Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Reliability-Aware Supervisory Flight Control of Quadrotors Using Causal Condition Estimation

Mohsen Rostami

msn.rostami@torontomu.ca

Toronto Metropolitan University

Julian Bardin

Toronto Metropolitan University

Alessandro Ciniello

Toronto Metropolitan University

Joon Chung

Toronto Metropolitan University

Research Article

Keywords: Reliability-aware control, Supervisory flight control, Causal condition estimation, Quadrotor fault tolerance, Rotor degradation

Posted Date: November 11th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11423089/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Reliability-Aware Supervisory Flight Control of Quadrotors Using Causal Condition Estimation

Mohsen Rostami, Julian Bardin, Alessandro Ciniello, and Joon Chung

Department of Aerospace Engineering, Toronto Metropolitan University, Toronto, ON M5B 2K3, Canada; msn.rostami@torontomu.ca (M.R.); jbardin@torontomu.ca (J.B.); alessandro.ciniello@torontomu.ca; j3chung@torontomu.ca (J.C.);
Correspondence: msn.rostami@torontomu.ca

Abstract: Quadrotor flight control under time-varying uncertainty is challenging because no single controller remains uniformly superior across nominal operation, payload variation, wind disturbance, and rotor degradation. This study presents a reliability-aware supervisory flight-control architecture that selects between adaptive proportional–integral–derivative control and linear quadratic regulation using causal condition estimation and requirement-conditioned reliability information. The framework combines controller-conditioned estimation of payload, wind magnitude, rotor effectiveness, and affected-rotor identity with reliability-based decision logic, asymmetric fault latching, hysteresis, dwell time, decision persistence, and bumpless command transfer. The estimator operates causally using a 0.50 s trailing window updated every 0.25 s. Once rotor degradation is confirmed, the fault state is permanently latched and post-fault reversal from linear quadratic regulation to adaptive PID is prohibited. The final architecture was evaluated exclusively in nonlinear simulation using 500 independent stratified missions comprising payload/wind-dominant, rotor-degradation-dominant, sequential mixed, and boundary scenarios, resulting in 2,000 paired simulations. The practical supervisor achieved 71.2% mission reliability, compared with 35.0% for fixed adaptive PID and 25.4% for fixed LQR, while matching the perfect-information oracle in all 500 missions. The improvement over both fixed controllers was statistically significant, and the practical architecture satisfied a predeclared 5 percentage-point noninferiority margin relative to the oracle. Successful-mission median tracking error increased by only 1.65% relative to the oracle. The estimator achieved 91.31% rotor-fault recall, a 0.326% false-positive rate, 0.393 s median causal detection delay, and 100% rotor-identity accuracy. These results demonstrate that causal condition estimation and asymmetric reliability-aware supervision can recover oracle-level mission reliability without requiring perfect operating-condition knowledge within the modeled operating envelope.

Keywords: Reliability-aware control; Supervisory flight control; Causal condition estimation; Quadrotor fault tolerance; Rotor degradation.

1. Introduction

Quadrotor unmanned aerial vehicles (UAVs) have become established platforms for inspection, mapping, delivery, surveillance, and research applications [1,2]. Idrissi et al. [1] survey the breadth of quadrotor applications together with associated architectural and control challenges, while Sonugür [2] reviews conventional and emerging control and simultaneous-localization-and-mapping approaches for quadrotor operation. Their usefulness nevertheless depends on maintaining stable and accurate flight despite nonlinear, underactuated, and strongly coupled dynamics. Mahony et al. [3] provide a widely used multirotor modeling, estimation, and control framework, and Pounds et al. [4] demonstrate how vehicle dynamics and actuator characteristics shape the behavior of a large quadrotor platform. Lee and Kim [5] further review trajectory-tracking control from modeling through experimental implementation. Taken together, this literature shows that practical operation must contend with changing payloads, aerodynamic disturbances, modeling uncertainty, sensing uncertainty, actuator limits, and propulsion degradation. These effects are not merely tracking disturbances: they can alter the closed-loop safety margin and move the vehicle into an operating region in which a different control law may be more reliable.

Classical proportional-integral-derivative (PID) control remains attractive because of its simple implementation and low computational demand, whereas linear quadratic regulation (LQR) provides a systematic state-feedback design

around a selected operating point [6]. Rinaldi et al. [6] compared representative quadrotor control approaches, illustrating the distinct performance characteristics of conventional designs. Xi et al. [7] examined flight-control behavior in the presence of wind, Razmi and Afshinfar [8] developed an adaptive sliding-mode approach for position and attitude control, Mechali et al. [9] proposed observer-based fixed-time nonsingular terminal sliding-mode control under uncertainty and disturbances, and Labbadi et al. [10] investigated robust tracking under external disturbances and model uncertainty. Collectively, these studies show that increasingly sophisticated controllers can improve robustness within their target conditions, but they also illustrate an important systems-level trade-off: a controller that performs strongly during nominal or disturbance-dominated operation need not remain the most reliable when actuator authority deteriorates. Conversely, a controller that tolerates degradation may sacrifice tracking performance during normal operation. This condition-dependent behavior motivates an architecture that selects among controllers according to demonstrated operational reliability rather than assuming that one fixed design is uniformly preferable.

Rotor and actuator faults are especially important because multirotors depend on coordinated thrust production for both force and moment generation. Jung and Bang [11] developed fault- and failure-tolerant model predictive control for a quadrotor, while Yang et al. [12] addressed varying load and actuator faults using sliding-mode fault-tolerant control. Wang et al. [13] considered active fault-tolerant control under actuator faults and model uncertainty, and Wang, van Kampen, and Chu [14] developed an incremental nonsingular terminal sliding-mode formulation for quadrotor fault tolerance. More recent approaches include adaptive neural fault-tolerant control [15], nonlinear observer-based sliding-mode control with hardware-in-the-loop validation [16], and disturbance-observer-enhanced adaptive fault-tolerant control [17]. These studies demonstrate substantial progress in recovering tracking performance after actuator degradation. However, most are formulated as standalone fault-tolerant controllers rather than as higher-level mechanisms for exploiting complementary reliability regions among heterogeneous controllers. They therefore do not directly address whether an adaptive controller and a fixed-gain controller can be used sequentially within the same mission to improve requirement satisfaction across changing operating conditions. Moreover, performance is often demonstrated through representative trajectories or selected fault cases, whereas supervisory decisions require an estimate of the probability that each candidate controller will satisfy phase-specific mission requirements across uncertain conditions.

A practical supervisory controller cannot assume perfect knowledge of payload, wind, or rotor effectiveness. Existing fault-detection and diagnosis methods include residual-based actuator-fault detection [18], motor-efficiency estimation [19], Kalman-filter architectures for actuator fault detection and diagnosis [20], adaptive observer-based fault detection and accommodation [21], data-fusion methods [22], and integrated schemes for actuator loss of effectiveness and combined actuator/sensor faults [23,24]. Li et al. [18] developed robust actuator-fault detection with guaranteed sensitivity, Chang and Cheng [19] estimated multirotor motor efficiency from onboard information, Amoozgar et al. [20] experimentally evaluated a two-stage Kalman filter for quadrotor actuator fault detection and diagnosis, and Lien et al. [21] combined adaptive observation with fault detection and fault-tolerant control. These results demonstrate that fault occurrence, severity, and affected-actuator information can be inferred from onboard measurements and command signals. However, such estimators are commonly developed within a particular closed-loop configuration. Because residuals and command-derived features are generated inside the feedback loop, their distributions can depend on the controller that is active. This controller dependence was observed directly during development of the present framework and is therefore treated explicitly here: separate estimator models are calibrated for APID- and LQR-generated data, and the model corresponding to the active feedback law is queried online. In this paper, this arrangement is referred to as controller-conditioned estimation. A further difficulty is that nominal flight, payload and wind disturbances, progressive rotor degradation, and transition periods do not necessarily form equally separable categorical classes. The proposed estimator therefore adopts a hierarchical structure that separates rotor-fault evidence from continuous payload, wind, and effectiveness estimation rather than relying on a single monolithic operating-mode classifier.

Supervisory control and gain scheduling provide established mechanisms for exploiting condition-dependent controller behavior [25-27]. Zuo and Yao [25] combined fault isolation with fault-tolerant control for an uncertain quadrotor, Adıgüzel et al. [26] incorporated a neural-network supervisor within a gain-scheduled quadrotor attitude controller, and Yang et al. [27] developed a supervisory fault-tolerant framework for uncertain nonlinear systems. For the present problem, however, controller selection alone is insufficient because the switching process must also be

protected. Classical bumpless-transfer studies show that replacing one dynamic controller with another can generate undesirable command discontinuities unless the incoming controller state or output is coordinated with the previously delivered input [28,29]. Kantue and Pedro [30] likewise emphasized the value of integrating fault estimation, accommodation, and reconfiguration for quadrotor control. Three issues remain especially relevant to the present architecture: diagnostic uncertainty can suppress a necessary safety switch, unrestricted bidirectional switching can produce chattering, and symmetric switching logic is physically inappropriate when a confirmed rotor loss is modeled as permanent rather than reversible.

A further gap concerns validation and the quantity used to make the supervisory decision. Masuda and Uchiyama [31] applied robust Monte Carlo model predictive control to quadrotor attitude control, while Wang et al. [32] used three-dimensional Monte Carlo modeling for UAS collision-risk assessment. Recent fault-tolerant-control studies have also introduced adaptive neural predefined-time accommodation [33], dynamic event-triggered neuroadaptive control [34], and observer-based backstepping integral nonsingular fast terminal sliding-mode control [35]. Despite these advances, mission-level reliability is seldom used as the central controller-selection quantity. Average tracking error alone cannot distinguish a controller that performs very accurately in relatively easy cases but fails frequently from one that provides more consistent requirement satisfaction across an uncertain mission set. A reliability-aware supervisor should therefore be evaluated using common randomized missions, phase-dependent success criteria, paired statistical comparisons, and confidence intervals. Its decision variable should represent the probability that each controller will satisfy the active operational requirements, rather than a heuristic based only on instantaneous tracking error or residual magnitude. The resulting practical architecture should also be compared with a perfect-information oracle to quantify how much of the theoretically available supervisory benefit remains after causal estimation and protected switching are introduced.

This study develops an estimator-driven, reliability-aware supervisory flight-control framework for nonlinear quadrotor missions with time-varying payload, wind, and rotor degradation. Adaptive PID (APID) and LQR are retained as a deliberately heterogeneous controller bank because their reliability regions are complementary. A causal hierarchical, controller-conditioned estimator uses onboard-accessible measurements, commands, and residual features to estimate payload, wind magnitude, rotor effectiveness, and affected-rotor identity. The supervisory layer evaluates requirement-conditioned controller reliability, applies reliability and performance hysteresis, decision persistence, and dwell time, and executes an offset-based bumpless transfer. Confirmed rotor degradation is permanently latched, creating an asymmetric APID-to-LQR safety transition and preventing post-fault reversal. The architecture is evaluated through 500 independent paired missions and 2,000 nonlinear simulations. The practical supervisor increases mission reliability from 35.0% for the best fixed controller to 71.2%, matches the perfect-information oracle in all 500 missions, and limits the successful-mission median tracking-error increase to 1.65%. These results demonstrate that causal estimation need not reproduce the operating condition perfectly; it must instead provide sufficiently timely and reliable information to support the correct safety decision within the modeled operating envelope.

2. Methodology

2.1. Framework overview and development protocol

The proposed architecture combines a nonlinear quadrotor plant, a deliberately heterogeneous controller bank, an offline requirement-conditioned reliability model, a causal controller-conditioned condition estimator, protected supervisory logic, and an offset-based command-transfer mechanism. Adaptive PID (APID) is used as the performance-oriented controller during nominal, payload, and wind-dominated operation, whereas a fixed hover-linearized LQR is retained as the degradation-resilient controller. The supervisory layer does not blend the two feedback laws continuously. Instead, it predicts the probability that each controller will satisfy the active phase requirements and authorizes a discrete controller transition only when the predicted advantage exceeds predeclared reliability or performance margins. Figure 1 illustrates this complete signal flow from onboard measurements and references through feature extraction, condition estimation, reliability-map interrogation, protected controller selection, bumpless transfer, nominal allocation, actuator dynamics, and the nonlinear plant.

The development protocol was separated from the final statistical assessment. First, constant-condition Monte Carlo data were used to construct the APID and LQR reliability maps. Second, a 40-mission requirement-aware development campaign was divided at the mission level into 24 calibration missions and 16 untouched validation missions. This dataset was used to establish controller complementarity, select the safe transfer architecture, train and validate the condition estimator, diagnose unsuccessful symmetric switching policies, and freeze the final asymmetric fault-latched policy. Third, the selected architecture was evaluated without further tuning on 500 newly generated independent missions. The final campaign therefore did not contribute to controller tuning, estimator fitting, map construction, threshold selection, or policy selection. All compared controllers were exposed to identical realizations through common random numbers, thereby preserving a paired experimental design.

2.2. Nonlinear quadrotor and actuator model

The nominal simulation vehicle is a representative 1.5-kg X-configuration quadrotor whose mass and principal inertias are based on the PX4/Gazebo Iris-class simulation model [36]. Figure 2(a) shows the in-house quadrotor prototype currently under development as the intended platform for future experimental implementation and validation of the proposed supervisory architecture, while Fig. 2(b) illustrates the simplified symmetric X-configuration adopted in the present numerical study. The prototype was not used to identify the parameters of the current simulation model; rather, it represents the planned next-stage hardware platform. Accordingly, the numerical model is Iris-inspired rather than an exact reproduction of a specific commercial airframe. Representative values are adopted for the arm length, actuator time constant, aerodynamic drag, rotor-speed limits, and reaction-torque coefficient to provide an internally consistent nonlinear simulation platform. The state vector contains inertial position $\mathbf{p}$, inertial velocity $\mathbf{v}$, 3-2-1 (ZYX) Euler angles $\boldsymbol{\eta} = [\varphi, \theta, \psi]^{\mathrm{T}}$, body angular velocity $\boldsymbol{\omega}$, and the four rotor angular speeds $\boldsymbol{\Omega}$. The resulting 16-state vector is

$$\mathbf{x} = [\mathbf{p}^{\mathrm{T}}\ \mathbf{v}^{\mathrm{T}}\ \boldsymbol{\eta}^{\mathrm{T}}\ \boldsymbol{\omega}^{\mathrm{T}}\ \boldsymbol{\Omega}^{\mathrm{T}}]^{\mathrm{T}} \tag{1}$$

Let $\mathbf{R}(\boldsymbol{\eta})$ denote the body-to-inertial rotation matrix, $\mathbf{E}(\boldsymbol{\eta})$ the Euler-rate transformation, $\mathbf{e}_3 = [0, 0, 1]^{\mathrm{T}}$, T the realized collective thrust, and $\boldsymbol{\tau}$ the realized body moment. The nonlinear equations of motion follow the standard Newton-Euler multirotor formulation [3,4]:

$$\begin{aligned} \dot{\mathbf{p}} &= \mathbf{v}, \\ m\dot{\mathbf{v}} &= -mg\mathbf{e}_3 + T\mathbf{R}(\boldsymbol{\eta})\mathbf{e}_3 \\ &\quad -\mathbf{R}(\boldsymbol{\eta})\mathbf{D}\mathbf{R}^{\mathrm{T}}(\boldsymbol{\eta})(\mathbf{v} - \mathbf{v}_w) + \mathbf{f}_d \end{aligned} \tag{2}$$

$$\dot{\boldsymbol{\eta}} = \mathbf{E}(\boldsymbol{\eta})\boldsymbol{\omega} \tag{3}$$

$$\mathbf{J}\dot{\boldsymbol{\omega}} = \boldsymbol{\tau} - \boldsymbol{\omega} \times (\mathbf{J}\boldsymbol{\omega}) + \boldsymbol{\tau}_d \tag{4}$$

where m and $\mathbf{J}$ are the instantaneous mass and inertia tensor, $\mathbf{D}$ is the body-axis linear-drag matrix, $\mathbf{v}_w$ is the inertial wind velocity, and $\mathbf{f}_d$ and $\boldsymbol{\tau}_d$ are externally applied force and moment disturbances. This model retains the nonlinear attitude kinematics, force-direction coupling, gyroscopic term, time-varying mass and inertia, aerodynamic drag, sensor noise, actuator saturation, and motor dynamics used throughout the study. Such a model is consistent with established multirotor formulations [3,4], but the uncertainty schedules and supervisory components described below are specific to the present study.

The generalized control command is $\mathbf{u} = [T \quad \tau_\phi \quad \tau_\theta \quad \tau_\psi]^{\mathrm{T}}$. A nominal X-configuration allocation matrix $\mathrm{B_0}$ converts $\mathbf{u}$ into commanded squared rotor speeds. The true allocation matrix includes the rotor-effectiveness vector $\boldsymbol{\eta}_r = [\eta_1, \eta_2, \eta_3, \eta_4]^{\mathrm{T}}$, so that

$$\mathbf{u}_a = \mathbf{B}(\boldsymbol{\eta}_r)\boldsymbol{\Omega}^2, \qquad \mathbf{B}(\boldsymbol{\eta}_r) = \mathbf{B}_0 \mathrm{diag}(\boldsymbol{\eta}_r) \tag{5}$$

The allocator uses the nominal matrix $\mathbf{B}_0$; therefore, loss of effectiveness is not supplied directly to the low-level controllers or compensated through explicit control reallocation. Commanded rotor speeds are limited to 0-900 rad/s. Each motor follows a first-order model with nominal time constant $\tau_m = 0.03$ s, rotor-speed-rate saturation of 2.5×10^4 rad/s², and state limiting at the physical speed bounds:

$$\dot{\Omega}_i = \mathrm{sat}\left(\frac{\Omega_{i,c} - \Omega_i}{\tau_{m,i}}\right) \tag{6}$$

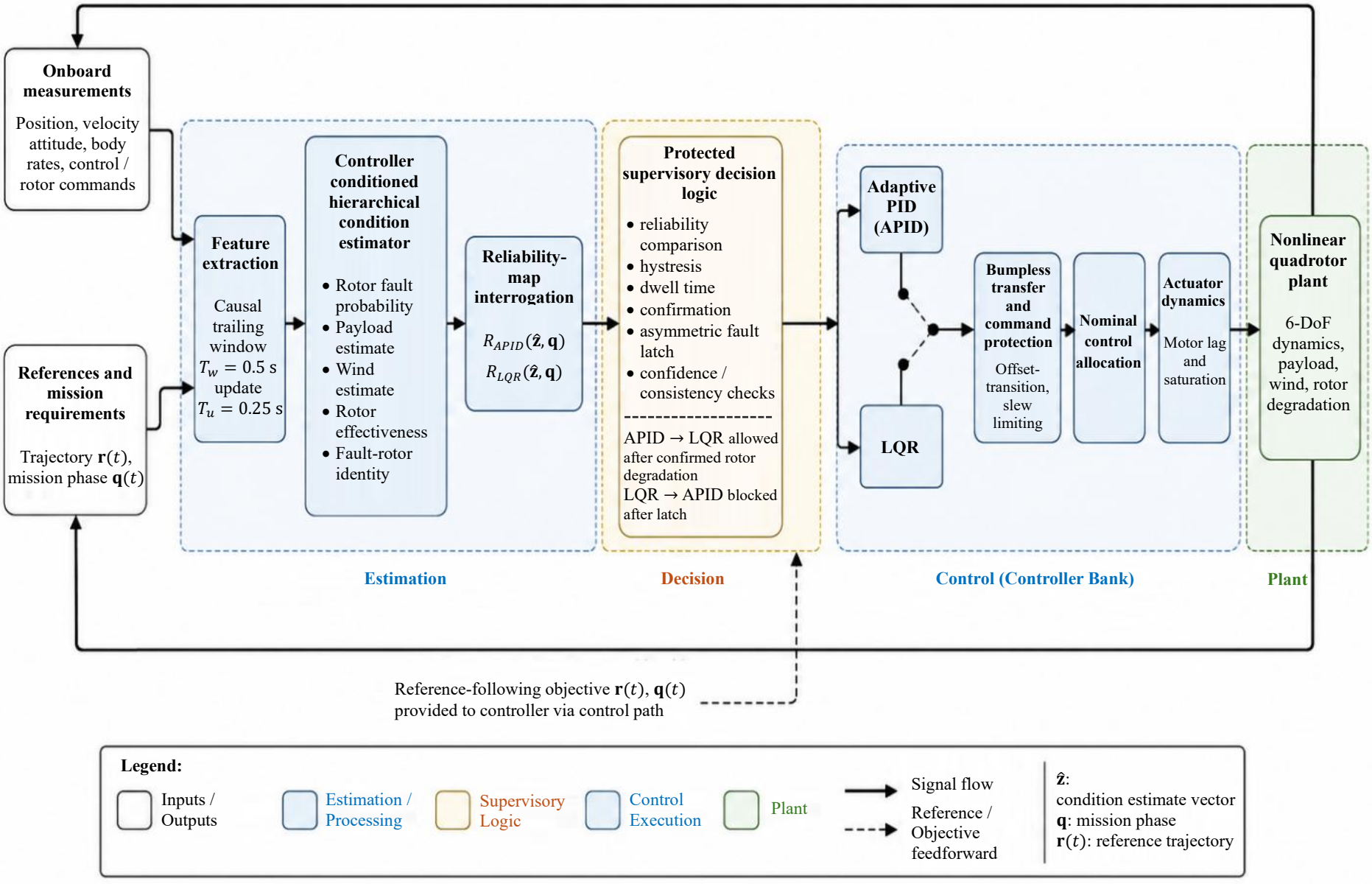


Figure 1. Complete signal flow of the proposed reliability-aware supervisory flight-control architecture, from onboard measurements and mission references through feature extraction, causal condition estimation, reliability-based controller selection, bumpless transfer, control allocation, actuator dynamics, and the nonlinear quadrotor plant.

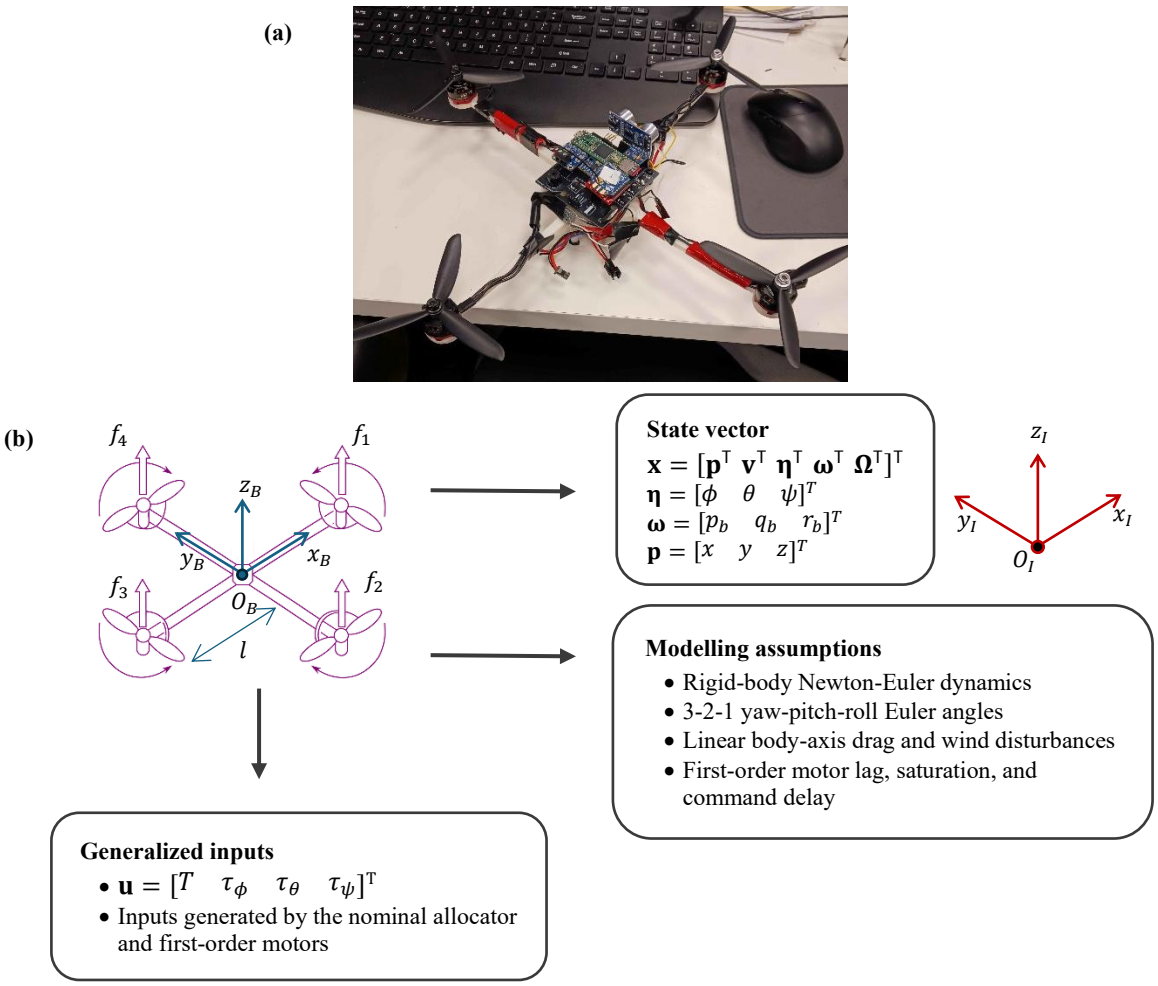


Figure 2. Vehicle platform and modelling abstraction: (a) in-house quadrotor prototype currently under development for future experimental implementation and validation of the proposed supervisory controller; (b) simplified symmetric X-configuration model used in the present nonlinear simulation, including the principal state variables, generalized inputs, and modelling assumptions.

The nominal vehicle is a representative 1.5-kg quadrotor whose mass, principal inertias, and propulsion scaling are based on the widely used PX4/Gazebo Iris-class simulation model [36]. The plant was subsequently simplified to a symmetric X configuration and assigned representative arm length, actuator time constant, aerodynamic drag, rotor-speed limits, and torque coefficient to provide an internally consistent nonlinear simulation platform. The resulting model is therefore Iris-inspired rather than an exact reproduction of a specific commercial airframe.

The nominal mass properties, propulsion coefficients, actuator parameters, aerodynamic drag, and command constraints of the simulated quadrotor are summarized in Table 1. These values define the representative Iris-inspired X-configuration vehicle used throughout the nonlinear simulation study. Total thrust and moment commands are subjected to the common physical limits listed in the table.

Table 1. Nominal quadrotor parameters and command limits.

Category	Parameter	Symbol	Value	Unit
Mass properties	Nominal mass	m_0	1.50	kg
Mass properties	Nominal inertia tensor	J_0	diag(0.029125, 0.029125, 0.055225)	kg·m²
Geometry	Arm length	l	0.23	m
Propulsion	Thrust coefficient	k_f	8.54858×10^{-6}	N·s²/rad²
Propulsion	Reaction-torque coefficient	k_m	1.60×10^{-7}	N·m·s²/rad²
Actuator	Nominal motor time constant	τ_m	0.03	s
Actuator	Maximum rotor speed	ω_{max}	900	rad/s

Actuator	Rotor-speed-rate limit	$\lvert\dot{\omega}\rvert_{\max}$	2.5×10^4	rad/s²
Aerodynamics	Linear drag matrix	D	diag(0.15, 0.15, 0.30)	N·s/m
Command limit	Maximum commanded tilt	$\theta_{\text{tilt,max}}$	35	deg
Command limit	Maximum body-moment command	$\tau_{\max}$	⌈1.40⌉⌈1.40⌉⌈0.35⌉	N·m
Command limit	Collective-thrust saturation	$T_{\max}$	$0.98(4\ k_f\ \omega_max^2) \approx 27.14$	N

The numerical controller gains, control-allocation model, reliability maps, and estimator calibration used in this study are associated with the nominal configuration and should not be interpreted as directly transferable to an arbitrary multirotor without recalibration. For another quadrotor with the same four-rotor topology, changes in mass, inertia, arm length, thrust and torque coefficients, or propulsion bandwidth would generally require retuning of the APID and LQR gains and regeneration of the corresponding reliability maps and estimator calibration data. The supervisory design principles—causal condition estimation, reliability-based controller selection, fault latching, and protected bumpless transfer—are not tied to the numerical parameters of the present airframe; however, their specific implementation remains dependent on the vehicle configuration, controller bank, and available actuator geometry. A change in rotor count or geometric configuration would therefore additionally require modification of the control-allocation model and the physics-based faulted-rotor decoding templates.

The continuous dynamics are integrated using a fixed-step fourth-order Runge-Kutta scheme with a 1-ms integration step. A fixed-step implementation was selected to maintain a common deterministic time grid across the actuator dynamics, measurement and controller updates, estimator windows, switching logic, and paired Monte Carlo realizations. This avoids interpolation and event-timing variability associated with adaptive-step integration and ensures that controller and supervisory events occur at identical sample instants across paired configurations. Measurements and controller commands are updated every 5 ms, giving five plant-integration steps per control interval.

Each mission lasts 20 s and follows the helical reference defined in Table 2. Zero-mean measurement noise is added independently to the position, velocity, Euler-angle, and body-rate channels. The corresponding nominal standard deviations are also summarized in Table 2 and are multiplied by the sampled sensor-noise severity for each mission.

Table 2. Simulation timing, reference-trajectory, and nominal measurement-noise settings.

Category	Quantity	Value	Unit
Simulation timing	Integration step	1	ms
Simulation timing	Measurement / controller update interval	5	ms
Simulation timing	Mission duration	20	s
Reference trajectory	Helix radius	0.70	m
Reference trajectory	Angular rate	0.30	rad/s
Reference trajectory	Initial altitude	0.80	m
Reference trajectory	Vertical climb rate	0.04	m/s
Reference trajectory	Yaw reference	Tangent-aligned	-
Measurement noise	Position standard deviation	[0.005; 0.005; 0.008]	m
Measurement noise	Velocity standard deviation	[0.010; 0.010; 0.015]	m/s
Measurement noise	Euler-angle standard deviation	[0.10; 0.10; 0.15]	deg
Measurement noise	Body-rate standard deviation	[0.25; 0.25; 0.30]	deg/s

2.3. Time-varying uncertainty and degradation scenarios

Each mission contains five ordered phases: nominal flight, payload transition, wind transition, rotor-degradation ramp, and degraded flight. Payload, wind, and rotor loss are introduced through smoothstep schedules rather than discontinuous parameter steps. Payload is applied between approximately 3.5 and 7.0 s and removed between approximately 9.5 and 12.0 s. Wind begins between 6.5 and 7.5 s, reaches its sampled magnitude after a 0.7-1.3-s ramp, and is removed around 10-12 s. Rotor degradation begins later in the mission and evolves over a 2-3-s ramp to its sampled final effectiveness, after which it remains permanent. The faulted rotor is selected uniformly from the four rotors, and wind direction is sampled uniformly in the horizontal plane. Figure 3 illustrates a representative sequential mixed mission and the temporal ordering of the uncertainty and degradation inputs.

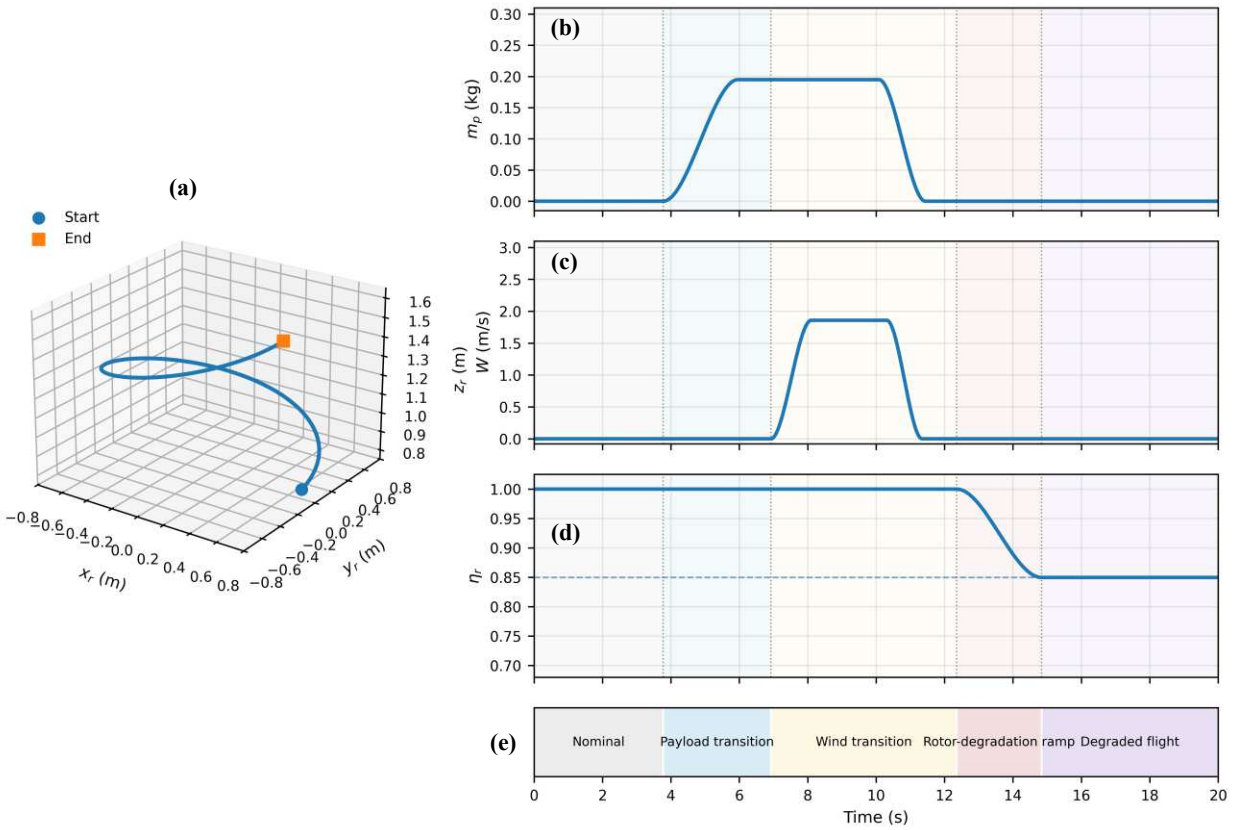


Figure 3. Representative sequential mixed mission used to illustrate the validation profile: (a) 20-s helical reference trajectory; (b) smoothstep payload schedule; (c) smoothstep wind-magnitude schedule; (d) permanent rotor-effectiveness reduction; and (e) corresponding mission phases.

Four mission strata were predeclared to provide coverage of the controllers' complementary regions. Payload/wind-dominant missions use large payload and wind disturbances with only mild rotor degradation. Rotor-degradation-dominant missions use small payload and wind disturbances but severe loss of effectiveness. Sequential mixed missions combine large payload and wind disturbances with subsequent moderate-to-severe rotor degradation. Boundary/ambiguous missions populate the transition region between the principal controller domains. Table 3 summarizes the final sampling envelope.

Table 3. Final mission strata and sampled operating-condition ranges.

Stratum	Missions	Payload (kg)	Wind (m/s)	Final rotor effectiveness	Noise scale	Rotor start (s)
Payload/wind dominant	125	0.20-0.30	1.50-3.00	0.96-1.00	1.00-2.50	13.5-14.5
Rotor-degradation dominant	125	0-0.08	0-0.80	0.70-0.84	1.00-3.00	10.5-11.5
Sequential mixed	188	0.15-0.30	1.50-3.00	0.70-0.88	1.00-3.00	11.8-12.8
Boundary/ambiguous	62	0.08-0.20	0.50-2.20	0.90-0.97	1.50-3.00	11.5-12.8

Independent nuisance uncertainties are also applied to the baseline vehicle. A normalized nuisance severity of 0.20 multiplies uniformly distributed fractional bounds, resulting in mass, inertia, thrust-coefficient, torque-coefficient, motor-time-constant, and drag perturbations of up to approximately ±6%, ±5%, ±4%, ±4%, ±6%, and ±8%, respectively. Initial position, velocity, attitude, angular-rate, and rotor-speed perturbations are sampled within ±[0.02, 0.02, 0.015] m, ±[0.02, 0.02, 0.015] m/s, ±0.5°, ±1°/s, and ±5 rad/s. No artificial command delay is applied in the final supervisory campaign because delay was not a dimension of the frozen reliability maps.

2.4. APID-LQR controller bank

2.4.1. Adaptive PID controller

The APID controller uses cascaded position and attitude-control structure consistent with conventional quadrotor flight-control implementations [3,4,6]. With position error $\mathbf{e}_p = \mathbf{p}_r - \mathbf{p}$, velocity error $\mathbf{e}_v = \mathbf{v}_r - \mathbf{v}$, and bounded position-error integral $\boldsymbol{\xi}_p$, the commanded inertial acceleration is

$$\mathbf{a}_c = \mathbf{a}_r + g\mathbf{e}_3 + \mathbf{K}_{p,p}\mathbf{e}_p + \mathbf{K}_{i,p}\boldsymbol{\xi}_p + \mathbf{K}_{d,p}\mathbf{e}_v \tag{7}$$

The corresponding force command $\mathbf{F}_c = m_0\mathbf{a}_c$ is limited by the minimum vertical-force and maximum-tilt constraints and converted to a desired attitude $\mathbf{R}_c$. Desired body rates are obtained from successive attitude references. The attitude loop uses attitude error $\mathbf{e}_\eta$, body-rate error $\mathbf{e}_\omega$, and integral $\boldsymbol{\xi}_\eta$ to calculate

$$\boldsymbol{\alpha}_c = \mathbf{K}_{p,\eta}\mathbf{e}_\eta + \mathbf{K}_{i,\eta}\boldsymbol{\xi}_\eta + \mathbf{K}_{d,\eta}\mathbf{e}_\omega \tag{8}$$

$$T_c = \mathbf{F}_c^{\top}\mathbf{R}(\boldsymbol{\eta})\mathbf{e}_3, \qquad \boldsymbol{\tau}_c = \mathbf{J}_0\boldsymbol{\alpha}_c + \boldsymbol{\omega} \times (\mathbf{J}_0\boldsymbol{\omega}) \tag{9}$$

The nominal position gains are $\mathbf{K}_{p,p=diag(8,8,10)}$, $\mathbf{K}_{i,p=diag(0.20,0.20,0.50)}$, and $\mathbf{K}_{d,p=diag(5,5,6)}$. The nominal attitude gains are $\mathbf{K}_{p,\eta=diag(12,12,8)}$, $\mathbf{K}_{i,\eta=diag(0.05,0.05,0.03)}$, and $\mathbf{K}_{d,\eta\,=\,diag(5,5,4)}$. Each axis stores the adaptive parameter vector $\boldsymbol{\vartheta}_j = [K_{p,j}, K_{i,j}, K_{d,j}]$. The current gains generate the present command, and the adaptive law updates the gains for the next 5-ms control sample. For regressor $\boldsymbol{\varphi}_j = [e_j, \xi_j, \dot{e}_j]$, the bounded normalized update has the generic form

$$\dot{\boldsymbol{\vartheta}}_j = \mathrm{Proj}\{\boldsymbol{\Gamma}_j \frac{\varepsilon_j \bar{\boldsymbol{\Phi}}_j}{1 + \bar{\boldsymbol{\Phi}}_j^{\top}\bar{\boldsymbol{\Phi}}_j} - \sigma\big(\boldsymbol{\vartheta}_j - \boldsymbol{\vartheta}_{j,0}\big)\} \tag{10}$$

where $\bar{\boldsymbol{\varphi}}_j$ is normalized by predefined error scales, ε_j is a dead-zone filtered combination of proportional and derivative errors, $\boldsymbol{\Gamma}_j$ is the adaptation-rate matrix, σ is leakage toward the nominal gains, and $\mathrm{Proj}\{\cdot\}$ enforces elementwise bounds. Proportional and derivative gains are constrained between 45% and 190% of their nominal values; the integral gains are constrained between zero and four times nominal. Gain-rate limits, conditional integration, adaptation freezing during adverse saturation, and 20-ms first-order filtering of position, velocity, and body-rate measurements protect the adaptive loop from noise and large transients. Euler angles are left unfiltered to avoid wrap-related phase artifacts.

2.4.2. Fixed LQR controller

The LQR is designed from the 12-state hover linearization containing position, velocity, attitude, and body-rate perturbations, following standard linearized quadrotor state-feedback formulations [3,4,6]. The control increment is

$$\delta\mathbf{u} = -\mathbf{K}\,\delta\mathbf{x}, \tag{11}$$

where $\mathbf{K}$ is obtained from the continuous algebraic Riccati equation. The state and input weights are generated using Bryson-style allowable deviations in which the diagonal weights are scaled according to prescribed allowable state and control deviations [37]: 1 m in position, [1.5, 1.5, 1.0] m/s in velocity, [20°, 20°, 45°] in attitude, [100°, 100°, 80°]/s in body rates, and [0.5m₀g, 1, 1, 0.30] in collective-thrust and moment increments. The horizontal errors are rotated by the reference yaw before feedback. A nonlinear feedforward force $m_0(\mathbf{a}_r + g\mathbf{e}_3)$ is converted into the instantaneous thrust direction, while the LQR increment provides the stabilizing collective-thrust and moment corrections. The resulting closed-loop hover model is controllable and asymptotically stable. Both APID and LQR use the same measurements, reference generator, command limits, nominal allocator, motor model, sensor noise, and numerical integration, so differences arise from the feedback laws rather than inconsistent plant interfaces.

2.5. Requirement-conditioned mission reliability

Controller selection is based on the probability of satisfying phase-dependent operational requirements rather than on instantaneous tracking error alone. Let $S_i = 1$ indicate that controller i completes a randomized trial while satisfying the global safety envelope and all requirements applicable to the active phase. For a condition vector $\mathbf{z}$ and phase q, controller reliability is defined as

$$R_i(\mathbf{z}, q) = \Pr(S_i = 1 \mid \mathbf{z}, q) \tag{12}$$

The offline reliability database contains two complementary map families: payload-wind maps parameterized by payload mass and wind magnitude, and rotor-noise maps parameterized by minimum rotor effectiveness and sensor-

noise scale. At every map cell, reliability is the fraction of paired Monte Carlo replicates satisfying the applicable label. Requirement-conditioned maps were generated by relabeling each completed baseline trial as successful only when the original mission-safety label passed and the position RMSE was below the predeclared phase limit. Wilson intervals, median position RMSE, control effort, and command smoothness were retained with each cell. Bilinear interpolation is used online, and queries are clamped to the validated map domain.

For the active phase, one map family is designated as the primary requirement map and the other remains a secondary safety map. The reliability score used by the supervisor is conservative:

$$\hat{R}_i = \min\left(\hat{R}_{i,\text{primary}}, \hat{R}_{i,\text{secondary}}\right) \tag{13}$$

whereas predicted tracking RMSE and other performance quantities are combined using the maximum of the two projections. This construction prevents a controller from being selected solely because it performs well in one uncertainty projection while exhibiting poor safety reliability in the other. Table 4 lists the phase-specific success requirements. Mission success additionally requires finite states and satisfaction of the global position, attitude, altitude, and actuator-saturation limits.

Table 4. Predeclared phase-specific mission requirements.

Phase	Primary map	RMSE limit (m)	Peak limit (m)	Roll/pitch limit	Allocation saturation limit	Final error limit (m)
Nominal	Payload-wind	0.12	0.25	35°	0.10	-
Payload transition	Payload-wind	0.15	0.30	35°	0.20	-
Wind transition	Payload-wind	0.18	0.35	40°	0.25	-
Rotor-degradation ramp	Rotor-noise	0.22	0.45	50°	0.30	-
Degraded flight	Rotor-noise	0.30	0.60	55°	0.30	0.30

2.6. Causal hierarchical condition estimator

The practical supervisor replaces the oracle's true operating condition with estimates obtained exclusively from onboard-accessible measurements, references, delivered generalized commands, rotor commands, and the identity of the currently active controller. No true payload, wind, rotor effectiveness, or faulted-rotor identity is supplied to the estimator. A 0.50-s trailing window is updated every 0.25 s. The estimate obtained after appending a sample is made available only at the next 5-ms control update, thereby preserving causality. At least 80% of the window must have been generated under the currently active controller; otherwise, the previous estimate is retained and marked ambiguous.

The estimator timescale was selected relative to the nominal actuator dynamics. For the nominal motor time constant of $\tau_m = 0.03$ s, the 0.25-s estimator update interval spans approximately 8.3 motor time constants, while the complete 0.50-s feature window spans approximately 16.7 time constants. Consequently, ordinary command-following transients associated with the nominal first-order motor response decay substantially within the estimator observation interval and are less likely to be interpreted as persistent loss of effectiveness. In addition, rotor degradation is not inferred from actuator lag alone: the detector jointly uses rotor-command spread and asymmetry, moment-residual magnitude and variability, angular-acceleration activity, command-rate information, and attitude/rate errors. Persistent degradation therefore produces a structured multi-feature signature rather than a single transient command–response mismatch.

This separation of timescales is specific to the propulsion bandwidth considered here; substantially slower actuators would change the residual statistics and would require corresponding recalibration of the actuator model and estimator. Candidate features include tracking-error statistics, normalized collective thrust and moments, rotor-command spread and asymmetry, command-rate activity, and residuals derived from the nominal translational and rotational dynamics. With filtered velocity and body-rate derivatives, the force and moment residuals are

$$\mathbf{r}_F = m_0(\hat{\mathbf{a}} + g\mathbf{e}_3) - T_c\mathbf{R}(\boldsymbol{\eta})\mathbf{e}_3 \tag{14}$$

$$\mathbf{r}_\tau = \mathbf{J}_0\hat{\boldsymbol{\alpha}} + \boldsymbol{\omega} \times (\mathbf{J}_0\boldsymbol{\omega}) - \boldsymbol{\tau}_c \tag{15}$$

Thirty window features were initially evaluated. The frozen rotor detector uses ten controller-conditioned features emphasizing rotor spread, command asymmetry, moment-residual magnitude and variability, angular-acceleration activity, command rate, and attitude/rate errors. The continuous regressors use a 16-feature set that additionally

includes thrust-to-weight offset, total and horizontal force residuals, force-residual directionality, position and velocity errors, and thrust variability.

Separate models are fitted for APID and LQR because the residual distributions were found to depend on the active feedback law. Rotor degradation is detected using balanced L2-regularized logistic regression. Features are standardized using the calibration data, class weights balance fault and non-fault windows, and the coefficients are obtained by a toolbox-independent Newton/iteratively reweighted least-squares algorithm. The regularization coefficient is selected from {0.01, 0.10, 1.00} using five-fold mission-level cross-validation. Windows with minimum rotor effectiveness below 0.985 are labelled as positive examples, windows at or above 0.995 are labelled as negative examples, and the narrow boundary region is omitted from binary training. The final detector uses an activation threshold of 0.25 and a deactivation threshold of 0.20.

Payload mass, wind magnitude, and minimum rotor effectiveness are estimated using separate standardized ridge regressions with $\lambda = 0.10$. Outputs are constrained to the validated domains [0, 0.35] kg, [0, 4] m/s, and [0.50, 1.00], respectively. Once rotor degradation is detected, affected-rotor identity is decoded from the exponentially averaged signed moment-residual vector. The four normalized allocation templates are (-,+,-), (-,-,+), (+,-,-), and (+,+,+) for rotors 1-4. The rotor whose template has the largest cosine similarity is selected when the residual norm and cosine confidence exceed their minimum thresholds. This physics-based decoder avoids fitting a separate multiclass model and preserves a direct connection to the nominal allocation geometry.

An out-of-distribution diagnostic is calculated from standardized detector and regression features. The generic estimate is considered in-distribution when the standardized RMS is no greater than 5 and the maximum absolute standardized feature is no greater than 10. The estimator also records ambiguity near the rotor-detection threshold, controller-window purity, fault probability, and rotor-identity confidence. Sensor-noise severity is supplied as a normalized covariance/health index from an assumed sensor-monitoring layer; the actual noise realization remains unknown. These quantities are logged for diagnostic analysis, but the final safety policy does not allow generic uncertainty gating to veto a persistent rotor-fault indication.

2.7. Reliability-aware supervisory decision logic

At each supervisory update, the estimated condition $\hat{\mathbf{z}} = [\hat{m}_p, \widehat{W}, \hat{\eta}_{min}, \hat{r}_f]$ is used to query the APID and LQR requirement-conditioned maps. When no rotor fault is detected, the rotor-effectiveness vector is set to unity. When a fault is detected, the estimated minimum effectiveness is assigned to the decoded rotor. The direct controller recommendation is

$$i^\star = \underset{i \in \{\mathrm{APID},\, \mathrm{LQR}\}}{\operatorname{argmax}} \; \hat{R}_i(\hat{\mathbf{z}}, q) \tag{16}$$

If the two controllers have equal reliability within numerical tolerance, the predicted position RMSE is used as the tie-break metric. The recommendation is then passed through the frozen safe-switching layer. A reliability-driven change requires an improvement greater than $h_R = 0.02$. When the reliability difference is within that margin, a performance-driven change requires a predicted RMSE improvement greater than $h_e = 0.005$ m. An eligible candidate must persist for 0.10 s, the active controller must have been retained for at least 0.25 s, and no previous command transfer may remain active.

The perfect-information oracle uses the same reliability maps, hysteresis, persistence, dwell time, and command-transfer mechanism but queries them using the true condition. It therefore represents the achievable upper bound of the frozen APID-LQR bank under the same decision safeguards, rather than an unconstrained instantaneous switcher. The practical and oracle supervisors start with APID.

2.8. Permanent fault latch and asymmetric safety policy

Initial online experiments showed that symmetric APID↔LQR switching was unsuitable for permanent rotor loss. Short controller-conditioned windows produced temporary uncertainty after a switch, which could request a return to APID and generate repeated reversals. Conversely, a single generic confidence gate could block a necessary APID-to-LQR transition when degradation itself drove the features outside the nominal calibration cloud. The final policy therefore separates fault evidence from generic estimator confidence and imposes directionality based on the physical persistence of the modeled fault.

Rotor evidence is considered active when the estimated degradation probability exceeds 0.25. Evidence may survive interruptions of up to 0.35 s and is latched after 0.10 s of confirmation; affected-rotor identity is not required before latching. Once latched at t_f, the fault state satisfies

$$F_r(t) = 1, \qquad t \geq t_f \tag{17}$$

and post-latch LQR-to-APID transitions are prohibited. Before fault confirmation, APID is explicitly retained during nominal, payload, and wind operation. After confirmation, generic out-of-distribution or ambiguous-condition flags may not veto an APID-to-LQR safety transition, although the frozen reliability, performance, confirmation, dwell, and transfer protections remain active. For conservative map lookup, the decision effectiveness is

$$\eta_{\mathrm{dec}} = \max(0.50, \hat{\eta}_{\mathrm{latched}} - 0.07) \tag{18}$$

The 0.07 allowance accounts for online severity-estimation error and biases the decision toward the degradation-resilient controller. This conservative severity estimate influences the reliability query; it does not modify the plant, allocator, or controller gains.

2.9. Bumpless transfer and command protection

When a controller transition is authorized at time t_s, the incoming command is coordinated with the previously delivered plant input rather than blended with the continually changing outgoing command. Let $\mathbf{u}_{new}(t)$ be the incoming controller output and $\mathbf{u}(t_s^-)$ the last delivered command. The initial offset is

$$\mathbf{b}_0 = \mathbf{u}(t_s^-) - \mathbf{u}_{\mathrm{new}}(t_s) \tag{19}$$

For normalized transfer time $s = clamp[(t - t_s)/T_b, 0, 1]$, the delivered target is

$$\mathbf{u}_{\mathrm{tgt}}(t) = \mathbf{u}_{\mathrm{new}}(t) + \rho(s)\mathbf{b}_0 \tag{20}$$

$$\rho(s) = 1 - (10s^3 - 15s^4 + 6s^5), \qquad T_b = 0.10 \text{ s} \tag{21}$$

Although this construction produces a smooth transition from the pre-switch command to the incoming controller output, it does not blend two simultaneously evolving controller commands; only the fixed offset computed at the switching instant is decayed. The quintic decay gives continuous endpoint slope and curvature and makes the first handover sample exactly continuous. The target command is first subjected to the common physical limits. A per-channel slew limiter then restricts the normalized delivered-command increment to 0.01 of the corresponding command range per 5-ms sample. If the transfer finishes before the delivered command reaches the moving target, the limiter remains active until catch-up is complete. The command is saturated again after slew limiting for numerical safety. The method follows the objective of classical bumpless-transfer designs [28,29] while retaining the incoming controller's dynamics rather than continuously blending two moving closed-loop commands.

2.10. Final Monte Carlo campaign and statistical analysis

The publication-scale campaign contains 500 independent missions and four paired configurations per mission: fixed APID, fixed LQR, the protected perfect-information oracle, and the frozen causal conservative supervisor. The 500 missions comprise 125 payload/wind-dominant, 125 rotor-degradation-dominant, 188 sequential mixed, and 62 boundary/ambiguous cases. They are arranged into five independently shuffled 100-mission batches with approximately preserved stratum proportions. This ordering permits interpretable convergence analysis without allowing one mission class to dominate an early prefix of the sample sequence. The campaign contains 2,000 nonlinear simulations. A checkpoint is written after every controller run, and no result from this campaign is used to change the architecture.

For configuration i, reliability is estimated as $\hat{R}_i = s_i/n$, where s_i is the number of successful missions among $n = 500$ paired cases. Two-sided 95% Wilson intervals quantify binomial uncertainty. The practical supervisor is compared with each fixed controller using the exact paired McNemar test based on discordant mission outcomes. Because two fixed-controller superiority hypotheses were predeclared, the familywise significance level of 0.05 is controlled using a Bonferroni-adjusted threshold of 0.025 for each comparison.

The retained oracle gain is defined relative to the best fixed reference as

$$G_{\mathrm{ret}} = \frac{\hat{R}_{\mathrm{practical}} - \hat{R}_{\mathrm{fixed}}}{\hat{R}_{\mathrm{oracle}} - \hat{R}_{\mathrm{fixed}}} \tag{22}$$

Paired mission resampling with 10,000 bootstrap replicates provides 95% confidence intervals for G_{ret} and for $\Delta R = \hat{R}_{practical} - \hat{R}_{oracle}$. Oracle noninferiority is declared when the one-sided 95% lower bootstrap bound for ΔR exceeds the predeclared margin of -0.05. In addition to reliability, the final evidence package reports phase-specific failures, rescue and harmful-loss counts, successful-mission position RMSE, controller agreement, causal detection delay, estimator errors, switch count, post-latch reversals, normalized command-transition increments, and Monte Carlo convergence at 20, 40, 60, 80, 100, 150, 200, 250, 300, 400, and 500 missions.

The architecture was considered validated only if it retained at least 70% of the oracle gain, passed the 5-percentage-point oracle-noninferiority test, significantly outperformed both fixed controllers, limited the median successful-mission RMSE increase relative to the oracle to 10%, maintained median and maximum switch counts no greater than one and two, respectively, produced no post-latch reversal, achieved rotor recall of at least 85%, kept the rotor false-positive rate below 5%, and maintained median causal detection delay below 0.75 s. These gates were declared before the final 500-mission results were generated. For reproducibility, Table 3 summarizes the frozen estimator, supervisory-decision, command-transfer, and statistical-validation settings used throughout the final campaign. No parameter listed in Table 5 was modified after the architecture was frozen and before the 500-mission evaluation was completed.

Table 5. Frozen estimator, supervisory, transfer, and final-validation settings.

Component	Parameter	Frozen value
Causal estimator	Window / update interval	0.50 s / 0.25 s
Controller conditioning	Minimum active-controller purity	0.80
Rotor detector	Activation / deactivation probability	0.25 / 0.20
Fault evidence	Grace / latch confirmation	0.35 s / 0.10 s
Conservative lookup	Effectiveness allowance	0.07
Selection	Reliability / RMSE hysteresis	0.02 / 0.005 m
Selection	Minimum dwell / candidate confirmation	0.25 s / 0.10 s
Transfer	Offset-decay duration	0.10 s
Command protection	Normalized per-channel step limit	0.01 per 5-ms sample
Final validation	Missions / paired simulations	500 / 2,000
Bootstrap	Paired replicates	10,000
Noninferiority	Practical-oracle margin	-0.05

3. Results and Discussion

3.1. Controller complementarity and supervisory-policy development

The numerical development first verified that APID and LQR occupy genuinely complementary operating regions. Adaptive PID was consistently preferred during nominal, payload, and wind phases because it preserved lower tracking errors and substantially higher payload–wind reliability. LQR, by contrast, retained substantially more missions after rotor loss of effectiveness. This complementarity is not visible from a single aggregate tracking metric: the same controller that is advantageous early in a sequential mission can become the less reliable choice after degradation. The resulting supervisory problem is therefore temporal and requirement dependent rather than a conventional gain-selection problem.

The pilot ablation in Fig. 4 shows why both estimation and decision-policy design were necessary. The perfect-information oracle raised pilot reliability from 37.5% for APID and 35.0% for LQR to 77.5%. A first practical implementation with a generic confidence gate remained at 37.5% because low-confidence or out-of-distribution windows repeatedly prevented the APID-to-LQR safety transition. Removing the gate improved reliability to 47.5%, but symmetric switching generated as many as 11 controller changes and failed all 15 sequential mixed missions. Permanent fault latching eliminated these reversals. The fault-latched, asymmetric, and conservative policies each recovered 77.5% reliability with at most one switch, and the conservative policy was frozen because it produced the lowest calibration RMSE among the equally reliable policies.

This ablation isolates an important control-design result. Estimator accuracy alone did not guarantee mission reliability; the meaning of uncertainty had to be interpreted in the context of fault permanence. A transient low-confidence estimate during payload or wind operation should inhibit a performance-motivated switch, whereas persistent rotor evidence should enable a one-way safety transition even if the complete feature vector is temporarily outside the nominal calibration envelope. The final policy therefore treats APID-to-LQR and LQR-to-APID decisions asymmetrically and prohibits reversal after confirmed permanent degradation.

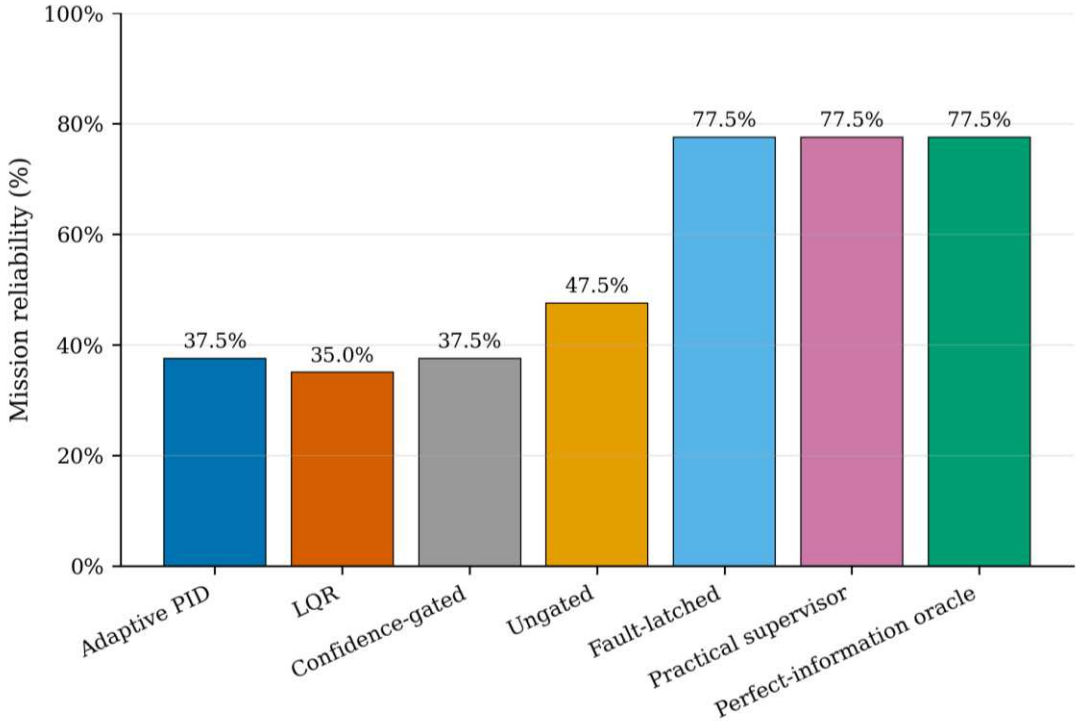


Figure 4. Pilot-stage reliability ablation from fixed controllers and naïve estimator-driven supervision to the final fault-latched conservative policy.

3.2. Hierarchical estimator validation and causal online behavior

The initial four-class operating-mode classifier achieved only 57.8% balanced accuracy on the protected holdout because the ambiguous/unknown class combined transitions and overlapping physical conditions rather than representing one coherent mode. The hierarchical reformulation resolved this limitation by detecting rotor degradation separately, estimating payload, wind, and rotor effectiveness continuously, and using low confidence as an unknown-state gate instead of a learned physical class. Offline holdout replay achieved 98.81% rotor recall, 99.88% specificity, 99.70% precision, and a 0.12% false-positive rate. The residual-sign decoder identified the affected rotor with 100% conditional accuracy, while payload and wind mean absolute errors (MAEs) were 0.042 kg and 0.233 m/s, respectively. Figure 5 compares the protected offline result with the final causal online campaign. Rotor recall decreased modestly from 98.81% to 91.31%, as expected when centered offline windows were replaced by trailing causal windows and the estimator was exposed to controller transitions and a broader independent mission distribution. Specificity remained above 99.6%, and the final false-positive rate was only 0.326%. The degradation detector therefore retained a high-confidence separation between damaged and non-damaged operation without assuming true fault timing or rotor effectiveness.

The online estimator produced one missed mild-fault mission and nine false-alarm missions among 500 cases. The missed case ended with a minimum rotor effectiveness of approximately 0.979 and did not create an oracle–practical outcome disagreement. The false alarms were concentrated in payload/wind-dominant cases with mild sampled effectiveness variation; all nine missions still satisfied their requirements. These residual errors are therefore relevant to estimator calibration but did not reduce the final reliability. The more consequential output was the causal detection delay: the median was 0.393 s and the 95th percentile was 0.587 s, both within the predeclared 0.75-s limit.

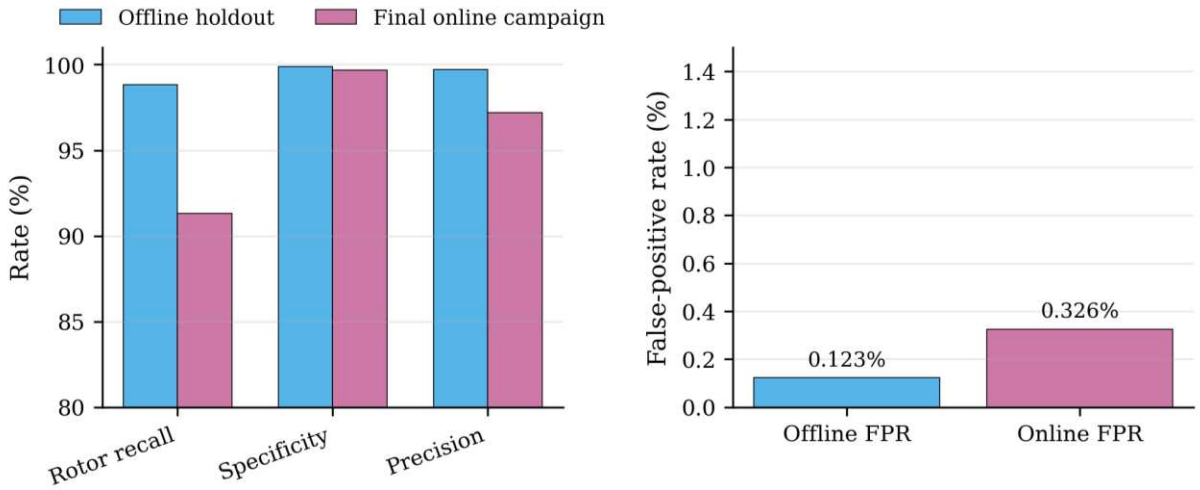


Figure 5. Rotor-degradation detector performance in the offline holdout replay and the independent causal online campaign. Online specificity is reported as one minus the false-positive rate.

3.3. Final mission reliability and paired statistical evidence

The publication-scale campaign comprised 500 independent missions and 2,000 paired simulations. The practical supervisor completed 356 missions, corresponding to 71.2% reliability with a 95% Wilson interval of 67.1–75.0%. Adaptive PID completed 175 missions (35.0%, 30.9–39.3%), while LQR completed 127 missions (25.4%, 21.8–29.4%). The perfect-information oracle also completed 356 missions. As shown in Fig. 6, the practical and oracle intervals are identical because their binary mission outcomes matched in every realization, whereas both are clearly separated from the fixed-controller intervals.

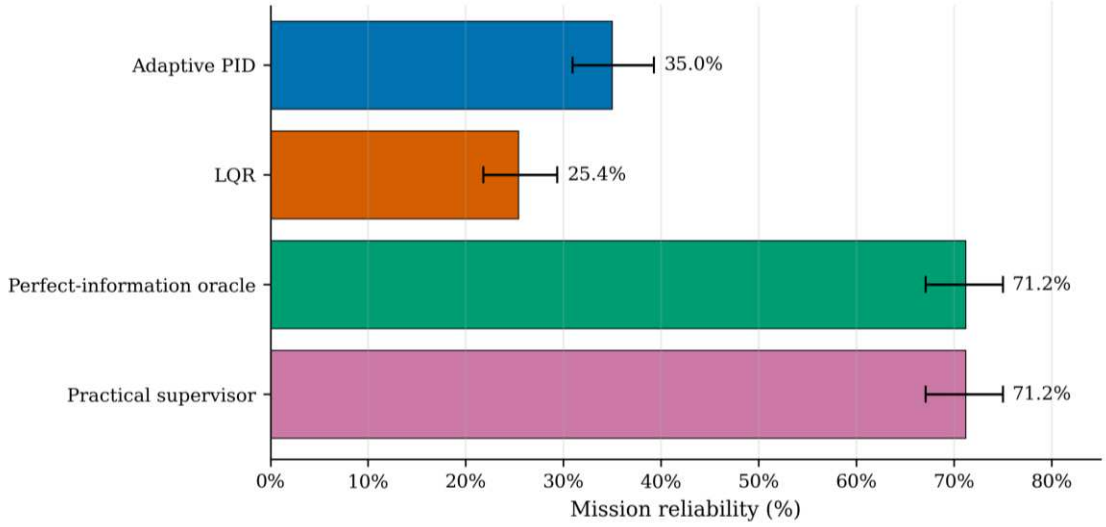


Figure 6. Final mission reliability with two-sided 95% Wilson confidence intervals.

The absolute reliability improvement relative to APID was 36.2 percentage points and relative to LQR was 45.8 percentage points. These differences were not produced by sacrificing missions already completed by the fixed controllers. As shown in Fig. 7, the practical supervisor achieved 181 additional successes relative to APID and 229 additional successes relative to LQR, whereas neither fixed controller achieved a success in any mission failed by the supervisor. Exact McNemar tests gave $p = 6.53 \times 10^{-55}$ against APID and $p = 2.32 \times 10^{-69}$ against LQR. within the 500-mission campaign, the practical supervisor strictly expanded the observed success sets of both fixed-controller candidates.

The practical–oracle comparison contained 356 joint successes and 144 joint failures, with no discordant outcomes. Consequently, the retained oracle gain was exactly 100%, and the paired bootstrap distribution collapsed to one for

retained gain and zero for the practical-minus-oracle reliability difference. Figure 8 summarizes the corresponding predeclared acceptance tests. The one-sided lower confidence bound for the reliability difference was zero, comfortably above the −5-percentage-point noninferiority margin, while retained gain exceeded the 70% threshold by 30 percentage points. These results support the claim that the causal estimator and fault-latched decision policy recovered the full reliability opportunity identified by perfect operating-condition knowledge. Table 6 consolidates the corresponding mission-level reliability statistics and successful-mission tracking performance for all four final configurations, including the Wilson confidence intervals and median position RMSE values.

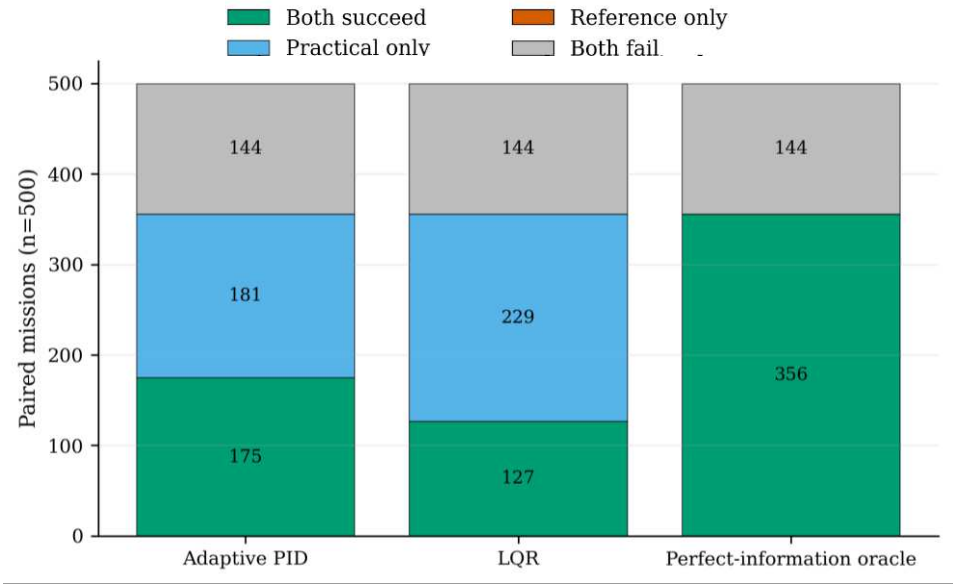


Figure 7. Paired mission outcomes comparing the practical supervisor with each benchmark controller.

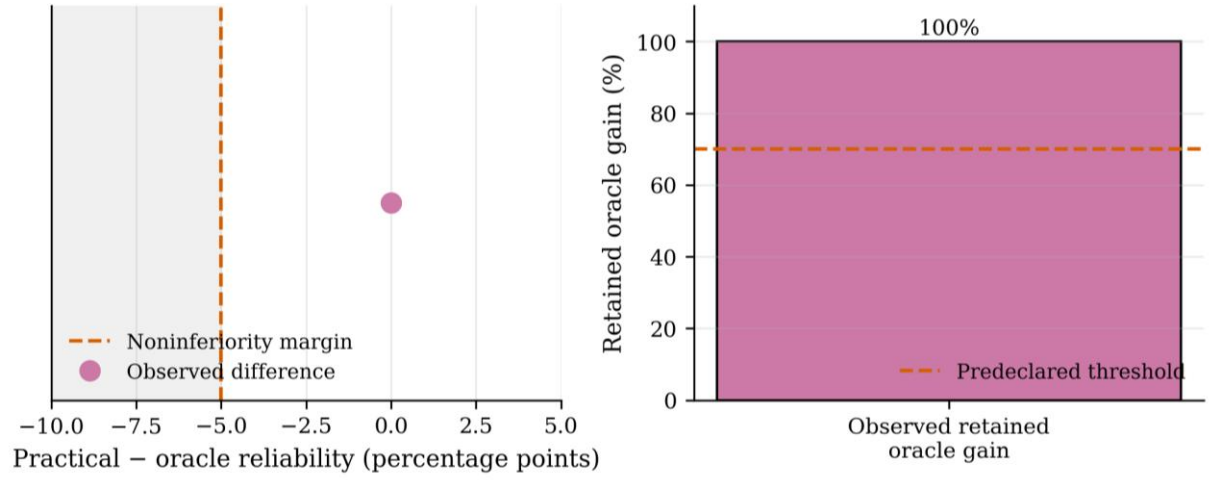


Figure 8. Predeclared noninferiority and retained-oracle-gain acceptance evidence.

Table 6. Final reliability and successful-mission tracking performance.

Configuration	Successes	Reliability	95% Wilson interval	Median successful RMSE (m)
Adaptive PID	175/500	35.0%	30.9–39.3%	0.0872
LQR	127/500	25.4%	21.8–29.4%	0.1244
Perfect-information oracle	356/500	71.2%	67.1–75.0%	0.1014
Practical supervisor	356/500	71.2%	67.1–75.0%	0.1030

3.4. Mission-stratum performance and operating-region interpretation

The aggregate improvement arose from different mechanisms in the four mission strata (Fig. 9). In payload/wind-dominant cases, APID, the oracle, and the practical supervisor each achieved 92.8% reliability, whereas fixed LQR achieved only 10.4%. The practical controller therefore preserved APID's strong disturbance-rejection behavior rather than switching unnecessarily to the degradation-oriented candidate. The nine remaining failures in this stratum occurred during payload or wind transitions and were shared with the oracle.

Rotor-degradation-dominant missions produced the opposite ordering. APID failed all 125 cases, while LQR completed 49.6% and both supervisors completed 51.2%. The 1.6-percentage-point improvement above fixed LQR reflects cases in which APID remained advantageous before the degradation transition and the protected handover occurred after sufficient fault evidence had accumulated. The final-effectiveness trend in Fig. 10 shows that reliability decreased as available rotor authority was reduced; practical and oracle curves remained coincident throughout the sampled range. The convergence of all candidate curves at the most severe conditions indicates a controller-bank authority boundary rather than a supervisory-selection error.

Sequential mixed missions provided the strongest evidence for supervision. APID completed none of the 188 cases because its nominal and disturbance advantages did not persist after substantial rotor loss. Fixed LQR completed only 12.2% because its poorer payload/wind behavior consumed tracking margin before the fault. The practical supervisor completed 60.6%, exactly matching the oracle. This result demonstrates that controller complementarity must be exploited in time: selecting the correct fixed controller at mission start cannot reproduce an architecture that remains with APID during payload and wind phases and then performs one protected transition to LQR after confirmed degradation.

All 62 boundary/ambiguous missions were completed by the practical and oracle supervisors, compared with 95.2% for APID and 46.8% for LQR. Figure 11 maps the payload–wind dominant subset over payload mass and wind magnitude. The APID and practical maps are nearly identical because the final policy deliberately retains APID unless persistent rotor evidence justifies the safety transition. The heatmaps also show that the residual failures are concentrated near combinations of high payload and high wind rather than being distributed randomly.

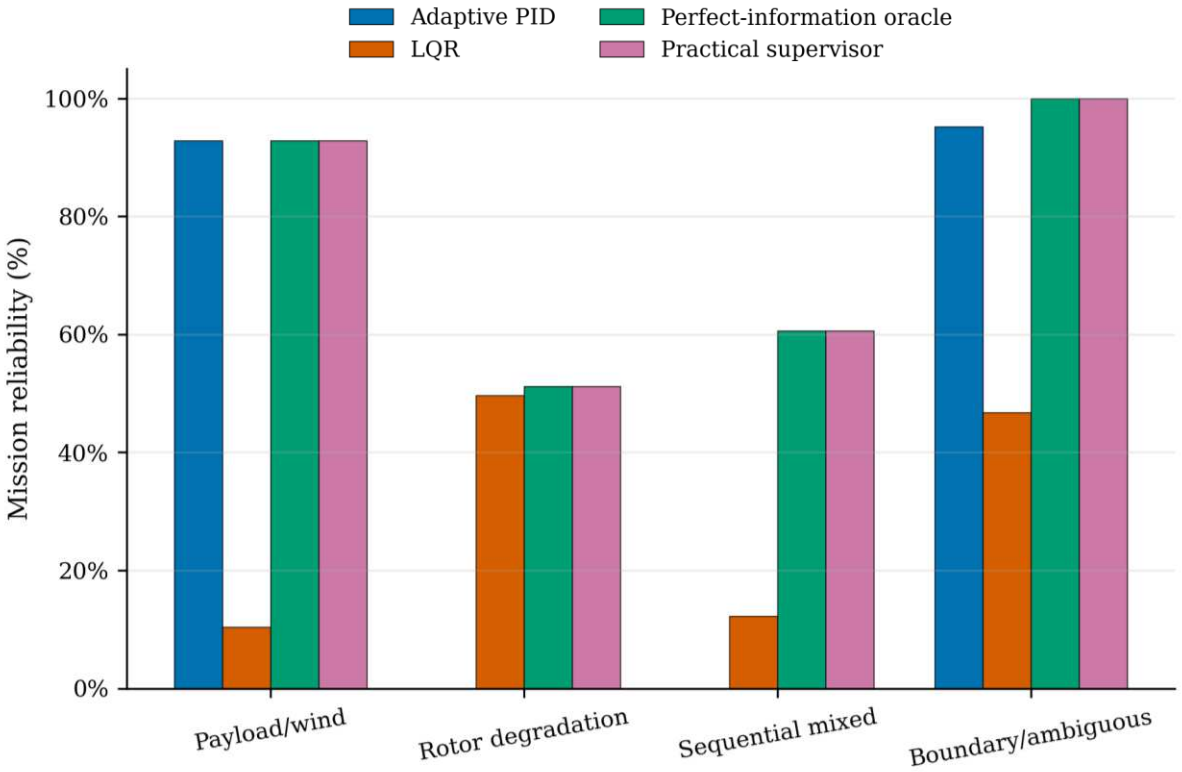


Figure 9. Mission reliability by predeclared operating stratum.

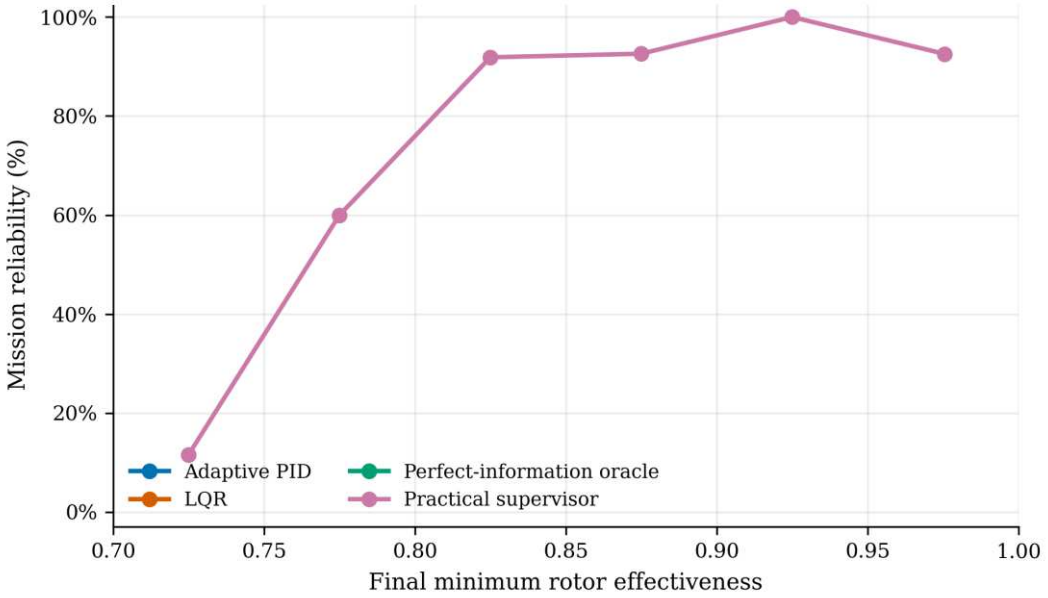


Figure 10. Reliability as a function of final minimum rotor effectiveness for missions containing rotor degradation.

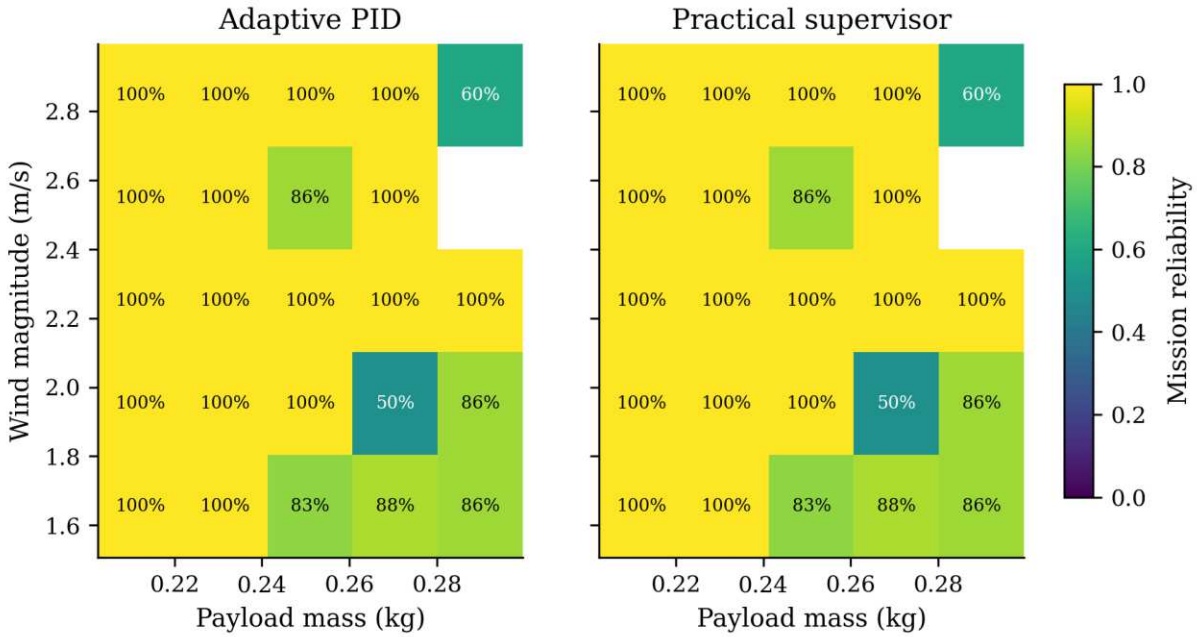


Figure 11. Binned payload–wind operating-region reliability for fixed APID and the practical supervisor.

3.5. Tracking performance and practical–oracle equivalence

Reliability improvement was not obtained through unbounded tracking-error tolerance. Conditional on mission success, APID had the smallest median position RMSE at 0.0872 m because its successful subset is dominated by payload/wind and boundary cases. The corresponding medians were 0.1244 m for LQR, 0.1014 m for the oracle, and 0.1030 m for the practical supervisor (Fig. 12). The practical architecture therefore improved mission reliability by more than 36 percentage points relative to APID while increasing successful-mission median RMSE by only 0.0158 m relative to APID and 0.00167 m relative to the oracle.

A paired comparison is more informative than the marginal distributions because practical and oracle completed the same 356 missions. Figure 13 plots their RMSE values for these jointly successful cases. The points remain tightly

clustered around the identity line. The mean practical-minus-oracle RMSE difference was 0.00068 m and the median was 0.00043 m, corresponding to less than one millimetre. The relative increase in the median successful-mission RMSE was 1.65%, well below the predeclared 10% limit.

The small practical penalty is consistent with its causal nature. The oracle can begin evaluating the degraded reliability map immediately from true effectiveness, whereas the practical system must accumulate a trailing measurement window and confirm fault evidence. Nevertheless, the protected switch typically occurred before the available tracking margin was exhausted. Exact agreement in controller choice at every instant was therefore unnecessary; it was sufficient for the practical decision to occur within the time interval in which the safer controller could still satisfy the phase requirements.

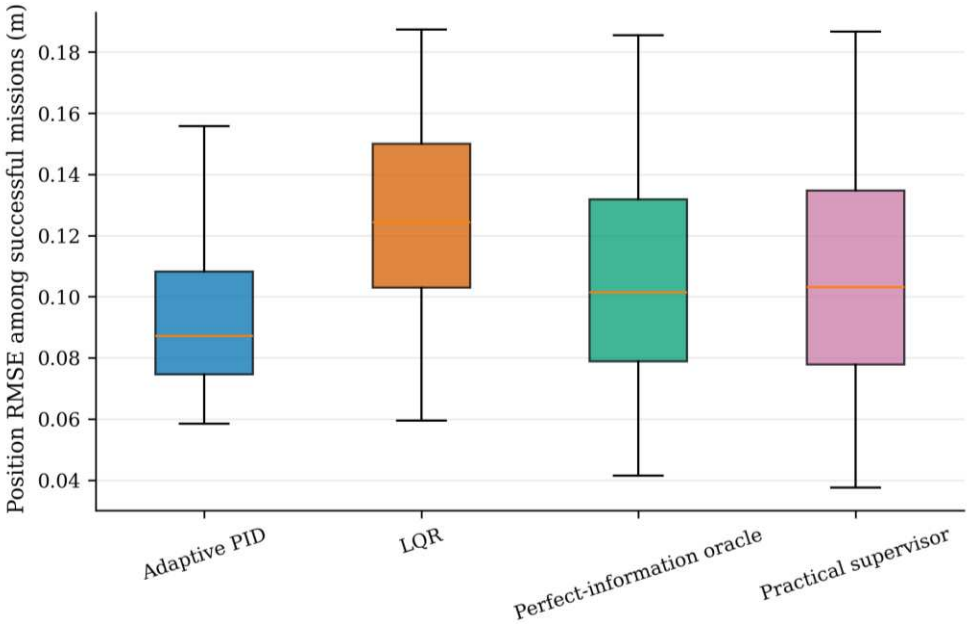


Figure 12. Distribution of position RMSE among successful missions for the four final configurations.

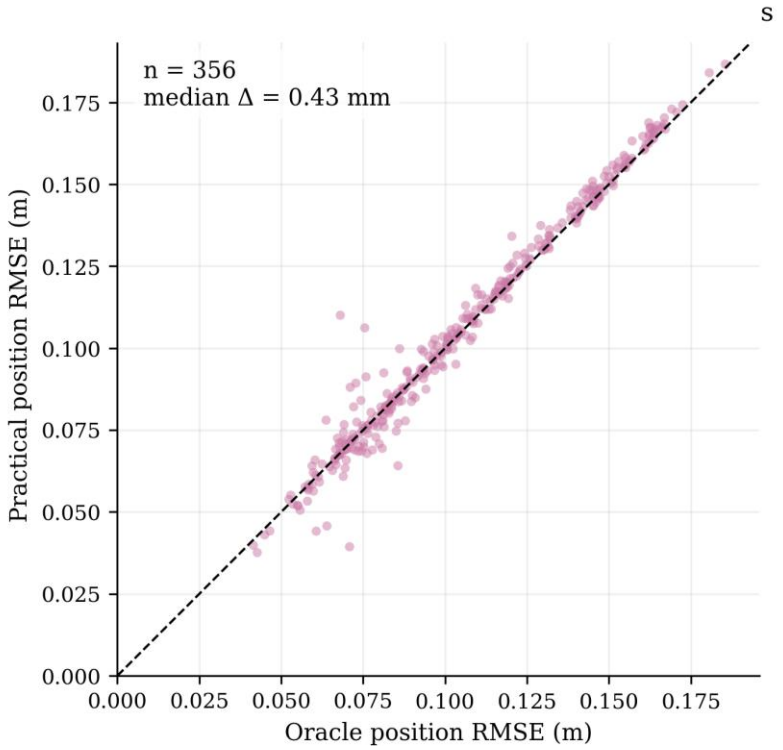


Figure 13. Paired position RMSE for the 356 missions completed by both the practical supervisor and the perfect-information oracle.

3.6. Estimator errors, detection delay, and controller agreement

The continuous condition estimates remained sufficiently accurate for reliability-map interrogation. Across the final campaign, the mean mission-level payload MAE was 0.0432 kg, wind MAE was 0.373 m/s, and minimum-effectiveness MAE was 0.0240. Figure 14 compares the mission-level error distributions with the design targets. Payload and wind remained within their 0.05-kg and 0.50-m/s limits. The effectiveness error also satisfied the 0.03 target in the final campaign after the conservative 0.07 decision margin reduced sensitivity to optimistic point estimates.

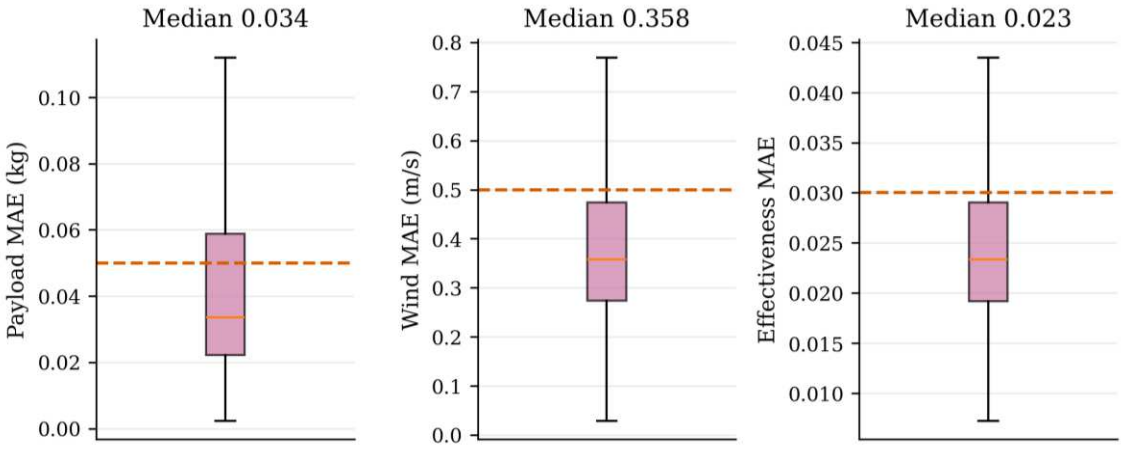


Figure 14. Mission-level payload, wind, and rotor-effectiveness estimation errors. Dashed lines indicate the design targets.

Detection delays are shown by stratum in Fig. 15. Sequential and degradation-dominant missions had similar central delays because both contained sustained progressive rotor loss. Payload/wind cases produced a smaller set of finite delays associated primarily with mild effectiveness variation or false alarms. The median delay over all detected faults was 0.393 s, with a 95th percentile of 0.587 s. No stratum approached the 0.75-s acceptance threshold at its median. The practical supervisor agreed with the oracle's active controller for 92.57% of mission time on average and 98.59% at the median (Fig. 16). The median absolute switch-timing error was 0.195 s. Missions with lower time-wise agreement often included intervals in which the oracle selected LQR for a mild or short-lived degradation indication while the practical controller remained with APID; these differences did not change the phase-level outcome. This provides a useful interpretation of the estimator's role: its objective is not exact reconstruction of every hidden parameter, but timely discrimination of decisions that change requirement satisfaction. Table 7 summarizes the final causal-estimator metrics against their predeclared acceptance criteria. All evaluated estimator quantities satisfied their respective thresholds, including rotor-fault recall, false-positive rate, causal detection delay, payload and wind estimation errors, rotor-effectiveness error, and affected-rotor identification accuracy.

Table 7. Final causal-estimator performance and comparison with the predeclared acceptance criteria.

Estimator metric	Final value	Acceptance criterion	Outcome
Rotor recall	91.31%	$\geq 85\%$	Pass
Rotor false-positive rate	0.326%	$\leq 5\%$	Pass
Median causal detection delay	0.393 s	≤ 0.75 s	Pass
Payload MAE	0.0432 kg	≤ 0.05 kg	Pass
Wind MAE	0.373 m/s	≤ 0.50 m/s	Pass
Effectiveness MAE	0.0240	≤ 0.03	Pass
Rotor-identity accuracy	100.0%	$\geq 90\%$	Pass

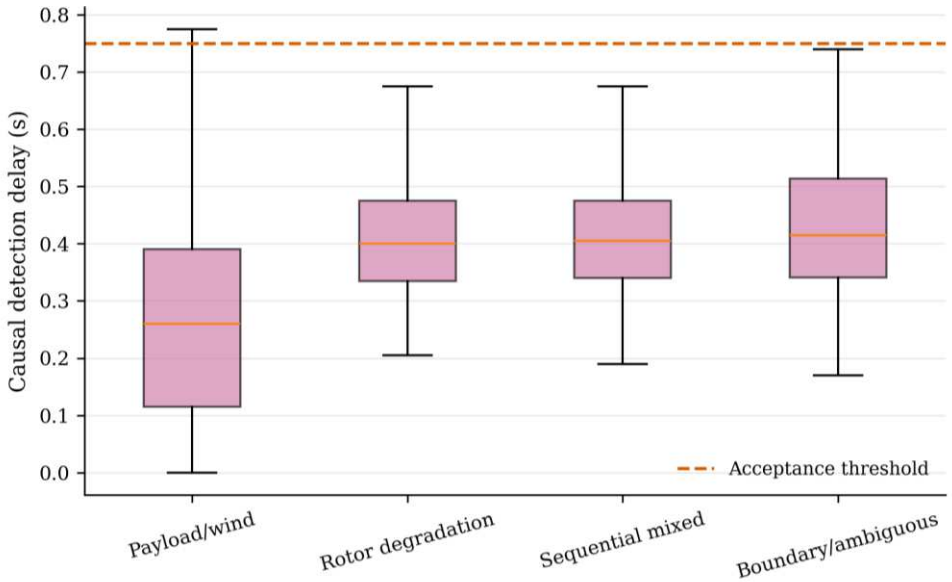


Figure 15. Distribution of causal rotor-degradation detection delay by mission stratum. The dashed line marks the 0.75-s acceptance limit.

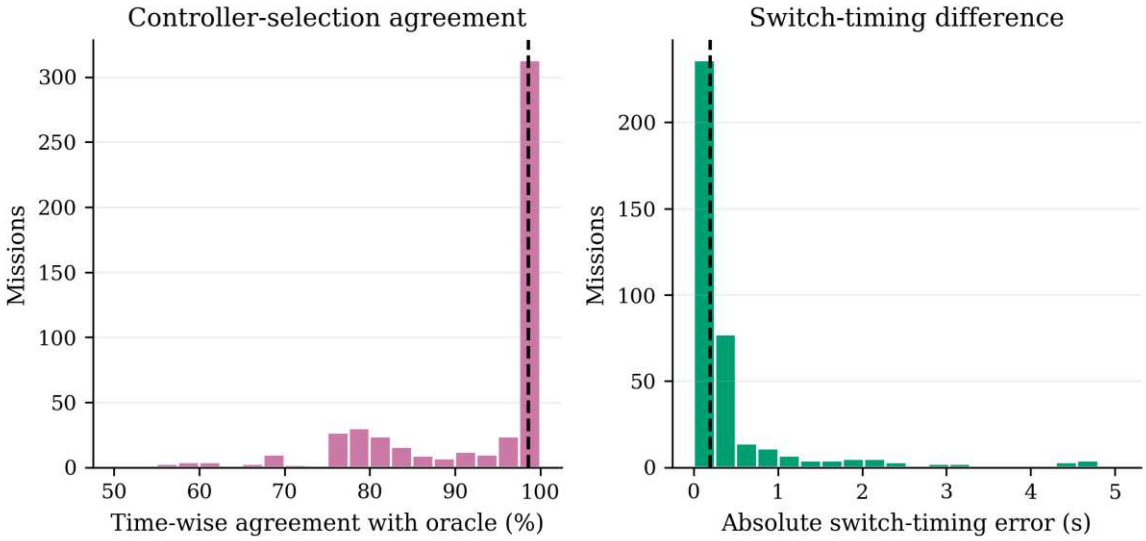


Figure 16. Time-wise controller agreement with the oracle and absolute switch-timing error.

3.7. Switching behavior and bumpless-transfer performance

The final policy produced a median and maximum switch count of one, with zero post-latch LQR-to-APID reversals. This is a decisive improvement over the ungated symmetric pilot, which reached 11 switches and failed every sequential mission. The fault latch converted the controller sequence into a physically consistent APID-to-LQR safety handover whenever permanent rotor degradation was confirmed. No transfer saturation was observed in either the oracle or practical configuration.

Figure 17 compares the normalized transition increment and post-switch position error. The practical supervisor had a median P95 normalized transition increment of 0.0189 and a maximum normalized increment of 0.0307. The corresponding median post-switch peak position error was 0.1125 m. These values are close to the oracle despite the practical switch occurring later and from estimated conditions. The offset-decay handover therefore preserved command continuity without obscuring the reliability effect of the controller change.

The final results also clarify why generic convex command blending was rejected during development. Blending continuously varying APID and LQR commands reduced the initial jump but introduced additional transition motion and increased post-switch tracking peaks. The frozen offset-based transfer instead cancels the incoming command

discontinuity at the switching instant and then decays only the fixed offset. Combined with the normalized slew limiter, this maintained bounded increments while allowing the incoming controller to assume authority rapidly.

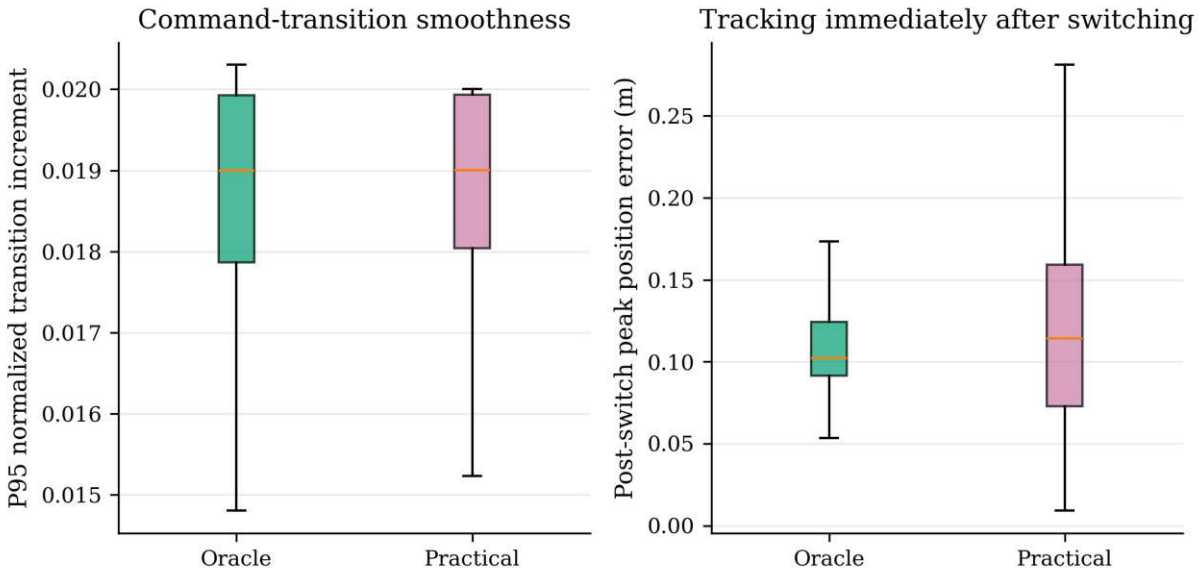


Figure 17. Bumpless-transfer quality for the oracle and practical supervisor, including normalized transition increments and post-switch tracking peaks.

3.8. Convergence, shared failures, and limitations

The reliability estimates converged rapidly enough to support the selected campaign size. Figure 18 shows that the practical and oracle estimates were identical at every checkpoint and stabilized near 70–71% after approximately 150–200 missions. The final value of 71.2% remained inside a narrow 67.1–75.0% Wilson interval. The fixed-controller curves also stabilized well below the supervisory result, indicating that the conclusion was not driven by one late batch or by the ordering of mission strata.

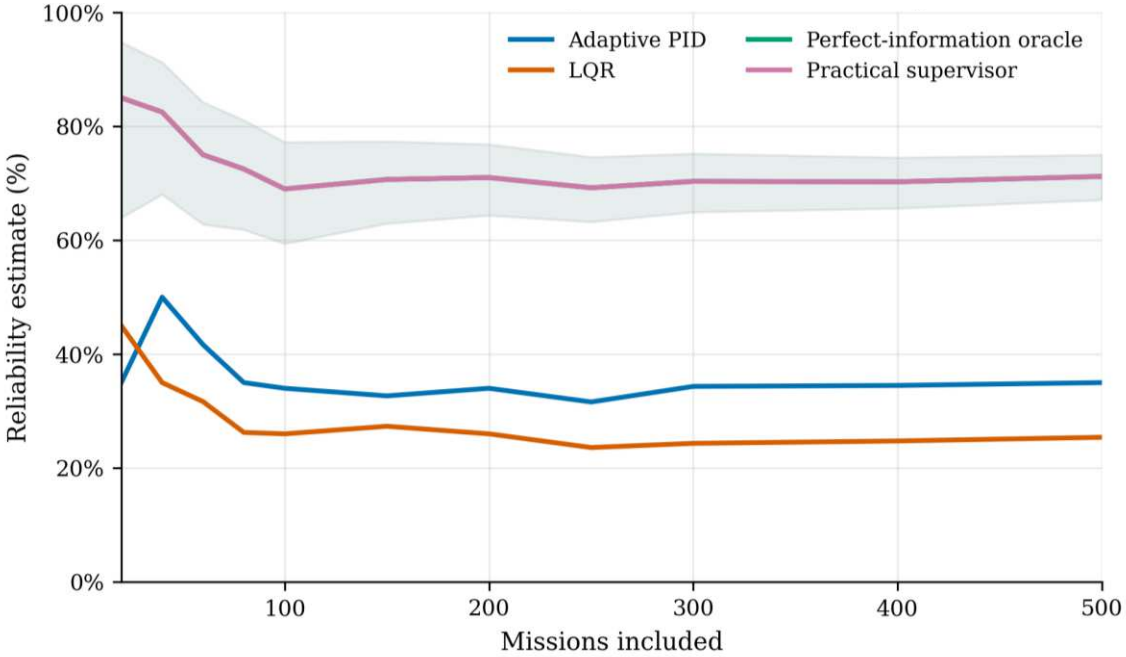


Figure 18. Monte Carlo convergence of reliability for the four final configurations. Shading denotes Wilson intervals for the two supervisory configurations.

All 144 practical failures were shared with the oracle. Figure 19 shows the first failed phase. The dominant limitations occurred during the rotor-degradation ramp and degraded-flight phase; 124 of the 144 shared failures, or 86.1%, were associated with degraded operation in sequential or degradation-dominant missions. Because the practical controller reproduced the oracle's outcome in every realization, these cases cannot be attributed to causal estimation, confidence logic, switch timing, or command transfer. They represent operating points for which neither APID nor LQR provided sufficient phase-level margin.

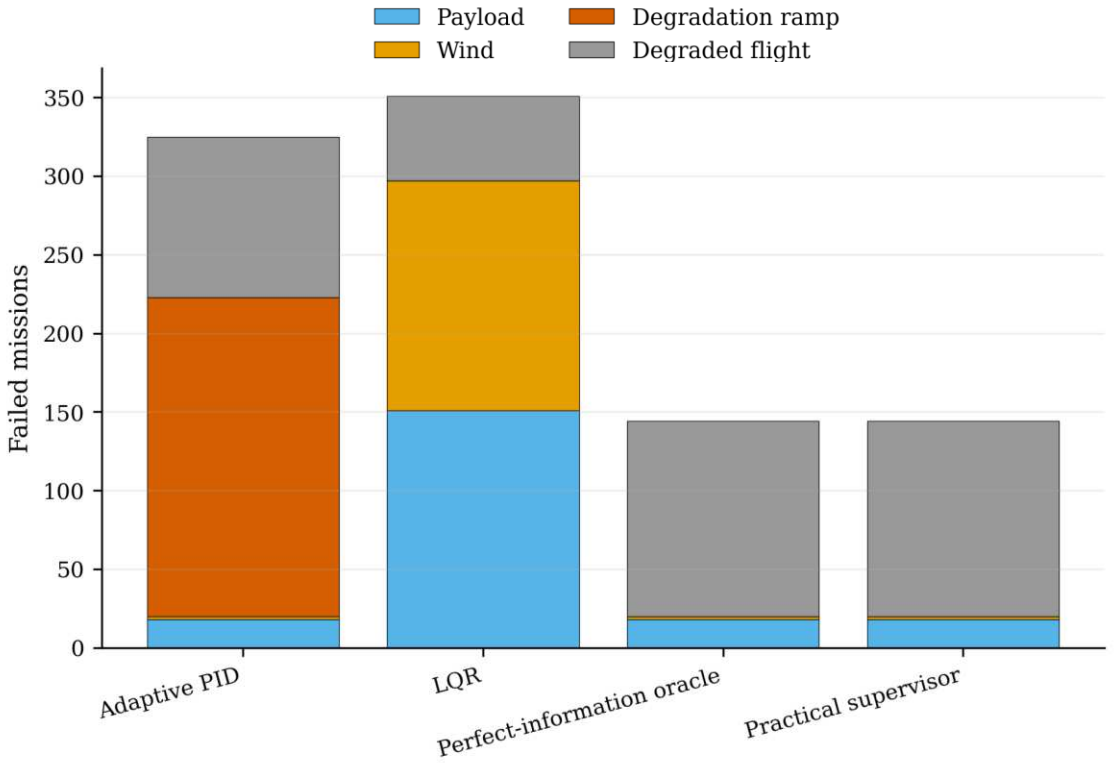


Figure 19. First failed mission phase for each final configuration. The oracle and practical supervisor share the same failure counts.

Taken together, the final statistical, tracking, switching, and estimator results satisfied all predeclared validation criteria for the frozen architecture. As summarized in Table 8, the practical supervisor retained 100% of the oracle reliability gain, passed the 5-percentage-point noninferiority criterion, significantly outperformed both fixed controllers in the paired comparisons, limited the relative increase in median successful-mission RMSE to 1.65%, maintained a median and maximum switch count of one, and produced no post-latch reversals. The final campaign therefore satisfied every acceptance gate established before the 500-mission evaluation.

Table 8. Predeclared final-architecture acceptance assessment.

Acceptance item	Observed result	Criterion	Outcome
Retained oracle gain	100.0%	$\geq 70\%$	Pass
Oracle noninferiority	Lower bound = 0.0 pp	Above -5 pp	Pass
Superiority vs both fixed controllers	Exact paired tests	$p < 0.025$ each	Pass
Relative median RMSE increase	1.65%	$\leq 10\%$	Pass
Median / maximum switches	1 / 1	≤ 1 / ≤ 2	Pass
Post-latch reversals	0	$= 0$	Pass

Despite satisfying all predeclared acceptance criteria, several limitations define the scope of these conclusions. The study used a validated nonlinear simulation rather than flight hardware, sampled uncertainties from prescribed bounded distributions, and considered permanent single-rotor loss of effectiveness without explicit control reallocation. Reliability is therefore conditional on the phase requirements and operating envelope defined in the

methodology. In addition, the APID and LQR gains, reliability maps, and controller-conditioned estimator were calibrated for the nominal four-rotor X-configuration plant family considered in this study. Transfer to an airframe with substantially different mass properties, geometry, propulsion coefficients, or actuator bandwidth would require controller retuning and regeneration or adaptation of the associated reliability and estimator models. In particular, although the final campaign included motor-time-constant perturbations of approximately $\pm 6\%$, substantially slower propulsion dynamics could produce command–response residuals that differ from those represented in the present calibration set. Extension to such systems should incorporate the altered actuator dynamics explicitly, or augment the estimator with online propulsion-bandwidth estimation. Changes in rotor count or configuration would additionally require revised control allocation and fault-identity templates. Broader validation should therefore examine cross-airframe transfer, propulsion-bandwidth variation, hardware timing, unmodelled structural dynamics, multiple or recoverable actuator faults, and operating envelopes beyond the present payload, wind, and degradation ranges.

Within these bounds, the evidence supports the intended contribution. Requirement-aware supervision more than doubled the reliability of the best fixed candidate, the causal estimator retained the full oracle gain, and the permanent fault latch removed the chattering failure mode exposed by symmetric switching. The complete development sequence also illustrates that estimator performance, switching logic, and command-transition design must be evaluated jointly; high classification accuracy alone would not have predicted the failure of the initial confidence-gated or ungated supervisors.

A direct next step is implementation of the frozen supervisory architecture on the in-house quadrotor prototype shown in Fig. 2(a), beginning with experimental identification of the vehicle mass properties, propulsion coefficients, actuator bandwidth, and sensor characteristics, followed by controller retuning, estimator recalibration, hardware-in-the-loop testing, and flight validation.

4. Conclusion

This study developed and validated a causal, reliability-aware supervisory flight-control architecture for quadrotors operating under time-varying payload, wind, and rotor-degradation conditions. The framework was motivated by the complementary behavior of adaptive PID and LQR: adaptive PID provided superior tracking under nominal, payload, and wind conditions, whereas LQR offered greater resilience after rotor effectiveness loss. A perfect-information supervisor first established the reliability benefit achievable through controller selection. This upper-bound architecture was then converted into a practical system using a controller-conditioned hierarchical estimator, requirement-based reliability maps, asymmetric decision logic, permanent fault latching, and protected bumpless transfer.

The estimator used only onboard-accessible measurements and commands within a causal 0.50 s trailing window. It estimated payload, wind magnitude, rotor effectiveness, and affected-rotor identity while supporting a directional APID-to-LQR safety transition. Development studies showed that generic confidence gating could suppress necessary switches, whereas unrestricted symmetric switching produced chattering and mission failures. The final latched policy eliminated post-fault reversals and limited every mission to at most one controller transition.

In the final 500-mission campaign, the practical supervisor achieved 71.2% reliability, exactly matching the perfect-information oracle and substantially exceeding fixed adaptive PID and LQR, which achieved 35.0% and 25.4%, respectively. No mission completed by either fixed controller was lost by the practical supervisor. The improvements were statistically significant, and the practical architecture passed the predeclared 5 percentage-point noninferiority test relative to the oracle. Its successful-mission median tracking error was only 1.65% higher than that of the oracle. Online estimation remained effective, with 91.31% rotor-fault recall, a 0.326% false-positive rate, 0.393 s median detection delay, and 100% affected-rotor identification accuracy.

The remaining failures were shared by the oracle and practical supervisors and were concentrated in severe degraded-flight phases, indicating limitations of the APID–LQR controller bank rather than the estimation or supervisory logic. Future work should therefore address higher-authority fault-tolerant controllers, actuator reallocation, cross-airframe and propulsion-system transfer, and online actuator-dynamics identification. A direct next step will be implementation of the frozen supervisory architecture on the in-house quadrotor prototype shown in Fig. 2(a), beginning with

experimental identification of its mass properties, propulsion coefficients, actuator bandwidth, and sensor characteristics, followed by controller retuning, estimator recalibration, hardware-in-the-loop testing, and flight validation. The framework also provides a reproducible pathway for integrating mission requirements, estimator confidence, and statistically validated controller selection within safety-critical autonomous systems. Overall, the results demonstrate that causal condition estimation can replace perfect uncertainty knowledge while preserving reliability-aware supervisory benefits.

Author Contributions: Conceptualization, M.R., J.B., A.C., and J.C.; Methodology, M.R., J.B., A.C., and J.C.; Software, M.R.; Formal analysis, M.R., J.B., and A.C.; Writing - original draft, M.R.; Preparation, M.R., J.B., A.C.; Data curation, M.R.; Writing - review and editing, J.B. and A.C., and J.C. All authors have read and agreed to the published version of the manuscript.

Funding: The authors declare none.
Conflicts of Interest: The authors declare no conflicts of interest.
Competing interests: The authors declare none.

Nomenclature

Notation convention. Bold lowercase letters denote vectors, bold uppercase letters denote matrices, $(\cdot)^{\mathsf{T}}$ denotes transpose, and a hat denotes an estimated quantity. SI units are shown where a single physical unit applies; "mixed" denotes generalized force–moment or gain quantities whose units depend on the channel.

A. Roman symbols

Symbol	Definition	SI unit
$\mathbf{a}_c$	Commanded inertial translational acceleration	m s^{-2}
$\mathbf{a}_r$	Reference/feedforward translational acceleration	m s^{-2}
$\hat{\mathbf{a}}$	Estimated translational acceleration	m s^{-2}
$\mathbf{B}(\boldsymbol{\eta}_r)$	Rotor-effectiveness-dependent control-allocation matrix	mixed
$\mathbf{B}_0$	Nominal control-allocation matrix	mixed
$\mathbf{b}_0$	Initial generalized-command offset used for bumpless transfer	mixed
$\mathbf{D}$	Translational drag-coefficient matrix	N s m^{-1}
$\mathbf{E}(\boldsymbol{\eta})$	Transformation from body angular rates to Euler-angle rates	–
$\mathbf{e}_3$	Inertial vertical unit vector, $[0\ 0\ 1]^{\mathsf{T}}$	–
$\mathbf{e}_p$	Position-tracking error	m
$\mathbf{e}_v$	Velocity-tracking error	m s^{-1}
$\mathbf{e}_\eta$	Attitude-tracking error	rad
$\mathbf{e}_\omega$	Body-angular-rate tracking error	rad s^{-1}
$\mathbf{f}_d$	Unmodelled translational disturbance force	N
$\mathbf{F}_c$	Commanded inertial force vector	N
F_r	Latched binary rotor-fault state	–
g	Gravitational acceleration	m s^{-2}
G_{ret}	Fraction of the perfect-information oracle reliability gain retained by the practical supervisor	–
h_R	Reliability hysteresis threshold	–
h_e	Predicted-RMSE tie-break hysteresis threshold	m
i	Controller or rotor index, according to context	–
$i^\star$	Controller selected by the reliability-aware decision rule	–
$\mathbf{J}$	Actual vehicle inertia matrix	kg m^2
$\mathbf{J}_0$	Nominal vehicle inertia matrix	kg m^2
$\mathbf{K}$	LQR state-feedback gain matrix	mixed
$\mathbf{K}_{p,p}$, $\mathbf{K}_{i,p}$, $\mathbf{K}_{d,p}$	Proportional, integral, and derivative gain matrices of the position loop	mixed
$\mathbf{K}_{p,\eta}$, $\mathbf{K}_{i,\eta}$, $\mathbf{K}_{d,\eta}$	Proportional, integral, and derivative gain matrices of the attitude loop	mixed
m	Actual total vehicle mass	kg
m_0	Nominal vehicle mass used by the controllers and estimator residuals	kg
m_p	Payload mass	kg
$\hat{m}_p$	Estimated payload mass	kg
$\mathbf{p}$	Vehicle position vector in the inertial frame	m
$\mathbf{p}_r$	Reference position vector	m
q	Active mission phase or requirement index	–
$\mathbf{R}(\boldsymbol{\eta})$	Rotation matrix from body coordinates to inertial coordinates	–
$R_i(\mathbf{z}, q)$	Requirement-conditioned reliability of controller i at condition $\mathbf{z}$ and phase q	–
$\hat{R}_i$	Interpolated or estimated reliability score of controller i	–
$\mathbf{r}_F$	Translational force residual used by the causal estimator	N
$\mathbf{r}_\tau$	Rotational moment residual used by the causal estimator	N m
S_i	Binary requirement-satisfaction indicator for controller i	–
s	Normalized bumpless-transfer time, $\mathrm{clamp}[(t - t_s)/T_b, 0,1]$	–
T	Total rotor-generated thrust	N
T_c	Commanded collective thrust	N
T_b	Bumpless-transfer offset-decay duration	s

T_{confirm}	Candidate-controller or fault-evidence confirmation interval	s
T_{dwell}	Minimum controller dwell time	s
t	Time	s
t_f	Time at which rotor degradation is permanently latched	s
t_s	Authorized controller-switching time	s
$\mathbf{u}$	Delivered generalized control command	mixed
$\mathbf{u}_a$	Actual generalized force–moment vector produced by the rotors	mixed
$\mathbf{u}_{\mathrm{new}}$	Command generated by the incoming controller	mixed
$\mathbf{u}_{\mathrm{tgt}}$	Bumpless-transfer target command	mixed
$\mathbf{v}$	Vehicle inertial velocity vector	m s^{-1}
$\mathbf{v}_r$	Reference velocity vector	m s^{-1}
$\mathbf{v}_w$	Wind-velocity vector	m s^{-1}
$\widehat{W}$	Estimated wind magnitude	m s^{-1}
$\mathbf{x}$	Nonlinear quadrotor state vector	mixed
$\delta\mathbf{x}$	Perturbation state used in the hover-linearized LQR model	mixed
$\delta\mathbf{u}$	LQR generalized-control increment	mixed
$\mathbf{z}$	Operating-condition vector used for reliability-map lookup	mixed
$\hat{\mathbf{z}}$	Causally estimated operating-condition vector	mixed

B. Greek symbols

Symbol	Definition	SI unit
$\boldsymbol{\alpha}_c$	Commanded angular-acceleration vector	rad s^{-2}
$\widehat{\boldsymbol{\alpha}}$	Estimated angular-acceleration vector	rad s^{-2}
$\boldsymbol{\eta}$	Euler-angle vector, $[\phi\ \theta\ \psi]^{\mathsf{T}}$	rad
$\boldsymbol{\eta}_r$	Rotor-effectiveness vector	–
$\hat{\eta}_{\mathrm{min}}$	Estimated minimum rotor effectiveness	–
$\hat{\eta}_{\mathrm{latched}}$	Rotor-effectiveness estimate stored after fault confirmation	–
η_{dec}	Conservative rotor-effectiveness value used for reliability-map lookup	–
$\boldsymbol{\xi}_p$	Bounded position-error integral state	m s
$\boldsymbol{\xi}_\eta$	Bounded attitude-error integral state	rad s
$\boldsymbol{\omega}$	Body-angular-rate vector	rad s^{-1}
$\boldsymbol{\Omega}$	Rotor-speed vector	rad s^{-1}
Ω_i	Actual speed of rotor i	rad s^{-1}
$\Omega_{i,c}$	Commanded speed of rotor i	rad s^{-1}
$\boldsymbol{\tau}$	Actual control-moment vector	N m
$\boldsymbol{\tau}_c$	Commanded control-moment vector	N m
$\boldsymbol{\tau}_d$	External disturbance-moment vector	N m
$\tau_{m,i}$	Motor time constant of rotor i	s
$\rho(s)$	Quintic bumpless-transfer offset-decay function	–
$\boldsymbol{\vartheta}_j$	Adaptive PID parameter vector for axis j	mixed
$\boldsymbol{\vartheta}_{j,0}$	Nominal adaptive-parameter vector for axis j	mixed
$\boldsymbol{\phi}_j$	Adaptive-law regressor for axis j	mixed
$\bar{\boldsymbol{\phi}}_j$	Scale-normalized adaptive-law regressor	–
ε_j	Dead-zone-filtered adaptive error signal	–
$\boldsymbol{\Gamma}_j$	Adaptation-rate matrix for axis j	mixed
σ	Leakage coefficient toward the nominal adaptive gains	s^{-1}

λ	Regularization coefficient in logistic or ridge regression	–
ΔR	Reliability difference between the practical and oracle supervisors	percentage points or –

C. Operators, decorations, and subscripts

Notation	Meaning
$(\cdot)^{\mathsf{T}}$	Matrix or vector transpose
$\dot{(\cdot)}$	Time derivative
$\widehat{(\cdot)}$	Estimated quantity
$\bar{(\cdot)}$	Normalized or averaged quantity, according to context
$(\cdot)^{-}$	Left-hand limit immediately before a switching instant
$\mathrm{diag}(\cdot)$	Diagonal-matrix operator
$\mathrm{sat}(\cdot)$	Physical saturation operator
$\mathrm{Proj}\{\cdot\}$	Projection onto the prescribed adaptive-parameter bounds
$\mathrm{clamp}(\cdot)$	Restriction of a scalar to a specified interval
$\Pr(\cdot)$	Probability operator
argmax	Argument that maximizes the specified quantity
$\min(\cdot)$, $\max(\cdot)$	Minimum and maximum operators
Subscript 0	Nominal or baseline quantity
Subscript c	Commanded quantity
Subscript d	Disturbance quantity
Subscript i	Controller or rotor index
Subscript j	Adaptive axis or channel index
Subscript p	Position-loop or payload quantity, according to context
Subscript r	Reference quantity or rotor-effectiveness quantity, according to context
Subscript w	Wind quantity
Subscripts “primary” and “secondary”	Primary requirement map and secondary safety map
Subscripts “oracle”, “practical”, and “fixed”	Perfect-information supervisor, causal practical supervisor, and best fixed-controller reference

D. Abbreviations

Notation	Meaning
APID	Adaptive proportional–integral–derivative control
CI	Confidence interval
FDI	Fault detection and identification
FPR	False-positive rate
FTC	Fault-tolerant control
HIL	Hardware-in-the-loop
IRLS	Iteratively reweighted least squares
LQR	Linear quadratic regulator/regulation
MAE	Mean absolute error
MC	Monte Carlo
OOD	Out-of-distribution
PID	Proportional–integral–derivative control
RMSE	Root-mean-square error
RMS	Root mean square
UAV	Unmanned aerial vehicle

References

1. Idrissi, M., Salami, M., Annaz, F.: A review of quadrotor unmanned aerial vehicles: applications, architectural design and control algorithms. Journal of Intelligent & Robotic Systems 104, 22 (2022). https://doi.org/10.1007/s10846-021-01527-7
2. Sonugür, G.: A review of quadrotor UAV: control and SLAM methodologies ranging from conventional to innovative approaches. Robotics and Autonomous Systems 161, 104342 (2023). https://doi.org/10.1016/j.robot.2022.104342
3. Mahony, R., Kumar, V., Corke, P.: Multirotor aerial vehicles: modeling, estimation, and control of quadrotor. IEEE Robotics & Automation Magazine 19(3), 20–32 (2012). https://doi.org/10.1109/MRA.2012.2206474
4. Pounds, P., Mahony, R., Corke, P.: Modelling and control of a large quadrotor robot. Control Engineering Practice 18(7), 691–699 (2010). https://doi.org/10.1016/j.conengprac.2010.02.008
5. Lee, H., Kim, H.J.: Trajectory tracking control of multirotors from modelling to experiments: a survey. International Journal of Control, Automation and Systems 15(1), 281–292 (2017). https://doi.org/10.1007/s12555-015-0289-3
6. Rinaldi, M., Primatesta, S., Guglieri, G.: A comparative study for control of quadrotor UAVs. Applied Sciences 13(6), 3464 (2023). https://doi.org/10.3390/app13063464
7. Xi, H., Zhang, D., Zhou, T., Yang, Y., Wei, Q.: An anti-wind modeling method of quadrotor aircraft and cascade controller design based on improved extended state observer. International Journal of Control, Automation and Systems 19(3), 1363–1374 (2021). https://doi.org/10.1007/s12555-019-0878-7
8. Razmi, H., Afshinfar, S.: Neural network-based adaptive sliding mode control design for position and attitude control of a quadrotor UAV. Aerospace Science and Technology 91, 12–27 (2019). https://doi.org/10.1016/j.ast.2019.04.055
9. Mechali, O., Xu, L., Huang, Y., Shi, M., Xie, X.: Observer-based fixed-time continuous nonsingular terminal sliding mode control of quadrotor aircraft under uncertainties and disturbances for robust trajectory tracking: theory and experiment. Control Engineering Practice 111, 104806 (2021). https://doi.org/10.1016/j.conengprac.2021.104806
10. Labbadi, M., Boukal, Y., Cherkaoui, M.: Robust tracking control for a quadrotor subjected to external disturbances and model uncertainties. PLOS ONE 18(4), e0283195 (2023). https://doi.org/10.1371/journal.pone.0283195
11. Jung, W., Bang, H.: Fault and failure tolerant model predictive control of quadrotor UAV. International Journal of Aeronautical and Space Sciences 22(3), 663–675 (2021). https://doi.org/10.1007/s42405-020-00331-1
12. Yang, P., Wang, Z., Zhang, Z., Hu, X.: Sliding mode fault tolerant control for a quadrotor with varying load and actuator fault. Actuators 10(12), 323 (2021). https://doi.org/10.3390/act10120323
13. Wang, B., Shen, Y., Zhang, Y.M.: Active fault-tolerant control for a quadrotor helicopter against actuator faults and model uncertainties. Aerospace Science and Technology 99, 105745 (2020). https://doi.org/10.1016/j.ast.2020.105745
14. Wang, X., van Kampen, E.-J., Chu, Q.: Quadrotor fault-tolerant incremental nonsingular terminal sliding mode control. Aerospace Science and Technology 95, 105514 (2019). https://doi.org/10.1016/j.ast.2019.105514
15. Gao, B., Liu, Y.-J., Liu, L.: Adaptive neural fault-tolerant control of a quadrotor UAV via fast terminal sliding mode. Aerospace Science and Technology 129, 107818 (2022). https://doi.org/10.1016/j.ast.2022.107818
16. Ahmadi, K., Asadi, D., Merheb, A., Nabavi-Chashmi, S.Y., Tutsoy, O.: Active fault-tolerant control of quadrotor UAVs with nonlinear observer-based sliding mode control validated through hardware-in-the-loop experiments. Control Engineering Practice 137, 105557 (2023). https://doi.org/10.1016/j.conengprac.2023.105557
17. Hu, X., Wang, B., Shen, Y., Fu, Y., Li, N.: Disturbance observer-enhanced adaptive fault-tolerant control of a quadrotor UAV against actuator faults and disturbances. Drones 7(8), 541 (2023). https://doi.org/10.3390/drones7080541
18. Li, Y., Zhu, X., Yin, G.: Robust actuator fault detection for quadrotor UAV with guaranteed sensitivity. Control Engineering Practice 138, 105588 (2023). https://doi.org/10.1016/j.conengprac.2023.105588
19. Chang, C.-C., Cheng, T.-H.: Motor-efficiency estimation and control of multirotors comprising a cooperative transportation system. IEEE Access 11, 36566–36578 (2023). https://doi.org/10.1109/ACCESS.2023.3266510
20. Amoozgar, M.H., Chamseddine, A., Zhang, Y.M.: Experimental test of a two-stage Kalman filter for actuator fault detection and diagnosis of an unmanned quadrotor helicopter. Journal of Intelligent & Robotic Systems 70, 107–117 (2013). https://doi.org/10.1007/s10846-012-9757-7
21. Lien, Y.-H., Peng, C.-C., Chen, Y.-H.: Adaptive observer-based fault detection and fault-tolerant control of quadrotors under rotor failure conditions. Applied Sciences 10(10), 3503 (2020). https://doi.org/10.3390/app10103503
22. Hamadi, H., Lussier, B., Fantoni, I., Francis, C.: Data fusion fault-tolerant strategy for a quadrotor UAV under sensors and software faults. ISA Transactions 129, 520–539 (2022). https://doi.org/10.1016/j.isatra.2022.01.007
23. Baldini, A., Felicetti, R., Freddi, A., Monteriù, A.: Fault-tolerant control of a variable-pitch quadrotor under actuator loss of effectiveness and wind perturbations. Sensors 23(10), 4907 (2023). https://doi.org/10.3390/s23104907
24. Okada, K.F.Á., Morais, A.S., Ribeiro, L., Amaral da Luz, C.M., Tofoli, F.L., Lima, G.V., Lopes, L.C.O.: Fault-tolerant control for quadcopters under actuator and sensor faults. Sensors 24(22), 7299 (2024). https://doi.org/10.3390/s24227299
25. Zuo, L., Yao, L.: Fault isolation and fault-tolerant control for the uncertain quadrotor UAV system. International Journal of Control, Automation and Systems 22(1), 301–310 (2024). https://doi.org/10.1007/s12555-022-0249-7
26. Adıgüzel, F., Kurtuluş, K., Türker, T.: A gain scheduling attitude controller with NN supervisor for quadrotor UAVs. International Journal of Control, Automation and Systems 22, 3777–3791 (2024). https://doi.org/10.1007/s12555-024-0458-3
27. Yang, H., Jiang, B., Staroswiecki, M.: Supervisory fault tolerant control for a class of uncertain nonlinear systems. Automatica 45(10), 2319–2324 (2009). https://doi.org/10.1016/j.automatica.2009.06.019

28. Bendtsen, J.D., Stoustrup, J., Trangbæk, K.: Bumpless transfer between observer-based gain scheduled controllers. International Journal of Control 78(7), 491–504 (2005). https://doi.org/10.1080/00207170500111028
29. Zheng, K., Başar, T., Bentsman, J.: H∞ bumpless transfer under controller uncertainty. IEEE Transactions on Automatic Control 54(7), 1718–1723 (2009). https://doi.org/10.1109/TAC.2009.2020648
30. Kantue, P., Pedro, J.O.: Integrated fault-tolerant control of a quadcopter UAV with incipient actuator faults. International Journal of Applied Mathematics and Computer Science 32(4), 601–617 (2022). https://doi.org/10.34768/amcs-2022-0042
31. Masuda, K., Uchiyama, K.: Attitude control system for quadrotor using robust Monte Carlo model predictive control. Actuators 13(11), 443 (2024). https://doi.org/10.3390/act13110443
32. Wang, C.H.J., Tan, S.K., Low, K.H.: Three-dimensional (3D) Monte-Carlo modeling for UAS collision risk management in restricted airport airspace. Aerospace Science and Technology 105, 105964 (2020). https://doi.org/10.1016/j.ast.2020.105964
33. Ranjan, S., Majhi, S.: Adaptive neural predefined-time attitude control of an uncertain quadrotor UAV with actuator fault. IEEE Transactions on Circuits and Systems II: Express Briefs 71(12), 4939–4943 (2024). https://doi.org/10.1109/TCSII.2024.3433430
34. Sarkar, N., Deb, A.K.: Dynamic event-triggered neuroadaptive fault-tolerant control of quadrotor UAV with a novel cosine kernel. Aerospace Science and Technology 155, 109643 (2024). https://doi.org/10.1016/j.ast.2024.109643
35. Mallavalli, S., Fekih, A.: An observer-based backstepping integral nonsingular fast terminal sliding mode fault-tolerant control design for quadrotors under different types of actuator faults. International Journal of Control, Automation and Systems 21, 4015–4031 (2023). https://doi.org/10.1007/s12555-022-0951-5
36. PX4 Development Team: Iris quadrotor simulation model. PX4-SITL Gazebo Classic. https://github.com/PX4/PX4-SITL_gazebo-classic/blob/main/models/iris/iris.sdf.jinja. Accessed 27 Sept 2026
37. Bryson, A.E., Jr., Ho, Y.-C.: Applied Optimal Control: Optimization, Estimation, and Control. Hemisphere Publishing, New York (1975)