doi: 10.82124/CEAS-GNC-2026-010 CEAS-GNC-2026-010

Control Barrier Functions on a Pseudo Control Level for Safe Flight Control

Simon Hafner Research Associate, Institute of Flight System Dynamics, Technical University of Munich, Garching, Germany. simon.hafner@tum.de

Johannes Autenrieb Research Scientist, Department of Flight Dynamics and Simulation, Institute of Flight Systems, DLR (German Aerospace Center), Braunschweig, Germany. Johannes.Autenrieb@dlr.de

Agnes Steinert Postdoctoral Researcher, Institute of Flight System Dynamics, Technical University of Munich, Garching, Germany. agnes.steinert@tum.de

Florian Holzapfel Professor, Institute of Flight System Dynamics, Technical University of Munich, Garching, Germany. florian.holzapfel@tum.de

ABSTRACT

Control Barrier Function (CBF) have gained popularity in ensuring the safety of systems with complex control laws, specifically outside the aerospace community. This paper proposes an approach for adding CBF to a pseudo control-based control law to ensure, e.g., envelope protections of aircraft. Applying the safety filter at the pseudo control level rather than the control input level offers several advantages for applicability in the aerospace domain, including reduced execution time. The proposed algorithm is validated in simulation using a linear short-period ADMIRE fighter and a linear hexacopter model. The execution time of the three approaches is compared using a microcontroller.

Keywords: Control Barrier Function, Envelope Protections, Flight Control, Incremental Nonlinear Dynamic Inversion

1 Introduction

In recent years, the Control Barrier Function (CBF) in the form of safety filters has gained significant popularity in the controls community [1]. Typical fields of application are providing safety for "black-box" controllers like neural networks or as collision avoidance strategies [2, 3]. In the area of flight control, their main fields of application are guidance tasks such as geo-fencing and flight envelope protection [4, 5]. CBF have been used in combination with Incremental Nonlinear Dynamic Inversion (INDI) controllers to realize an angle-of-attack protection by [6–8]. Monitoring a control law for unsafe commands is called run-time assurance. This run-time assurance can be implemented using control barrier functions [9].

Safety is a major concern in aerospace systems, which are often considered safety-critical systems. Even though the CBF can be used to guarantee the safety of complex control laws, CBFs are less widespread in the aerospace domain compared to other fields. However, another important aspect of theoretical safety properties is the algorithm's and its implementation's certifiability. The complexity of the underlying optimization problem in the CBF safety filter can prevent certification. Two main aspects

are significant here: The worst-case execution time and the convergence guarantees. Both are issues for standard CBF approaches.

In over-actuated aerospace systems, a widespread control approach separates the control law into an outer controller and a control allocation. The outer control law provides desired pseudo controls to the control allocation. The control allocation then allocates the pseudo controls to actual control inputs while considering the limits of the actuators. This is then called constrained control allocation. The resulting optimization problem is a box-constrained optimization problem for simple control input limits.

In recent years, the increasing use of complex aerial systems with highly redundant and heterogeneous actuator configurations has gained significant attention in the flight control community. Examples include modern transport and fighter aircraft with multiple control surfaces, as well as multirotor and hybrid aerial vehicles with large numbers of actuators [10, 11]. As a consequence, control allocation has become an important and well-established topic in flight control, particularly for systems, where actuator redundancy can be exploited to achieve performance and robustness objectives [12].

Control allocation is a widely used concept in overactuated aerospace systems such as aircraft and multicopters. In such systems, desired forces or moments can be generated by multiple combinations of actuator inputs due to a rectangular control effectiveness matrix [13]. As a result, control design is often performed on a pseudo-control level, where a performance-oriented controller generates abstract pseudo-control commands, such as required forces or moments [14]. While this separation is highly effective for performance control, many aerospace systems are subject to additional safety-critical constraints. Typical examples include flight envelope protection for fixed-wing aircraft, where limits on angle of attack or load factors must be enforced, as well as attitude or velocity limits for multirotor vehicles [4, 15]. These safety constraints are commonly expressed as bounds on system states or outputs and must be respected under all operating conditions [5, 16].

In recent years, CBFs have gained increasing attention in control theory as a systematic framework for enforcing such safety constraints. CBFs encodes safety requirements by defining a safe set and ensuring its forward invariance, thereby guaranteeing that the system state remains within admissible bounds. In practice, this concept is often implemented as a QP-based min-norm safety filter that modifies the control input generated by a nominal controller only when safety violations are imminent [1, 5].

Two approaches to apply the CBF are common in the literature. The most common one is to directly apply the safety filter at the control input level. The other approach is to integrate the safety filter with the control allocation in one optimization problem [17]. This concept, referred to as CBF-based control allocation, has been proposed in [17].

While this unified formulation avoids redundant consideration of actuator limits, it significantly increases the complexity of the optimization problem due to the additional linear inequality constraints introduced by the CBF. As a consequence, standard control allocation algorithms such as Redistributed Scaled Pseudoinverse (RSPI) or Sequential Least Squares Active Set (SLS-AS), which are tailored to box-constrained problems, cannot be applied directly and must be replaced by general-purpose quadratic programming solvers. This raises concerns regarding execution time and predictability, which are particularly critical when the optimization is executed on the control input level, where hard real-time guarantees are required, especially for potentially open-loop unstable systems. Furthermore, ensuring feasibility and solution existence for such combined optimization problems across all operating conditions remains challenging and remains an open issue.

One possible approach to addressing the challenges posed by the combined treatment of overactuation and safety constraints is to simplify and decompose the underlying problem. By separating the handling of performance objectives, actuator constraints, and safety requirements across different control levels, the overall optimization complexity can be reduced while retaining the desirable properties of each approach. Moreover, careful control system design decisions can mitigate potential negative effects,

such as performance degradation or feasibility issues, and limit their impact to an acceptable level. This perspective motivates the approach proposed in this work, which integrates CBFs into the control architecture in a structured manner that preserves real-time feasibility and robustness while maintaining safety guarantees.

The core contribution of this paper is a CBF-based safety filter at the pseudo control level before control allocation. This approach offers reduced computational complexity while maintaining identical safety performance. The evaluation using two models and execution time measurements on a microcontroller confirms the contribution.

2 Control Barrier Functions – Background

2.1 Nonlinear Dynamic Inversion

Let's consider the following overactuated input-affine nonlinear system

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{G}(\boldsymbol{x})\boldsymbol{u}, \tag{1a}$$

$$\boldsymbol{y} = \boldsymbol{z}(\boldsymbol{x}), \tag{1b}$$

where $\boldsymbol{x} \in \mathbb{R}^{n_x}$ is the state vector, $\boldsymbol{u} \in \mathbb{R}^m$ is the control input vector, and $\boldsymbol{y} \in \mathbb{R}^n$ is the output vector with $m > n$, as well as $f : \chi \to \mathbb{R}^{n_x}$, $g : \chi \to \mathbb{R}^{n_x \times m}$, and $z : \chi \to \mathbb{R}^n$ are sufficiently smooth functions.

The main principle of feedback linearization, often referred to in the flight control community as Nonlinear Dynamic Inversion (NDI), is to linearize the input–output relation of a nonlinear input-affine system by means of exact state feedback and coordinate transformation rather than Jacobian linearization. Consequently, the dynamics of the controlled outputs are transformed into chains of integrators, enabling the application of linear control design techniques [18].

By introducing a suitable feedback-linearizing coordinate transformation for the system in (1), the input–output dynamics can be rewritten in terms of the pseudo-control variables as

$$\boldsymbol{\nu} = \boldsymbol{f}_\nu(\boldsymbol{x}) + \underbrace{\boldsymbol{G}_\nu(\boldsymbol{x})\boldsymbol{u}}_{\boldsymbol{\tau}}, \tag{2}$$

where $\boldsymbol{f}_\nu(\boldsymbol{x})$ denotes the transformed drift vector field and $\boldsymbol{G}_\nu(\boldsymbol{x})$ the associated control effectiveness matrix. The vector $\boldsymbol{\tau} \in \mathbb{R}^n$ represents the virtual input and is defined as $\boldsymbol{\tau} = \boldsymbol{G}_\nu(\boldsymbol{x})\boldsymbol{u}$, i.e., the control-dependent component of the pseudo-control.

The vector $\boldsymbol{\nu} \in \mathbb{R}^n$ is the pseudo control vector, which is defined as

$$\boldsymbol{\nu} = \begin{bmatrix} {}^{(r_1)}y_1 & {}^{(r_2)}y_2 & \cdots & {}^{(r_n)}y_n \end{bmatrix}^T, \tag{3}$$

where ${}^{(r_i)}y_i$ denotes the r_i-th derivative of the output y_i [19, p. 267]. Therefore, r_i represents the relative degree of the output y_i. The desired pseudo-control input $\boldsymbol{\tau}_{\text{des}}$ can be generated via feedback linearization using standard linear control design methods, such as PID or Linear Quadratic Regulator (LQR).

A control law corresponding to the desired pseudo-control $\boldsymbol{\tau}_{\text{des}}$ is obtained as

$$\boldsymbol{u} = \boldsymbol{G}_\nu^{-1}(\boldsymbol{x})\underbrace{(\boldsymbol{\nu} - \boldsymbol{f}_\nu(\boldsymbol{x}))}_{\boldsymbol{\tau}_{\text{des}}}, \tag{4}$$

provided that $\boldsymbol{G}_\nu(\boldsymbol{x})$ is square and nonsingular. In many flight control applications, however, a unique mapping from $\boldsymbol{\tau}_{\text{des}}$ to the actual control inputs $\boldsymbol{u}$ does not exist. In particular, flight vehicles are often overactuated, i.e., $m > n$, which leads to a non-square control effectiveness matrix $\boldsymbol{G}_\nu(\boldsymbol{x})$. In such cases, the determination of $\boldsymbol{u}$ is formulated as a control allocation problem, where the desired pseudo-control $\boldsymbol{\tau}_{\text{des}}$ is distributed among the available actuators while respecting actuator constraints.

2.2 Control Allocation

In general, the control allocation problem can be interpreted as a constrained convex optimization problem, since magnitude constraints on each available control effector $u_i(t)$ must be considered. We define the following admissible set of control inputs $\mathbb{U}$:

$$\mathbb{U} := \{\boldsymbol{u} \in \mathbb{R}^m \mid \underline{\boldsymbol{u}} \leq \boldsymbol{u} \leq \overline{\boldsymbol{u}}\}. \tag{5}$$

where $\underline{\boldsymbol{u}}$ and $\overline{\boldsymbol{u}}$ are the lower and upper actuator bounds, respectively. Further, for simplicity of notation, the control effectiveness matrix $\boldsymbol{G}_\nu(x)$ is from now on referred to as $\boldsymbol{B}$. The control allocation problem is typically formulated as

$$\begin{aligned} \boldsymbol{u}_{\text{cmd}} = \arg\min_{\boldsymbol{u}} &\|\boldsymbol{B}\boldsymbol{u} - \boldsymbol{\tau}_{\text{des}}\|_2^2 \\ \text{s.t. } &\underline{\boldsymbol{u}} \leq \boldsymbol{u} \leq \overline{\boldsymbol{u}}, \end{aligned} \tag{6}$$

where $\|\cdot\|_2$ represents the ℓ_2-norm and $\boldsymbol{\tau}_{\text{des}}$ is the desired pseudo control vector provided by the performance-oriented controller. This formulation allows the control law to focus on performance objectives on the pseudo-control level, while the overactuation and actuator constraints are handled separately by the control allocation.

2.3 Control Barrier Functions

If applied to flight control applications, the standard integration of CBF-based safety filters would enforce system safety at the control command level after a potential control allocation system and directly before the actuators. At this stage, the safety filter enforces model-based inequality constraints derived from the CBF formulation in addition to the box constraints imposed by the actuator limits. The goal of the safety filter is to minimize the deviation between the allocated control input and a safe control input while satisfying all constraints. With respect to the considered class of overactuated vehicle, an example of a corresponding control structure is illustrated in Fig. 1.

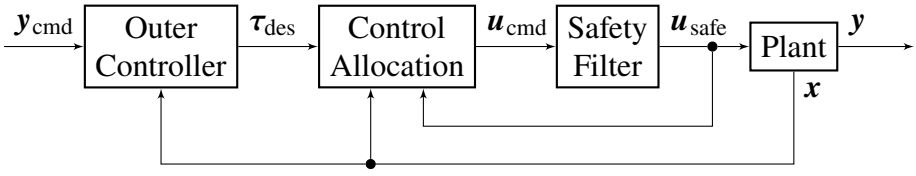

Fig. 1 Block diagram of a closed-loop control structure using a standard CBF-based safety filter on control input level after the control allocation.

The safety filter modifies the control input commanded by the control allocation by solving

$$\begin{aligned} \boldsymbol{u}_{\text{safe}} = \arg\min_{\boldsymbol{u}} &\|\boldsymbol{u} - \boldsymbol{u}_{\text{cmd}}\|_2^2 \\ \text{s.t. } &\boldsymbol{A}_{\text{u}}\boldsymbol{u} \leq \boldsymbol{b}_{\text{u}}, \\ &\underline{\boldsymbol{u}} \leq \boldsymbol{u} \leq \overline{\boldsymbol{u}}, \end{aligned} \tag{7}$$

where $\boldsymbol{u}_{\text{cmd}}$ is the control input vector provided by the control allocation and $\boldsymbol{A}_{\text{u}}$ and $\boldsymbol{b}_{\text{u}}$ are obtained from the CBF formulation and the system dynamics. For overactuated systems, this approach may lead to significant deviations from the desired pseudo control, as the safety filter operates independently of the control allocation and can substantially alter the control inputs, potentially degrading closed-loop performance.

An alternative approach is to combine control allocation and safety filtering into a single optimization problem. Since both problems can be formulated as constrained convex optimization problems, a unified quadratic program can be constructed that simultaneously enforces actuator limits and safety constraints. The resulting closed-loop structure is illustrated in Fig. 2 and can be formulated as

$$\begin{aligned} \boldsymbol{u}_{\text{safe}} = \arg\min_{\boldsymbol{u}} & \|\boldsymbol{B}\boldsymbol{u} - \boldsymbol{\tau}_{\text{des}}\|_2^2 \\ \text{s.t. } & \boldsymbol{A}_{\text{u}}\boldsymbol{u} \leq \boldsymbol{b}_{\text{u}}, \\ & \underline{\boldsymbol{u}} \leq \boldsymbol{u} \leq \overline{\boldsymbol{u}}. \end{aligned} \tag{8}$$

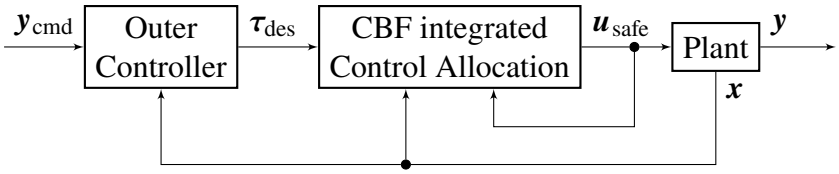

Fig. 2 Block diagram of a closed-loop control structure using a combined CBF control allocation.

2.4 The Mathematical Concept of Control Barrier Functions

In many flight control applications, safety-critical requirements such as flight envelope protection or collision avoidance must be enforced in addition to performance objectives. Classical approaches often address these constraints through heuristic saturation strategies, e.g., reference limiting [5]. Although effective in practice, these methods generally lack formal guarantees of constraint satisfaction and do not explicitly incorporate safety as a closed-loop invariant property of the controlled system.

For the nonlinear input-affine system given in (1), safety is encoded through a continuously differentiable function $h : \mathbb{R}^{n_x} \to \mathbb{R}$ defining the safe set

$$S = \{\boldsymbol{x} \in \mathbb{R}^{n_x} \mid h(\boldsymbol{x}) \geq 0\}. \tag{9}$$

In a flight control context, S may represent the admissible flight envelope, a restricted operational domain, or any set of states satisfying structural, aerodynamic, or performance limitations.

The boundary and interior of the safe set are defined as

$$\partial S = \{\boldsymbol{x} \mid h(\boldsymbol{x}) = 0\}, \tag{10}$$

$$\text{int}(S) = \{\boldsymbol{x} \mid h(\boldsymbol{x}) > 0\}. \tag{11}$$

Inside $\text{int}(S)$ the system operates safely, whereas on the boundary ∂S the safety constraint is active. To guarantee forward invariance of S, trajectories reaching the boundary must not leave the set, i.e., the system dynamics must not point outward at ∂S.

The time derivative of $h(\boldsymbol{x})$ along system trajectories is given by

$$\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u}. \tag{12}$$

A function $h(\boldsymbol{x})$ is called a CBF [20, 21] if there exists an extended class $\mathcal{K}$ function $\alpha(\cdot)$ such that

$$L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq -\alpha(h(\boldsymbol{x})). \tag{13}$$

A common choice is $\alpha(h) = \gamma h$ with $\gamma > 0$, which yields the condition

$$L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq -\gamma h(\boldsymbol{x}). \tag{14}$$

On the boundary ∂S, where $h(\boldsymbol{x}) = 0$, this reduces to $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) \geq 0$, ensuring that trajectories cannot exit the safe set. For $h(\boldsymbol{x}) > 0$, the term $-\gamma h(\boldsymbol{x}) < 0$ allows $\dot{h}$ to be temporarily negative while still driving the system away from the boundary in an exponential-type manner, thereby preserving forward invariance.

In practice, the CBF condition is enforced via a real-time quadratic program (QP). Given a nominal control input $\boldsymbol{u}^*(t)$ provided by a performance-oriented controller, the safety filter computes

$$\begin{aligned} \underset{\boldsymbol{u}(t) \in \mathbb{R}^m}{\arg\min} \quad & \|\boldsymbol{u}(t) - \boldsymbol{u}^*(t)\|_2^2 \\ \text{subject to} \quad & L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq -\alpha(h(\boldsymbol{x})), \\ & A_u \boldsymbol{u} + b \leq 0 \end{aligned} \tag{15}$$

which minimally modifies the nominal input while ensuring constraint satisfaction and respecting actuator limitations.

In many flight systems, however, operational constraints exhibit a relative degree greater than one with respect to the control input. For example, angle-of-attack or load-factor limits may depend on states whose first time derivative does not explicitly contain the control input. In such cases, $L_g h(\boldsymbol{x}) = 0$, i.e., the control input does not appear in $\dot{h}$, and the standard CBF condition cannot directly impose a constraint on $\boldsymbol{u}$.

For constraints with relative degree greater than one, Higher-Order Control Barrier Function (HOCBF) [22] are employed. For a constraint function $h(\boldsymbol{x}, t)$ of relative degree d, auxiliary functions are defined recursively as

$$\Psi_0 = h, \tag{16}$$

$$\Psi_1 = \dot{\Psi}_0 + \alpha_1(\Psi_0), \tag{17}$$

$$\vdots$$

$$\Psi_d = \dot{\Psi}_{d-1} + \alpha_d(\Psi_{d-1}), \tag{18}$$

with class $\mathcal{K}$ functions α_i, often chosen to $\alpha_i = \gamma_i \Psi_{i-1}$. The HOCBF condition requires

$$\Psi_d(\boldsymbol{x}, t) \geq 0, \tag{19}$$

which again results in a linear inequality constraint in the control input that can be incorporated into the QP formulation.

3 Control Barrier Functions on a Pseudo Control Level

The solution proposed in this paper is to separate the safety filter and control allocation again. However, the safety filter is moved between the outer controller and control allocation. The safety filter is then implemented on a pseudo control level, not on a control input level. This approach has been applied using INDI on underwater vehicles for obstacle avoidance by [23]. However, [23] do not consider the

control allocation and assume the forces and moments control input to the system. In this paper, the approach is applied to aircraft for envelope protection with a specific focus on certifiability aspects.

By splitting the tasks in this way, the following advantages are identified:

1) The control allocation problem is not altered. Therefore, the existing algorithms can be applied. Those algorithms have been specifically optimized for the box constraint problem. Thus, they are numerically more efficient than general solvers.
2) The optimization problem of the pseudo control safety filter is of lower dimension since the number of pseudo controls in overactuated systems is lower than the number of inputs. This reduces computational complexity and improves numerical robustness, making the approach more suitable for safety-critical applications.
3) Constraints and prioritization at the pseudo control level (e.g., roll moment vs. yaw moment) are more intuitive than at the actuator level (individual surfaces/propellers). If the command violates safety constraints, pseudo-controls can be prioritized over control inputs, ensuring that the most critical motion commands are maintained while less essential ones are relaxed. This is not directly possible when the safety filter is arranged behind the control allocation (CA).
4) Alternatively, a direction-preserving CA algorithm can be used. This allows only to change the direction of the commanded pseudo-control to fulfill safety requirements, but not when the input limits are reached.

Separating the CBF filter and control allocation may also simplify certification, as the algorithm is decomposed into simpler components that are easier to analyze and verify.

The proposed safety filter structure at the pseudo-control level is shown in Fig. 3. The optimization

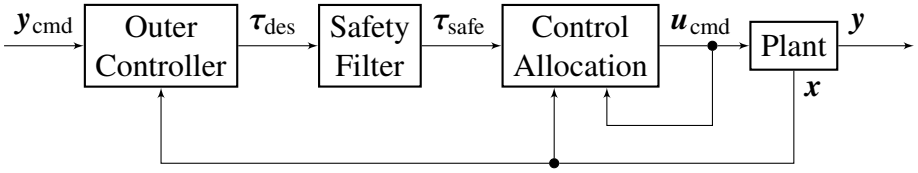

Fig. 3 Block diagram of a closed-loop control structure using the novel structure of a safety filter on a pseudo control level in front of the control allocation.

problem of the control allocation only changes in terms of the desired pseudo-control, leading to

$$\begin{aligned} \boldsymbol{u}_{\text{cmd}} = \arg\min_{\boldsymbol{u}} &\|\boldsymbol{B}\boldsymbol{u} - \boldsymbol{\tau}_{\text{safe}}\|_2^2 \\ \text{s.t. } &\underline{\boldsymbol{u}} \leq \boldsymbol{u} \leq \overline{\boldsymbol{u}}, \end{aligned} \tag{20}$$

where $\boldsymbol{\tau}_{\text{safe}}$ is the desired pseudo control vector filtered by the safety filter. If a non-direction-preserving CA is used, the direction of the pseudo-control vector may be changed, such that it points in an unsafe direction. Therefore, a direction-preserving control allocation should be used. Note that if the safety filter's constraints conflict with the input constraints, the input constraints take precedence, potentially leading to unsafe commands. However, this is a logical behavior, since the input constraints are usually derived from the physical system and will be enforced regardless of what the controller commands. The corresponding safety filter on the pseudo control level is then

$$\begin{aligned} \boldsymbol{\tau}_{\text{safe}} = \arg\min_{\boldsymbol{\tau}} &\|\boldsymbol{\tau} - \boldsymbol{\tau}_{\text{des}}\|_2^2 \\ \text{s.t. } &\boldsymbol{A}_\tau \boldsymbol{\tau} \leq \boldsymbol{b}_\tau, \end{aligned} \tag{21}$$

where $\boldsymbol{A}_\tau$ and $\boldsymbol{b}_\tau$ are obtained from the CBF. If the safety constraints only involve the outputs and their derivatives up to the relative degree and the system is transformed into the *Byrnes-Isidori-Normal form* [18, p. 78], the admissible set of pseudo controls can be expressed as simple box constraints. This further

simplifies the optimization problem of the safety filter to

$$\begin{aligned} \boldsymbol{\tau}_{\text{safe}} &= \arg\min_{\boldsymbol{\tau}} \|\boldsymbol{\tau} - \boldsymbol{\tau}_{\text{des}}\|_2^2 \\ \text{s.t. } & \boldsymbol{\tau}_{\min} \leq \boldsymbol{\tau} \leq \boldsymbol{\tau}_{\max}. \end{aligned} \tag{22}$$

3.1 Safety-Aware Limiting of the Pseudo Control

In the following, we assume that the safety constraints only involve the outputs, i.e., $y_{i,\min} \leq y_i \leq y_{i,\max}, i = 1, \ldots, n$, and do not depend on other outputs, meaning that each safety constraint acts on a single output channel only. Consequently, constraints of the form $y_1 + y_2 \leq c$, for example, are excluded, as they would couple multiple outputs within a single safety condition. This leads to independent bounds on each pseudo-control component.

This assumption allows the use of simple saturations rather than quadratic programming solvers. Those solvers are complex and hard to certify, whereas for saturations, it is easy to provide real-time guarantees. For a huge class of problems in flight control, this assumption can be made.

The core idea of the proposed approach is to enforce safety constraints at the pseudo-control level before the control allocation. Therefore, to guarantee safe operation, the admissible range of $\boldsymbol{\tau}_{\text{des}}$ must be determined based on the system's safety requirements. These safety requirements are typically defined in terms of output constraints, such as limits on angles and velocities.

If the constraints are defined independently for each output and its derivatives up to the relative degree, without coupling between different outputs, the admissible range of the virtual input $\boldsymbol{\tau}_{\text{des}}$ can be directly defined as

$$\boldsymbol{\tau}_{\min} \leq \boldsymbol{\tau}_{\text{des}} \leq \boldsymbol{\tau}_{\max}, \tag{23}$$

which are box constraints. Coupled constraints, such as $y_1 + y_2 \leq y_{\max}$, are possible, but the safety filter cannot be implemented as a simple saturation. Instead, solving an optimization problem would be required.

Those pseudo control bounds are derived from the output constraints using a CBF approach. For the sake of simplicity, we consider a relative degree of one for all outputs, i.e., $\boldsymbol{\nu} = \dot{\boldsymbol{y}}$. The higher-order case is discussed in the next section. Following the CBF framework, the upper output bound is encoded by the barrier function

$$h(\boldsymbol{y}) = \boldsymbol{y}_{\max} - \boldsymbol{y}, \tag{24}$$

with its derivative

$$\dot{h}(\boldsymbol{y}) = -\dot{\boldsymbol{y}} = -(\boldsymbol{f}_\nu(\boldsymbol{x}) + \boldsymbol{\tau}). \tag{25}$$

Imposing the CBF condition $\dot{h}(\boldsymbol{y}) \geq -\gamma h(\boldsymbol{y})$ gives

$$\boldsymbol{\tau} \leq \underbrace{-\boldsymbol{f}_\nu(\boldsymbol{x}) + \gamma(\boldsymbol{y}_{\max} - \boldsymbol{y})}_{\boldsymbol{\tau}_{\max}(\boldsymbol{x})}. \tag{26}$$

Similarly, the lower state bound is described by

$$h(\boldsymbol{y}) = \boldsymbol{y} - \boldsymbol{y}_{\min}, \qquad \dot{h}(\boldsymbol{y}) = \boldsymbol{f}_\nu(\boldsymbol{x}) + \boldsymbol{\tau} \geq -\gamma(\boldsymbol{y} - \boldsymbol{y}_{\min}), \tag{27}$$

which leads to

$$\boldsymbol{\tau} \geq \underbrace{-\boldsymbol{f}_\nu(\boldsymbol{x}) - \gamma(\boldsymbol{y} - \boldsymbol{y}_{\min})}_{\boldsymbol{\tau}_{\min}(\boldsymbol{x})}. \tag{28}$$

Together, these conditions define the admissible range for the virtual input

$$\boldsymbol{\tau}_{\min}(\boldsymbol{x}) \leq \boldsymbol{\tau} \leq \boldsymbol{\tau}_{\max}(\boldsymbol{x}). \tag{29}$$

Since the inequality constraints are box constraints, the safety filter can be implemented using simple component-wise saturation as

$$\boldsymbol{\tau}_{\text{safe}} = \text{sat}(\boldsymbol{\tau}_{\text{des}}, \boldsymbol{\tau}_{\min}(\boldsymbol{x}), \boldsymbol{\tau}_{\max}(\boldsymbol{x})). \tag{30}$$

This leads to a significant reduction in computational complexity compared to solving a quadratic programm (QP) as in Eqs. (7) and (8).

3.2 Using Higher-Order Control Barrier Functions (HOCBF)

For outputs with a relative degree higher than one, the admissible range of the virtual input $\boldsymbol{\tau}$ cannot be directly defined using a standard CBF. Instead, higher-order CBFs must be employed. Since each output may have a different relative degree, the higher-order CBF is constructed separately for each output y_i, where $i = 1, \ldots, n$. The CBF condition is then imposed on the highest-order function as

$$\dot{\Psi}_{r_i-1}(y_i) + \gamma_{r_i} \Psi_{r_i-1}(y_i) \geq 0. \tag{31}$$

How does this now translate into constraints on the pseudo-control level? The derivative of the highest-order barrier function can be expressed as

$$\dot{\Psi}_{r_i-1}(y_i) = -\overset{(r_i)}{y_i} + b(y_i, \overset{(1)}{y_i}, \ldots, \overset{(r_i-1)}{y_i}). \tag{32}$$

Also, $\Psi_{r_i-1}(y_i)$ is only dependent on the output y_i and its derivatives up to order $r_i - 1$. Further, by definition of the pseudo control, it holds that $\overset{(r_i)}{y_i} = \tau_i + f_{\nu_i}(\boldsymbol{x})$ according to Eq. (2). Therefore, the CBF condition can be rearranged to

$$\tau_i \leq b(y_i, \overset{(1)}{y_i}, \ldots, \overset{(r_i-1)}{y_i}) - f_{\nu_i}(\boldsymbol{x}) + \gamma_{r_i} \Psi_{r_i-1}(y_i) = \tau_{i,\max}. \tag{33}$$

For the lower bound, the same procedure can be applied to obtain $\tau_{i,\min}$.

4 Validation

The new proposed approach to apply the CBF on a pseudo control level is validated in simulation using two different models: First, the linearized short-period dynamics of the ADMIRE fighter aircraft, and second, a linear model of the attitude dynamics of a hexacopter. This section validates the proposed CBF safety filter on pseudo control level using two simulation examples:

- a linearized short-period model of the ADMIRE fighter aircraft, and
- a linear multiple input multiple output (MIMO) hexacopter attitude dynamics model

The purpose of using the two examples is as follows: The short-period model is selected primarily to illustrate the structural and implementation differences among the three different CBF strategies using a simple example. The hexacopter example represents an overactuated MIMO system, in which additional advantages of the proposed approach, such as pseudo-control prioritization and direction preservation, can be demonstrated.

For both examples, the three CBF-based safety filter implementations are compared:

- Case A: Standard CBF safety filter on the control input level (denoted with aft)

- Case B: Combined CBF control allocation (denoted with incl)
- Case C: CBF safety filter on the pseudo control level (denoted with bef)

In addition, a baseline case using a simple command saturation without CBF is included for reference.

4.1 The Short-Period ADMIRE Model

In order to analyze and validate the proposed methods, this work employs the ADMIRE model (*Aero-Data Model in a Research Environment*), which describes a small single-engine fighter aircraft with a delta-canard configuration [24]. The model is linearized around a low-speed operating point at Mach number $M = 0.22$ and an altitude of 3000 m, where the effectiveness of the control surfaces is known to be poor.

The short-period model serves as an example to demonstrate the implementation differences among CBF-based safety filter structures. Although the system is overactuated, it has only a single pseudo-control associated with the angle-of-attack dynamics. Consequently, advantages related to pseudo-control prioritization (3) and direction preservation (4) cannot be demonstrated in this example, since these effects only arise in systems with multiple pseudo-controls. These aspects are therefore addressed in the second example using a hexacopter model in Section 4.2. Besides that, the short-period example highlights an important advantage of the proposed approach: the reduced dimensionality of the safety filter at the pseudo-control level. Since only a single pseudo-control is involved, the admissible range can be computed directly as bounds or limits for the pseudo-control, whereas the standard and combined approaches require solving optimization problems in the higher-dimensional actuator space. Our approach only requires a control allocation.

The linearized aircraft dynamics are expressed as

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}_x\boldsymbol{u}, \tag{34a}$$

$$y = \boldsymbol{C}\boldsymbol{x} \tag{34b}$$

with the definitions

$$\boldsymbol{x} = \begin{bmatrix} \alpha & q \end{bmatrix}^T - \boldsymbol{x}_{\text{lin}}, \tag{35}$$

$$y = \alpha - y_{\text{lin}}, \tag{36}$$

$$\boldsymbol{u} = \begin{bmatrix} u_c & u_{re} & u_{le} & u_r \end{bmatrix}^T - \boldsymbol{u}_{\text{lin}}. \tag{37}$$

Here, α is the angle of attack and q the pitch rate. The vector $\boldsymbol{u}$ contains the deflections of the canard wings, the right and left elevons, and the rudder. The terms $x_{\text{lin}}, y_{\text{lin}}, u_{\text{lin}}$ denote the points of linearization.

The control surfaces are limited by

$$\delta_c \in [-55, 25] \cdot \tfrac{\pi}{180}, \tag{38}$$

$$\delta_{re}, \delta_{le}, \delta_r \in [-30, 30] \cdot \tfrac{\pi}{180}. \tag{39}$$

For the considered operating point, and using a simplified model suitable for allocation studies, the system matrices are given by

$$\boldsymbol{A} = \begin{bmatrix} -0.5432 & 0.9778 \\ 2.6221 & -0.5057 \end{bmatrix}, \quad \boldsymbol{B}_x = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1.6532 & -1.2735 & -1.2735 & 0.0024 \end{bmatrix}, \quad \boldsymbol{C} = \begin{bmatrix} 1 & 0 \end{bmatrix}. \tag{40}$$

Here, the actuator dynamics are neglected, and the control surfaces are treated as pure moment generators, such that their influence on $\dot{\alpha}$ is ignored. This corresponds to setting the first row of $\boldsymbol{B}_x$ equal to zero.

As mentioned before, the proposed CBF safety filter is applied at a pseudo-control level. Therefore, the system is transformed into a pseudo control form as follows:

$$y = \boldsymbol{C}\boldsymbol{x} = \alpha, \tag{41}$$

$$\dot{y} = \boldsymbol{C}\dot{\boldsymbol{x}} = \boldsymbol{C}\boldsymbol{A}\boldsymbol{x} + \cancelto{\boldsymbol{0}}{\boldsymbol{C}\boldsymbol{B}_x}\boldsymbol{u}, \tag{42}$$

$$\ddot{y} = \boldsymbol{C}\boldsymbol{A}\dot{\boldsymbol{x}} = \boldsymbol{C}\boldsymbol{A}^2\boldsymbol{x} + \boldsymbol{C}\boldsymbol{A}\boldsymbol{B}_x\boldsymbol{u} = f_\nu(\boldsymbol{x}) + \underbrace{\boldsymbol{B}_\nu\boldsymbol{u}}_{\tau} = \nu. \tag{43}$$

The derivation shows that the output α has a relative degree of two $r = 2$ and $\nu = \ddot{\alpha}$. Taking a closer look at the control effectiveness matrix on the pseudo control level, one finds that

$$\boldsymbol{B}_\nu = 0.9778 \cdot \begin{bmatrix} 1.6532 & -1.2735 & -1.2735 & 0.0024 \end{bmatrix} = b \cdot \boldsymbol{B}_{x,2}, \tag{44}$$

where $\boldsymbol{B}_{x,2}$ is the second row of $\boldsymbol{B}_x$ and $b = 0.9778$. In this example, it holds that the pitch acceleration generated by the inputs $\dot{q}_u = \boldsymbol{B}_{x,2}\boldsymbol{u}$ is directly proportional to the virtual input $\nu = b \cdot \dot{q}_u$. This means instead of allocating the virtual input τ related to the pseudo control $\ddot{\alpha}$, the control allocation can allocate the pitch acceleration $\dot{q}_u$ and then scale it by b to obtain the virtual input τ.

The control allocation in the pseudo-control-level safety filter and the standard approach are implemented by a pseudo-inverse with the following saturation. Note that for a single pseudo-control input, direction preservation is unnecessary, and no redistribution is possible. The optimization problems Eqs. (7) and (8) are implemented using the `quadprog` function from MATLAB configured to use an active set algorithm.

4.1.1 Derivation of the HOCBF

As the next step, the higher-order CBF is constructed to enforce angle-of-attack limits of

$$-3^\circ \leq \alpha \leq 3^\circ. \tag{45}$$

The barrier functions are defined as

$$\Psi_0(\alpha) = h(\boldsymbol{x}) = \alpha_{\max} - \alpha, \tag{46}$$

$$\Psi_1(\alpha) = \dot{\Psi}_0(\alpha) + \gamma_1\Psi_0(\alpha) = -\dot{\alpha} + \gamma_1(\alpha_{\max} - \alpha) \tag{47}$$

$$\Psi_2(\alpha) = \dot{\Psi}_1(\alpha) + \gamma_2\Psi_1(\alpha) = -\ddot{\alpha} + \gamma_1(-\dot{\alpha}) + \gamma_2\Psi_1(\alpha) = -\nu + b(\dot{\alpha}) + \gamma_2\Psi_1(\alpha) \tag{48}$$

As a next step, we insert $\nu = f_\nu(\boldsymbol{x}) + \tau$ from Eq. (43), and impose the CBF condition

$$\Psi_2(\alpha) = -f_\nu(\boldsymbol{x}) - \tau + b(\dot{\alpha}) + \gamma_2\Psi_1(\alpha) \geq 0. \tag{49}$$

The CBF condition is rearranged to

$$\tau \leq -f_\nu(\boldsymbol{x}) + b(\dot{\alpha}) + \gamma_2\Psi_1(\alpha) \tag{50}$$

which corresponds to Eq. (33). Finally, we insert $f_\nu(\boldsymbol{x}) = CA^2\boldsymbol{x}$, $b(\dot{\alpha}) = -\gamma\dot{\alpha}$ and $\Psi_1 = -\dot{\alpha} + \gamma_1(\alpha_{max} - \alpha)$ leading to

$$\tau \leq -CA^2\boldsymbol{x} - (\gamma_1 + \gamma_2)\dot{\alpha} + \gamma_2\gamma_1(\alpha_{max} - \alpha) \tag{51}$$

We estimate $\dot{\alpha}$ using $\dot{\alpha} = CA\boldsymbol{x}$. In practice, the estimate could be obtained using load factor measurements as commonly done [16].

Now, inserting the system matrices leads to

$$\boldsymbol{CA}^2 = \begin{bmatrix} 2.8590 & -1.0256 \end{bmatrix}, \tag{52}$$

$$\boldsymbol{CA} = \begin{bmatrix} -0.5432 & 0.9778 \end{bmatrix}. \tag{53}$$

By that, the bounds on the virtual input have been obtained. As mentioned before, the control allocation can allocate the pitch acceleration $\dot{q}_u$ instead of the virtual input τ. Therefore, the bounds on the pitch acceleration are obtained as

$$\frac{\tau_{\min}}{b} \leq \dot{q}_u \leq \frac{\tau_{\max}}{b}. \tag{54}$$

This transformation will be used in the simulation results.

For the standard CBF safety filter on the control input level and the combined CBF control allocation, the CBF conditions must be expressed in terms of the control inputs $\boldsymbol{u}$. Inserting the plant definition of $\tau = \boldsymbol{B}_v \boldsymbol{u}$, Eq. (43), into the CBF conditions leads to the inequality constraints

$$\underbrace{\begin{bmatrix} \boldsymbol{B}_v \\ -\boldsymbol{B}_v \end{bmatrix}}_{\boldsymbol{A}_\tau} \boldsymbol{u} \leq \underbrace{\begin{bmatrix} \tau_{\max}(\boldsymbol{x}) \\ -\tau_{\min}(\boldsymbol{x}) \end{bmatrix}}_{\boldsymbol{b}_\tau}. \tag{55}$$

4.1.2 Results

To demonstrate the different safety filter implementations, the following simulation cases are considered:

- Case 0 (Baseline): Angle-of-attack command limited by a simple saturation block to ±3° (denoted with sat).
- Case A–C (CBF-based cases): Step command in angle of attack from 0° to 7° with
  - A) standard CBF safety filter on the control input level (denoted with aft),
  - B) combined CBF control allocation (denoted with incl),
  - C) CBF safety filter on the pseudo control level (denoted with bef).

The baseline case is included to highlight that a simple command can lead to a more conservative, slower response. The proposed CBF safety filter at the pseudo-control level is validated in simulation and compared with the standard CBF safety filter at the control input level and the combined CBF control allocation. All three approaches are implemented in MATLAB/Simulink and use the same outer LQR controller to provide the desired pitch acceleration command $\dot{q}_{\text{des}}$. The LQR is designed to track a step command in angle of attack while minimizing the states and control inputs. The parameters of the LQR are chosen as

$$\boldsymbol{Q} = \text{diag}\,(50, 5)\,, \tag{56}$$

$$\boldsymbol{R} = 1. \tag{57}$$

The CBF parameters are set to $\gamma_1 = 3$ and $\gamma_2 = 40$, while the limits on the angle of attack are set to $\alpha_{\min} = -3°$ and $\alpha_{\max} = 3°$. Therefore, the step command violates the safety limits, and interception of the safety filters is expected. The resulting angle of attack response for the three different CBF implementations is shown in Fig. 4. From the simulation results, we can make the following key

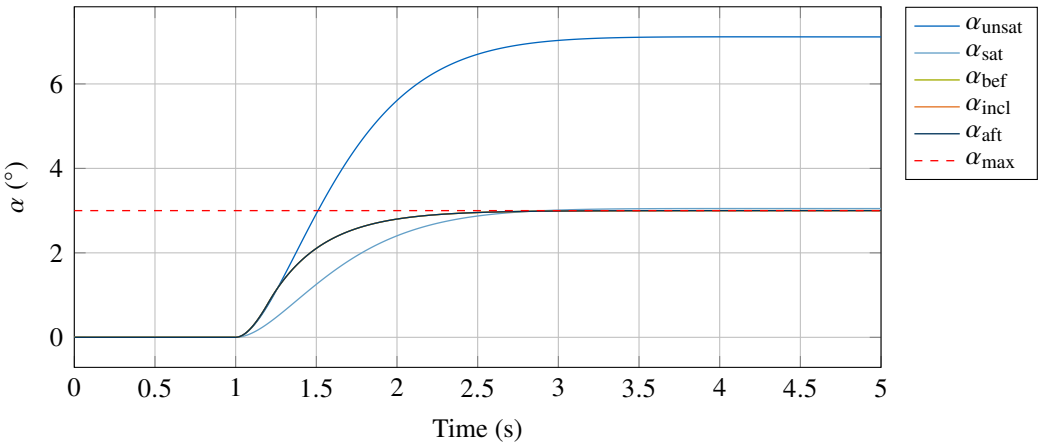

Fig. 4 Angle of attack (α) response for a step command with different CBF safety filter implementations.

observations:

- with simple command saturation, the angle-of-attack response is significantly slower than the ones including CBF.
- All three CBF-based approaches lead to identical angle-of attack responses and successfully prevent overshoot of the limit, hence enforce the safety constraints

The main differences between the approaches, therefore, lie not in performance but in implementation and computational effort. All three approaches successfully enforce the angle-of-attack limits, preventing any violations. The responses of all three methods are identical, demonstrating that the proposed CBF safety filter at the pseudo-control level achieves the same performance as the standard methods. This behavior is expected, since the system has only one pseudo-control input; thus, scaling or limiting the pseudo-control input is equivalent. Also, no input limits are active in this scenario.

4.2 Hexacopter

As a second example, the attitude dynamics of a Hexacopter (see Fig. 5) are used, which was also used in [25–27]. In this example, the differences between the approaches can be shown since it is an overactuated MIMO system. Specifically, a direction-preserving control allocation can be used here.

Fig. 5 AscTec Firefly Hexacopter

4.2.1 Model and Baseline Controller

The attitude dynamics of the hexacopter are modeled as a linear system around hover conditions. Specifically, the roll, pitch, and yaw dynamics are considered as chains of double integrators. The

state-space representation of the system is given by

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}_x\boldsymbol{B}_\nu\boldsymbol{u}, \tag{58a}$$
$$\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x}, \tag{58b}$$

where

$$\boldsymbol{x} = \begin{bmatrix} \phi & p & \theta & q & \psi & r \end{bmatrix}^T \tag{59}$$
$$\boldsymbol{y} = \begin{bmatrix} \phi & \theta & \psi \end{bmatrix}^T, \tag{60}$$
$$\boldsymbol{u} = \begin{bmatrix} T_1 & T_2 & T_3 & T_4 & T_5 & T_6 \end{bmatrix}^T. \tag{61}$$

Here, ϕ, θ, ψ are the roll, pitch, and yaw angles, while p, q, r are the corresponding angular rates. The control inputs T_i represent the normalized thrust generated by each of the six rotors. The system matrices are defined as

$$\begin{aligned} \boldsymbol{A} &= \operatorname{diag}\left(\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}\right), \quad \boldsymbol{B}_x = \operatorname{diag}\left(\begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}\right), \\ \boldsymbol{C} &= \operatorname{diag}\left(\begin{bmatrix} 1 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 \end{bmatrix}\right). \end{aligned} \tag{62}$$

The control effectiveness matrix $\boldsymbol{B}_\nu$ maps the control inputs to the pseudo controls and is given by

$$\boldsymbol{B}_\nu = \begin{bmatrix} -\frac{1}{2}Lk_F & -Lk_F & -\frac{1}{2}Lk_F & \frac{1}{2}Lk_F & Lk_F & \frac{1}{2}Lk_F \\ \frac{\sqrt{3}}{2}Lk_F & 0 & -\frac{\sqrt{3}}{2}Lk_F & -\frac{\sqrt{3}}{2}Lk_F & 0 & \frac{\sqrt{3}}{2}Lk_F \\ -k_M & k_M & -k_M & k_M & -k_M & k_M \end{bmatrix}, \tag{63}$$

where $L = 0.215\,\mathrm{m}$ is the distance from the center of mass to each rotor, $k_F = 5.8833\,\mathrm{rad/(m\,s^2)}$ is the acceleration coefficient, and $k_M = 0.112\,\mathrm{rad/s^2}$ is the moment coefficient.

Similar to the previous example, the system is transformed into pseudo control form, which is straightforward since the system is already in a chained double integrator form. Therefore, the transformed system is given by

$$\boldsymbol{\nu} = \ddot{\boldsymbol{y}} = \boldsymbol{B}_\nu\boldsymbol{u} = \boldsymbol{\tau}. \tag{64}$$

Note that $\boldsymbol{f}_\nu(\boldsymbol{x}) = \boldsymbol{A}_\nu = 0$ for the considered example system, and the outputs ϕ, θ, ψ all have a relative degree of two $r_i = 2$. The pseudo control vector is therefore

$$\boldsymbol{\nu} = \begin{bmatrix} \dot{p} & \dot{q} & \dot{r} \end{bmatrix}^T. \tag{65}$$

For hexacopter control, a cascaded controller is used, consisting of an outer angle loop and an inner angular rate loop. Both loops are proportional error controllers with the following gains

$$K_y = \operatorname{diag}([1.5, 1.5, 1]), \quad K_{\dot{y}} = \operatorname{diag}([15, 15, 10]). \tag{66}$$

The outputs of the inner loop are the desired angular accelerations $\dot{p}_{\mathrm{des}}, \dot{q}_{\mathrm{des}}, \dot{r}_{\mathrm{des}}$.

Since the system is overactuated, a control allocation should be deployed to map the desired angular accelerations to the rotor thrusts. The task of the control allocation is to solve Eq. (20), which can be solved using standard QP techniques, like Active Set or Interior Point methods [28]. Instead of standard solvers, specific algorithms for control allocation can be used, such as RSPI, which belongs to

the class of cascaded generalized inverses (CGI) methods. RSPI is closely related to the SLS-AS QP control allocation. One characteristic of RSPI is that it can be configured to preserve the direction of the desired virtual input when allocating the control inputs. In addition, RSPI has a low computational complexity, making it suitable for real-time applications. In the Section 4.2.3, the direction-preserving version is compared to the non-direction-preserving version to highlight its benefits. Therefore, the control allocation in the pseudo-control-level safety filter and the standard approach are implemented using the RSPI algorithm from [29]. The optimization problems Eqs. (7) and (8) are implemented using the `quadprog` function from MATLAB configured to use an active set algorithm.

Similar to the previous example, the system is transformed into pseudo control form, which is straightforward since the system is already in a chained double integrator form. Therefore, the transformed system is given by

$$\boldsymbol{\nu} = \boldsymbol{B}_\nu \boldsymbol{u} = \boldsymbol{\tau}. \tag{67}$$

Note that $\boldsymbol{f}_\nu(\boldsymbol{x}) = 0$ for the considered example system, and the outputs ϕ, θ, ψ all have a relative degree of two $r_i = 2$.

4.2.2 Derivation of the HOCBF

In the following, the higher-order CBF is constructed to enforce attitude angle limits of the hexacopter. Since all attitude angles have the same relative degree of two and are decoupled (chain of integrators), the derivation is identical for all three angles. Therefore, the derivation is shown only for the roll angle ϕ.

The barrier function for the upper roll angle limit is defined as

$$\Psi_{0,1}(\phi) = h_1(\boldsymbol{x}) = \phi_{\max} - \phi, \tag{68}$$

$$\Psi_{1,1}(\phi) = \dot{\Psi}_{0,1}(\phi) + \gamma_{1,1}\Psi_{0,1}(\phi) = -\dot{\phi} + \gamma_{1,1}(\phi_{\max} - \phi) = -p + \gamma_{1,1}(\phi_{\max} - \phi) \tag{69}$$

$$\Psi_{2,1}(\phi) = \dot{\Psi}_{1,1}(\phi) + \gamma_{2,1}\Psi_{1,1}(\phi) = -\dot{p} + \gamma_{1,1}(-p) + \gamma_{2,1}\Psi_{1,1}(\phi). \tag{70}$$

Inserting the plant definition of $\nu_1 = \tau_1$ leads to the upper bound on the virtual input

$$\Psi_{2,1}(\phi) = -\dot{p} - (\gamma_{1,1} + \gamma_{2,1})p + \gamma_{2,1}\gamma_{1,1}(\phi_{\max} - \phi) = \tau_{1,\max}(\boldsymbol{x}). \tag{71}$$

$$\tau_1 \leq -(\gamma_{1,1} + \gamma_{2,1})p + \gamma_{2,1}\gamma_{1,1}(\phi_{\max} - \phi) = \tau_{1,\max}(\boldsymbol{x}). \tag{72}$$

Similarly, the lower bound is derived as

$$\tau_1 \geq -(\gamma_{1,1} + \gamma_{2,1})p - \gamma_{2,1}\gamma_{1,1}(\phi - \phi_{\min}) = \tau_{1,\min}(\boldsymbol{x}). \tag{73}$$

The same procedure is applied to the pitch and yaw angles to obtain the complete set of bounds on the virtual input

$$\boldsymbol{\tau}_{\max}(\boldsymbol{x}) = -(\boldsymbol{\gamma}_1 + \boldsymbol{\gamma}_2)\begin{bmatrix} p \\ q \\ r \end{bmatrix} + \boldsymbol{\gamma}_2\boldsymbol{\gamma}_1\left(\begin{bmatrix} \phi_{\max} \\ \theta_{\max} \\ \psi_{\max} \end{bmatrix} - \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}\right), \tag{74}$$

where $\boldsymbol{\gamma}_1 = \mathrm{diag}([\gamma_{1,1}, \gamma_{1,2}, \gamma_{1,3}])$ and $\boldsymbol{\gamma}_2 = \mathrm{diag}([\gamma_{2,1}, \gamma_{2,2}, \gamma_{2,3}])$. The lower bound follows the same structure.

To get the inequality constraints for the standard CBF safety filter and the combined CBF control allocation, the plant definition $\boldsymbol{\tau} = \boldsymbol{B}_\nu \boldsymbol{u}$ is used according to Eq. (55).

4.2.3 Results

To evaluate the proposed CBF safety filter on pseudo control level, multiple closed-loop simulations of the hexacopter attitude control are performed. In those simulations, step commands in roll, pitch, and yaw angle are commanded to demonstrate

- the enforcement of safety limits on the attitude angles,
- the differences between the three CBF safety filter implementations, and
- the influence of a direction-preserving control allocation.

Note that only the standard and the proposed CBF safety filter implementations include a separate control allocation. Therefore, only those two approaches are influenced by direction-preserving control allocation.

The CBF parameters are set to

$$\boldsymbol{\gamma}_1 = \mathrm{diag}([3, 3, 1]), \quad \boldsymbol{\gamma}_2 = \mathrm{diag}([12, 12, 4]), \tag{75}$$

and the attitude angle limits are set to

$$\begin{bmatrix} \phi_{\min} & \theta_{\min} & \psi_{\min} \end{bmatrix}^T = \begin{bmatrix} -15^\circ & -20^\circ & -7^\circ \end{bmatrix}^T, \quad \begin{bmatrix} \phi_{\max} & \theta_{\max} & \psi_{\max} \end{bmatrix}^T = \begin{bmatrix} 15^\circ & 20^\circ & 7^\circ \end{bmatrix}^T. \tag{76}$$

The test cases consider synchronous step commands in pitch and roll angle from 0° to 25° at $t = 1\,\mathrm{s}$. While the yaw angle is commanded to 15° at $t = 5\,\mathrm{s}$. All commands violate the safety limits, so that interception by the CBF safety filters is expected. Furthermore, the simulation scenario in Fig. 6 is chosen such that the control input limits are reached. If the desired virtual input were fully feasible, i.e., achievable without saturating the actuators, all three methods would yield identical behavior, and no differences would be observable. By driving the system into actuator saturation, the differences of the investigated architectures become visible, as discussed in the following. The resulting attitude angle responses for the three different CBF implementations without a direction-preserving control allocation are shown in Fig. 6.

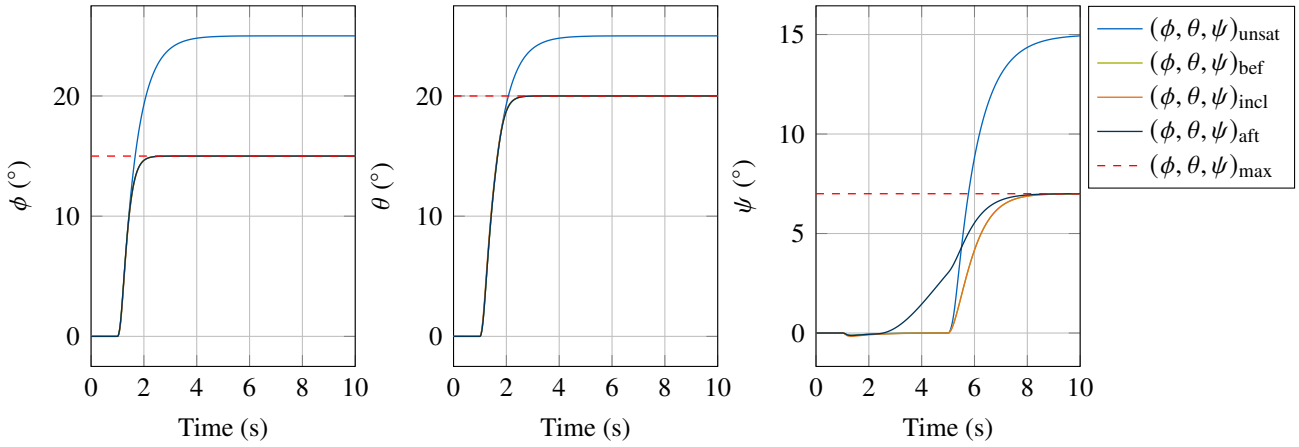

Fig. 6 Hexacopter attitude response with different CBF safety filter implementations

The results show that all three CBF safety filter implementations successfully enforce the attitude angle limits, preventing any violations. The control input limits are reached in those scenarios, which is required to demonstrate the different direction-preserving characteristics of the control allocation in Fig. 7. This is necessary because if the desired virtual input is feasible and can be reached by the control allocation, there will be no differences.

The response of the proposed CBF safety filter on pseudo control level closely matches that of the integrated CBF control allocation approach. However, the standard CBF safety filter on control input level diverges from the common behavior in the yaw angle response. Even before the step command at $t = 5\,\mathrm{s}$, the yaw angle starts to deviate from zero. This behavior can be explained by the fact that the safety filter modifies the control inputs after control allocation, as shown in Eq. (7) and Fig. 1. As a result, the safety filter fails to account for the desired virtual inputs during safety enforcement, leading to deviations from non-targeted states.

To further investigate the influence of the control allocation method, the same test case is repeated using a direction-preserving RSPI control allocation. The resulting attitude angle responses for the three different CBF implementations with direction-preserving control allocation are shown in Fig. 7. It can be

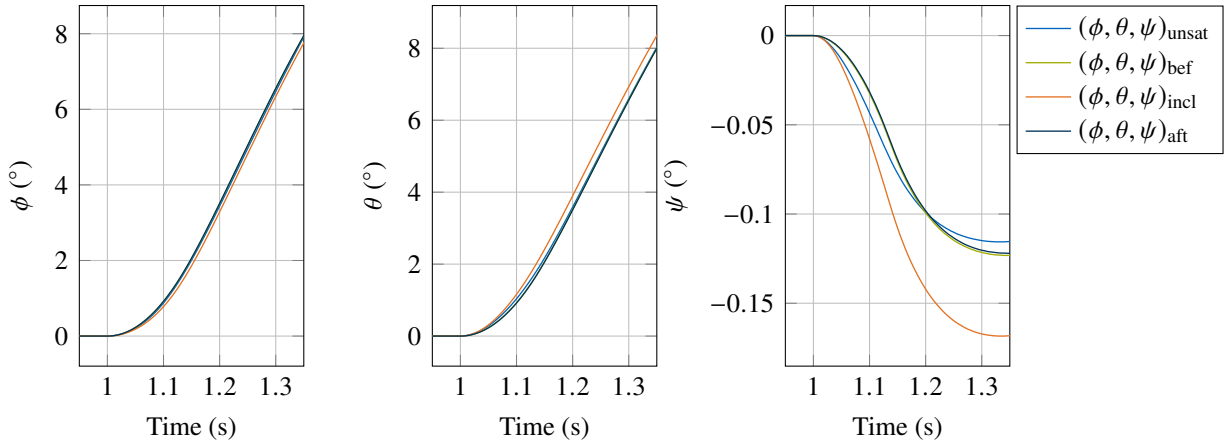

Fig. 7 Hexacopter attitude response with different CBF safety filter implementations with synchronous step inputs using a direction-preserving control allocation.

observed that the integrated approach, where CA and the safety filter are solved as a single optimization problem, deviates from the system's reference trajectory using a direction-preserving control allocation. While the approaches with a separate safety filter and direction preserving CA perfectly track the reference trajectory in ϕ and θ.

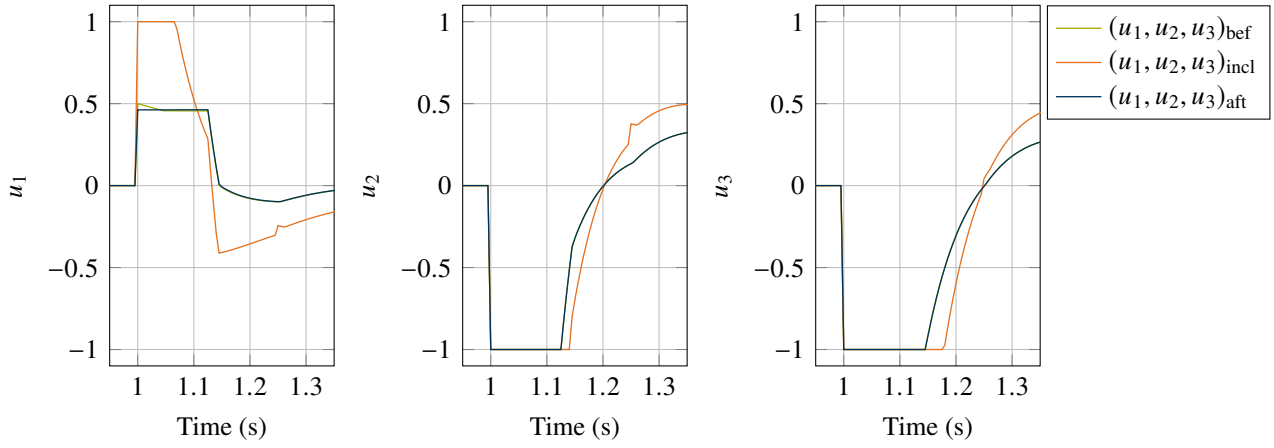

Fig. 8 Hexacopter control inputs with different CBF safety filter implementations with synchronous step inputs using a direction-preserving control allocation.

Overall, the results are similar to the previous case without direction preservation. The safety constraints are successfully enforced with nearly identical trajectories.

However, in the transient, the safety filter with integrated control allocation deviates from the other two approaches. It even leads the reference system without the safety filter in the pitch angle response,

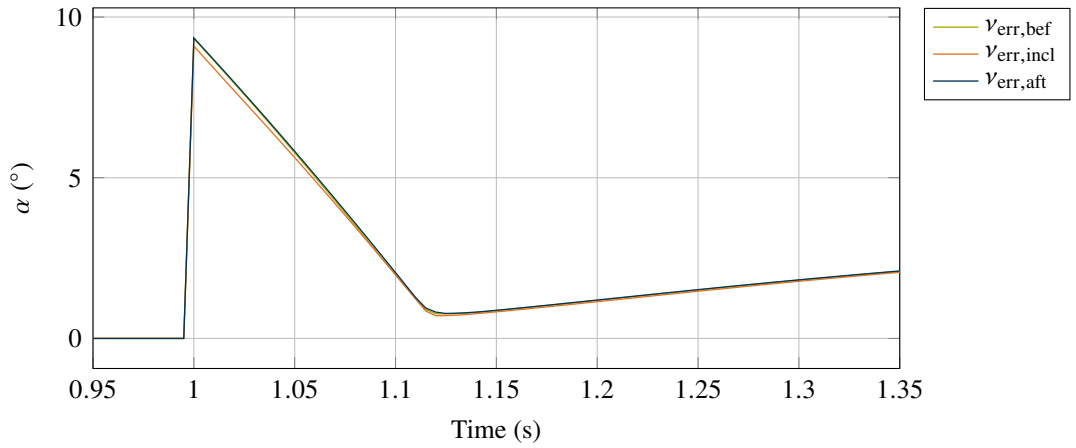

Fig. 9 Angle of attack (α) response for a step command with different CBF safety filter implementations.

while lagging behind in the roll angle response. The reason for this behavior is that the integrated approach is not direction-preserving; rather, it minimizes the deviation from the desired virtual input in a least-squares sense. Therefore, it can allocate control inputs that result in a virtual input with a direction opposite to the desired one. In contrast, the proposed CBF safety filter on pseudo control level with direction-preserving control allocation maintains the direction of the desired virtual input while enforcing safety.

4.3 Execution Time Evaluation

One advantage of the proposed CBF safety filter on pseudo control level is its lower computational complexity. To evaluate the execution time of the three different CBF safety filter types, they are implemented in Simulink and evaluated on an STM32F767ZI microcontroller running at a clock speed of 216 MHz. The execution time is measured using the Simulink processor-in-the-loop (PiL) functionality. The measured execution times always consider the combination of safety filter and control allocation.

The average and maximum execution times for each approach are summarized in Fig. 10. For that, the test case from Fig. 6 with 15 s and 30 000 steps is used. The results show that the proposed CBF safety

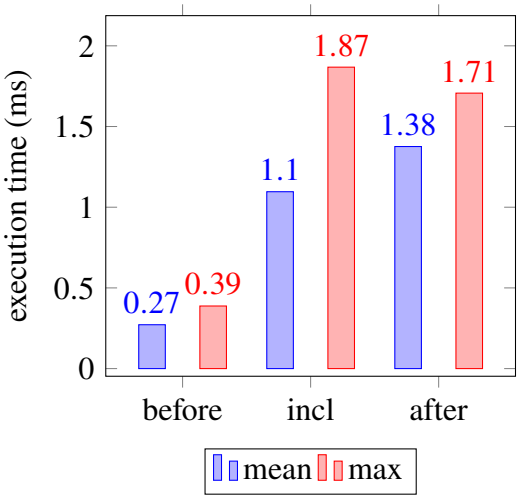

Fig. 10 Comparison of the execution time for the three CBF safety filter implementations on an STM32F767ZI.

filter on pseudo control level has a significantly lower execution time compared to the standard CBF safety filter and the combined CBF control allocation. The reduction in mean execution time is approximately

70 % compared to the combined approach and nearly 80 % compared to the standard approach. This demonstrates the computational efficiency of the proposed method, making it particularly suitable for real-time applications on resource-constrained embedded systems like aircraft.

5 Conclusion and Outlook

This paper presented a novel approach to implement CBFs for safety-critical control systems by applying the CBF conditions on a pseudo control level. This method effectively decouples safety filtering from control allocation, enabling the use of standard control allocation techniques. The proposed approach was validated through simulations using the linearized short-period dynamics of the ADMIRE fighter aircraft and the attitude dynamics of a hexacopter. The results demonstrated that the proposed CBF safety filter on pseudo control level achieves identical performance to standard CBF safety filters and integrated CBF control allocation methods in enforcing safety constraints. Furthermore, the proposed method significantly reduces the computational complexity, since the safety filter reduces to a simple saturation operation. We demonstrated a reduction in execution time on an embedded microcontroller. This makes the proposed approach particularly suitable for real-time applications on resource-constrained systems, such as aircraft.

Future work includes extending the proposed method to nonlinear systems and evaluating its performance in more complex scenarios.

Declaration of Use of Artificial Intelligence

Artificial Intelligence (AI) has been used to proofread, correct grammatical errors, and improve the conciseness of the manuscript. The service Grammarly was used for this purpose, with the last use on 26 March 2026.

References

[1] Aaron D. Ames, Samuel Coogan, Magnus Egerstedt, Gennaro Notomista, Koushil Sreenath, and Paulo Tabuada. Control Barrier Functions: Theory and Applications. In *2019 18th European Control Conference (ECC)*, pages 3420–3431, Naples, Italy, June 2019. IEEE. ISBN: 978-3-907144-00-8. doi: 10.23919/ECC.2019.8796030.

[2] Kim Peter Wabersich and Melanie N. Zeilinger. A predictive safety filter for learning-based control of constrained nonlinear dynamical systems. *Automatica*, 129:109597, 2021.

[3] Johannes Autenrieb and Mark Spiller. Decentralized cbf-based safety filters for collision avoidance of cooperative missile systems with input constraints. *arXiv:2510.06846 [eess.SY]*, Oct. 2025.

[4] Tamas G. Molnar, Suresh K. Kannan, James Cunningham, Kyle Dunlap, Kerianne L. Hobbs, and Aaron D. Ames. Collision avoidance and geofencing for fixed-wing aircraft with control barrier functions. *IEEE Transactions on Control Systems Technology*, 33(5):1493–1508, 2025. doi: 10.1109/TCST.2025.3536215.

[5] Johannes Autenrieb. Quadratic Programming Approach to Flight Envelope Protection Using Control Barrier Functions. *Journal of Guidance, Control, and Dynamics*, pages 1–12, Sept. 2025. ISSN: 0731-5090, 1533-3884. doi: 10.2514/1.G009203.

[6] Jingyao Huang, Jiaolong Liu, Hang Guo, and Wenxing Fu. Control Barrier Function Based Three-Channel Coupling Control of Air-Breathing Hypersonic Vehicle with AoA Constraint. In Lianqing Liu, Yifeng Niu, Wenxing Fu, and Yi Qu, editors, *Proceedings of 4th 2024 International Conference on Autonomous Unmanned Systems (4th ICAUS 2024)*, volume 1377, pages 284–294. Springer Nature Singapore, Singapore, 2025. ISBN: 978-981-96-3567-2 978-981-96-3568-9. doi: 10.1007/978-981-96-3568-9_27.

[7] Johannes Autenrieb and Hyo-Sang Shin. Sensor-based safety-critical control using an incremental control barrier function formulation via reduced-order approximate models. In *2025 American Control Conference (ACC)*, pages 374–381, 2025. doi: 10.23919/ACC63710.2025.11107913.

[8] Hannes Hofsäss and Florian Holzapfel. Runtime monitoring of flight control with incremental nonlinear dynamic inversion. In *2024 AIAA DATC/IEEE 43rd Digital Avionics Systems Conference (DASC)*, pages 1–10, 2024. doi: 10.1109/DASC62030.2024.10748693.

[9] Evangelos Huber, Muhammad Ahmed Hassan, Johannes Bender, Nils Schlautmann, Julius Hoffelner, and Florian Holzapfel. Run Time Assurance for a Hybrid Actuated Fly-by-Wire Aircraft: A Control Barrier Function Approach. In *2025 AIAA DATC/IEEE 44th Digital Avionics Systems Conference (DASC)*, pages 01–10, Montreal, QC, Canada, Sept. 2025. IEEE. ISBN: 979-8-3315-2519-4. doi: 10.1109/DASC66011.2025.11257188.

[10] Johannes Autenrieb and Hyo-Sang Shin. Complementary filter-based incremental nonlinear model following control design for a tilt-wing uav. *International Journal of Robust and Nonlinear Control*, Dec. 2024. doi: 10.1002/rnc.7743.

[11] Jiannan Zhang, Pranav Bhardwaj, Stefan A. Raab, Saurabh Saboo, and Florian Holzapfel. Control allocation framework for a tilt-rotor vertical take-off and landing transition aircraft configuration. In *AIAA AVIATION Forum, 2018 Applied Aerodynamics Conference*, 2018. doi: 10.2514/6.2018-3480, `https://arc.aiaa.org/doi/abs/10.2514/6.2018-3480`.

[12] Kenneth A. Bordignon. *Constrained control allocation for systems with redundant control effectors*. PhD thesis, Virginia Polytechnic Institute and State University, 1996.

[13] O. Härkegård. *Backstepping and control allocation with applications to flight control*. PhD thesis, Linköpings universitet, Linköping, Sweden, Apr. 2003. `http://www.diva-portal.org/smash/get/diva2:243491/FULLTEXT01.pdf`.

[14] Simon Hafner, Stephan Myschik, Jakob Bachler, and Florian Holzapfel. Fast pseudoinverse-based control allocation using variant of cholesky decomposition. *Journal of Guidance, Control, and Dynamics*, 48(8):1901–1914, 2025. doi: 10.2514/1.G008767.

[15] Agnes Christine Gabrys, Rasmus Steffensen, Rafael de Angelis Cordeiro, José Raul Azinheira, Alexandra Moutinho, and Florian Holzapfel. Integration of phase plane flight envelope protections in cascaded incremental flight control. *IFAC-PapersOnLine*, 52(12):429–435, 2019. ISSN: 2405-8963. 21st IFAC Symposium on Automatic Control in Aerospace ACA 2019. doi: https://doi.org/10.1016/j.ifacol.2019.11.281.

[16] Rasmus Steffensen, Agnes Christine Gabrys, and Florian Holzapfel. Flight Envelope Protections Using Phase Plane Limits and Backstepping Control. In *5th CEAS Conference on Guidance, Navigation and Control (EuroGNC 2019)*. CEAS, Apr. 2019.

[17] Ricardo de Castro. Safe and high-performance control allocation. *IEEE Transactions on Automatic Control*, 67(6):3120–3127, 2022. doi: 10.1109/TAC.2021.3096882.

[18] F. Holzapfel. *Nonlinear adaptive control of an unmanned aerial vehicle (Original German Title: Nichtlineare adaptive Regelung eines unbemannten Fluggerätes)*. PhD thesis, Technische Universität München, München, Germany, June 2004. `https://mediatum.ub.tum.de/?id=601905`.

[19] J.-J. E. Slotine and Weiping Li. *Applied Nonlinear Control*. Prentice Hall, Englewood Cliffs, N.J, 1991. ISBN: 978-0-13-040890-7.

[20] Aaron D. Ames, Jessy W. Grizzle, and Paulo Tabuada. Control barrier function based quadratic programs with application to adaptive cruise control. In *53rd IEEE Conference on Decision and Control*, pages 6271–6278, Los Angeles, CA, USA, Dec. 2014. IEEE. ISBN: 978-1-4673-6090-6 978-1-4799-7746-8 978-1-4799-7745-1. doi: 10.1109/CDC.2014.7040372.

[21] Aaron D. Ames, Xiangru Xu, Jessy W. Grizzle, and Paulo Tabuada. Control Barrier Function Based Quadratic Programs for Safety Critical Systems. *IEEE Transactions on Automatic Control*, 62(8):3861–3876, Aug. 2017. ISSN: 0018-9286, 1558-2523. doi: 10.1109/TAC.2016.2638961.

[22] Wei Xiao and Calin Belta. High-Order Control Barrier Functions. *IEEE Transactions on Automatic Control*, 67(7):3655–3662, July 2022. ISSN: 0018-9286, 1558-2523, 2334-3303. doi: 10.1109/TAC.2021.3105491.

[23] Yizong Chen, Zhiqiang Miao, Weiwei Zhan, and Yaonan Wang. Safety-Critical Control for Underwater Vehicles with Model Uncertainties and External Disturbances. In *2024 18th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pages 568–573, Dubai, United Arab Emirates, Dec. 2024. IEEE. ISBN: 979-8-3315-1849-3. doi: 10.1109/ICARCV63323.2024.10821583.

[24] Ola Härkegård and S. Torkel Glad. Resolving actuator redundancy—optimal control vs. control allocation. *Automatica*, 41(1):137–144, 2005. ISSN: 0005-1098. doi: https://doi.org/10.1016/j.automatica.2004.09.007.

[25] Guillermo P. Falconí and Florian Holzapfel. Adaptive fault tolerant control allocation for a hexacopter system. In *2016 American Control Conference (ACC)*, pages 6760–6766. IEEE, 2016. ISBN: 978-1-4673-8682-1. doi: 10.1109/ACC.2016.7526736.

[26] Hangxu Li, Simon Hafner, Stephan Myschik, Haichao Hong, and Florian Holzapfel. Rank-One-Update-Based Efficient Recursive Least-Square Filter for Adaptive Control Allocation. *Journal of Guidance, Control, and Dynamics*, 48(9):2118–2127, Sept. 2025. ISSN: 0731-5090, 1533-3884. doi: 10.2514/1.G008814.

[27] Simon Hafner, Andrea Dan Ryals, Hangxu Li, Lorenzo Pollini, and Florian Holzapfel. Kalman Filter Based Adaptive Control Allocation. In *2024 32nd Mediterranean Conference on Control and Automation (MED)*, pages 113–118, Chania - Crete, Greece, June 2024. IEEE. doi: 10.1109/MED61351.2024.10566161.

[28] Tor A. Johansen and Thor I. Fossen. Control allocation—A survey. *Automatica*, 49(5):1087–1103, 2013. ISSN: 00051098. doi: 10.1016/j.automatica.2013.01.035.

[29] Simon Hafner, Stephan Myschik, and Florian Holzapfel. Accelerating Sequential Least Squares Active Set Control Allocation. *Control Engineering Practice*, 166(January 2026):106621, 2026. ISSN: 0967-0661. doi: 10.1016/j.conengprac.2025.106621.