{\gamma_i} h, \ldots, \mathcal{L}_{\gamma_r} h \right]^T.
$$
where functions $h, \mathcal{L}_j h, \mathcal{L}_g h, u_{t_k}$ have suppressed argument $x(t)$, and $\omega_k = \omega_k(t - D_{k, i})$; then QP$_{t-D}$ in (56) can be written as

$$\arg \min_{\phi} \| \phi \|^2 \quad \text{subject to} \quad \hat{\beta}^T \phi \geq c.$$  \hspace{1cm} (A5)

Also, QP$_{t-D}$ can be written as

$$\arg \min_{v} \| v \|^2 \quad \text{subject to} \quad \hat{\beta}^T_{1:r+j} v + \beta^T_{r+j+1:r+i} \phi_{r+j+1:r+i} \geq c$$  \hspace{1cm} (A6)

where $\phi^*$ is the solution of (A5).

With the representations (A5), (A6), the rest of the proof follow from Lemma 1.

**Remark 8.** Lemma 2 states that the $i$-consistency of $I(x), h(x), \mathcal{L}_j h(x)$ and $\mathcal{L}_g h(x)$ at time $t$ implies that the $j$-th component of the solution of QP$_{t-D}$ is equivalent to the $j$-th component of the solution of QP$_{t-D}$ for $j < i$. This is intuitive since $i$-consistency implies that the predicted states, barrier function values, and Lie derivatives used in QP$_{t-D}$ and QP$_{t-D}$ are equivalent.

**Lemma 3.** For all $t > D_r$ and $\gamma \geq \frac{m+1}{m}$, if $I(x), h(x), \mathcal{L}_j h(x)$ and $\mathcal{L}_g h(x)$ are $r$-consistent at time $t$, then the controller (63), (58) achieves

$$\dot{h}(x(t)) + a(h(x(t))) \geq 0.$$  \hspace{1cm} (A7)

**Proof.** In what follows, we suppress the argument of $h(x(t))$ for brevity. When $h$ appears without an argument, it refers to $h(x(t))$. Now, for all $t > D_r$, it follows from Lemma 2 that

$$\dot{h}(x(t)) + a(h(x(t))) = z_r(t - D_r) + \sum_{j=1}^{r} \mathcal{L}_g h \hat{u}^{QP}_{j - D_r}$$  \hspace{1cm} (A8)

where $z_r(\cdot)$ is as defined in (62).

Now we consider the two cases for $\hat{u}^{QP}_{j - D_r}$ in (58).

**Case 1:** $\zeta_j(t - D_r) = \gamma_j(z_r(t - D_r)) \geq 0$.

Here, $\hat{u}^{QP}_{j - D_r} = 0$ for all $j = 1, \ldots, r$. In addition, $\gamma_j(z_r(t - D_r)) \geq 0 \Rightarrow z_r(t - D_r) \geq 0$. Therefore, (A8) becomes

$$\dot{h}(x(t)) + a(h(x(t))) = z_r(t - D_r) \geq 0$$  \hspace{1cm} (A9)

**Case 2:** $\zeta_j(t - D_r) = \gamma_j(z_r(t - D_r)) < 0$.

Here, $\gamma_j(z_r(t - D_r)) < 0 \Rightarrow z_r(t - D_r) < 0$ and $\gamma_j(z_r(t - D_r)) = \gamma z_r(t - D_r)$.

Also, by $r$-consistency of $I(x), h(x), \mathcal{L}_j h(x)$ and $\mathcal{L}_g h(x)$ (A8) becomes

$$\dot{h}(x(t)) + a(h(x(t))) = z_r(t - D_r) - \sum_{j=1}^{r} \mathcal{L}_g h \left( \frac{\gamma z_r(t - D_r)}{m+1} \right) \mathcal{L}_g h$$

$$= z_r(t - D_r) - \frac{\gamma z_r(t - D_r)}{m+1} \left( \sum_{j=1}^{r} \mathcal{L}_g h \right)^2$$  \hspace{1cm} (A12)

$$= z_r(t - D_r) \left( 1 - \frac{\gamma}{m+1} \right)$$  \hspace{1cm} (A13)
Lemma 4. If \( m \) satisfies (65) for some \( \delta > 0, \epsilon > 0 \), then the i-consistency of \( I(x), h(x), L_h(x) \) and \( L_C h(x) \) at \( t > D_i \) for some \( i \in \{1, \ldots, r\} \) implies (66).

Proof. For all \( t > D_i \)

\[
\dot{h}(x(t)) + a(h(x(t))) = L_f h + a(h) + \sum_{j=1}^{r} L_{g_j} h u_j(t - D_j) = L_f h + a(h) + \sum_{j=1}^{i} L_{g_j} h \left( u_0(p_j(t - D_j)) + \tilde{u}_j^{Q_{p_j}(\cdot)} \right) + \sum_{j=i+1}^{r} L_{g_j} h u_j(t - D_j) \tag{A16}
\]

\[
= L_f h + a(h) + \sum_{j=1}^{i} L_{g_j} h \left( u_0(p_j(t - D_j)) + \tilde{u}_j^{Q_{p_j}(\cdot)} \right) + \sum_{j=1}^{r} L_{g_j} h u_j(t - D_j) \tag{A17}
\]

\[
= z_i(t - D_i) + \sum_{j=1}^{i} L_{g_j} h u_j^{Q_{p_j}(\cdot)} + \sum_{j=1}^{r} L_{g_j} h u_j(t - D_i - D_{j,i}) \tag{A18}
\]

The equality in (A17) uses Lemma 2, and the equality in (A18) uses the definition of \( z_i(\cdot) \) in (62). Now, consider the two cases for \( \tilde{u}_j^{Q_{p_j}(\cdot)} \) in (58).

Case 1: \( \zeta_i(t - D_j) = \gamma_f(z_i(t - D_i)) + \sum_{k=1}^{r} L_{g_k} h(p_k(t - D_i)) \omega_k(t - D_i - D_{k,i}) \geq 0 \).

Here, \( \tilde{u}_j^{Q_{p_j}(\cdot)} = 0 \) for all \( j \) and (A18) becomes

\[
\dot{h}(x(t)) + a(h(x(t))) = z_i(t - D_i) + \sum_{j=i+1}^{r} L_{g_j} h u_j(t - D_i - D_{j,i}) \tag{A19}
\]

\[
\geq \gamma_f(z_i(t - D_i)) + \sum_{j=i+1}^{r} L_{g_j} h u_j(t - D_i - D_{j,i}) \tag{A20}
\]

\[
= \zeta_i(t - D_i) \tag{A21}
\]

\[
\geq 0 \tag{A22}
\]

where the equality in (A21) uses the the i-consistency of \( I(x(t)), h(x(t)), L_f h(x(t)) \) and \( L_C h(x(t)) \).

Case 2: \( \zeta_i(t - D_j) = \gamma_f(z_i(t - D_i)) + \sum_{k=1}^{r} L_{g_k} h(p_k(t - D_i)) \omega_k(t - D_i - D_{k,i}) < 0 \).

Here, (A18) becomes

\[
\dot{h}(x(t)) + a(h(x(t))) = z_i(t - D_i) - \frac{\zeta_i(t - D_j)}{\zeta_i(t - D_i)} \sum_{j=1}^{i} \left( L_{g_j} h \right)^2 + \sum_{j=i+1}^{r} L_{g_j} h(x(t)) \omega_j(t - D_i - D_{j,i}) \tag{A23}
\]

\[
= z_i(t - D_i) - \frac{\|L_{g_{1,i}} h(p_1(t - D_i))\|^2}{\zeta_i(t - D_i)} \zeta_i(t - D_i) + \sum_{j=i+1}^{r} L_{g_j} h(p_k(t - D_i)) \omega_k(t - D_i - D_{j,i}) \tag{A24}
\]

\[
= z_i(t - D_i) + \eta_i(t - D_i) - \rho_i(t - D_i) \zeta_i(t - D_i) \tag{A25}
\]

Now, consider the inequality in (65). It follows that

\[
m \geq -\frac{\|L_{g_{i}} h(p_i(t - D_i))\|^2}{\epsilon \delta^2} (z_i(t - D_i) + \eta_i(t - D_i) + \epsilon) \tag{A26}
\]

\[
m \geq -\frac{\|L_{g_{i}} h(p_i(t - D_i))\|^2}{\delta^2} \frac{(z_i(t - D_i) + \eta_i(t - D_i) + \epsilon)}{\epsilon + z_i(t - D_i) - \gamma_f(z_i(t - D_i))} \tag{A27}
\]
since \( m > 0 \) and \( s - \gamma_f(s) \geq 0 \) for all \( s \). For brevity, we suppress the arguments of \( L_G h, z_i, \) and \( \eta_i \). We have

\[
\frac{m\delta^2}{\|L_G h\|^2} (\varepsilon + z_i - \gamma_f(z_i)) \geq -(z_i + \eta_i + \varepsilon)
\]

(A28)

\[
\frac{m\delta^2}{\|L_G h\|^2} \gamma_f(z_i) - z_i \geq z_i + \eta_i + \varepsilon
\]

(A29)

since \( \gamma_f(z(t - D_i)) + \eta_i(t - D_i) = \zeta_i(t - D_i) < 0 \). Simplifying further, we get

\[
\delta^2 \left( 1 - \frac{z_i + \eta_i + \varepsilon}{\gamma_f(z_i) + \eta_i} \right) \geq \frac{\|L_G h\|^2}{m} \frac{z_i + \eta_i + \varepsilon}{\gamma_f(z_i) + \eta_i}
\]

(A30)

\[
\frac{\delta^2}{\delta^2 + \frac{\|L_G h\|^2}{m}} \geq \frac{z_i + \eta_i + \varepsilon}{\gamma_f(z_i) + \eta_i}
\]

(A31)

Note that \( \rho_i = \frac{\|L_{G_i} h\|^2}{\|L_{G_i} h\|^2 + \frac{1}{m} \|L_G h\|^2} > \frac{\delta^2}{\delta^2 + \frac{\|L_G h\|^2}{m}} \) if \( \|L_{G_i} h\| > \delta > 0 \). Therefore, we have

\[
\rho_i(t - D_i) > \frac{z_i(t - D_i) + \eta_i(t - D_i) + \varepsilon}{\gamma_f(z_i(t - D_i)) + \eta_i(t - D_i)}
\]

(A32)

\[
= \frac{z_i(t - D_i) + \eta_i(t - D_i) + \varepsilon}{\zeta_i(t - D_i)}
\]

(A33)

Substituting (A33) into (A25) gives

\[
\dot{h}(x(t)) + a(h(x(t))) > z_i(t - D_i) + \eta_i(t - D_i)
\]

\[
- \frac{z_i + \eta_i + \varepsilon}{\zeta_i(t - D_i)} \zeta_i(t - D_i)
\]

(A34)

\[
> -\varepsilon
\]

(A35)

The proof of Theorem 3 follows from Lemmas 2, 3 and 4.