Delay-Certified Readiness Barriers for Overactuated Multirotor Allocation Without Lipschitz Constants

Giuseppe Silano, *Senior Member, IEEE*

arXiv:2611.03490v1 [cs.RO] 2 Nov 2026

***Abstract*— Overactuated multirotors can produce the same wrench with many rotor-speed combinations, some of which leave the vehicle unable to change its wrench quickly, e.g., to reject a gust. Readiness barriers keep this ability, the readiness, above a floor, but only in continuous time; with sampled commands and varying speed-controller delays, the available robustness bound covers less than one millisecond. We remove this limit by exploiting the structure of the problem: each rotor responds monotonically and in closed form to its own torque, so the speeds reachable under uncertain switching times lie in a cheaply computed box, and the lowest readiness over that box sits at a corner known in advance. This yields a Lipschitz-free test that a command keeps the readiness above its floor until the next sample, plus a fallback torque that keeps the vehicle on a certified plan. In hexarotor simulations with 2–40 ms delays and ± 2 ms jitter, the filter never violated the floor. At 10 ± 2 ms, it matched a delay-compensating predictor within the design envelope and cost 2–30% more tracking error beyond it, where the predictor alone violated the floor in 2 to 16 of 16 missions.**

***Index Terms*— Control barrier functions, sampled-data systems, input delay, overactuated multirotors.**

I. Introduction

Overactuated multirotors carry more rotors than the components of the wrench (the force and moment acting on the body) they control, which enables full-pose tracking and physical interaction [1], [2]. A continuum of rotor speeds then produces the commanded wrench, and the allocator selects one, typically by minimizing effort subject to actuator limits [3], [4]. Effort minimization, however, is blind to the authority to *change* the wrench. Thrust and drag scale quadratically with spin rate [5], so a slow rotor has a shallow thrust slope and a fast one has spent its torque on drag; either way, the vehicle cannot respond quickly to a disturbance.

The drag-aware co-metric of [6] makes this authority measurable. A rotor changes its thrust at a rate given by its thrust slope, which grows with speed, times the acceleration it can still produce in either direction, which shrinks as drag absorbs its torque. The co-metric maps these rates through the allocation matrix to the set of achievable wrench rates, whose log-volume we call the *readiness*, in the spirit of manipulability [7]: it is low when a rotor is nearly stopped or saturated, and $-\infty$ when some wrench component can no longer change. Rather than maximizing it, which leads to discontinuous allocations, our previous work [8] keeps it above a floor with a Control Barrier Function (CBF) [9], [10] enforced by a torque-level Quadratic Program (QP). This *readiness barrier* certifies the floor, but, like most CBF guarantees, in continuous time; the present paper extends it to sampled, delayed implementations.

Flight controllers, however, hold each command for a sampling period and send it through Electronic Speed Controllers (ESCs) whose delay varies from sample to sample. Between two samples the barrier condition is not enforced, and with delay the filter acts on a past state. In [8], simulations of a hexarotor showed that the filter tolerates ESC delays of 1–10 ms, but its robustness bound [8, Prop. 13], built on Lipschitz constants of the barrier that are worst cases over the whole certified set, certifies only 0.63 ms there: the guarantee lags behind the observed robustness. At longer delays, or with more demanding maneuvers, it can violate the floor: at 40 ms on a nominal mission, and from 5 ms when the maneuver demands more than the floor was designed for (Section V). This paper closes the gap between what is observed and what is certified.

Our key observation is that the readiness barrier has enough structure to be checked exactly between samples, without Lipschitz constants. Each rotor responds to its own torque independently, monotonically, and in closed form, so the speeds reachable under any admissible switching times lie in a box whose corners are computed directly. Moreover, since a slow rotor lacks thrust slope and a fast one lacks torque margin, the contribution of each rotor to the readiness first grows and then shrinks with its speed (Lemma 2 and Fig. 1(c)), so the lowest readiness over the box sits at a corner known in advance. A command is then accepted only if the readiness provably stays above the floor until the next sample. Our contributions are:

1) An exact, closed-form minimum of the readiness over a box of rotor speeds (Proposition 3).
2) A delay certificate for bounded, order-preserving, per-ESC delays, with a hold torque that keeps the last certified plan safe and makes the test recursively feasible (Theorem 5), and a bound tracing its conservatism to the jitter and the command variation (Corollary 6).
3) An extension to bounded drag and speed-measurement errors (Proposition 7).
4) A certified allocation filter (Algorithm 1), compared on the hexarotor of [8] with predictor-based and Lipschitz-margin filters (Section V).

[1]Giuseppe Silano is with the Department of Power Generation Technologies and Materials, Ricerca sul Sistema Energetico (RSE), 20134 Milan, Italy, and with the Multi-robot Systems (MRS) group, Department of Cybernetics, Czech Technical University, 12135 Prague, Czech Republic (e-mail: giuseppe.silano@fel.cvut.cz).

This work was partially funded by the research fund for the Italian Electrical System (decree n. 388, Nov. 6th, 2024) and the GAČR project no. 27-00437M.

The authors used Google Gemini 4 and Anthropic Claude Opus 5.5 for proofreading and exposition refinement, but independently verified all outputs and assumes full responsibility for the article's content.

II. Related Work

Allocation and actuator authority. Allocation is classically a constrained least-squares or linear program [3]; [2] classifies multirotors by their input allocation and [4] reviews tilting platforms. These methods make the wrench attainable but do not preserve the authority to change it. Barriers on actuator quantities appear in CBF-based allocation for obstacle avoidance [11] and in servo-rate allocation for tiltrotors [12]; [8] and [13] floor the readiness with fixed and articulated rotors, and matrix CBFs [14] offer a directional alternative to the log-determinant.

CBFs under sampling and delay. CBF filters [9], [10] are designed in continuous time; for the readiness barrier, which has relative degree one, the issue is their sampled, delayed implementation. Sampled-data methods tighten the barrier condition with margins from global or local bounds [15], use approximate discrete-time models [16], or certify a given controller over the hold interval [17]; the local conditions of [15] need no Lipschitz constant but a supremum over the reachable set. For input delay, CBFs act on a predicted state [18], [19], made robust to prediction errors by input-to-state safety [20], robust formulations [19], delay estimation [21], or integral CBFs [22]; time-varying delays are treated in [23], and interval analysis handles measurement and actuation uncertainty [24].

Reachability and backup controllers. Monotone and mixed-monotone systems give cheap interval over-approximations of reachable sets [25], [26]. Backup CBFs integrate a backup controller over a horizon [27]; [28] adds sampled data, state uncertainty, and a known delay, and [29] bounds backup trajectories by mixed-monotone boxes and minimizes a concave barrier over their 2^n vertices.

Positioning. We specialize these ingredients: for general dynamics, the reachable set and the extremum of the barrier over it must be approximated, whereas for readiness barriers the set is a box computed in closed form, the nonconcave barrier is minimized exactly at one known vertex, and the infinite tail of the backup needs no integration.

III. Problem Formulation

This section recalls the readiness barrier of [8] and shows why sampling and delay break its guarantee.

A. Model and readiness barrier

We use the model and notation of [8]. An overactuated multirotor with n rotors produces the wrench $\mathbf{w} = \mathbf{A}\boldsymbol{\phi}(\mathbf{v}) \in \mathbb{R}^m$, $n > m$, with $\boldsymbol{\phi}(\mathbf{v}) = \mathbf{v} \odot |\mathbf{v}|$ (componentwise $\phi(v) = v|v|$), and the spin rate v_i of rotor i obeys

$$m_i \dot{v}_i = -b_i v_i |v_i| + \tau_i, \qquad |\tau_i| \le \bar{\tau}_i, \tag{1}$$

where $m_i > 0$ is the inertia of motor and propeller, $b_i > 0$ the aerodynamic drag coefficient, and τ_i the motor torque, bounded by $\bar{\tau}_i > 0$; in vector form, $\dot{\mathbf{v}} = \mathbf{f}(\mathbf{v}) + \mathbf{M}^{-1}\boldsymbol{\tau}$ with $\mathbf{M} = \operatorname{diag}(m_i)$, and $\operatorname{rank}\mathbf{A} = m$. The motor time scale is $T_0 = m_i/\sqrt{\bar{\tau}_i b_i}$. The symmetric acceleration capacity $\bar{a}_i(v_i) = (\bar{\tau}_i - b_i v_i^2)/m_i$ is the acceleration rotor i can still produce in either direction; it is positive on the open box $\mathcal{V} = \{\mathbf{v} : |v_i| < v_i^{\mathrm{sat}}\}$, where $v_i^{\mathrm{sat}} := \sqrt{\bar{\tau}_i/b_i}$ is the saturation speed, at which drag absorbs the full torque, and admissible torques render $\mathcal{V}$ forward invariant; and on $\mathcal{V}$ the drag-aware co-metric and readiness level are [6], [8]

$$\begin{gathered}\mathbf{D}(\mathbf{v}) = 4\mathbf{A}\boldsymbol{\Psi}(\mathbf{v})\mathbf{A}^\top, \quad \psi_i(v_i) = v_i^2 \bar{a}_i(v_i)^2, \\ L(\mathbf{v}) = \ln\det \mathbf{D}(\mathbf{v}),\end{gathered} \tag{2}$$

with $\boldsymbol{\Psi} = \operatorname{diag}(\psi_i)$ and $\mathbf{A}_{\bullet i}$ the ith column of $\mathbf{A}$. The weight ψ_i is a quarter of the squared largest rate, $2v_i\bar{a}_i$, at which rotor i can change its thrust $\phi(v_i)$ in either direction, so $\mathbf{D} = \mathbf{J}\operatorname{diag}(\bar{a}_i^2)\mathbf{J}^\top$, with $\mathbf{J}$ defined below, is the shape matrix of the ellipsoid $\{\mathbf{J}\operatorname{diag}(\bar{a}_i)\mathbf{u} : \|\mathbf{u}\| \le 1\}$ of wrench rates available at $\mathbf{v}$, and the readiness L is, up to a constant, twice the logarithm of its volume. Each ψ_i vanishes at standstill (no thrust slope) and at saturation (no torque margin) and peaks at the sweet spot $v_i^\star = v_i^{\mathrm{sat}}/\sqrt{3}$ (Lemma 2). Given a floor $\bar{\ell}$, the barrier is $h(\mathbf{v}) = L(\mathbf{v}) - \bar{\ell}$ and the certified set is $\mathcal{C} = \{\mathbf{v} \in \mathcal{V} : h(\mathbf{v}) \ge 0\}$. The floor is chosen in the window $\ell^\dagger < \bar{\ell} \le L^{\mathrm{op}}$. The lower end is the readiness left when the most dispensable rotor stops, so a floor above it keeps every rotor spinning; the upper end is the best readiness that exact tracking guarantees throughout the mission, so a floor below it does not force tracking errors. Formally, $\ell^\dagger = L^{\max} - \min_k \Delta_k$, where $\Delta_k = -\ln(1-\lambda_k)$ is the readiness lost when rotor k stops, with leverage $\lambda_k = \psi_k(v_k^\star)\mathbf{A}_{\bullet k}^\top \mathbf{S}^{-1}\mathbf{A}_{\bullet k}$, $\mathbf{S} = \sum_i \psi_i(v_i^\star)\mathbf{A}_{\bullet i}\mathbf{A}_{\bullet i}^\top$, $L^{\max} = \ln\det(4\mathbf{S})$, and $L^{\mathrm{op}} = \min_{\mathbf{w}\in\mathcal{W}} \max_{\mathbf{v}:\mathbf{A}\boldsymbol{\phi}(\mathbf{v})=\mathbf{w}} L(\mathbf{v})$ is the worst-case achievable readiness over the mission wrenches $\mathcal{W}$ [8, Thm. 1, eq. (10)]. Because $\bar{\ell} > \ell^\dagger$, trajectories in $\mathcal{C}$ keep the sign of every rotor speed, which by symmetry of (1) we take positive.

The allocation filter of [8] solves, at the measured state, the QP

$$\begin{aligned}\min_{\boldsymbol{\tau},\boldsymbol{\delta}} \ & \|\boldsymbol{\tau} - \boldsymbol{\tau}_{\mathrm{ref}}\|^2 + \rho\|\boldsymbol{\delta}\|^2 \\ \text{s.t. } & \nabla h(\mathbf{v})^\top \mathbf{a} \ge -\alpha h(\mathbf{v}), \quad \mathbf{J}(\mathbf{v})\mathbf{a} = \boldsymbol{\mu} + \boldsymbol{\delta}, \\ & |\boldsymbol{\tau}| \le \bar{\boldsymbol{\tau}}, \quad \mathbf{a} = \mathbf{f}(\mathbf{v}) + \mathbf{M}^{-1}\boldsymbol{\tau},\end{aligned} \tag{3}$$

where $\mathbf{J} = 2\mathbf{A}\operatorname{diag}(|\mathbf{v}|)$, $\boldsymbol{\mu} = \dot{\mathbf{w}}_{\mathrm{des}} + k_w(\mathbf{w}_{\mathrm{des}} - \mathbf{A}\boldsymbol{\phi}(\mathbf{v}))$ is the task demand for the desired wrench $\mathbf{w}_{\mathrm{des}}$, $\boldsymbol{\tau}_{\mathrm{ref}}$ is a nominal allocation (in [8] and here, the solution of (3) without the barrier row and with $\boldsymbol{\tau}_{\mathrm{ref}} = \mathbf{0}$, i.e., the effort-minimizing allocation), and $\alpha, k_w, \rho > 0$ are gains. In words, the QP stays close to the nominal allocation, tracks the wrench demand up to the penalized slack $\boldsymbol{\delta}$, and lets h decrease at most at rate αh, so the floor has priority over tracking. Applied in continuous time, (3) renders $\mathcal{C}$ forward invariant [8, Thm. 7].

B. Sampled data and actuation delay

On a flight controller, (3) is solved at $t_j = j\Delta t$ and its output passes through the ESCs with a delay that varies from sample to sample. We model this as follows.

Assumption 1. *The command $\boldsymbol{\tau}_j$ (components $\tau_{j,i}$), computed at t_j from the exact state $\mathbf{v}(t_j)$, is applied to rotor i on $[\sigma_{j,i}, \sigma_{j+1,i})$, where the unknown switching instants satisfy $\sigma_{j,i} \in [t_j + \underline{d},\, t_j + \bar{d}]$, $0 \le \underline{d} \le \bar{d}$, and $\sigma_{j,i} \le \sigma_{j+1,i}$*

(commands are not reordered; each ESC may switch at its own instant). The commands issued before $t = 0$ are known. The jitter width is $\Delta d := \overline{d} - \underline{d}$.

In words, each command reaches each ESC after an unknown delay between $\underline{d}$ and $\overline{d}$, the ESCs may switch at different instants, and no command overtakes an earlier one. The lower bound $\underline{d}$ includes the computation time of the filter. For a filter that ignores the delay, [8, Prop. 13, Cor. 14] absorbs a delay d by raising the floor by $\eta(d) = (\alpha\bar{H} + KVd)d + KVd/\alpha$, where $\bar{H} = L^{\max} - \bar{\ell}$ and K, V bound the second derivatives of h and the rotor accelerations on $\mathcal{C}$; keeping the raised floor below L^{op} caps the certified delay at $d^\star = 0.63$ ms for the hexarotor of [8]. A predictor removes the nominal delay, but its error under jitter again calls for a margin. We therefore pose the following problem.

Problem 1. *Under Assumption 1, given $\mathbf{v}_k$ and the committed commands $\{\boldsymbol{\tau}_j\}_{j<k}$, decide without Lipschitz constants whether a candidate $\boldsymbol{\tau}_k$ guarantees $h(\mathbf{v}(t)) \geq 0$ on $[t_k, t_{k+1}]$ for every admissible realization of the switching instants, and ensure that some candidate always passes.*

IV. A Lipschitz-Free Delay Certificate

This section solves Problem 1. The solution rests on three structural facts: every rotor is a scalar, monotone system with a closed-form flow; each ψ_i is unimodal; and L is monotone in every ψ_i.

A. Three structural facts

Lemma 1 (Monotone rotor flow)**.** *Under a constant torque τ_i, the solution $t \mapsto v_i(t)$ of (1) is monotone in t, and $v_i(t)$ is nondecreasing in $v_i(0)$ and in τ_i. For $v_i(0) = v_0 \geq 0$, let $v^\infty = \sqrt{|\tau_i|/b_i}$ and $\omega = \sqrt{|\tau_i| b_i}/m_i$. If $\tau_i > 0$, $v_i(t) = v^\infty \tanh(\omega t + \operatorname{artanh}(v_0/v^\infty))$ for $v_0 < v^\infty$, with $\coth$ and arcoth for $v_0 > v^\infty$; if $\tau_i = 0$, $v_i(t) = v_0/(1 + b_i v_0 t/m_i)$; if $\tau_i < 0$, $v_i(t) = v^\infty \tan(\arctan(v_0/v^\infty) - \omega t)$ up to the zero crossing t_z, and $-v^\infty \tanh(\omega(t - t_z))$ after it. For $v_0 < 0$, $v_i(t; v_0, \tau_i) = -v_i(t; -v_0, -\tau_i)$.*

Proof. The right-hand side of (1) is locally Lipschitz in v_i and nondecreasing in τ_i. Solutions of a scalar autonomous equation are monotone in time, and order is preserved in the initial condition and in the input by the comparison principle for scalar (hence monotone) systems [25]. The closed forms follow by substitution into (1). □

Lemma 1 makes every rotor speed under a piecewise-constant torque computable in closed form, and ordered: more torque or a faster start never yields a slower rotor. We write $\boldsymbol{\Phi}_t(\mathbf{v}, \boldsymbol{\tau})$ for the flow of (1) under the constant torque $\boldsymbol{\tau}$. Since $\dot{v}_i \leq (\bar{\tau}_i - b_i v_i^2)/m_i$, an admissible torque never drives a rotor above v_i^{sat}.

Lemma 2 (Unimodality)**.** *On $[0, v_i^{\mathrm{sat}}]$, ψ_i is increasing on $[0, v_i^\star]$ and decreasing on $[v_i^\star, v_i^{\mathrm{sat}}]$, with $v_i^\star = v_i^{\mathrm{sat}}/\sqrt{3}$. Hence, for $0 \leq l \leq u \leq v_i^{\mathrm{sat}}$, $\min_{v \in [l,u]} \psi_i(v) = \min\{\psi_i(l), \psi_i(u)\}$.*

Proof. $\psi_i'(v) = 2v\bar{a}_i(v)(\bar{\tau}_i - 3b_i v^2)/m_i$ [8, Lemma 3], which is positive on $(0, v_i^\star)$ and negative on $(v_i^\star, v_i^{\mathrm{sat}})$. A unimodal function attains its minimum over an interval at an endpoint. □

For a box $\mathcal{B} = [\underline{\mathbf{v}}, \overline{\mathbf{v}}]$ of spin rates with $\overline{\mathbf{v}} \leq v^{\mathrm{sat}}$, define

$$\underline{\psi}_i(\mathcal{B}) = \begin{cases} \min\{\psi_i(\underline{v}_i), \psi_i(\overline{v}_i)\}, & \underline{v}_i > 0, \\ 0, & \underline{v}_i \leq 0, \end{cases} \tag{4}$$

and $\underline{L}(\mathcal{B}) = \ln\det\left(4\mathbf{A}\,\mathrm{diag}(\underline{\psi}(\mathcal{B}))\mathbf{A}^\top\right)$, with $\ln\det = -\infty$ for a singular matrix.

Proposition 3 (Exact box minimum)**.** *With $\underline{\psi}$ as in (4), if $\underline{\mathbf{v}} > \mathbf{0}$ and $\overline{\mathbf{v}} \leq v^{\mathrm{sat}}$, then $\min_{\mathbf{v} \in \mathcal{B}} L(\mathbf{v}) = \underline{L}(\mathcal{B})$. If $\underline{v}_i \leq 0$ for some i, then $\underline{L}(\mathcal{B}) \leq \ell^\dagger < \bar{\ell}$.*

Proof. For $\mathbf{v} \in \mathcal{B}$, Lemma 2 gives $\psi_i(v_i) \geq \underline{\psi}_i(\mathcal{B})$ for every i, so, by (2), $\mathbf{D}(\mathbf{v}) \succeq 4\mathbf{A}\,\mathrm{diag}(\underline{\psi})\mathbf{A}^\top$ and $L(\mathbf{v}) \geq \underline{L}(\mathcal{B})$ by monotonicity of $\ln\det$ on the positive semidefinite cone. Equality holds at the vertex that selects, for each i, the endpoint attaining $\underline{\psi}_i$; this vertex lies in $\mathcal{B}$. If $\underline{\psi}_i = 0$, then $\underline{\psi}_j \leq \psi_j(v_j^\star)$ for $j \neq i$ gives $4\mathbf{A}\,\mathrm{diag}(\underline{\psi})\mathbf{A}^\top \preceq 4(\mathbf{S} - \psi_i(v_i^\star)\mathbf{A}_{\bullet i}\mathbf{A}_{\bullet i}^\top)$, and the matrix determinant lemma yields $\underline{L} \leq L^{\max} + \ln(1 - \lambda_i) = L^{\max} - \Delta_i \leq \ell^\dagger$, the dropout gap of [8, Thm. 1]. □

In words, the lowest readiness over a box is at the corner where each rotor sits at the end of its range with the smaller weight; finding it costs $2n$ scalar evaluations and one $m \times m$ log-determinant instead of a search over 2^n corners. Any conservatism thus lies in the box, not in the minimization.

B. Reachable boxes under jittered delay

Fix a step k. Commands $\boldsymbol{\tau}_j$ with $j < k$ are already committed, $\boldsymbol{\tau}_k$ is the candidate, and for $j > k$ we set $\boldsymbol{\tau}_j = \boldsymbol{\tau}^{\mathrm{h}}$, the *sweet-spot hold torque*

$$\tau_i^{\mathrm{h}} = b_i (v_i^\star)^2 = \bar{\tau}_i/3, \tag{5}$$

which makes $v^\star$ an equilibrium of (1). Under $\boldsymbol{\tau}^{\mathrm{h}}$, every rotor moves toward the speed of maximum readiness, so readiness can only grow and nothing after the window needs checking. This hypothesis on future commands only decides whether $\boldsymbol{\tau}_k$ is accepted; it plays no role in the guarantee over the current interval (Theorem 5(i)).

Let $\mathcal{I}(t) = \{j : t_j + \underline{d} \leq t < t_{j+1} + \overline{d}\}$ and define the torque envelopes $\tau_i^+(t) = \max_{j \in \mathcal{I}(t)} \tau_{j,i}$ and $\tau_i^-(t) = \min_{j \in \mathcal{I}(t)} \tau_{j,i}$. Let $\mathbf{v}^\pm$ be the solutions of (1) from $\mathbf{v}^\pm(t_k) = \mathbf{v}_k := \mathbf{v}(t_k)$ under $\boldsymbol{\tau}^\pm$.

Lemma 4 (Envelopes)**.** *Under Assumption 1, the torque applied to rotor i at time t is $\tau_{j,i}$ for some $j \in \mathcal{I}(t)$, and $\mathbf{v}^-(t) \leq \mathbf{v}(t) \leq \mathbf{v}^+(t)$ for all $t \geq t_k$ and every admissible realization of $\{\sigma_{j,i}\}$.*

Proof. If $\tau_{j,i}$ is applied to rotor i at t, then $\sigma_{j,i} \leq t < \sigma_{j+1,i}$, so $t_j + \underline{d} \leq t$ and $t < t_{j+1} + \overline{d}$; hence $j \in \mathcal{I}(t)$ and $\tau_i^-(t) \leq \tau_i(t) \leq \tau_i^+(t)$ for every i. The rotors are decoupled and each is monotone in its input (Lemma 1), so the comparison principle [25] gives the ordering. □

The envelopes (Figure 1) are piecewise constant, with breakpoints in $\{t_j + \underline{d}, t_j + \overline{d}\}_j$. Let $t_k = \theta_0 < \theta_1 <$

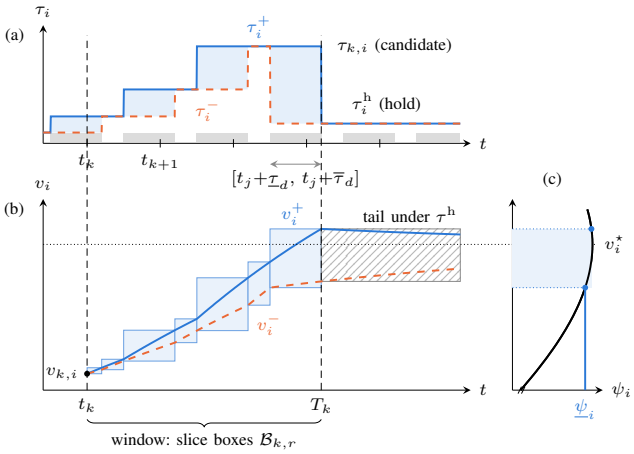


Fig. 1: Certificate of one rotor at step k (illustrative timing). (a) Commands switch at unknown instants in the gray intervals; $\tau_i^{\pm}$ take the extreme command. (b) The envelopes $v_i^{\pm}$ and slice boxes $\mathcal{B}_{k,r}$ enclose every admissible trajectory; after T_k the hold torque drives both toward $v_i^\star$. (c) ψ_i is unimodal, so its minimum over a box side is at an endpoint.

$\cdots < \theta_R = T_k$ collect these breakpoints within the window $[t_k, T_k]$, $T_k := t_{k+1} + \overline{d}$. On each slice $[\theta_r, \theta_{r+1}]$ the envelopes are constant, so $\mathbf{v}^{\pm}$ are monotone there (Lemma 1) and the reachable states lie in the slice box

$$\mathcal{B}_{k,r} = \big[\mathbf{v}^-(\theta_r) \wedge \mathbf{v}^-(\theta_{r+1}),\ \mathbf{v}^+(\theta_r) \vee \mathbf{v}^+(\theta_{r+1})\big], \quad (6)$$

with $\wedge, \vee$ the componentwise minimum and maximum. The step-k certificate, the lowest readiness the vehicle can reach over the window, is

$$\mathcal{L}_k(\boldsymbol{\tau}_k) = \min_{0 \le r < R} \underline{L}(\mathcal{B}_{k,r}). \quad (7)$$

It needs the exact flows of Lemma 1 at the $R+1$ breakpoints and one $m \times m$ log-determinant per slice.

Theorem 5 (Delay certificate)**.** *Let Assumption 1 hold and $\bar{\ell} > \ell^\dagger$.*

(i) *If $\mathcal{L}_k(\boldsymbol{\tau}_k) \ge \bar{\ell}$, then $h(\mathbf{v}(t)) \ge 0$ for all $t \in [t_k, t_{k+1}]$, for every admissible realization of the switching instants and whatever commands are issued after t_k.*

(ii) *If $\mathcal{L}_k(\boldsymbol{\tau}_k) \ge \bar{\ell}$, then $\mathcal{L}_{k+1}(\boldsymbol{\tau}^{\mathrm{h}}) \ge \bar{\ell}$.*

(iii) *If $\mathcal{L}_0(\boldsymbol{\tau}_0) \ge \bar{\ell}$ and every later command either passes its certificate or equals $\boldsymbol{\tau}^{\mathrm{h}}$, then $h(\mathbf{v}(t)) \ge 0$ for all $t \ge 0$; by (ii), a certified command exists at every step.*

Proof. (i) For $t < t_{k+1}$, $j \in \mathcal{I}(t)$ implies $t_j \le t - \underline{d} < t_{k+1}$, i.e., $j \le k$: only committed commands and $\boldsymbol{\tau}_k$ act on $[t_k, t_{k+1}]$, so the hold hypothesis is immaterial there. By Lemma 4 and the monotonicity of $\mathbf{v}^{\pm}$ on each slice, $\mathbf{v}(t) \in \mathcal{B}_{k,r}$ in (6) for $t \in [\theta_r, \theta_{r+1}]$. Since $\mathcal{L}_k \ge \bar{\ell} > \ell^\dagger$, Proposition 3 excludes $\underline{v}_i \le 0$ and gives $L(\mathbf{v}(t)) \ge \underline{L}(\mathcal{B}_{k,r}) \ge \bar{\ell}$.

(ii) With $\boldsymbol{\tau}_{k+1} = \boldsymbol{\tau}^{\mathrm{h}}$, steps k and $k+1$ use the same command sequence, so $\boldsymbol{\tau}^{\pm}(t)$ coincide for $t \ge t_{k+1}$. By Lemma 4, $\mathbf{v}_k^-(t_{k+1}) \le \mathbf{v}_{k+1} \le \mathbf{v}_k^+(t_{k+1})$, where $\mathbf{v}_k^{\pm}$, $\mathbf{v}_{k+1}^{\pm}$ are the envelopes of the two steps, and by Lemma 1 $\mathbf{v}_k^- \le \mathbf{v}_{k+1}^- \le \mathbf{v}_{k+1}^+ \le \mathbf{v}_k^+$ for $t \ge t_{k+1}$. On $[t_{k+1}, T_k]$ the breakpoints of step $k+1$ are those of step k plus t_{k+1}, so every slice box of step $k+1$ lies in one of step k. On $[T_k, T_{k+1}]$, $\boldsymbol{\tau}^{\pm} \equiv \boldsymbol{\tau}^{\mathrm{h}}$, under which each rotor moves monotonically toward $v_i^\star$ without crossing it, so $\psi_i(v_{k+1,i}^{\pm}(t))$ is nondecreasing (Lemma 2) and every $\underline{\psi}_i$ there is at least $\underline{\psi}_i(\mathcal{B}_{k,R-1})$, which contains $[v_{k+1,i}^-(T_k), v_{k+1,i}^+(T_k)]$; Loewner monotonicity gives $\underline{L} \ge \underline{L}(\mathcal{B}_{k,R-1}) \ge \bar{\ell}$.

(iii) Let $k' \le k$ be the last step whose command passed. All commands after k' equal $\boldsymbol{\tau}^{\mathrm{h}}$, the hypothesis of $\mathcal{L}_{k'}$, so the slice boxes of step k' contain $\mathbf{v}(t)$ on $[t_{k'}, T_{k'}]$ (Lemma 4), and after $T_{k'}$ every rotor moves monotonically toward $v_i^\star$ from a point of $\mathcal{B}_{k',R-1}$, as in (ii). Hence $L(\mathbf{v}(t)) \ge \bar{\ell}$ until a new command passes, and induction over the passing steps concludes. □

In practice, (i) says that an accepted command is safe until the next sample whatever the ESCs do, (ii) that the hold torque is always available as a certified fallback, and (iii) that the closed loop never leaves $\mathcal{C}$: a passing certificate covers its command followed by $\boldsymbol{\tau}^{\mathrm{h}}$ forever. Theorem 5 does not involve L^{op}, but it needs a certified start: a state in $\mathcal{C}$ is not enough, because commands already in the pipeline may leave $\mathcal{C}$ before any new command acts.

Corollary 6 (Source of conservatism)**.** *For $t \in [t_k, T_k]$ and every i,*

$$0 \le v_i^+(t) - v_i^-(t) \le \frac{\Delta d}{m_i} \sum_{j \in \mathcal{K}_k(t)} |\tau_{j,i} - \tau_{j-1,i}|, \quad (8)$$

where $\mathcal{K}_k(t) = \{j : [t_j + \underline{d}, t_j + \overline{d}) \cap [t_k, t] \ne \emptyset\}$ and $\boldsymbol{\tau}_j = \boldsymbol{\tau}^{\mathrm{h}}$ for $j > k$. In particular, for a known constant delay ($\Delta d = 0$), $\mathbf{v}^+ = \mathbf{v}^-$ is the exact trajectory under the hold hypothesis.

Proof. The spread $e_i = v_i^+ - v_i^-$ starts at zero and, since ϕ is increasing, $m_i \dot{e}_i \le \tau_i^+ - \tau_i^-$, which is at most the sum of $|\tau_{j,i} - \tau_{j-1,i}|$ over consecutive pairs in $\mathcal{I}(t)$. A pair $j-1, j$ is in $\mathcal{I}(t)$ exactly when $t \in [t_j + \underline{d}, t_j + \overline{d})$, of length Δd; integrating gives (8). □

Corollary 6 explains when the certificate is conservative. The envelope spread, the part of the box width due to the uncertain switching, is Δd times the total variation of the commands still in flight (about $\overline{d}/\Delta t$ of them), the candidate, and the transition $|\tau_i^{\mathrm{h}} - \tau_{k,i}|$ to the hold torque. It vanishes for a known delay, leaving only the motion within each slice, and for $\Delta d > 0$ grows with the delay through the number of commands in flight. Smooth commands therefore make the certificate tighter (Section V).

C. Measurement error and drag uncertainty

The certificate uses only monotonicity, so bounded model and measurement errors enter as wider envelopes. Suppose the filter measures $\hat{\mathbf{v}}_k$ with $|\hat{v}_{k,i} - v_{k,i}| \le \varepsilon$ and the plant drag satisfies $b_i \in [b_i^-, b_i^+]$. For $v_i > 0$ the flow of (1) is nonincreasing in b_i. Integrate $\mathbf{v}^+$ from $\hat{\mathbf{v}}_k + \varepsilon\mathbf{1}$ with b_i^- and $\mathbf{v}^-$ from $\hat{\mathbf{v}}_k - \varepsilon\mathbf{1}$ with b_i^+. Under $\boldsymbol{\tau}^{\mathrm{h}}$ the plant

no longer settles at $v^\star$ but between $v_{i,-}^{\mathrm{eq}} = (\tau_i^{\mathrm{h}}/b_i^+)^{1/2}$ and $v_{i,+}^{\mathrm{eq}} = (\tau_i^{\mathrm{h}}/b_i^-)^{1/2}$, so we add the tail box $\mathcal{B}_{k,\infty} = [\mathbf{v}^-(T_k) \wedge \mathbf{v}_-^{\mathrm{eq}}, \mathbf{v}^+(T_k) \vee \mathbf{v}_+^{\mathrm{eq}}]$ to (7). In (4), $\bar{a}_i$ is clipped at zero, so that ψ_i stays unimodal on $[0,\infty)$ and boxes reaching beyond v_i^{sat} are rejected.

Proposition 7 (Robust certificate)**.** *Let $b_i^- > b_i/3$. With these envelopes and the tail box, Theorem 5(i) and (iii) hold for every plant drag in $[\mathbf{b}^-, \mathbf{b}^+]$ and every measurement error bounded by ε. If $\varepsilon = 0$, Theorem 5(ii) holds as well.*

Proof. A passing certificate has $\mathbf{v}^- > \mathbf{0}$ on the window. The true state starts in $[\hat{\mathbf{v}}_k - \varepsilon\mathbf{1}, \hat{\mathbf{v}}_k + \varepsilon\mathbf{1}]$ and, for positive speeds, the right-hand side of (1) is nonincreasing in b_i and nondecreasing in τ_i, so the comparison principle brackets it between $\mathbf{v}^{\pm}$: this gives (i). After T_k, each true rotor moves monotonically from $[v_i^-(T_k), v_i^+(T_k)]$ toward its equilibrium in $[v_{i,-}^{\mathrm{eq}}, v_{i,+}^{\mathrm{eq}}]$, so it stays in $\mathcal{B}_{k,\infty}$: this gives (iii). For (ii) with $\varepsilon = 0$, the slices on $[t_{k+1}, T_k]$ nest as in Theorem 5, and the step-$(k+1)$ tail box lies in $\mathcal{B}_{k,\infty}$. □

The certificate can pass only if $\underline{L}([\mathbf{v}_-^{\mathrm{eq}}, \mathbf{v}_+^{\mathrm{eq}}]) \geq \bar{\ell}$ (which needs $b_i^- > b_i/3$); for the hexarotor with $\pm 10\%$ drag this value is -10.16, far above the floors of Section V. Two remarks delimit the result. First, the certified floor is the nominal readiness (2) of the true speeds, not that of the perturbed plant. The worst-case weight $v_i^2 \max\{0, \bar{\tau}_i - b_i^+ v_i^2\}^2/m_i^2$ of [8, Thm. 10] is unimodal too and can replace ψ_i in (4); Section V reports the readiness of the true plant. Second, with $\varepsilon > 0$ the re-measured box at t_{k+1} may exceed the step-k envelopes by 2ε, so a fresh certificate may fail; applying $\boldsymbol{\tau}^{\mathrm{h}}$ then continues the last certified plan, which remains safe by (iii).

Any constant backup whose equilibrium box passes the certificate would also work, e.g., one that produces the current wrench; $\boldsymbol{\tau}^{\mathrm{h}}$ is the one whose tail never lowers the readiness.

D. Certified allocation filter

Algorithm 1 wraps the QP (3). The QP is solved at the state predicted at $t_k + \hat{d}$, with $\hat{d} = (\underline{d} + \overline{d})/2$, by integrating the committed commands with the exact flows. Its output is applied if it passes the certificate. Otherwise the command is blended toward $\boldsymbol{\tau}^{\mathrm{h}}$, which passes by Theorem 5(ii) on the exact model. Bisection on the blend factor $\gamma \in [0,1]$ keeps only certified points, so the applied command passes, although the smallest certified γ is not guaranteed: the certificate need not be monotone in γ, since blending shrinks the increment from $\boldsymbol{\tau}_k$ to $\boldsymbol{\tau}^{\mathrm{h}}$ in Corollary 6 but may enlarge the one from $\boldsymbol{\tau}_{k-1}$ to $\boldsymbol{\tau}_k$. If the QP fails (e.g., an infeasible barrier row at a noisy measurement) or $\boldsymbol{\tau}^{\mathrm{h}}$ fails a fresh check, $\boldsymbol{\tau}^{\mathrm{h}}$ is applied, which continues the last certified plan (Theorem 5(iii), Proposition 7). The fallback is task-agnostic and perturbs the wrench, in particular the collective; a wrench-preserving backup with a monotone tail is left for future work.

Algorithm 1 Delay-certified readiness filter (step k)

Require: $\mathbf{v}_k$, committed $\{\boldsymbol{\tau}_j\}_{j<k}$, $[\underline{d}, \overline{d}]$, $\bar{\ell}$, N_b; a passing certificate at some earlier step

$\hat{\mathbf{v}} \leftarrow$ exact flow of $\mathbf{v}_k$ to $t_k + \hat{d}$ under committed commands
$\boldsymbol{\tau}_{\mathrm{ref}} \leftarrow$ effort-minimizing allocation at $\hat{\mathbf{v}}$; $\boldsymbol{\tau}^{\mathrm{QP}} \leftarrow$ QP (3) at $\hat{\mathbf{v}}$
if the QP is solved and $\mathcal{L}_k(\boldsymbol{\tau}^{\mathrm{QP}}) \geq \bar{\ell}$ **then**
  return $\boldsymbol{\tau}^{\mathrm{QP}}$
end if
if $\mathcal{L}_k(\boldsymbol{\tau}^{\mathrm{h}}) < \bar{\ell}$ or the QP failed **then**
  return $\boldsymbol{\tau}^{\mathrm{h}}$ {retained plan}
end if
$\gamma^- \leftarrow 0$, $\gamma^+ \leftarrow 1$
for N_b iterations **do**
  $\gamma \leftarrow (\gamma^- + \gamma^+)/2$
  if $\mathcal{L}_k((1-\gamma)\boldsymbol{\tau}^{\mathrm{QP}} + \gamma\boldsymbol{\tau}^{\mathrm{h}}) \geq \bar{\ell}$ **then**
    $\gamma^+ \leftarrow \gamma$
  else
    $\gamma^- \leftarrow \gamma$
  end if
end for
return $(1-\gamma^+)\boldsymbol{\tau}^{\mathrm{QP}} + \gamma^+\boldsymbol{\tau}^{\mathrm{h}}$

V. Simulation Study

The simulations answer four questions:

- **Q1** How tight is the box bound (Section V)?
- **Q2** Does the filter keep the floor across delays and jitter, and at what cost (Sections V-A and V-B)?
- **Q3** How does it compare with Lipschitz margins (Section V-B)?
- **Q4** Does it survive model and measurement errors and independent ESC timing (Sections V-C and V-D)?

Platform and missions. We use the coplanar hexarotor of [8], overactuated with respect to its $m = 4$ controlled wrench components (collective thrust and three moments): $n = 6$ identical nondimensional motors $\bar{\tau}_i = b_i = m_i = 1$, with $T_0 = 0.32$ s, and $\mathbf{A}_{\bullet i} = (1,\, l\sin\beta_i,\, -l\cos\beta_i,\, \kappa_{\mathrm{y}}(-1)^{i+1})$ with $\beta_i = 30^\circ + 60^\circ(i-1)$, $l = 0.25$, and $\kappa_{\mathrm{y}} = 0.016$, so that $L^{\max} = -10.128$ and $\ell^\dagger = -11.226$. The mission is the roll-moment reversal of [8]: fixed collective c (fraction of the sweet-spot collective) and a sinusoidal roll moment of amplitude β (in units of $\phi(v_i^\star)/4$) and period $2T_0$, flown for $2T_0$. On *certifiable* missions the demand stays within the design envelope ($L^{\mathrm{op}} > \ell^\dagger$, computed on 41 points of one period) and the floor follows the rule of [8], $\bar{\ell} = \ell^\dagger + \frac{1}{2}(L^{\mathrm{op}} - \ell^\dagger)$. Since floors are fixed at design while maneuvers are commanded online, we also fly *readiness-priority* missions, whose demand exceeds the envelope: the floor $\bar{\ell} = \ell^\dagger + 0.15 > L^{\mathrm{op}}$ cannot be held while tracking exactly [8, Prop. 2], so every filter must trade tracking for readiness.

Implementation. The filter runs at $\Delta t = 2$ ms; the plant is integrated piecewise between switching instants (fourth-order Runge–Kutta, steps $\leq 10^{-4}T_0$). Each command switches at

$\sigma_{j,i} = \sigma_j = \max\{t_j + \hat{d} + \epsilon_j, \sigma_{j-1}\}$, with ϵ_j uniform in $[-\Delta d/2, \Delta d/2]$ (per-rotor instants in Section V-D). Runs start at the readiness-maximizing allocation of $\mathbf{w}_{\mathrm{des}}(0)$ with the queue at its equilibrium torque, and the first certificate passed in every run. All filters use the gains of [8] ($\alpha(h) = \alpha h$, $\alpha = 5$ unless stated, $k_w = 8$, $\rho = 10^4$), OSQP [30], and the same missions and jitter realizations; Algorithm 1 uses $N_b = 14$.

Baselines. All are built on (3). The *RB filter*, i.e., (3) at the measured state [8], shows what happens if the delay is ignored. *Pred*, i.e., (3) at the state predicted at $t_k + \hat{d}$ with exact flows (predictor feedback [19], [22] without robustness margin), isolates delay compensation. The others add the conventional safety margin ζ to the barrier row. *SD-Lip*: the RB filter with $\zeta = KV\Delta t$, as in the global conditions of [15]. *Pred-Lip*: Pred with $\zeta = KV(\Delta t + \Delta d/2)$, a heuristic allowance for the hold interval and half the jitter (it does not bound the prediction error, which accumulates as in (8)). For these two, K (a bound on $\|\nabla(\nabla h^\top \mathbf{f})\|_1 + \|\nabla^2 h\|_1 \max_i \bar{\tau}_i/m_i$) and V (on $\|\dot{\mathbf{v}}\|_\infty$) are sampled on $\mathcal{C}$: $KV = 212\,T_0^{-2}$, about half of the global $398\,T_0^{-2}$ of [8]. *Pred-Lip-G*: the same margin with the global constants. *Pred-Lip-Loc*: the same margin with $K_k V_k$ sampled at every step at $\mathbf{v}_k$ and 32 random points of the box of radius $V_{\max}(\Delta t + \bar{d})$ around it, $V_{\max} = \max_i 2\bar{\tau}_i/m_i$, which contains the speeds reachable over the window. None of these margins is a certificate or reproduces the robust predictors of [19], [22]; when the barrier row is infeasible, they apply the torque that maximizes $\dot{h}$.

Metrics. The worst margin $\min_t h$ over a run, recorded at every integration step, is what the filter guarantees: a negative value means that the readiness fell below the floor. The Root Mean Square (RMS) wrench error (Euclidean norm, nondimensional) relative to Pred, e_w (median over missions of the paired ratio), isolates the price of the extra safety, because every filter enforces the floor by giving up tracking and Pred is the cheapest filter that compensates the delay. The fallback rate shows how often the certificate overrides the QP.

Q1–*Tightness.* Over one hold interval without delay, on 400 random states, the gap between the box bound and the true minimum grows linearly in Δt, that of a second-order Lipschitz bound with the constant sampled on $\mathcal{C}$ quadratically: the Lipschitz bound is tighter at 0.5 ms (median gap 2.6×10^{-4} vs. 4.8×10^{-4} nats), the box bound 1.9 and 11 times tighter at 2 and 10 ms.

A. Delay and jitter sweep

Q2–Algorithm 1 keeps the floor at every delay and jitter tested, whereas ignoring the delay fails as the delay grows. Figure 2 sweeps the nominal delay from 2 to 40 ms on two missions with $c = 0.75$: a certifiable one ($\beta = 1.5$, $\bar{\ell} = -11.061$) and a readiness-priority one ($\beta = 2.0$, $L^{\mathrm{op}} = -11.405$, $\bar{\ell} = -11.076$). Each of 20 delay–jitter configurations ($\hat{d} \in \{2, 5, 10, 20, 40\}$ ms, $\Delta d \in \{0, 1, 2, 4\}$ ms) is flown on both missions with five jitter realizations: Algorithm 1 keeps the floor in all 200 runs.

On the certifiable mission, Algorithm 1 coincides with Pred in all 100 runs, while the RB filter and SD-Lip violate at 40 ms ($\min h = -0.18$ and -0.19), consistent with the failure near 29 ms reported in [8]. On the readiness-priority mission, the RB filter violates from 5 ms (-0.62 at 40 ms); Pred does not violate at $\alpha = 5$ ($\min h \geq 0.015$), Pred-Lip keeps $\min h \geq 0.27$ at 1.5–3.3 times its wrench error, and Algorithm 1 $\min h \geq 0.038$ at 1.0–1.65 times. The fallback of Algorithm 1 is never active without jitter, as Corollary 6 suggests, and at $\Delta d = 4$ ms it is active in 11% of the steps at 2 ms and 35% at 40 ms, because a longer window holds more commands in flight.

B. Monte Carlo missions

Q2–Beyond the design envelope, delay compensation alone is not enough, and the certificate keeps the floor at a moderate cost. Table I reports 50 missions with $c \sim \mathcal{U}[0.7, 0.9]$ and $\beta \sim \mathcal{U}[1.0, 2.5]$, flown at 10 ± 2 ms with $\alpha = 5$ and $\alpha = 20$; 34 are certifiable and 16 are flown with the readiness-priority floor. On the exact model, the zero violations of Algorithm 1 confirm the implementation of Theorem 5. On certifiable missions its median wrench error equals that of Pred (fallback in 2% and 0.1% of the steps; worst mission at $\alpha = 5$: 1.6 times Pred). Beyond the envelope the predictor's safety depends on the gain: Pred violates in 2 of 16 missions at $\alpha = 5$ and in all 16 at $\alpha = 20$, while Algorithm 1 holds the floor at a median wrench-error increase of 19% and 30%.

Q3–The Lipschitz margins are also safe, with a cost that depends strongly on the gain and on the constants, and without a guarantee. Pred-Lip costs 7.7 and 2.4 times the median wrench error of Pred at $\alpha = 5$, but 1.00 and 1.35 times at $\alpha = 20$: near $v^\star$, where $\nabla h \approx \mathbf{0}$ and $h \leq L^{\max} - \bar{\ell}$, a constant margin ζ keeps the barrier row feasible only if $\alpha(L^{\max} - \bar{\ell}) \geq \zeta$, and $\alpha(L^{\max} - \bar{\ell})$ spans 3.6–5.4 at $\alpha = 5$ and 14–22 at $\alpha = 20$, against $\zeta = 2.65$ (Pred-Lip) and $4.97\,T_0^{-1}$ (Pred-Lip-G). Accordingly, Pred-Lip-G is infeasible in 60% of the steps at $\alpha = 5$ and in none at $\alpha = 20$. Pred-Lip-Loc costs 1.21 and 1.45 times Pred at $\alpha = 5$ (Algorithm 1: 1.00 and 1.19) and 1.00 and 1.19 times at $\alpha = 20$ (1.00 and 1.30), with $\min h \geq +0.026$: at the high gain, a local Lipschitz margin is cheaper than the certificate beyond the envelope, but its sampled constants give no guarantee.

Thin buffer. With $\bar{\ell} = \ell^\dagger + 0.9(L^{\mathrm{op}} - \ell^\dagger)$ on the certifiable missions, the RB filter violates in 15 of 34 missions at $\alpha = 20$, but Pred never ($\min h \geq +0.053$ and $+0.009$); Algorithm 1 costs 1.29 and 1.00 times Pred. On certifiable missions, the certificate thus adds a guarantee, not observed safety.

C. Model mismatch and measurement error

Q4–With drag and measurement errors, the robust certificate still keeps the floor, and also the readiness of the true plant. Table II repeats the $\alpha = 5$ Monte Carlo with a plant whose drag coefficients deviate by up to $\pm 10\%$ per rotor and with speed measurements corrupted by uniform

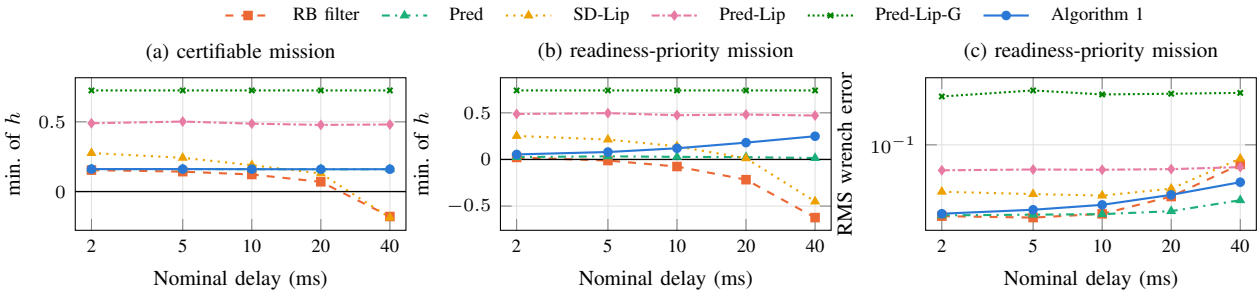


Fig. 2: Nominal-delay sweep at jitter ± 2 ms ($\Delta d = 4$ ms), $\alpha = 5$, synchronous switching, five jitter realizations per point; $c = 0.75$, (a) $\beta = 1.5$, $\bar{\ell} = -11.061$, (b), (c) $\beta = 2$, $\bar{\ell} = -11.076$. Pred-Lip-Loc was not run in this sweep. (a), (b): worst $\min_t h$ over the realizations (below zero: floor violated); (c): median RMS wrench error. Algorithm 1 stays above the floor at every delay, while the RB filter and SD-Lip fail at long delays; in (a), Algorithm 1 and Pred coincide.

TABLE I: Monte Carlo missions at 10 ± 2 ms. Viol.: missions with $\min_t h < 0$ (bold); $h_{\min}$: worst $\min_t h$; e_w: median (90th percentile) over missions of the RMS wrench error relative to Pred, whose absolute median (sweet-spot collective $= 2$) is 0.0023, 0.0022 (certifiable) and 0.0257, 0.0202 (readiness-priority) at $\alpha = 5$, 20; act.: % of steps with active fallback (Algorithm 1) or infeasible barrier row (Lipschitz baselines). Algorithm 1 never violates and matches Pred on certifiable missions; beyond the envelope, Pred violates and the Lipschitz margins are safe but, except Pred-Lip-Loc at $\alpha = 20$, costlier.

	$\alpha = 5$				$\alpha = 20$			
Filter	viol.	$h_{\min}$	e_w	act.	viol.	$h_{\min}$	e_w	act.
certifiable missions (34)								
RB filter	0	+0.014	2.35 (3.05)	0	**3**	**−0.035**	2.72 (3.36)	0
Pred	0	+0.098	1.00 (1.00)	0	0	+0.013	1.00 (1.00)	0
Pred-Lip	0	+0.408	7.70 (12.29)	0	0	+0.135	1.00 (1.46)	0
Pred-Lip-Loc	0	+0.204	1.21 (2.77)	0	0	+0.058	1.00 (1.00)	0
Pred-Lip-G	0	+0.581	141.59 (285.72)	60	0	+0.225	1.11 (2.88)	0
Alg. 1	0	+0.130	1.00 (1.20)	2.0	0	+0.015	1.00 (1.00)	0.1
readiness-priority missions (16)								
RB filter	**16**	**−0.111**	0.94 (1.07)	0	**16**	**−0.106**	1.15 (1.45)	0
Pred	**2**	**−0.004**	1.00 (1.00)	0	**16**	**−0.022**	1.00 (1.00)	0
Pred-Lip	0	+0.480	2.39 (3.78)	0	0	+0.109	1.35 (1.78)	0
Pred-Lip-Loc	0	+0.156	1.45 (1.70)	0	0	+0.026	1.19 (1.32)	0
Pred-Lip-G	0	+0.642	12.47 (26.60)	0.1	0	+0.224	1.71 (2.68)	0
Alg. 1	0	+0.121	1.19 (1.35)	30	0	+0.023	1.30 (1.59)	32

errors of up to $\varepsilon = 0.005\, v^{\text{sat}}$. Algorithm 1-R uses the robust certificate of Proposition 7 with the true bounds. Pred violates in 3 of 16 readiness-priority missions, neither variant of Algorithm 1 in any (beyond the envelope, 1-R costs 31% instead of 22%). As Proposition 7 anticipates, the hold torque fails a fresh check in 0.8% of the steps and is then applied as the continuation of the last certified plan. Evaluated with the plant drag, only Algorithm 1-R keeps the readiness above the floor ($h_{\min}^{\text{true}} \geq +0.026$; Pred and the nominal Algorithm 1: -0.135 and -0.028).

Source of the cost. On the readiness-priority reversal ($c = 0.75$, $\beta = 2$) at 10 ± 2 ms, the RB filter crosses the floor ($\min h = -0.075$), and Pred skims it ($+0.031$). Algorithm 1 keeps $\min h = +0.130$ at a wrench error of 0.0209 against 0.0165 for Pred (fallback in 27% of the steps). The cost is mostly in the collective, which the hold torque perturbs: the RMS collective error rises from 0.0046 to 0.0088, the moment error from 0.0158 to 0.0190. Following Corollary 6, adding $10\|\boldsymbol{\tau} - \boldsymbol{\tau}_{k-1}\|^2$ to the QP cost halves the fallback rate (12%) and lowers the wrench error to 0.0188 ($\min h = +0.078$).

D. Independent ESC jitter

Q4–Real ESCs switch at slightly different instants. With independent per-rotor offsets $\epsilon_{j,i}$ in the Monte Carlo of Table I, Pred still holds the floor on certifiable missions, but beyond the envelope it violates in 11 of 16 missions at $\alpha = 5$ (2 with synchronous switching; $\min h$ down to -0.27) and in all 16 at $\alpha = 20$. Algorithm 1 keeps the floor in all 100 runs at 1.09 and 1.02 times the wrench error of Pred, because Lemma 4 covers the skew by construction.

E. Computation

The certificate is cheap enough for a flight controller. At 10 ± 2 ms it covers 7 slices, each with $2n$ closed-form flows and one 4×4 Cholesky factorization, some 3×10^3 operations in all. Written in Python and compiled to machine code with Numba, a just-in-time compiler for numerical Python, it takes 9.6 μs (median; 99th percentile 14.6 μs) on one core of an Intel Core Ultra 7 255H CPU under Ubuntu 24.04. A full filter step (prediction, two OSQP solves, certificates) takes 0.57 ms (median; 99th percentile 1.37 ms, maximum 1.42 ms) with $N_b = 14$, against 0.48 ms for Pred. These are

TABLE II: As Table I with $\pm 10\%$ drag mismatch and 0.5% speed error, $\alpha = 5$. $h_{\min}^{\text{true}}$: with the plant drag; unc.: runs (% of steps) with a failed fresh hold check. Absolute e_w of Pred: 0.0062, 0.0266. Only Algorithm 1-R keeps the true-plant readiness above the floor.

Filter	viol.	$h_{\min}$	$h_{\min}^{\text{true}}$	e_w	act.	unc.
certifiable missions (34)						
Pred	0	$+0.090$	$\mathbf{-0.018}$	1.00 (1.00)	0	–
Pred-Lip	0	$+0.399$	$+0.224$	3.61 (6.15)	0	–
Alg. 1	0	$+0.124$	$\mathbf{-0.004}$	1.00 (1.13)	1.8	6 (0.1)
Alg. 1-R	0	$+0.147$	$+0.026$	1.00 (1.27)	3.3	8 (0.1)
readiness-priority missions (16)						
Pred	**3**	$\mathbf{-0.007}$	$\mathbf{-0.135}$	1.00 (1.00)	0	–
Pred-Lip	0	$+0.471$	$+0.370$	2.32 (3.34)	0	–
Alg. 1	0	$+0.121$	$\mathbf{-0.028}$	1.22 (1.38)	27	16 (1.8)
Alg. 1-R	0	$+0.155$	$+0.033$	1.31 (1.54)	32	16 (2.1)

not hard bounds; on flight hardware, the computation time must be measured and included in $\underline{d}$.

VI. Conclusion

Readiness barriers can be certified under sampling and uncertain actuation delay without Lipschitz constants: because rotors respond monotonically and in closed form and the readiness is unimodal in each rotor speed, the worst case over all reachable speeds costs a few closed-form evaluations, and the sweet-spot hold torque provides a certified fallback.

For a practitioner, the certificate turns a delay known only as an interval into a guarantee: it wraps any allocation QP, takes microseconds, and matched the tracking error of an uncertified predictor when the floor was compatible with the mission. It pays off when demands may exceed the design envelope or ESCs switch at different instants, where the predictor alone violated the floor; otherwise the predictor was safe, and local Lipschitz margins can be cheaper at high gains, without a guarantee.

Limitations. The guarantee needs the rotor model of [8], a certified start, hard delay bounds including computation, no reordering or dropped packets, and exact arithmetic; with measurement error, only the last certified plan is guaranteed, on the nominal readiness. The evidence is simulation of one hexarotor. Next steps are flight tests with measured ESC delays, and a wrench-preserving backup.

References

[1] M. Ryll *et al.*, "A Novel Overactuated Quadrotor Unmanned Aerial Vehicle: Modeling, Control, and Experimental Validation," *IEEE Trans. Control Syst. Technol.*, vol. 23, pp. 540–556, 2015.

[2] M. Hamandi *et al.*, "Design of Multirotor Aerial Vehicles: A Taxonomy Based on Input Allocation," *Int. J. Robot. Res.*, vol. 40, pp. 1015–1044, 2021.

[3] T. A. Johansen *et al.*, "Control Allocation—A Survey," *Automatica*, vol. 49, pp. 1087–1103, 2013.

[4] F. Ciresola *et al.*, "Control Input Allocation for Tilting Multirotors – A Review," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, 2026, pp. 656–663.

[5] R. Mahony *et al.*, "Multirotor Aerial Vehicles: Modeling, Estimation, and Control of Quadrotor," *IEEE Robot. Autom. Mag.*, vol. 19, no. 3, pp. 20–32, 2012.

[6] A. Franchi, "Drag-Aware Aerodynamic Manipulability for Torque-Limited Redundant Multirotors: Aerodynamic Promptness Based on the Symmetric Acceleration Capacity," arXiv:2603.07998, 2026.

[7] T. Yoshikawa, "Manipulability of Robotic Mechanisms," *Int. J. Robot. Res.*, vol. 4, pp. 3–9, 1985.

[8] G. Silano, "Readiness Barrier Functions: Forward-Invariant Control Authority for Overactuated Multirotor Allocation," *IEEE Robotics. and Autom. Lett.*, 2026, arXiv:2608.16335.

[9] A. D. Ames *et al.*, "Control Barrier Function Based Quadratic Programs for Safety Critical Systems," *IEEE Trans. Autom. Control*, vol. 62, no. 8, pp. 3861–3876, 2017.

[10] ——, "Control Barrier Functions: Theory and Applications," in *Proc. Eur. Control Conf. (ECC)*, 2019, pp. 3420–3431.

[11] H. Yang *et al.*, "Safety-Critical Control Allocation for Obstacle Avoidance of Quadrotor Aerial Photography," *IEEE Control Syst. Lett.*, vol. 8, pp. 1973–1978, 2024.

[12] S. Lee *et al.*, "CBF-Based Control Allocation for Omnidirectional Tiltrotor Aerial Robots," arXiv:2610.22057, 2026.

[13] G. Silano *et al.*, "Tilt as a Certified Resource: Preserving Motor Wrench-Rate Authority on Articulated Multirotors," arXiv:2609.21580, 2026.

[14] P. Ong *et al.*, "Matrix Control Barrier Functions," *IEEE Trans. Autom. Control*, 2026.

[15] J. Breeden *et al.*, "Control Barrier Functions in Sampled-Data Systems," *IEEE Control Syst. Lett.*, vol. 6, pp. 367–372, 2022.

[16] A. J. Taylor *et al.*, "Safety of Sampled-Data Systems with Control Barrier Functions via Approximate Discrete Time Models," in *Proc. IEEE Conf. Decis. Control (CDC)*, 2022, pp. 7127–7134.

[17] G. Bahati *et al.*, "Sample-and-Hold Safety with Control Barrier Functions," in *Proc. Amer. Control Conf. (ACC)*, 2024, pp. 5169–5176.

[18] M. Jankovic, "Control Barrier Functions for Constrained Control of Linear Systems with Input Delay," in *Proc. Amer. Control Conf. (ACC)*, 2018, pp. 3316–3321.

[19] T. G. Molnar *et al.*, "Safety-Critical Control With Input Delay in Dynamic Environment," *IEEE Trans. Control Syst. Technol.*, vol. 31, pp. 1507–1520, 2023.

[20] ——, "Input-to-State Safety with Input Delay in Longitudinal Vehicle Control," *IFAC-PapersOnLine*, vol. 55, pp. 312–317, 2022.

[21] Y. Kim *et al.*, "Robust Safety-Critical Control for Input-Delayed System with Delay Estimation," in *Proc. Eur. Control Conf. (ECC)*, 2024, pp. 2218–2223.

[22] A. K. Kiss *et al.*, "Integral Control Barrier Functions With Input Delay: Prediction, Feasibility, and Robustness," *IEEE Control Syst. Lett.*, vol. 10, pp. 2077–2082, 2026.

[23] I. Abel *et al.*, "Safety-Critical Control of Systems with Time-Varying Input Delay," *IFAC-PapersOnLine*, vol. 54, no. 18, pp. 169–174, 2021.

[24] Y. Zhang *et al.*, "Control Barrier Function Meets Interval Analysis: Safety-Critical Control with Measurement and Actuation Uncertainties," in *Proc. Amer. Control Conf. (ACC)*, 2022, pp. 3814–3819.

[25] D. Angeli *et al.*, "Monotone Control Systems," *IEEE Trans. Autom. Control*, vol. 48, pp. 1684–1698, 2003.

[26] S. Coogan, "Mixed Monotonicity for Reachability and Safety in Dynamical Systems," in *Proc. IEEE Conf. Decis. Control (CDC)*, 2020, pp. 5074–5085.

[27] Y. Chen *et al.*, "Backup Control Barrier Functions: Formulation and Comparative Study," in *Proc. IEEE Conf. Decis. Control (CDC)*, 2021, pp. 6835–6841.

[28] A. Singletary *et al.*, "Control Barrier Functions for Sampled-Data Systems with Input Delays," in *Proc. IEEE Conf. Decis. Control (CDC)*, 2020, pp. 804–809.

[29] M. Abate *et al.*, "Enforcing Safety at Runtime for Systems with Disturbances," in *Proc. IEEE Conf. Decis. Control (CDC)*, 2020, pp. 2038–2043.

[30] B. Stellato *et al.*, "OSQP: An Operator Splitting Solver for Quadratic Programs," *Math. Program. Comput.*, vol. 12, pp. 637–672, 2020.