UAV Autonomy & Robust Control Testbed

Robust PID Stabilization, Vibration Mitigation, and Deterministic Failsafe Behavior on a 250 mm Quadcopter

Rajat Gupta

Abstract

This report presents the design, implementation, and experimental evaluation of a 250 mm quadrotor testbed developed to study classical control robustness under asymmetric mass loading, mechanical vibration, and communication loss. Independent PID control loops executed on an embedded flight controller provide real-time stabilization, while a deterministic failsafe state machine ensures predictable behavior during signal loss. Additional sim-to-real autonomy experiments were prototyped using NVIDIA Isaac Sim and deployed to a companion computer, highlighting practical limitations imposed by latency and sensor noise.

1 System Overview

The platform consists of a lightweight carbon-fiber quadcopter configured for rapid iteration and instrumentation. All flight-critical control logic executes directly on the flight controller to guarantee hard real-time performance. A companion computer is used exclusively for non-real-time autonomy prototyping and data logging.

1.1 Hardware Components

Subsystem	**Component Details**
Airframe	250 mm carbon-fiber quadcopter frame
Motors	Readytosky 2306-class brushless motors $\times 4$
Propellers	5-inch tri-blade FPV propellers
ESC	Vishnu 50A ESC (DShot600)
Flight Controller	Brahma F4 MK-III
Companion Computer	Raspberry Pi 4B
Navigation	u-blox NEO-6M GPS module
Power	4S LiPo battery

Table 1: Primary system components

2 Control Strategy

Stabilization is achieved through independent PID control loops for roll, pitch, and yaw, executing entirely on the flight controller. The control objective prioritized disturbance rejection, vibration tolerance, and predictable transient response over aggressive maneuverability.

PID gains were tuned empirically using step-response testing and sustained hover evaluation. Particular emphasis was placed on limiting derivative gain to avoid amplification of high-frequency vibration-induced IMU noise. Dynamic notch filtering was applied to suppress dominant motor–propeller harmonic bands while preserving adequate phase margin.

The control law implemented for each axis is:

$$u(t) = K_p e(t) + K_i \int e(t)\, dt + K_d \frac{de(t)}{dt} \tag{1}$$

where:

- K_p provides immediate corrective torque proportional to attitude error.
- K_i compensates for steady-state biases caused by asymmetric loading and thrust imbalance.
- K_d damps oscillatory behavior by reacting to angular rate changes.

2.1 Disturbance Rejection: Asymmetric Mass Experiment

To evaluate robustness under non-ideal conditions, a 5.67 g coin was affixed to one arm of the quadcopter, introducing a static asymmetric mass disturbance.

This perturbation produced the following effects:

- **Mechanical Imbalance:** Motors operated at unequal steady-state thrust levels to maintain hover.
- **Harmonic Resonance Shift:** Added mass altered arm resonance frequencies, increasing IMU sensor noise.
- **Vibration Hardening:** Increased proportional gain and deployment of notch filters mitigated induced oscillations.

The experiment demonstrated stable hover capability despite persistent structural asymmetry.

3 Deterministic Failsafe Design

A deterministic failsafe state machine was implemented to handle communication loss scenarios, prioritizing controlled descent over mission continuation. All failsafe logic executes directly on the flight controller to guarantee bounded response latency.

3.1 Failsafe Implementation

- **Link Detection:** RSSI monitoring triggers failsafe at approximately 15–25% signal strength.
- **Attitude Neutralization:** Roll and pitch commands are driven toward 0° within 100–150 ms.
- **Controlled Descent:** Linear throttle decay produces a descent deceleration of approximately 0.4–0.6 $\mathrm{m/s^2}$.
- **Ground Detection:** A Z-axis accelerometer spike triggers motor disarm at a touchdown velocity of 1.2–1.8 m/s.

Overall failsafe detection latency remained below 150 ms, ensuring predictable and repeatable descent behavior without free-fall.

4 Sim-to-Real Transfer: Autonomous Guidance

A learning-based guidance layer was prototyped to reduce pilot workload during non-nominal conditions. Policies were trained in NVIDIA Isaac Sim using domain randomization over mass distribution, sensor noise, and communication latency.

4.1 Architecture

- **Training Environment:** Isaac Sim with randomized physics parameters.
- **Deployment Target:** Raspberry Pi 4B generating low-frequency reference commands.
- **Safety Arbitration:** Manual RC input always overrides autonomous commands.
- **Control Isolation:** Stabilization and failsafe logic remain exclusively on the flight controller.

Integration was prototyped but not deployed in flight. Communication latency exceeding 50 ms and vibration-induced sensor noise produced a significant sim-to-real gap. Future work requires tighter domain randomization and hardware-in-the-loop validation.

5 Experimental Results

5.1 Hover and Stability

Metric	Clean	Coin-Loaded
Hover Drift (XY)	10–15 cm	15–25 cm
Roll Std. Dev.	0.5–0.9°	0.9–1.6°
Pitch Std. Dev.	0.5–0.8°	0.8–1.5°
Yaw Drift	low	low–moderate
Stable Hover Duration	20–60 s	15–45 s

Table 2: Hover performance comparison

5.2 IMU Noise and Vibration

Metric	Clean	Pre-filter	Filtered
Gyro Noise Floor	0.04–0.08 dps	0.10–0.20 dps	0.08–0.16 dps
Dominant Band	120–200 Hz	100–300 Hz	attenuated
PSD Reduction	—	—	5–12 dB
D-term Stability	stable	borderline	stable

Table 3: IMU vibration characteristics

5.3 Failsafe Performance

Metric	Value (Range)
RSSI Threshold	15–25%
Detection Latency	100–150 ms
Attitude Neutralization	250–450 ms
Descent Deceleration	0.4–0.6 $\mathrm{m/s^2}$
Touchdown Velocity	1.2–1.8 m/s
Outcome	Controlled descent, no free-fall

Table 4: Failsafe performance metrics

6 Conclusion

This testbed demonstrates that classical PID control, when carefully tuned and supported by appropriate filtering and deterministic failsafe logic, can maintain robust stability under asymmetric loading, vibration, and communication loss. The work highlights practical limits of sim-to-real autonomy transfer in small UAVs and underscores the importance of real-time isolation and hardware-level safety guarantees.