



MASTER ERASMUS MUNDUS
EMARO+ “EUROPEAN MASTER IN ADVANCED ROBOTICS”

2024 / 2025

Master Thesis Report

Presented by

Emanuele Bua Odetti

On 26/08/2025

**Mechatronic Design of Aerial Robots for Agile
Maneuvers and Collaborative Transportation**

Jury

President:	Olivier Kermorgant	Professor (ECN)
External jury:	Claudio Pacchierotti	CNRS researcher (CNRS)
Supervisors: (EMARO)	Marco Tognon Gianluca Corsini Szymon Bielenin Olivier Kermorgant Marco Baglietto	ISFP researcher (INRIA) Research Engineer (CNRS) PhD student (INRIA) Professor (ECN) Professor (UNIGE)

Laboratory: Rainbow team at IRISA/Inria, Rennes

Abstract

This thesis presents the mechatronic design, modeling, and experimental validation of a novel quadrotor platform aimed at collaborative aerial transportation and agile flight research. The project was motivated by the obsolescence of existing laboratory Unmanned Aerial Vehicle (UAV)s, which lacked the agility, Thrust-to-Weight Ratio (TWR), and modularity required for modern multi-agent and dynamic applications.

A complete design workflow was developed, combining:

- a hardware-in-the-loop test bench for precise thrust and torque identification,
- a custom UAV configurator for component selection and performance envelope analysis,

The test bench provided high-fidelity motor models and accurate actuator allocation matrices, while the configurator enabled evaluation of achievable linear and angular accelerations under different mass and geometry configurations.

The proposed quadrotor platform, named *BumbleBee*, features a compact frame, low center of mass, and modular electronics integration, achieving significant improvements in maneuverability and actuation authority. Simulation-based cascade PID tuning was successfully transferred to real flight tests, validated through square trajectory tracking, sensitivity analyses, and the CIRP maneuver. Comparative experiments against the legacy MKQuad platform demonstrated more than a two-fold improvement in angular acceleration, alongside higher TWR and dynamic responsiveness.

Furthermore, the platform's ability to execute highly agile maneuvers was demonstrated through differential-flatness-based trajectory generation. Real-world experiments included a 360-degree flip and a powerloop, confirming that the combination of high TWR and low inertia enables precise execution of aggressive flight patterns. These demonstrations underline the platform's suitability for research in agile aerial robotics and cooperative transportation.

This thesis delivers a comprehensive methodology bridging hardware characterization, computational design tools, and real-world flight validation. The resulting platform establishes a robust foundation for future work in agile flight, and multi-robot collaboration.

Keywords: Quadrotor UAV, Mechatronic Design, Thrust-to-Weight Ratio, Cascade PID, Differential Flatness, Agile Maneuvers, Collaborative Aerial Transportation

Acknowledgements

The completion of this thesis would not have been possible without the guidance, support, and encouragement of numerous people and institutions.

First, I am profoundly grateful to my supervisors, Marco Tognon, Gianluca Corsini, and Szymon Bielenin, for their expert guidance, constructive feedback, and constant availability throughout this research journey. I also wish to thank Olivier Kermorgant and Marco Baglietto for their valuable advice and for fostering an inspiring academic environment.

I am deeply indebted to the members of the Rainbow team at IRISA/Inria, Rennes, whose collaborative spirit and stimulating discussions significantly enriched this work.

I gratefully acknowledge the EMARO+ programme, École Centrale de Nantes, and Università di Genova for providing the opportunity, resources, and academic framework that made this research possible.

On a personal note, I would like to express my heartfelt appreciation to my family and friends for their patience, encouragement, and unwavering belief in me during the challenges of this endeavor.

Notations

General and Physical Quantities

Symbol	Description
C_f, C_τ	Thrust and torque coefficients
TWR	Thrust-to-Weight Ratio
SE(3)	Special Euclidean Group in 3 dimensions
SO(3)	Special Orthogonal Group in 3 dimensions
RPM	Revolutions Per Minute
x, y, z	Translational coordinates (scalars)
ϕ, θ, ψ	Rotational angles (roll, pitch, yaw)
$f_i = C_f \omega_i^2$	Rotor thrust
$\tau_i = C_\tau \omega_i^2$	Rotor drag torque
$\mathcal{F}_W = \{x, y, z\}$	World frame
$\mathcal{F}_B = \{x_b, y_b, z_b\}$	Body frame
$\mathbf{p} = [x, y, z]_{\mathcal{W}}^\top$	Position in the world frame
$\boldsymbol{\eta} = [\phi, \theta, \psi]_{\mathcal{B}}^\top$	Euler angles (roll, pitch, yaw)
$\mathbf{R}_{B \rightarrow W}$	Rotation matrix from body to world frame
$\boldsymbol{\omega}_B = [\omega_x, \omega_y, \omega_z]^\top$	Body angular velocity
m	Mass of the quadrotor
\mathbf{J}	Inertia matrix in the body frame
ω_i	Angular velocity of the i -th rotor
$\boldsymbol{\Omega}^2 = [\omega_1^2, \omega_2^2, \omega_3^2, \omega_4^2]^\top$	Squared motor speeds
g	Gravitational acceleration
T	Total thrust (scalar)
$\boldsymbol{\tau}_B$	Control torque vector in body frame
$\mathbf{u} = [T, \tau_\phi, \tau_\theta, \tau_\psi]^\top$	Control input vector
$\tau_\phi, \tau_\theta, \tau_\psi$	Torques around roll, pitch, yaw
\mathbf{A}	Allocation matrix
\mathbf{A}^{-1}	Mixer matrix
α	Arm angle w.r.t. body x_b -axis in X-configuration
ℓ	Distance from quadrotor center to each motor
n	Number of motors
$m_{\text{component}}$	Mass of a specific component

Control and Error Quantities

Symbol	Description
ϕ_d	Desired roll
θ_d	Desired pitch
T_d	Desired thrust
$k_{p_x}, k_{p_y}, k_{p_z}$	Translational proportional gains
$k_{d_x}, k_{d_y}, k_{d_z}$	Translational derivative gains
$k_{i_x}, k_{i_y}, k_{i_z}$	Translational integral gains
\mathbf{R}	Current orientation of the quadrotor
\mathbf{R}_d	Desired orientation of the quadrotor
$\boldsymbol{\omega}_d$	Desired angular velocity
$\mathbf{e}_R = \frac{1}{2}(\mathbf{R}_d^\top \mathbf{R} - \mathbf{R}^\top \mathbf{R}_d)^\vee$	Attitude error vector
$(\cdot)^\vee$	Inverse of the hat map
$\mathbf{e}_\omega = \boldsymbol{\omega}_B - \mathbf{R}^\top \mathbf{R}_d \boldsymbol{\omega}_d$	Angular velocity error vector
$\mathbf{k}_R, \mathbf{k}_\omega$	Control gain vectors
$\hat{\boldsymbol{\omega}}_B$	Skew-symmetric matrix of $\boldsymbol{\omega}_B$
$\dot{\boldsymbol{\omega}}_d$	Desired angular acceleration
$\omega_{i_{\text{error}}} = \omega_{i_d} - \omega_i$	Rotor i -th velocity error
u_i	Control input for the i -th motor
$k_{p_\omega}, k_{i_\omega}, k_{d_\omega}$	Rotor velocity control gains
$\mathbf{y} = [x, y, z, \psi]^\top$	Flat outputs
\mathbf{a}_d	Desired translational acceleration
$\ddot{\mathbf{p}}_d$	Desired position acceleration
$\mathbf{k}_z = [0, 0, 1]^\top$	Unit vector along world Z -axis
\mathbf{f}_d	Desired thrust vector
$\mathbf{x}_c = [\cos \psi_d, \sin \psi_d, 0]^\top$	Projected heading in the inertial XY -plane
$\boldsymbol{\omega}_d^\times = \mathbf{R}_d^\top \dot{\mathbf{R}}_d$	Desired angular velocity (skew form)
$\dot{\boldsymbol{\omega}}_d^\times$	Desired angular acceleration (skew form)

Abbreviations

ADC	Analog-to-Digital Converter
BB	BumbleBee
BBFP	BumbleBee Full Power
BBLP	BumbleBee Limited Power
BEMT	Blade Element Momentum Theory
CCW	counterclockwise
CW	clockwise
CIRP	Constant Increasing Ramp Path
COM	Center of Mass
DAQ	Data acquisition
DoF	Degree of Freedom
ENU	East-North-Up
ESC	Electronic Speed Controller
FA	Fully Actuated
FC	flight controller
FLU	Forward-Left-Up
GPS	Global Positioning System
IMU	inertial measurement unit
INDI	Incremental Nonlinear Dynamic Inversion
LIO	LiDAR-Inertial Odometry
MKQ	MKQuad
MPC	Model predictive control
MTOW	Maximum Take-off Weight
NHFC	Near-Hovering Flight Controller
NMPC	Nonlinear Model predictive control
OD	Omnidirectional
PID	Proportional Integral Derivative
POM	GenoM3 component for localization
RMSE	Root Mean Square Error

RPM	Revolutions Per Minute
TWR	Thrust-to-Weight Ratio
UA	Under Actuated
UAV	Unmanned Aerial Vehicle
VIO	Visual-Inertial Odometry

List of Figures

1.1	Novel quadrotor platform developed in this work	15
3.1	Quadrotor in X configuration.	26
3.2	Cascade PID control architecture for quadrotor UAV. The outer loop regulates position, the middle loop stabilizes attitude, and the inner loop tracks the desired motor speeds.	30
4.1	Vertical motor mount with force-torque sensor installed at the base.	35
4.2	Electronics mounting plate featuring standoffs for ESC and FC installation. . . .	36
4.3	Overview of the propulsion Test-bench setup. The system integrates a thrust/-torque sensor beneath the motor and logs measurements during steady-state rotation.	38
4.4	Step input test with tuned PID ($P=5$, $I=300$, $D=0$): Measured vs reference velocity.	40
4.5	Ramp input test: system tracking a staircase velocity profile.	41
4.6	Zoomed view of steady-state convergence. Error remains below 5 Hz.	41
4.7	Commanded vs. measured motor velocity during characterization.	42
4.8	Average debiased forces: F_x , F_y , and F_z vs. motor speed.	43
4.9	Average debiased torques: M_x , M_y , and M_z vs. motor speed.	43
4.10	Estimated thrust coefficient C_f vs. commanded motor speed.	44
4.11	Estimated torque coefficient C_τ vs. commanded motor speed.	45
4.12	Fitting result for thrust data: the linear-plus-quadratic model improves prediction accuracy across the measured range.	46
4.13	Fitting result for torque data: the extended model captures deviations from ideal quadratic behavior, particularly at higher velocities.	46
4.14	Comparison of thrust vs. motor speed: measured data, fitted model, and manufacturer datasheet reference.	47
4.15	Ratio of datasheet to measured thrust values as a function of speed.	48
4.16	Measured thrust vs. motor speed with ground effect at 10 cm.	48
5.1	3D attainable acceleration envelope with hover planes.	54
5.2	3D attainable torque envelope with hover planes.	55
5.3	2D cross-section of the attainable acceleration envelope.	55
5.4	2D cross-section of the attainable torque envelope.	56
5.5	Comparison of linear and angular acceleration across platforms.	56
5.6	Final CAD Configuration (Design D): Motors and Battery Mounted Below . . .	59
5.7	Final CAD Render of the BB Platform	60

6.1	Quadrotor software architecture implemented using <code>telekyb3</code> . Components include state estimation, control, trajectory generation, and hardware interfacing. Courtesy of Gianluca Corsini [1].	63
6.2	Assembled quadrotor platform used in experiments. Modular layout with on-board compute, flight controller, and protective shell visible.	64
7.1	Comparison of desired vs. measured state along the square trajectory: NHFC (controller) vs. POM (state estimator).	69
7.2	Time evolution of tracking errors for position and attitude components.	69
7.3	MKQuad – Comparison between controller reference (NHFC) and measured state (POM) during CIRP execution.	72
7.4	BumbleBee (Limited Power) – NHFC vs. POM state tracking during CIRP. . .	72
7.5	BumbleBee (Full Power) – NHFC vs. POM state tracking during CIRP.	73
7.6	Real linear vs. angular acceleration data during CIRP for MKQuad, BB (Limited Power), and BB (Full Power).	74
7.7	Quantitative comparison across the three configurations MKQ, BBLP, and BBFP—based on thrust capability, angular acceleration, thrust-to-weight ratio, and mean inertia.	75
7.8	Chronophotography of the powerloop maneuver. The drone starts from point A , completes the first vertical loop reaching point B , and then performs a second loop in the opposite direction.	78
7.9	3D trajectory of the power-loop maneuver: Near-Hovering Flight Controller (NHFC) desired vs. GenoM3 component for localization (POM) measured. . . .	79
7.10	Position, attitude, and angular velocity evolution during the power-loop maneuver.	80
7.11	Position, attitude, and angular velocity evolution during the flip maneuver. . . .	81
7.12	Commanded and measured rotor velocities during the flip maneuver.	81

List of Tables

2.1	Summary of physical specifications and performance metrics of surveyed UAV platforms.	19
2.2	Summary of onboard components and sensors for surveyed UAV platforms.	20
4.1	Bill of materials for the propulsion Test-bench.	37
5.1	Comparison of COM for Different Layouts	59
6.1	Bill of materials for the quadrotor platform.	64
7.1	RMSE, mean error, and percentage overshoot for position and attitude tracking during the square trajectory simulation.	70
7.2	RMSE, mean error, and percentage overshoot for position and attitude tracking during real-world square trajectory.	70

Contents

List of Figures	8
List of Tables	10
Contents	11
1 Introduction	14
1.1 Motivation	14
1.2 Contribution	15
2 State of the Art	16
2.1 The diversity of modern UAV design	16
2.2 UAV taxonomy by control capabilities	17
2.2.1 Underactuated UAVs	17
2.2.2 Fully actuated UAVs	17
2.2.3 Omnidirectional UAVs	18
2.2.4 UAVs in research	18
2.3 Benchmarking quadrotor platforms for design and performance metrics	18
2.3.1 Key design metrics	21
2.4 Advanced test benches for UAV motor characterization and propulsion analysis .	21
2.4.1 Evolution of UAV motor test benches	22
2.4.2 Dynamic and wind-tunnel testing	22
2.4.3 Active dynamometers and automated characterization	23
2.4.4 Relevance to UAV design and control	23
2.5 High-Performance and aggressive quadrotor flight	24
2.5.1 Hardware for extreme agility	24
2.5.2 Advanced control strategies	24
2.5.3 Performance benchmarks	25
3 Model and Control	26
3.1 Quadrotor UAV	26
3.2 Mathematical Model	27
3.2.1 Kinematic Model	27
3.2.2 Dynamic Model	28
3.2.3 Mixer	29
3.3 Cascade PID Control	30
3.3.1 Outer Loop: Position Control	30
3.3.2 Inner Loop: Attitude Control (Geometric Control)	30
3.3.3 Motor Velocity PID Control	31
3.3.4 Cascade Tuning Considerations	31

3.4	Differential Flatness	32
3.4.1	Flatness-Based Trajectory Generation	32
4	Test-bench	34
4.1	Test-bench Design	34
4.1.1	Design Considerations	34
4.1.2	Mechanical Structure	35
4.1.3	Operational Notes	36
4.2	Test-bench Implementation	36
4.2.1	Mechanical and Electronic Setup	37
4.2.2	Control and Measurement Workflow	38
4.2.3	Aerodynamic Parameter Identification	39
4.2.4	Support for Control Validation and PID Tuning	39
4.3	Test-bench experiments	39
4.3.1	Motor PID Tuning	39
4.3.2	Thrust and Torque Characterization	42
4.3.3	Comparison with Manufacturer Datasheet	47
5	Methodology and Design	50
5.1	Configurator	50
5.1.1	Design Requirements and Initial Analysis	50
5.1.2	Component Selection Tool	50
5.1.3	From Configuration to Performance Estimation	51
5.1.4	Envelope Analysis	53
5.2	Quadrotor Design	57
5.2.1	Initial Design Considerations	57
5.2.2	Minimum Motor Distance Calculation	57
5.2.3	Lower Center of Mass	59
6	Implementation	61
6.1	Software Stack: <code>telekyb3</code>	61
6.1.1	Framework Design Principles	61
6.1.2	Core Components	61
6.1.3	Custom Modifications	62
6.1.4	Perception and Estimation Modules	62
6.1.5	Middleware and Execution Environment	62
6.1.6	Summary	63
6.2	Quadrotor Platform	63
6.2.1	Mechanical and Electronic Setup	63
6.2.2	Software and Control Integration	64
6.2.3	Simulation Implementation	65
6.2.4	Advanced Control Execution	66
7	Experiments	67
7.1	Overview	67
7.2	Simulation-Based Controller Tuning	68
7.2.1	Tuning Strategy	68
7.2.2	Validation Using a Square Trajectory	68
7.3	Real-World Validation	70
7.3.1	Performance Metrics	70
7.4	Comparative Evaluation	71

7.4.1	CIRP Test: Constant Increasing Ramp Path	71
7.4.2	CIRP Analysis	73
7.5	Agile Flight Demonstrations	76
7.5.1	Trajectory Generation for Agile Maneuvers	76
7.5.2	Powerloop Maneuver	78
7.5.3	Flip Maneuver	80
8	Conclusion	82
9	Future Works	84
	Bibliography	86

Introduction

1.1 Motivation

Over the course of this thesis, one major project was carried out: the redesign of a quadrotor platform. The efforts were initiated in response to the obsolescence of the aerial platforms currently in use within the INRIA research laboratory. These systems, originally developed over 15 years ago, no longer reflect the capabilities, constraints, or modularity required for modern aerial robotics research. As the field has rapidly advanced in both hardware and software, continuing to rely on legacy platforms would undermine the accuracy, scalability, and relevance of future experimental work.

The current quadrotor platform used in the laboratory environment suffers from several structural and functional deficiencies that limit its suitability for modern robotics research, particularly in multi-agent and agile flight experiments.

- **Excessive size for constrained environments:** The platform's large dimensions are not well-matched to the indoor testing arena, which is bounded by the field of view of the motion capture system. This spatial constraint makes it difficult to fly multiple drones simultaneously or to execute complex maneuvers without risking collisions.
- **Poor maneuverability:** The wide footprint contributes to high inertia and reduced responsiveness, preventing the drone from executing agile or aggressive flight trajectories. This compromises its use in research scenarios where high-performance tracking or rapid stabilization is critical.
- **Limited actuation authority:** The propulsion system lacks the necessary control authority in terms of force and torque generation. Consequently, the drone is unable to effectively counteract external disturbances such as abrupt maneuvers or environmental forces, making it unsuitable for operation in unstructured or outdoor settings.
- **Restricted payload integration:** The current design offers limited capacity to integrate additional hardware, such as advanced sensors or experimental modules. Any increase in payload significantly degrades flight performance and stability.

These limitations have motivated a comprehensive mechatronic redesign of the platform, aiming to enhance agility, reduce physical footprint, and improve thrust and torque capabilities. The goal is to develop a new aerial system optimized for high-performance research in both single-agent and multi-agent scenarios, adaptable to confined indoor spaces while maintaining robustness for outdoor deployments.

1.2 Contribution

The primary objective of this thesis is to modernize and optimize the aerial robotic platforms currently used within the research laboratory through a set of targeted technical contributions:

- **Hardware-in-the-loop test bench:** A dedicated experimental setup was developed to enable precise characterization of motors and identification of aerodynamic parameters. By directly measuring the thrust and torque coefficients (C_f and C_τ), this test bench improves simulation fidelity and supports the construction of accurate actuator allocation matrices essential for reliable flight control.
- **Custom UAV configurator:** A comprehensive software tool was designed to predict and validate the performance of future UAV designs. By integrating empirical data obtained from hardware tests, the configurator generates high-fidelity models of UAV flight dynamics and control responses, thereby shortening development cycles and enhancing platform robustness.
- **Novel quadrotor platform:** A compact quadrotor UAV was designed and validated, tailored specifically for experimental research in agile control strategies. The new design achieves significant gains in TWR, maneuverability, and modularity, making it particularly suited for dynamic flight experiments in constrained environments.
- **Software framework enhancement:** Practical modifications were introduced into the open-source software framework `telekyb3`¹. In particular, the position and attitude controller was extended to allow for asymmetric tuning of gains along the X and Y axes, improving control precision and flexibility for trajectory tracking and stabilization tasks.

Together, these contributions establish a robust experimental foundation that bridges theoretical control developments with practical validation, enabling future advancements in autonomous flight, high-performance control systems, and multi-sensor robotic integration.

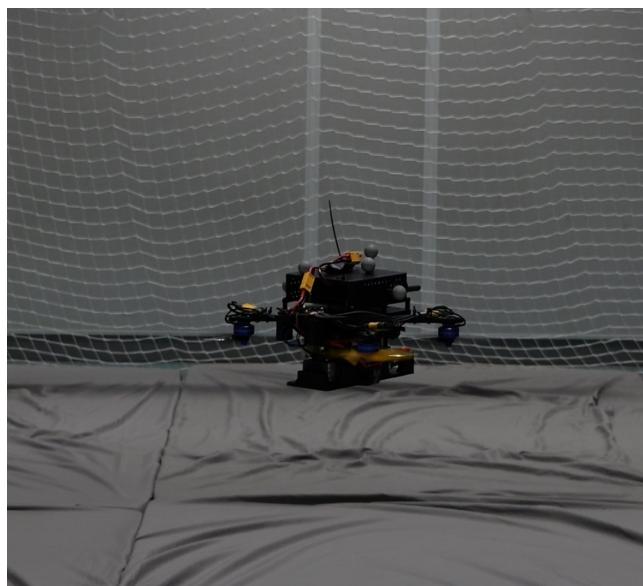


Figure 1.1: Novel quadrotor platform developed in this work

¹<https://git.openrobots.org/projects/telekyb3>

State of the Art

2.1 The diversity of modern UAV design

Unmanned Aerial Vehicles (UAVs), particularly multirotor platforms such as quadrotors, have seen a rapid evolution in their design, driven by advances in propulsion technology, lightweight structures, embedded computing, and sensor integration. These developments have enabled an increasing range of missions and flight profiles, leading to the emergence of a diverse set of UAV configurations that vary significantly in size, layout, actuation, and control complexity.

While the classical quadrotor remains the most common UAV archetype due to its mechanical simplicity and stable hover capabilities, numerous variants have emerged to meet more demanding requirements. These include hexarotors with tilted propellers for directional force generation, coaxial designs for compactness, and variable-tilt architectures for dynamic reconfiguration during flight. Such configurations are not merely structural adaptations; they enable fundamentally different dynamic behaviors, extending beyond the capabilities of traditional quadrotors.

The design space of UAVs is characterized by several critical trade-offs. Increasing thrust-to-weight ratio (TWR) improves agility and payload capacity, but often at the expense of endurance or control robustness. Enlarging the frame allows for better separation of rotors, improving torque authority, but results in higher inertia and a larger operational footprint. Similarly, platforms with extensive sensing payloads—such as LiDAR units, stereo cameras, or depth sensors—offer enhanced autonomy but require additional structural support and power, influencing flight performance and control authority.

This variety is further compounded by differences in the number and orientation of actuators. Traditional quadrotors are underactuated systems: they generate thrust only along a fixed axis (typically aligned with the body z -axis) and must tilt their entire body to produce lateral motion. In contrast, some platforms incorporate tilted or independently actuated rotors that allow generation of forces and moments along more directions, enabling decoupled control of orientation and translation. As these capabilities increase, so do the possibilities for executing aggressive or constrained maneuvers, such as precise path tracking in cluttered environments or maintaining a fixed pose during interaction tasks.

Given this high degree of architectural and functional variability, it becomes essential to adopt a formal taxonomy that organizes UAV designs according to their control capabilities. Classifying UAVs solely by geometric layout or application domain is insufficient for reasoning about their dynamic behavior or controllability. Instead, a categorization based on the number of independent control inputs relative to the six degrees of freedom in $SE(3)$ provides a principled framework for comparing designs.

This motivates the structure of the next section (Section 2.2), where we introduce a taxonomy grounded in actuation completeness. This classification distinguishes between underactuated platforms, those unable to control all six degrees of freedom simultaneously, and fully

actuated or omnidirectional UAVs, which achieve greater control authority through additional or reoriented actuators. This control-centric perspective is instrumental for understanding the limitations of conventional platforms and guiding the development of more agile and capable UAVs, such as the one proposed in this thesis.

2.2 UAV taxonomy by control capabilities

The classification of multirotor UAVs is fundamentally based on their actuation scheme and the number of independent control inputs available. This directly determines how many Degree of Freedom (DoF) in the six-dimensional configuration space $SE(3)$ can be actively governed. The main categories include Under Actuated (UA), Fully Actuated (FA), and Omnidirectional (OD) systems. The distinction lies in the system's ability to generate control inputs that span a subset or the entirety of the wrench space, the space of force and moment vectors that can be produced independently along the six axes of motion.

2.2.1 Underactuated UAVs

Conventional multirotors, such as standard quadrotors and coplanar hexarotors, are typically Under Actuated. A rigid body in three-dimensional space has six DoFs, three translational and three rotational, but an UA platform has fewer independent control inputs than DoF. For example, a classical quadrotor provides four control inputs (propeller thrusts) to govern six DoFs. Since all propellers are aligned along parallel axes, usually vertical, the vehicle cannot generate lateral forces without reorienting its body. In practice, lateral translation is achieved by rolling or pitching, thus coupling translational and rotational motion [2, 3].

The UA configuration is mechanically simple, lightweight, and energy-efficient because all thrust contributes to lift in hover. Despite their inherent coupling, cascaded Proportional Integral Derivative (PID) controllers and geometric control strategies provide stable flight and trajectory tracking near hover [4]. These platforms are dominant in commercial and research applications where endurance, simplicity, and moderate agility are sufficient.

2.2.2 Fully actuated UAVs

Fully Actuated UAVs are designed to produce a full-rank allocation matrix, enabling independent control of all six DoF [5, 6]. Achieving full actuation requires either adding actuators or modifying rotor orientations to allow thrust vectoring. Two primary approaches exist:

1. Fixed-tilt designs introduce non-coplanar propellers at constant angles (e.g., canted hexarotors), allowing lateral force generation without tilting the main body. This provides decoupled force and moment control but sacrifices some efficiency because a portion of thrust is not aligned with gravity in hover [7].
2. Variable-tilt designs use actively actuated mechanisms, such as servo-driven tilt rotors, to reorient propeller axes during flight. This dynamic thrust vectoring allows switching between efficient hover and FA configurations for tasks like aerial manipulation or aggressive maneuvers [8].

FA multirotors can translate and rotate independently, enabling maneuvers impossible for UA vehicles, such as pure lateral flight while maintaining a fixed orientation. However, these benefits come with increased mass, mechanical complexity, and reduced energy efficiency.

2.2.3 Omnidirectional UAVs

Omnidirectional UAVs are a subset of FA platforms that achieve holonomic flight, i.e., the ability to produce forces in any spatial direction regardless of orientation [5, 9]. Omnidirectionality typically requires either a sufficient number of propellers with diverse orientations or actively reconfigurable rotors to ensure the attainable wrench space encompasses the origin in all directions.

Research prototypes such as Voliro [9] and the ODQuad [10] demonstrate this capability using six or more tilted or gimbaled rotors, allowing operations like wall-parallel flight, precise surface inspection, and agile path following in constrained environments. OD designs are particularly advantageous for aerial manipulation and interaction tasks but remain complex and less energy-efficient compared to conventional configurations.

2.2.4 UAVs in research

In summary, UA UAVs dominate commercial and research applications due to their efficiency and simplicity, FA systems provide decoupled 6-DoF control for high-performance and manipulation tasks, and OD UAVs extend these capabilities to truly holonomic motion at the cost of higher complexity. This taxonomy is crucial for understanding the design trade-offs in multirotor platforms and for positioning the proposed quadrotor within the current state of the art.

2.3 Benchmarking quadrotor platforms for design and performance metrics

To guide and validate the design of the proposed quadrotor, we conducted a detailed benchmarking of state-of-the-art UAV platforms developed in both academic research and high-performance applications. This comparative study identifies prevailing mechatronic architectures, performance trade-offs, and component selections that define the current frontier of agile and efficient multirotor design. By analyzing physical specifications, propulsion systems, on-board computing, and sensor configurations, we establish reference performance envelopes that serve as the foundation for the component choices in this work [11, 12, 13, 14, 15].

The survey focuses on fundamental design metrics, total mass, wheelbase, TWR, top speed, linear acceleration, and payload potential, alongside the onboard electronics suite, which includes flight controllers, embedded computers, sensors, and communication hardware. These parameters are central to UAV mission capability, agility, and autonomy level. The platforms span a wide range, from compact research quadrotors for agile maneuvers to heavier configurations designed for payload or advanced perception.

Tables 2.1 and 2.2 summarize the results of this survey. While some specifications are not fully disclosed in the literature, the data provides valuable insights into the design space of research UAVs and the performance ranges typically achieved in cutting-edge platforms. This benchmarking process directly informs the selection of motors, propellers, batteries, and structural dimensions for the quadrotor developed in this thesis.

Name	Weight	Wheelbase	Total Thrust	TWR	Speed	Acceleration
SUPER [16]	1500g	280 mm	94.4 N	ideal 6.3 real 5	> 20m/s	N/A
Agilicious [11]	775g	250 mm/330mm	38 N	5	36m/s	N/A
OmniNxt [17]	660g	151 mm	N/A	N/A	N/A	N/A
S-drone [18]	2500g	450mm	N/A	N/A	N/A	N/A
FLA platform [12]	3000g	450mm	62.8 N	N/A	N/A	N/A
ASL flight [19]	2400g	650mm	N/A	N/A	N/A	N/A
MRS Platform [13]	4800g	500mm	N/A	N/A	N/A	N/A
Unnamed [20]	N/A	N/A	N/A	N/A	N/A	N/A
DIY 7-Inch FPV [21]	800g	300mm	98.1 N	5	100 km/h	N/A
Borinot [14]	990g	370mm	96.6N	3.5 (4.7 without limb)	12m/s	N/A
Agile UAV [15]	600g	280mm	15 N	5	5m/s	8 m/s ²

Table 2.1: Summary of physical specifications and performance metrics of surveyed UAV platforms.

Reference	Motors	FC	PC	Sensor/Camera	Propellers	ESC	Battery
[16]	T-Motor F90	PX4	intel NUC 12	LIDAR Livox Mid 360	N/A	N/A	N/A
[11]	T-Motor Velox V2306 V2.0	TMotor F7 Flight Controller	Nvidia Jetson TX2	Intel realSense T265 tracking camera	SFP 5148	F55A Pro II 3-6S 4-in-1 ESC	Tattu R-Line 4s 1800mAh 120C
[17]	N/A	Nxt-FC	Nvidia Jetson Orin	Multi-fisheye camera set	N/A	N/A	N/A
[18]	MN-series T-Motor	Pixhawk4 flight controller	UP core 01/16	Downward-facing cameras; Downward-facing Benewake TFmini Lidar	N/A	N/A	5000mAh 4S battery
[12]	DJI E600 3508 415 kV	N/A	Intel NUC i7	Hokuyo 2D LIDAR; Downward-facing Garmin rangefinder	DJI E600	DJI E600	N/A
[19]	N/A	N/A	Intel NUC i7	Intel RealSense ZR300 depth imaging sensor	13-in 4.5pitch	N/A	N/A
[13]	N/A	N/A	Intel NUC i7	Ouster OS1 Lidar; Intel RealSense D435i depth camera	N/A	N/A	N/A
[20]	N/A	N/A	Intel NUC i7	Intel RealSense T265 and D435 cameras	N/A	N/A	N/A
[21]	2806.5 Motors	F405 Flight Controller	N/A	N/A	7-inch	45A 4-in-1 ESC	4S 5200mAh
[14]	T-Motor F90-1300KV	PX4	Intel NUC7-i7-DNKE	Optical flow, IMU, downward-facing camera	7-inch	TMOTOR F35A 6S	6S 3.0Ah 60C
[15]	N/A	TMotor F7	Raspberry Pi 5	Motion capture markers (not used for control), IMU	5-inch	N/A	4S 1800mAh 100C

Table 2.2: Summary of onboard components and sensors for surveyed UAV platforms.

A closer inspection of the surveyed platforms reveals several common characteristics and trade-offs among quadrotors currently used in research. Most platforms are relatively compact, with wheelbases generally below 400 mm, confirming their possible usage in agile flight scenarios or in constrained indoor environments. However, in many cases the total thrust or TWR is either not reported or estimated indirectly from available motor-propeller combinations and manufacturer datasheets, rather than from experimental thrust stand measurements. This highlights a gap in the literature regarding standardized performance reporting.

For instance, platforms like the FLA [12] and MRS [13] are structurally larger and more suited for perception applications, but their thrust-to-weight ratios are either unspecified or inferred to be low, making them less ideal for high agility tasks. Others, such as Borinot [14], offer modularity but suffer from reduced performance once peripheral payloads are added, lowering the effective TWR. The Agilicious platform [11], while well suited for dynamic maneuvers, has a limited payload margin, and its performance envelope is tightly coupled to its small airframe and electronics.

Moreover, some configurations exhibit a large physical footprint relative to their thrust capability, as seen in the ASL and S-drone platforms, which may limit their use in cluttered or GPS-denied environments. In contrast, DIY racing-style drones, such as the 7-inch FPV setup [21], show extremely high thrust margins and speeds exceeding 100 km/h, but their onboard autonomy is limited, making them less suitable for reproducible experimentation or perception-driven control.

Overall, while each system has been tailored for specific experimental needs, none of the surveyed designs simultaneously satisfies compactness, high TWR, integrated sensing, and flexibility for advanced control development. These insights motivated the development of a new platform presented in this thesis, designed to maximize thrust authority and responsiveness while maintaining a compact and modular structure compatible with onboard and offboard sensing pipelines.

2.3.1 Key design metrics

The following metrics capture the fundamental trade-offs in quadrotor design and serve as a basis for comparing platforms and guiding component selection [4, 2, 11]:

- **Thrust-to-Weight Ratio (TWR)** – Higher TWR (≥ 3) ensures agility, payload margin, and vertical acceleration.
- **Payload Capacity** – Defined by thrust reserve; affects Maximum Take-off Weight (MTOW) and endurance.
- **Stability and Responsiveness** – Quantified via disturbance rejection, attitude bandwidth, and rise time.
- **Control Accuracy** – Measured as position/attitude tracking error; critical for precise tasks and formation flight.

These metrics directly influenced the selection of motors, propellers, and power systems for the platform developed in this thesis.

2.4 Advanced test benches for UAV motor characterization and propulsion analysis

The precise characterization of brushless DC motors and propeller assemblies is fundamental for the design, optimization, and control of quadrotor and multirotor UAVs. Accurate knowledge

of thrust, torque, efficiency, and dynamic response directly influences the selection of propulsion components, the tuning of control algorithms, and the prediction of flight performance. To this end, the development of advanced test benches, often called motor dynamometers or thrust stands, has become a critical step in both research and industrial UAV workflows [22, 23, 24, 25].

2.4.1 Evolution of UAV motor test benches

Early UAV motor characterization relied on simple static thrust stands, where the motor-propeller assembly was mounted vertically on a load cell to measure axial thrust. While suitable for basic verification, these setups offered limited insight into torque generation, transient behavior, and efficiency under realistic flight loads. Modern research has moved toward integrated, multi-axis test benches capable of simultaneously capturing thrust, torque, rotational speed, electrical input, and in some cases, aerodynamic effects [23, 26].

Static test benches remain essential for baseline characterization. They typically employ:

- **Load Cells for Thrust and Torque:** High-precision strain-gauge load cells measure forces along the motor axis (for thrust) and perpendicular to the shaft (for torque). [25].
- **Rotational Speed Measurement:** Optical encoders, Hall-effect sensors, or high-rate tachometers capture motor Revolutions Per Minute (RPM), essential for deriving propeller efficiency and dynamic response curves.
- **Electrical Monitoring:** Voltage and current are monitored with high-speed shunt or Hall sensors, or precision power analyzers, enabling accurate computation of input power and efficiency maps.
- **Thermal Sensors:** Infrared sensors or thermocouples are often integrated to track motor and ESC heating, which affects performance under sustained high load.

Static thrust stands are widely used for generating TWR curves, evaluating hover efficiency, and validating simulation models. However, they cannot fully replicate the aerodynamic conditions encountered in high-speed or maneuvering flight. This limitation has driven the development of dynamic and wind-tunnel test benches.

2.4.2 Dynamic and wind-tunnel testing

Dynamic test benches extend capability by replicating in-flight conditions, including forward airflow and transient load changes. A three-axis load-cell system embedded in a 40 cm wind tunnel has been used to measure thrust and torque under varying airspeeds [24]. These measurements enable the derivation of aerodynamic coefficients and the validation of Blade Element Momentum Theory (BEMT) models. Similarly, a dedicated motor-propeller characterization bench operating under airflow up to 15 m/s demonstrated how propeller efficiency deteriorates significantly in crossflow regimes, an essential consideration for high-speed multirotor flight [26].

Wind-tunnel integration is essential for aggressive UAVs, where rapid attitude changes and translational motion produce significant aerodynamic effects such as blade flapping, inflow distortion, and dynamic thrust variations. Dynamic test benches allow researchers to identify performance deviations from static testing and incorporate these effects into accurate simulation and controller tuning.

2.4.3 Active dynamometers and automated characterization

The most advanced UAV motor test benches employ active dynamometers, such as eddy-current or hysteresis brakes, to load the motor shaft across its full speed–torque envelope [23]. These systems enable:

- **Efficiency Mapping:** Generating motor and propeller performance maps over a grid of RPM and torque points.
- **Transient Response Analysis:** Capturing acceleration and deceleration characteristics for controller feedforward tuning.
- **ESC and Powertrain Evaluation:** Testing the complete propulsion chain under realistic conditions, including voltage drop, ESC saturation, and regenerative braking (in bidirectional setups).

Automated data acquisition is a defining feature of state-of-the-art test benches. High-resolution Data acquisition (DAQ) systems, often using National Instruments hardware or 24-bit microcontroller Analog-to-Digital Converter (ADC), synchronize thrust, torque, RPM, current, voltage, and thermal measurements at high sampling rates. This allows for both static sweeps and dynamic profiles to be executed automatically, generating large datasets for machine learning, system identification, or optimization workflows [23].

2.4.4 Relevance to UAV design and control

The integration of test bench data into UAV development is multifold:

- **Component Selection:** Accurate thrust and efficiency data inform motor and propeller choice to achieve desired TWR and endurance.
- **Model Validation:** Empirical measurements validate dynamic simulation models, bridging the gap between datasheet performance and real-world behavior.
- **Controller Tuning:** Bench-derived motor dynamics (e.g., step response, throttle-to-thrust mapping) support precise feedforward and PID/Model predictive control (MPC) tuning.
- **Mission Optimization:** High-fidelity performance maps enable predictive energy management for tasks requiring high agility or extended endurance.

Research has repeatedly demonstrated that manufacturer-provided specifications can deviate significantly from real-world measurements [22, 25]. In-house characterization ensures that design choices are grounded in realistic performance metrics. For the quadrotor developed in this thesis, test bench data provided the foundation for selecting the optimal propulsion system, determining safe operating envelopes, and enabling aggressive maneuvers with confidence.

The state of the art in UAV motor test benches shows a clear evolution from simple static thrust stands to sophisticated, multi-axis, and actively-loaded dynamometers. Modern systems combine high-speed sensing, automated data acquisition, and dynamic testing capabilities to capture the complete propulsion performance envelope. These tools are indispensable for UAV research, allowing the precise design, validation, and optimization of platforms ranging from stable aerial photography drones to highly agile racing and research quadrotors.

2.5 High-Performance and aggressive quadrotor flight

Aggressive maneuvers represent the extreme end of quadrotor flight, where the vehicle operates close to its dynamic limits with high linear accelerations, large attitude excursions, and rapid rotational rates. Such flight includes aerobatic flips, powerloops, high-speed dives, tight slaloms, and rapid gap traversal. Executing these maneuvers requires a combination of specialized hardware capable of delivering exceptional thrust and responsiveness, and advanced control algorithms able to maintain precise stability under nonlinear and fast-changing conditions [4, 11, 15].

2.5.1 Hardware for extreme agility

Quadrotors performing aggressive maneuvers are characterized by:

- **High Thrust-to-Weight Ratio (TWR):** Platforms for agile flight typically exhibit TWR values exceeding 3:1, providing sufficient margin to accelerate vertically at multiple g's and recover quickly from dives [11, 15]. This is enabled by high-power brushless motors, lightweight propellers, and high-discharge LiPo batteries or hybrid propulsion.
- **Lightweight and Rigid Frames:** Carbon-fiber structures minimize inertia while maintaining rigidity to withstand high loads and impacts. Reduced inertia enhances angular acceleration and shortens maneuver response times.
- **Optimized Aerodynamics and Layout:** Minimizing drag through compact layouts and symmetric geometries improves performance during forward or inverted flight. Some research platforms use bidirectional motors for reversible thrust, enabling sustained inverted maneuvers [11].
- **High-Speed Sensing and Computing:** Aggressive flight demands low-latency inertial measurement unit (IMU)s (500–1000 Hz), fast electronic speed controllers (ESCs), and control loops capable of several hundred hertz to maintain stability at extreme rotation rates [4].
- **Power System Considerations:** High-C batteries or reliable hybrid systems are required to support brief but intense current draws. Endurance is often sacrificed for raw performance, with flight times of only 3–5 minutes in extreme conditions.

2.5.2 Advanced control strategies

Hardware alone is insufficient to achieve aggressive maneuvers. Classical PID controllers designed for hover are ineffective when the vehicle performs flips or operates far from its nominal equilibrium. Modern control solutions include:

- **Geometric Nonlinear Control:** Controllers formulated on $SE(3)$ or $SO(3)$ provide globally valid attitude control and remain effective under large rotations, enabling flips and perching maneuvers [4].
- **Differentially Flat Trajectory Planning:** Exploiting quadrotor differential flatness allows minimum-snap or time-optimal trajectory generation that respects physical limits for aggressive motions [4].

- **Model Predictive Control (MPC):** Nonlinear Model predictive control (NMPC) has been applied for high-speed flight, handling actuator limits and aerodynamic effects in real time [15, 27].
- **Adaptive and Incremental Control:** Techniques such as Incremental Nonlinear Dynamic Inversion (INDI) and adaptive gain tuning enhance performance under rapid dynamics and varying conditions, including voltage drop or aerodynamic drag.

High-speed flight also requires accurate state estimation at high update rates. Laboratory demonstrations often employ motion capture for sub-millimeter accuracy, while autonomous outdoor platforms rely on high-rate IMU fusion with visual-inertial odometry or event cameras [11].

2.5.3 Performance benchmarks

State-of-the-art quadrotors have demonstrated remarkable feats in both research and racing environments:

- **Speed and Acceleration:** Agile quadrotors routinely exceed 140 km/h, with some research prototypes achieving 0–100 km/h in under 2 s [11, 15].
- **Aerobatic Maneuvers:** Platforms can execute flips, rolls, and powerloops autonomously with precise trajectory tracking, as demonstrated in ETH Zurich’s Flying Machine Arena [4].
- **Trajectory Tracking Accuracy:** Research drones achieve centimeter-level tracking even at 8 m/s flight, maintaining stable control during high-g turns [11].
- **Aggressive Perching and Gap Flight:** Specialized controllers allow drones to settle on vertical surfaces or pass through narrow openings using rapid attitude changes and precise braking.

These demonstrations highlight the integration of cutting-edge propulsion, lightweight structures, and nonlinear control methods, enabling quadrotors to achieve levels of agility comparable to aerobatic fixed-wing aircraft. Aggressive flight research not only advances aerial robotics capabilities but also provides insights for faster, more robust UAVs for inspection, racing, and autonomous navigation in complex environments.

Model and Control

3.1 Quadrotor UAV

A quadrotor is one of the simplest and most widely adopted multirotor aerial platforms due to its mechanical simplicity and high maneuverability. It consists of four independent rotors mounted on a rigid cross airframe (Fig. 3.1). Each rotor generates lift along the local body Z -axis, allowing the vehicle to hover, translate, and rotate in three-dimensional space [3, 28].

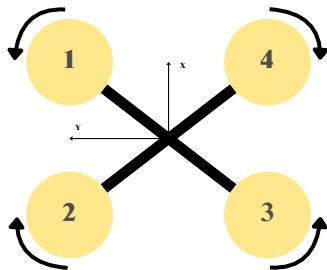


Figure 3.1: Quadrotor in X configuration.

The platform can be arranged in either a *plus* (“+”) or *X* configuration. In this work, the *X* configuration is adopted because it is widely preferred for agile flight maneuvers. In this setup, two rotors rotate clockwise (CW) and the other two counterclockwise (CCW), arranged so that diagonally opposite rotors spin in the same direction. This configuration enables yaw control by exploiting the net reaction torques of the rotors.

A quadrotor is a six-degrees-of-freedom (6-DOF) system: three translational coordinates (x, y, z) and three rotational angles (ϕ, θ, ψ), representing roll, pitch, and yaw respectively. However, it has only four independent control inputs (the four rotor speeds), which makes it an UA system [3]. This underactuation introduces several implications for control:

- **Coupled dynamics:** Translational motion is inherently coupled to attitude. To generate horizontal accelerations, the quadrotor must tilt in the desired direction of motion, which simultaneously modifies its orientation.
- **Stabilization challenges:** Unlike FA aerial platforms, a standard quadrotor cannot independently generate forces in all directions. As a result, aggressive maneuvers or external disturbances require precise and continuous feedback control to simultaneously regulate both position and orientation.
- **Reduced maneuverability:** Rapid trajectory changes or lateral accelerations are limited by the maximum tilting angles and the available TWR. The vehicle cannot instantaneously move in any direction without first reorienting itself, which imposes a natural limit on achievable accelerations and turning rates.

- **Path planning implications:** Feasible trajectories must satisfy these dynamic and actuation constraints. Planning algorithms must consider that the quadrotor's thrust is always aligned with its body z -axis, ensuring that the required accelerations are achievable given the vehicle's orientation and thrust limits.

The basic control principles are as follows:

- **Vertical motion (Altitude):** Adjusting the collective thrust of all four rotors.
- **Roll (ϕ):** Increasing thrust on one lateral rotor pair while decreasing the opposite pair.
- **Pitch (θ):** Increasing thrust on the front or rear rotor pair while decreasing the opposite pair.
- **Yaw (ψ):** Varying the speed difference between CW and CCW rotors to generate a net torque.

3.2 Mathematical Model

To describe the dynamics of the quadrotor, we adopt the Newton–Euler formalism [3]. The following assumptions are made for modeling:

1. The airframe is rigid, and the center of gravity coincides with the origin of the body frame.
2. Rotor thrust and drag are proportional to the square of the angular velocity: $f_i = C_f \omega_i^2$, $\tau_i = C_\tau \omega_i^2$.
3. Aerodynamic effects such as blade flapping and induced inflow are neglected.
4. The inertia matrix is diagonal due to geometric symmetry.

3.2.1 Kinematic Model

Two reference frames are defined for the quadrotor motion:

- **World frame** $\mathcal{F}_W = \{x, y, z\}$ following the East-North-Up (ENU) convention. This frame is inertial and used to express position and translational dynamics.
- **Body-fixed frame** $\mathcal{F}_B = \{x_b, y_b, z_b\}$, centered at the quadrotor's center of mass and following the Forward-Left-Up (FLU) convention. This frame is used to express velocities, torques, and angular dynamics.

The quadrotor state is defined as:

$$\mathbf{p} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}_{\mathcal{W}}, \quad \boldsymbol{\eta} = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}_{\mathcal{B}} \quad (3.1)$$

where \mathbf{p} is the position in the world frame and $\boldsymbol{\eta}$ are the Euler angles (roll–pitch–yaw) describing the orientation of the body frame \mathcal{F}_B with respect to \mathcal{F}_W .

Rotation Matrix The rotation matrix from the body frame to the world frame, using ZYX Euler angles, is:

$$\mathbf{R}_{\mathcal{B} \rightarrow \mathcal{W}} = \begin{bmatrix} c_\theta c_\psi & -c_\theta s_\psi & s_\theta \\ c_\phi s_\psi + s_\phi s_\theta c_\psi & c_\phi c_\psi - s_\phi s_\theta s_\psi & -s_\phi c_\theta \\ s_\phi s_\psi - c_\phi s_\theta c_\psi & s_\phi c_\psi + c_\phi s_\theta s_\psi & c_\phi c_\theta \end{bmatrix} \quad (3.2)$$

where $c_\star = \cos(\star)$ and $s_\star = \sin(\star)$.

Angular Velocity in Body Frame The body angular velocity $\boldsymbol{\omega}_B = [\omega_x, \omega_y, \omega_z]^T$ relates to the Euler angle rates as:

$$\boldsymbol{\omega}_B = \begin{bmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \phi & \cos \theta \sin \phi \\ 0 & -\sin \phi & \cos \theta \cos \phi \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (3.3)$$

3.2.2 Dynamic Model

Let m be the mass of the quadrotor and \mathbf{J} the diagonal inertia matrix expressed in the body frame \mathcal{F}_B . Let ω_i be the angular velocity of the i -th rotor.

Translational Dynamics (in World Frame) The translational dynamics are expressed in the world frame \mathcal{F}_W :

$$m\ddot{\mathbf{p}} = \mathbf{R}_{\mathcal{B} \rightarrow \mathcal{W}} \begin{bmatrix} 0 \\ 0 \\ T \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \quad (3.4)$$

where $\mathbf{p} = [x, y, z]_{\mathcal{W}}^T$ and

$$T = \sum_{i=1}^4 C_f \omega_i^2 \quad (3.5)$$

is the collective thrust along \mathbf{z}_b , rotated into the world frame by $\mathbf{R}_{\mathcal{B} \rightarrow \mathcal{W}}$.

Rotational Dynamics (in Body Frame) The rotational dynamics are expressed in the body frame \mathcal{F}_B as:

$$\mathbf{J} \dot{\boldsymbol{\omega}}_B + \boldsymbol{\omega}_B \times (\mathbf{J} \boldsymbol{\omega}_B) = \boldsymbol{\tau}_B \quad (3.6)$$

where

$$\boldsymbol{\omega}_B = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}, \quad \boldsymbol{\tau}_B = \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix}, \quad \mathbf{J} = \begin{bmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix} \quad (3.7)$$

with:

- $\boldsymbol{\omega}_B$: angular velocity in the body frame,
- $\boldsymbol{\tau}_B$: control torque vector expressed in the body frame.
- \mathbf{J} : body-frame inertia matrix (assumed diagonal due to symmetry),

3.2.3 Mixer

In multirotor systems, the thrust and torque generated by each rotor must be mapped into a global control input vector consisting of total thrust and body torques. This mapping is governed by the control allocation matrix, which relates the individual rotor forces to the collective control commands.

Let the control input vector be:

$$\mathbf{u} = \begin{bmatrix} T \\ \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} \in \mathbb{R}^4 \quad (3.8)$$

where:

- T is the total thrust along the body z -axis,
- $\tau_\phi, \tau_\theta, \tau_\psi$ are the torques around the roll, pitch, and yaw axes,
- All quantities are expressed in the body-fixed frame \mathcal{F}_B .

Let $\boldsymbol{\Omega}^2 = [\omega_1^2, \omega_2^2, \omega_3^2, \omega_4^2]^T$ be the squared motor speeds. Then, the mapping from squared motor speeds to control inputs is defined as:

$$\mathbf{u} = \mathbf{A} \boldsymbol{\Omega}^2 \quad (3.9)$$

where $\mathbf{A} \in \mathbb{R}^{4 \times 4}$ is the allocation matrix, whose rows represent the contributions of each motor to the thrust and torques.

For a quadrotor in X configuration, with first motors placed at an angle α with respect to the body x_b -axis, and assuming:

- All rotors generate thrust proportional to $C_f \omega_i^2$,
- Each rotor is placed at distance ℓ from the center of mass,
- The drag torque is proportional to $C_\tau \omega_i^2$,

the allocation matrix becomes:

$$\mathbf{A} = \begin{bmatrix} C_f & C_f & C_f & C_f \\ \ell C_f \sin \alpha & \ell C_f \sin \alpha & -\ell C_f \sin \alpha & -\ell C_f \sin \alpha \\ -\ell C_f \cos \alpha & \ell C_f \cos \alpha & \ell C_f \cos \alpha & -\ell C_f \cos \alpha \\ -C_\tau & C_\tau & -C_\tau & C_\tau \end{bmatrix} \quad (3.10)$$

This matrix structure reflects the physical contributions of each rotor to the overall motion:

- The first row sums all thrusts.
- The second and third rows compute torques about the x - and y -axes via the lever arm and rotor position.
- The fourth row captures reaction torques due to rotor drag.

Inversion and Motor Command Computation To compute the squared motor speeds needed to achieve a desired control input \mathbf{u} , the allocation matrix must be inverted:

$$\boldsymbol{\Omega}^2 = \mathbf{A}^{-1} \mathbf{u} \quad (3.11)$$

This inversion is performed at each control iteration and serves as the final step in the control pipeline, translating high-level thrust and torque commands into low-level motor references.

3.3 Cascade PID Control

Quadrotor control is often implemented using a cascade PID architecture, which separates the fast attitude dynamics from the slower translational dynamics [3]. Additionally, an inner-most motor speed control loop ensures that the computed motor speed commands are accurately tracked by the Electronic Speed Controller (ESC).

A schematic of the cascade control structure is shown in Fig. 3.2.

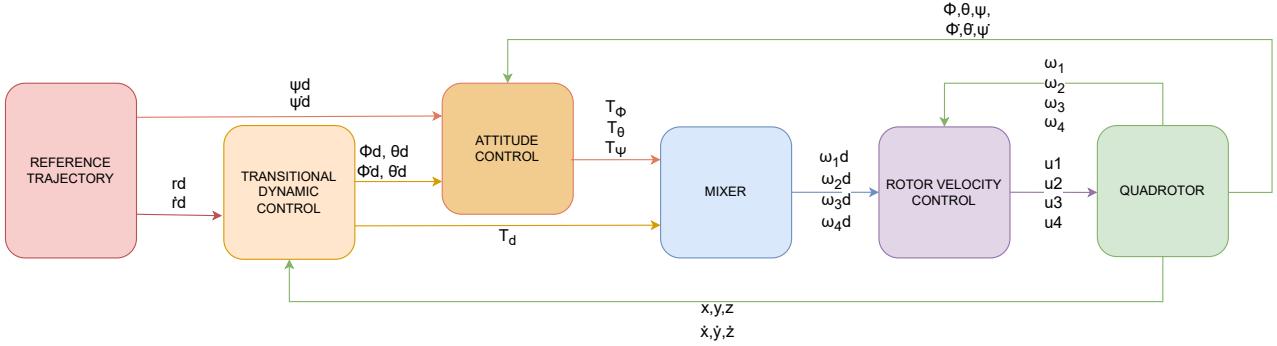


Figure 3.2: Cascade PID control architecture for quadrotor UAV. The outer loop regulates position, the middle loop stabilizes attitude, and the inner loop tracks the desired motor speeds.

3.3.1 Outer Loop: Position Control

The outer loop governs translational motion along x , y , and z using PID controllers. It outputs desired roll ϕ_d , pitch θ_d , and collective thrust T_d to achieve the commanded position:

$$\phi_d = k_{p_x}(x_r - x) + k_{d_x}(\dot{x}_r - \dot{x}) + k_{i_x} \int (x_r - x) dt \quad (3.12)$$

$$\theta_d = k_{p_y}(y_r - y) + k_{d_y}(\dot{y}_r - \dot{y}) + k_{i_y} \int (y_r - y) dt \quad (3.13)$$

$$T_d = mg + k_{p_z}(z_r - z) + k_{d_z}(\dot{z}_r - \dot{z}) + k_{i_z} \int (z_r - z) dt \quad (3.14)$$

3.3.2 Inner Loop: Attitude Control (Geometric Control)

Unlike traditional PID controllers based on Euler angles, the inner-loop attitude stabilization in this work is implemented using a geometric control law defined directly on the special orthogonal group $\text{SO}(3)$. This coordinate-free approach avoids singularities and ambiguities associated with Euler angles and quaternions, and ensures almost-global asymptotic stability of the closed-loop system [29].

Let $\mathbf{R} \in \text{SO}(3)$ denote the current orientation of the quadrotor, and $\boldsymbol{\omega}_B \in \mathbb{R}^3$ its body angular velocity. The desired orientation \mathbf{R}_d is obtained from the outer-loop position controller, as described in the flatness-based trajectory generation (Section 3.4). The desired angular velocity is denoted by $\boldsymbol{\omega}_d$.

The attitude tracking error is defined using the rotation matrix error:

$$\mathbf{e}_R = \frac{1}{2} (\mathbf{R}_d^\top \mathbf{R} - \mathbf{R}^\top \mathbf{R}_d)^\vee \quad (3.15)$$

where $(\cdot)^\vee : \text{SO}(3) \rightarrow \mathbb{R}^3$ is the inverse of the hat map. The angular velocity error is:

$$\mathbf{e}_\omega = \boldsymbol{\omega}_B - \mathbf{R}^\top \mathbf{R}_d \boldsymbol{\omega}_d \quad (3.16)$$

The control moment applied to the quadrotor is then:

$$\boldsymbol{\tau}_B = -\mathbf{k}_R \mathbf{e}_R - \mathbf{k}_\omega \mathbf{e}_\omega + \mathbf{\omega}_B \times \mathbf{J} \mathbf{\omega}_B - \mathbf{J} (\hat{\mathbf{\omega}}_B \mathbf{R}^\top \mathbf{R}_d \mathbf{\omega}_d - \mathbf{R}^\top \mathbf{R}_d \dot{\mathbf{\omega}}_d) \quad (3.17)$$

where:

- $\mathbf{k}_R, \mathbf{k}_\omega$ are control gain vectors,
- \mathbf{J} is the diagonal inertia matrix,
- $\hat{\mathbf{\omega}}_B$ is the skew-symmetric matrix of $\mathbf{\omega}_B$,
- $\dot{\mathbf{\omega}}_d$ is the desired angular acceleration from trajectory planning.

This geometric controller ensures that both orientation and angular velocity tracking errors converge exponentially to zero, provided the initial attitude error is less than 180° . It is robust to aggressive maneuvers and enables recovery from large attitude deviations (e.g., inverted flight) as demonstrated in [29].

The computed control moment $\boldsymbol{\tau}_B$, together with the collective thrust T_d , is then mapped to the squared motor speeds ω_i^2 via the inverse of the allocation matrix.

3.3.3 Motor Velocity PID Control

The innermost loop ensures that each motor reaches the desired angular velocity computed by the attitude controller. This is typically implemented in the motor ESCs using PID control:

$$\omega_{i_{\text{error}}} = \omega_{i_d} - \omega_i \quad (3.18)$$

$$u_i = k_{p_\omega} \omega_{i_{\text{error}}} + k_{i_\omega} \int \omega_{i_{\text{error}}} dt + k_{d_\omega} \frac{d}{dt} \omega_{i_{\text{error}}} \quad (3.19)$$

where u_i is the control input for the i -th motor.

This motor-level PID loop ensures accurate torque generation, compensating for motor dynamics, propeller drag, and electrical delays.

3.3.4 Cascade Tuning Considerations

- The motor velocity PID loop is the fastest and is tuned first to ensure rapid speed tracking.
- The attitude (inner) loop is tuned next, with high bandwidth and damping for fast stabilization.
- The position (outer) loop is tuned last and made slower, to avoid exciting the inner dynamics.
- Integral terms in the position and motor loops help counteract steady-state errors due to gravity and drag.

This three-layer cascade PID architecture is widely used in both research and commercial quadrotor platforms due to its robustness, modularity, and ability to handle both hover and moderate dynamic maneuvers [3].

3.4 Differential Flatness

Quadrotor dynamics are known to be differentially flat, meaning that all states and control inputs can be expressed as algebraic functions of a small set of flat outputs and their derivatives [28]. For a standard quadrotor, the flat outputs are:

$$\mathbf{y} = \begin{bmatrix} x \\ y \\ z \\ \psi \end{bmatrix}, \quad (3.20)$$

where $\mathbf{p} = [x, y, z]^\top$ is the position in the world frame \mathcal{F}_W and ψ is the yaw angle. All other states, including roll ϕ , pitch θ , body angular velocities $\boldsymbol{\omega}_B$, and motor speeds ω_i , can be reconstructed from these flat outputs and their derivatives up to fourth order.

3.4.1 Flatness-Based Trajectory Generation

Let the reference trajectory be given by \mathbf{p}_d and ψ_d , together with their derivatives up to snap $\ddot{\mathbf{p}}_d$. By leveraging differential flatness, the required thrust, orientation, and angular velocities can be computed directly from these trajectories.

Thrust Direction and Magnitude The desired translational acceleration, \mathbf{a}_d , including gravity is defined as:

$$\mathbf{a}_d = \ddot{\mathbf{p}}_d + g\mathbf{k}_z \quad (3.21)$$

where $\ddot{\mathbf{p}}_d$ is the desired position acceleration, g is the gravitational constant, and $\mathbf{k}_z = [0, 0, 1]^\top$ is the unit vector along the world Z -axis.

The required thrust vector, \mathbf{f}_d , is obtained as:

$$\mathbf{f}_d = m \mathbf{a}_d \quad (3.22)$$

where m is the UAV mass. The direction of the desired thrust is given by the unit vector:

$$\hat{\mathbf{f}}_d = \frac{\mathbf{f}_d}{\|\mathbf{f}_d\|} \quad (3.23)$$

while its magnitude, T_d , is:

$$T_d = \|\mathbf{f}_d\| \quad (3.24)$$

Desired Orientation The desired rotation matrix $\mathbf{R}_d \in SO(3)$ is computed using the thrust direction and the desired yaw heading. The projected heading in the inertial XY -plane is:

$$\mathbf{x}_c = \begin{bmatrix} \cos \psi_d \\ \sin \psi_d \\ 0 \end{bmatrix} \quad (3.25)$$

Then the desired body y -axis and x -axis are [30]:

$$\mathbf{y}_B = \frac{\mathbf{z}_B \times \mathbf{x}_c}{\|\mathbf{z}_B \times \mathbf{x}_c\|}, \quad (3.26)$$

$$\mathbf{x}_B = \mathbf{y}_B \times \mathbf{z}_B \quad (3.27)$$

$$\mathbf{R}_d = [\mathbf{x}_B \ \mathbf{y}_B \ \mathbf{z}_B] \quad (3.28)$$

This ensures the thrust aligns with \mathbf{z}_B while satisfying the yaw reference ψ_r .

Desired Angular Velocity and Acceleration Using rotation matrix kinematics:

$$\dot{\boldsymbol{\omega}}_d^x = \mathbf{R}_d^\top \dot{\mathbf{R}}_d \quad (3.29)$$

The desired angular acceleration, $\ddot{\boldsymbol{\omega}}_d^x$, is obtained by differentiating in time the desired angular velocity $\dot{\boldsymbol{\omega}}_d^x$. These expressions ultimately depend on the jerk $\ddot{\mathbf{p}}_d$ and snap $\dddot{\mathbf{p}}_d$ of the reference trajectory.

Feedforward Control Inputs Differential flatness allows direct computation of the feedforward control input vector:

$$\mathbf{u}_d = \begin{bmatrix} T_d \\ \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} m\|\ddot{\mathbf{p}}_d + g\mathbf{k}_z\| \\ \mathbf{J}\dot{\boldsymbol{\omega}}_d + \boldsymbol{\omega}_d \times (\mathbf{J}\boldsymbol{\omega}_d) \end{bmatrix} \quad (3.30)$$

This can be combined with the cascade feedback controller to enhance trajectory tracking during aggressive maneuvers.

Test-bench

4.1 Test-bench Design

The Test-bench was developed to satisfy a clear and rigorous set of requirements aimed at enabling precise measurement of thrust and torque produced by various motor-propeller configurations. Its primary role is to empirically validate the performance data found in manufacturers' datasheets, which serve as critical input to the UAV configurator tool developed in this thesis.

4.1.1 Design Considerations

The architecture of the Test-bench reflects a balance between functional versatility, measurement accuracy, and operational safety. The core design elements include:

- **Modular FC–ESC Integration:** To minimize voltage drops and signal latency, the flight controller (FC) and ESC are mounted in close proximity to the motor. The base platform accommodates both Chimera and Tawaki flight controllers, adhering to the standard 30.5 mm × 30.5 mm mounting pattern, thus supporting a wide range of off-the-shelf FCs.
- **Flexible Electronics Layout:** A dedicated electronics bay supports modular swapping of controller boards, promoting ease of configuration and maintenance. The design ensures robust cable management and spatial decoupling between power and signal paths, reducing electromagnetic interference and improving operational stability.
- **Accurate Motor–Sensor Mounting Stack:** The structural layout consists of an aluminum profile base, on top of which a six-axis force/torque sensor is rigidly installed. The motor is then mounted directly onto the sensor. This vertical stack ensures that all loads pass cleanly through the sensor without mechanical distortion or off-axis effects. The configuration also allows for inverted propeller mounting, which minimizes ground effect and ensures the aerodynamic forces measured are representative of free-flight conditions.
- **Power Protection and Safety:** A fast-blow fuse is integrated into the power line to guard against transient overcurrent conditions, particularly those induced by rapid ESC switching or fault states. A dedicated hardware kill switch enables immediate system shutdown in case of emergency, ensuring operator safety and preventing damage to test components.

4.1.2 Mechanical Structure

The Test-bench comprises two principal structural elements, each designed to ensure accurate measurement, ease of integration, and mechanical stability during testing:

- **Motor–Sensor Mounting Tower:** A vertically oriented structural tower serves as the core load-bearing element for the thrust and torque measurement system. The high-precision multi-axis force–torque sensor is rigidly mounted onto the aluminum profile base, while the brushless motor is installed directly atop the sensor. This stacking ensures that all forces and torques generated by the motor–propeller system are transmitted through the sensor with minimal structural compliance or misalignment. The vertical clearance below the propeller is sufficient to suppress aerodynamic ground effects, thereby improving measurement fidelity.

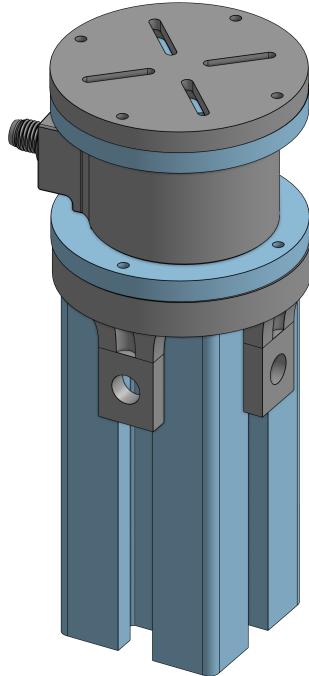


Figure 4.1: Vertical motor mount with force–torque sensor installed at the base.

- **Electronics Support Plate:** A dedicated base plate hosts the ESC and FC, both mounted on vibration-isolating standoffs to ensure thermal dissipation and mechanical decoupling. The layout is designed for modularity, supporting rapid substitution of FC models such as Chimera and Tawaki. Cable routing is carefully managed to physically separate high-current and low-voltage signal lines, reducing electromagnetic interference and ensuring robust operation under dynamic load conditions.

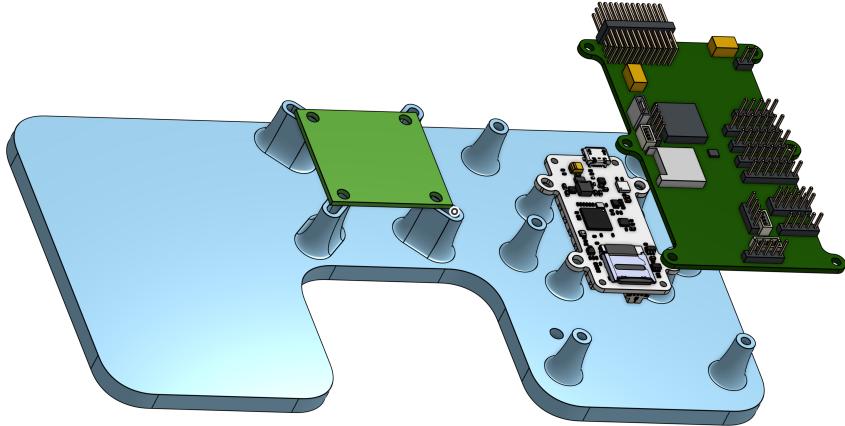


Figure 4.2: Electronics mounting plate featuring standoffs for ESC and FC installation.

4.1.3 Operational Notes

To ensure accurate and repeatable results, the motor–propeller pair is installed with the propeller facing downward. This orientation reduces the influence of ground effect on thrust measurements, which could otherwise distort aerodynamic force profiles. All electronic components are firmly secured to minimize the impact of vibration and dynamic oscillations during high-speed operation. The Test-bench setup supports consistent operation across different motor–propeller configurations, making it a versatile platform for characterization and validation tasks.

4.2 Test-bench Implementation

Accurate characterization of motor dynamics is critical in aerial robotics, as it underpins both high-fidelity simulation and robust control performance. Precise identification of thrust and torque responses enables the formulation of reliable dynamic models, which are essential for validating design choices and predicting vehicle behavior under diverse flight conditions. An experimentally validated motor model also facilitates effective PID tuning, ensuring that the propulsion system reacts predictably to control inputs and contributes to overall flight stability.

To meet these objectives, a dedicated Test-bench was developed to experimentally evaluate the integration of the flight controller (FC), electronic speed controllers (ESCs), and brushless motors of the proposed quadrotor platform. The system automates the measurement of thrust and torque, allowing the extraction of the aerodynamic coefficients C_f (thrust) and C_τ (torque). These coefficients directly inform the actuator allocation matrix \mathbf{A} , which maps motor speeds to the corresponding body forces and torques employed in the control architecture.

The Test-bench adopts a modular, plug-and-play design that enables rapid replacement of propulsion components and straightforward reconfiguration for alternative platforms or test scenarios. This modularity not only streamlines component benchmarking but also supports

future experimental campaigns involving multi-motor calibration and low-level controller tuning.

By combining hardware modularity with software-driven automation, the Test-bench ensures safe, repeatable, and high-precision propulsion testing. The resulting aerodynamic coefficients are directly integrated into both the simulation environment and the control pipeline, effectively closing the loop between hardware characterization and high-fidelity UAV control design. Moreover, the platform provides a reliable framework for iterative development, supporting both model validation and controller optimization.

4.2.1 Mechanical and Electronic Setup

The hardware setup consists of structural framing, propulsion electronics, sensing modules, and supporting infrastructure. The mechanical frame is built using standard aluminum profiles with 3D-printed interfaces to mount the propulsion unit and sensors securely. Table 4.1 summarizes the key components.

Component Category	Specification
Structural Frame	4 × aluminum profiles (0.5 m × 45 mm × 45 mm) 1 × aluminum profile (1 m × 45 mm × 45 mm)
Sensing	Botasy Medusa 6-axis force/torque sensor
Control Electronics	Chimera flight controller Skystars KO45A ESC
Propulsion Unit	AOS Supernova 2207 1980KV motor Gemfan GF51466R propeller
Support Components	Custom 3D-printed mounting interface 6S LiPo battery USB cable for data and power

Table 4.1: Bill of materials for the propulsion Test-bench.

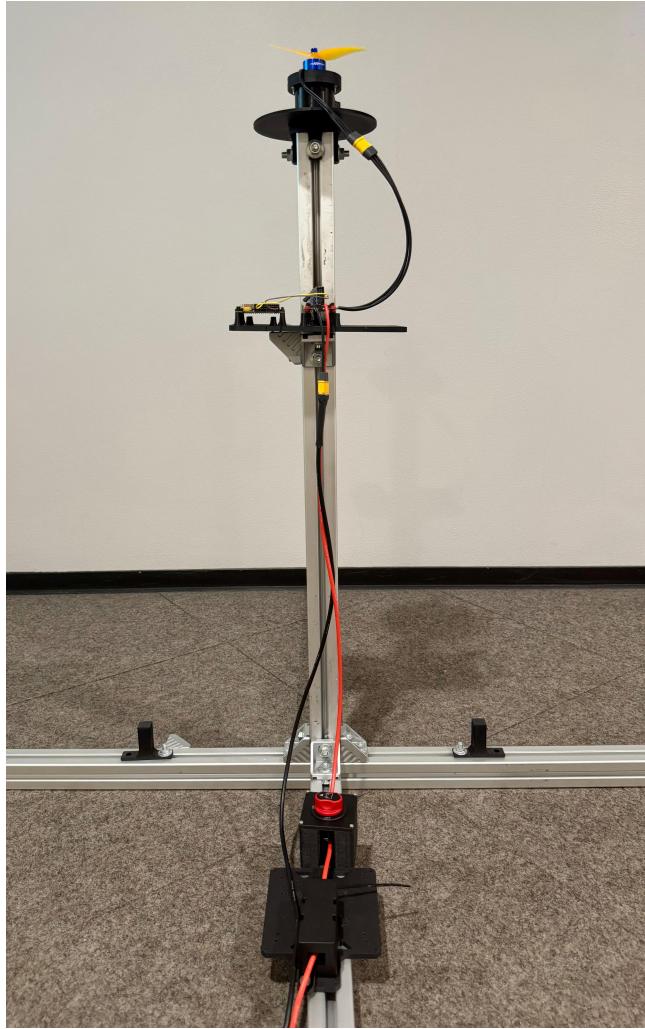


Figure 4.3: Overview of the propulsion Test-bench setup. The system integrates a thrust/torque sensor beneath the motor and logs measurements during steady-state rotation.

4.2.2 Control and Measurement Workflow

A custom Python script interfaces with the flight controller via the `rotorcraft` module and synchronizes data collection from the Botasys Medusa force-torque sensor. The routine executes a programmable sweep of motor speeds to characterize aerodynamic behavior. The workflow is divided into the following stages:

1. **Initialization:** Communication with the FC and sensor is established. PID gains and test parameters (e.g., speed range, timing) are loaded.
2. **Motor Activation:** The motor is ramped up to a commanded angular velocity ω_i , drawn from a predefined reference set $\{\omega_1, \omega_2, \dots\}$.
3. **Steady-State Logging:** After a settling time, force and torque data are collected while the motor maintains a constant speed and safety thresholds are checked. If exceeded, the test halts.
4. **Shutdown and Recalibration:** The motor is stopped, and the sensor bias is updated to compensate for drift.
5. **Logging:** Data is logged for offline processing.

4.2.3 Aerodynamic Parameter Identification

For each test run, the average values of vertical thrust force f_z and torque τ_z are computed. The following static models are used to fit the data:

$$f_i = C_f \cdot \omega^2, \quad \tau_i = C_\tau \cdot \omega^2 \quad (4.1)$$

Using least-squares regression on the collected (ω_i^2, f_i) and (ω_i^2, τ_i) pairs, the coefficients C_f and C_τ are estimated. Invalid samples (e.g., NaNs, saturation outliers) are filtered prior to curve fitting.

4.2.4 Support for Control Validation and PID Tuning

The Test-bench also functions as a diagnostic platform for evaluating PID performance and low-level controller dynamics. By configuring the system to begin data acquisition immediately after the motor command is issued, without waiting for steady-state conditions, it becomes possible to capture the full step response, including transient phenomena. This enables detailed analysis of control latency, motor response time, and overshoot, which are critical for accurate tuning and validation of feedback controllers.

As a result, the platform supports dual operating modes: an identification mode, focused on extracting steady-state aerodynamic coefficients, and a control validation mode, aimed at characterizing the system's dynamic behavior during transients.

4.3 Test-bench experiments

This Section presents the methodology used to experimentally characterize the motor-propeller system via a custom-designed hardware-in-the-loop test bench. A closed-loop PID tuning strategy was implemented to ensure accurate and stable control of the motor speed during testing, enabling reliable data acquisition across a wide range of operational conditions. This Section also describes the procedure used to experimentally determine the thrust and torque coefficients (C_f and C_τ) of selected motor-propeller combination using the custom-built test bench. Measurements were taken across a range of motor speeds, and the resulting force and torque data were used to fit a quadratic model. Currently, only the quadratic term is considered, but future work will involve more complete aerodynamic modeling by including both linear and quadratic terms. The Section concludes with a comparison between measured data and manufacturer specifications, along with hypotheses explaining any observed discrepancies.

4.3.1 Motor PID Tuning

To ensure rapid and stable velocity tracking, a closed-loop PID tuning procedure was performed using the Test-bench. The primary goal was to minimize overshoot and steady-state error while achieving fast convergence to commanded speeds.

The tuning strategy followed the incremental approach also known as manual or heuristic tuning [31]: starting from $K_{p\omega}=0$ and a low $K_{i\omega}$, the integral term was gradually increased to eliminate steady-state error. Subsequently, $K_{p\omega}$ was raised to improve responsiveness and dampen oscillations. The derivative term $K_{d\omega}$ was kept at zero. The final gains used were: $K_{p\omega}=5$, $K_{i\omega}=300$, $K_{d\omega}=0$. These values produced satisfactory performance across the range of operating conditions relevant to this study, although no fine-tuning for optimality was performed.

Step Response Evaluation

Figure 4.4 shows the system response to a step input under time-based control mode. The following performance metrics were extracted:

- **Rise Time:** approximately 63.92 ms
- **Settling Time:** approximately 100.62 ms
- **Overshoot:** approximately 5.76%
- **Steady-State Error:** below 0.45 Hz

These results confirm the controller's ability to react quickly and accurately to abrupt changes in desired velocity, with minimal transients.

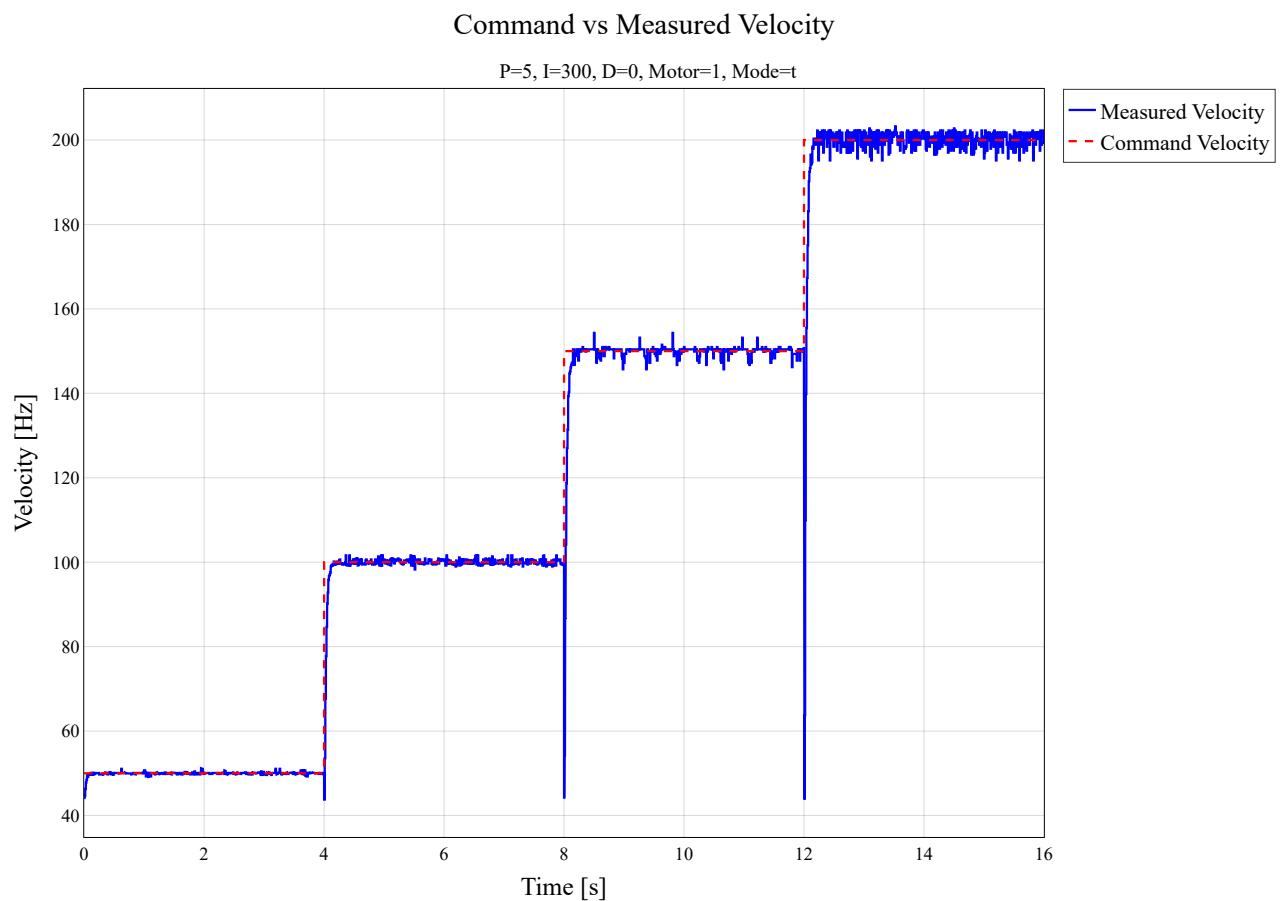


Figure 4.4: Step input test with tuned PID ($P=5, I=300, D=0$): Measured vs reference velocity.

Ramp Tracking Performance

The staircase trajectory test (Figure 4.5) evaluates the controller's ability to track a continuously increasing frequency profile. The system demonstrated strong tracking behavior, maintaining phase alignment throughout the ramp. The average tracking error over the duration of the test was below 1.086 Hz, with no noticeable time delay.

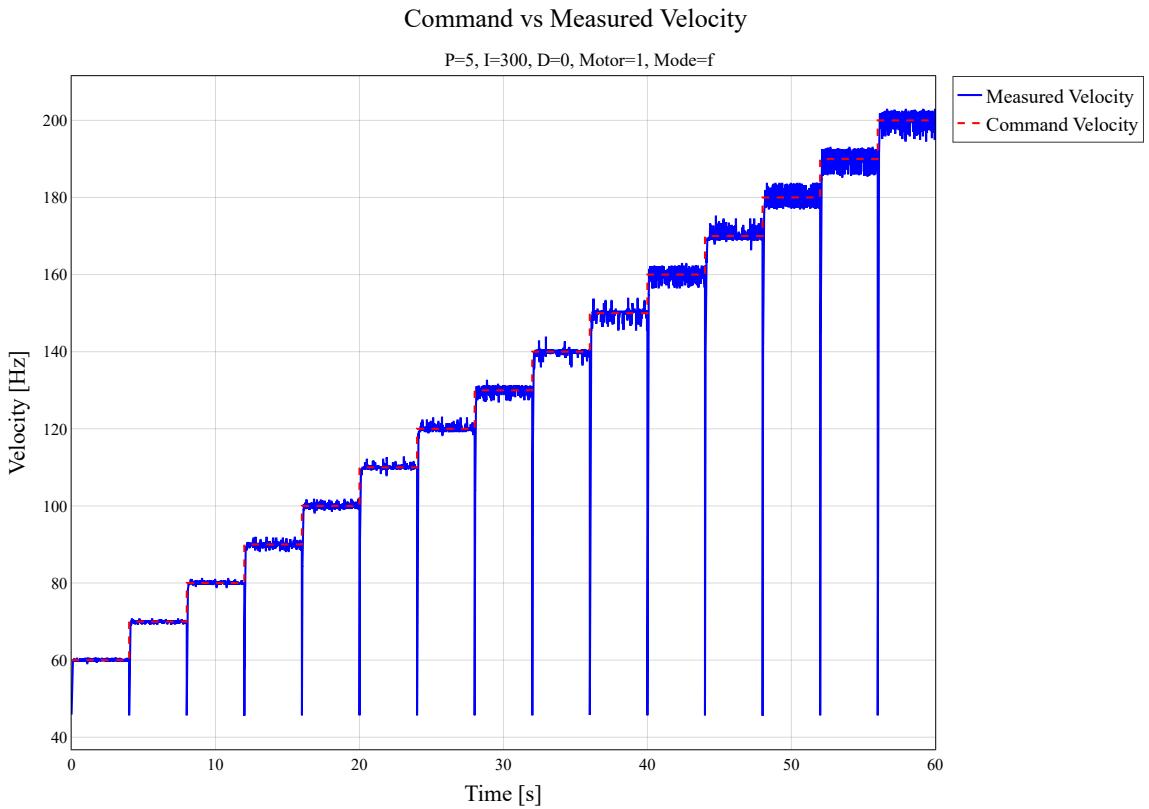


Figure 4.5: Ramp input test: system tracking a staircase velocity profile.

Steady-State Accuracy

A zoomed view on the steady-state region is provided in Figure 4.6. The error remained consistently under 5 Hz, confirming the controller's stability under constant velocity.

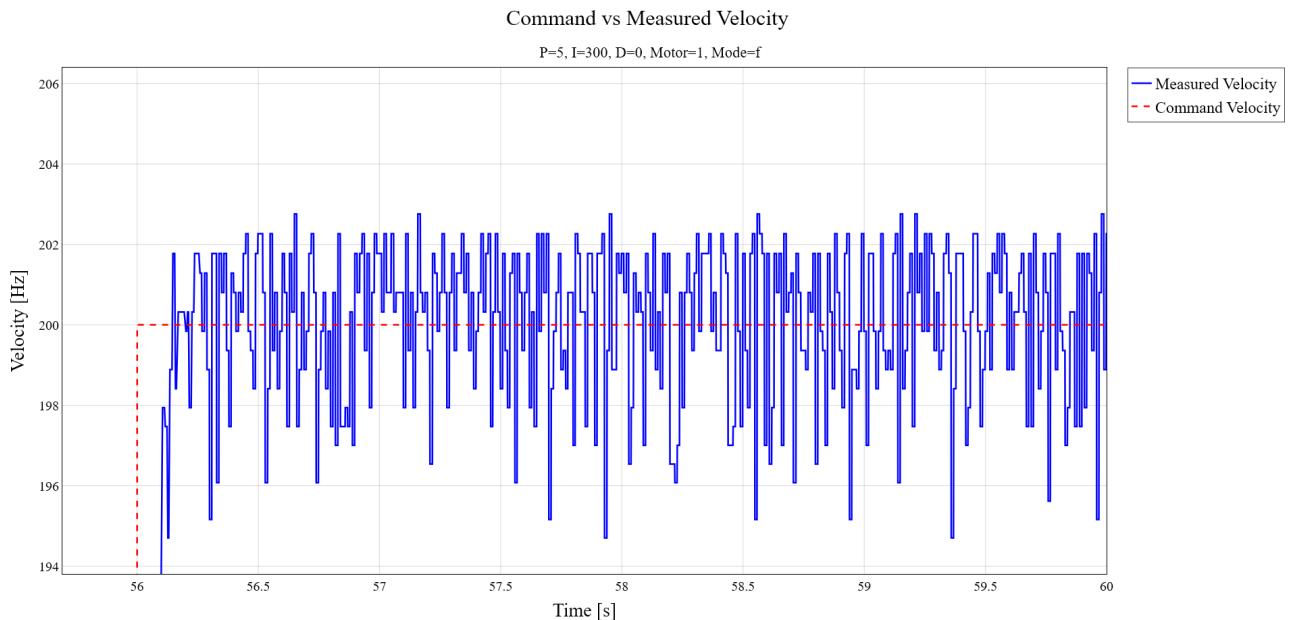


Figure 4.6: Zoomed view of steady-state convergence. Error remains below 5 Hz.

These results validate the selected PID gains and demonstrate that the controller maintains consistent behavior across both step and ramp inputs. The low steady-state error and fast

response times justify the use of this configuration for dynamic thrust and torque measurements in the subsequent phases of the Test-bench experiments.

4.3.2 Thrust and Torque Characterization

Following the motor PID tuning (see Section 4.3.1), the propulsion system was experimentally characterized using the setup detailed in Section 4.2. The objective was to empirically derive the aerodynamic coefficients C_f and C_τ , which represent thrust and torque generation as a function of motor speed. These coefficients are essential for actuator allocation and dynamic modeling.

The experiment consisted of applying a staircase velocity profile in 11 discrete steps, ranging from 50 Hz to 550 Hz in increments of 50 Hz. At each step, the motor was allowed to reach steady state, and the average force and torque values were logged after bias correction. The velocity tracking performance is shown in Figure 4.7, confirming the system's ability to reliably execute the desired input sequence.

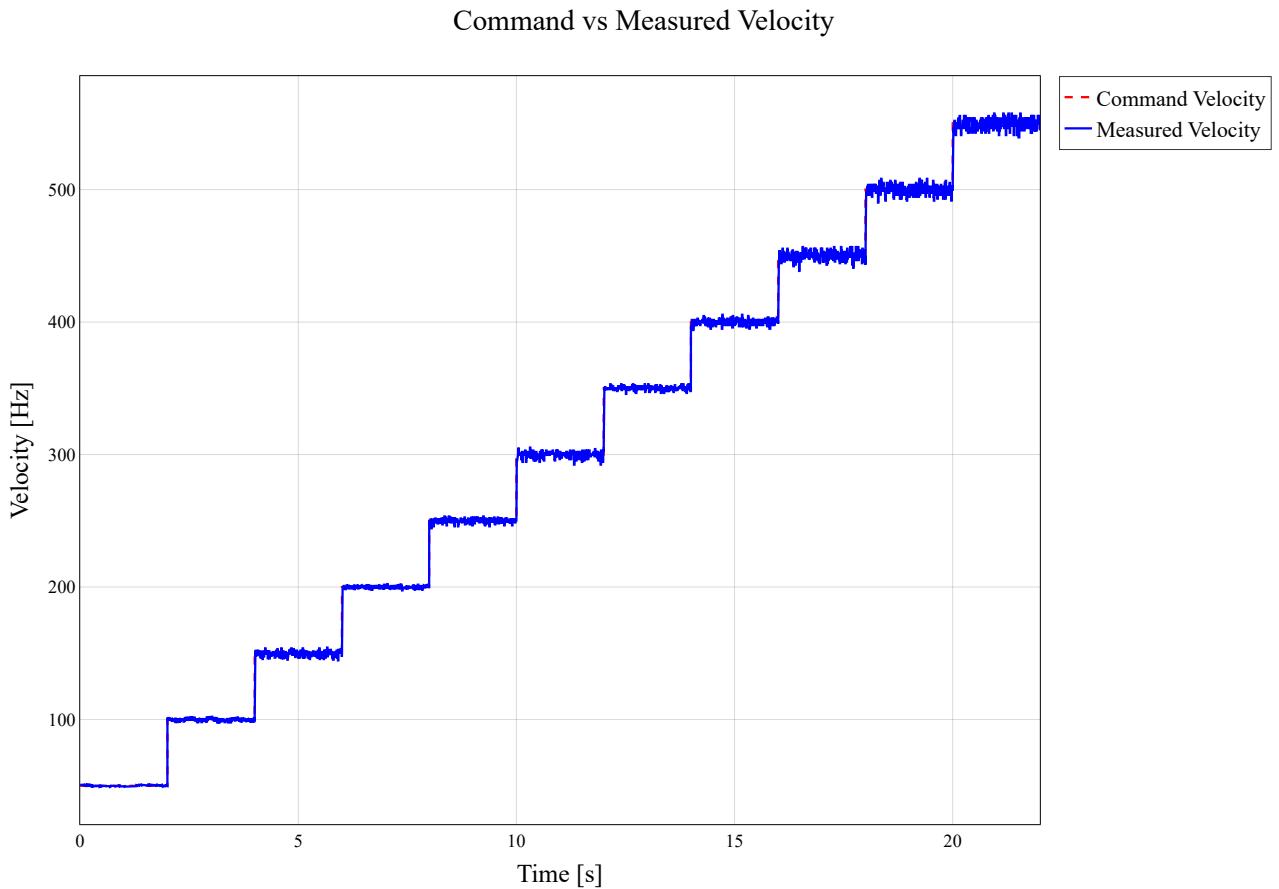


Figure 4.7: Commanded vs. measured motor velocity during characterization.

Figures 4.8 and 4.9 present the averaged, debiased force and torque data as functions of the commanded velocity. The observed negative trend in F_z (thrust direction) is due to the inverted "push" propeller configuration, in which the thrust vector is directed downward.

Debiased Forces vs Commanded Velocity

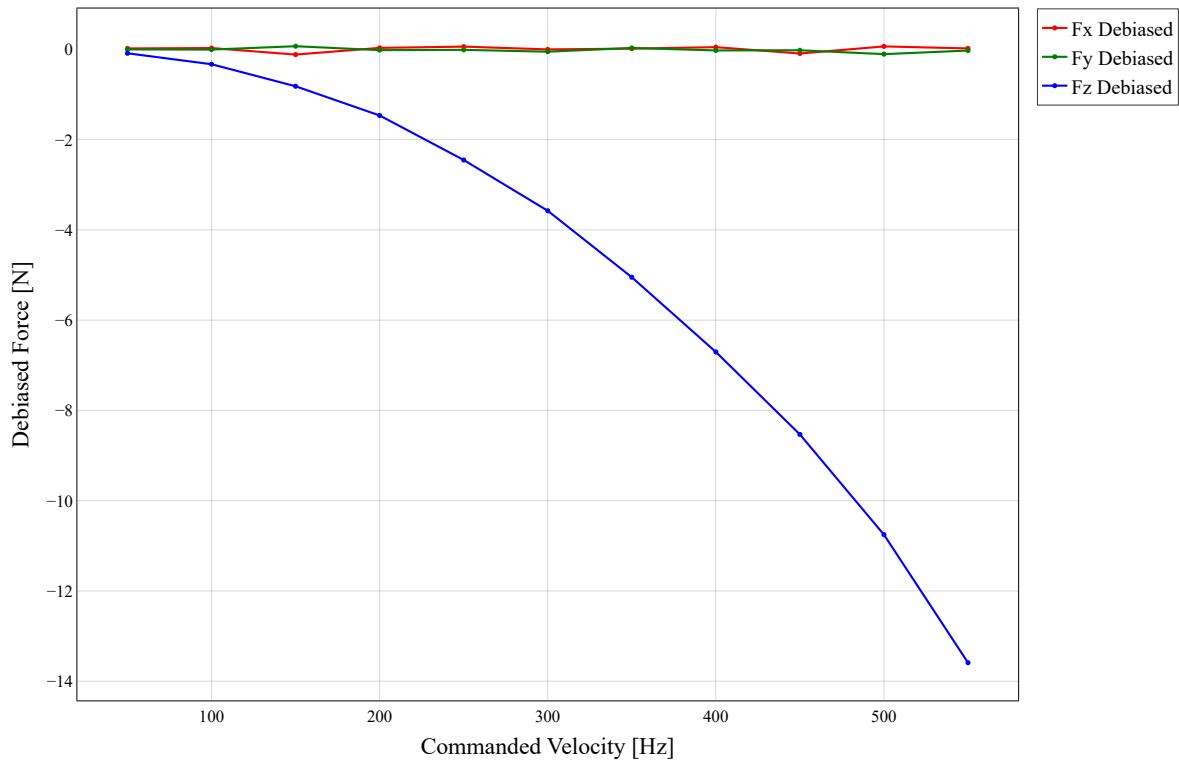


Figure 4.8: Average debiased forces: F_x , F_y , and F_z vs. motor speed.

Debiased Torques vs Commanded Velocity

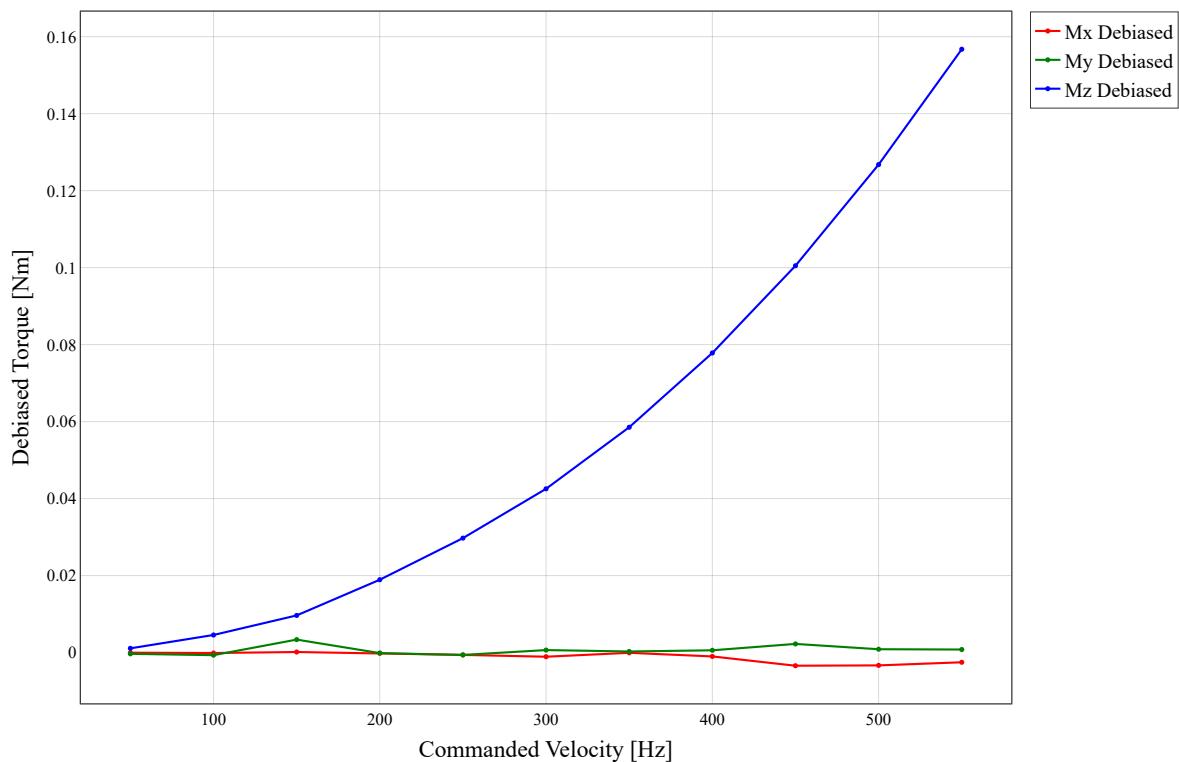


Figure 4.9: Average debiased torques: M_x , M_y , and M_z vs. motor speed.

Quadratic Model Estimation

The initial approach assumed a purely quadratic relationship between force/torque and motor speed, modeled as:

$$f_i = C_f \omega_i^2, \quad \tau_i = C_\tau \omega_i^2$$

A least-squares regression yielded the following average aerodynamic coefficients:

$$C_f = -3.94 \times 10^{-5} \text{ N/Hz}^2, \quad C_\tau = 4.76 \times 10^{-7} \text{ Nm/Hz}^2$$

The trends of C_f and C_τ across the velocity range are shown in Figures 4.10 and 4.11, respectively. However, the non-constant behavior across frequencies suggested that a purely quadratic model may not fully capture the system dynamics.

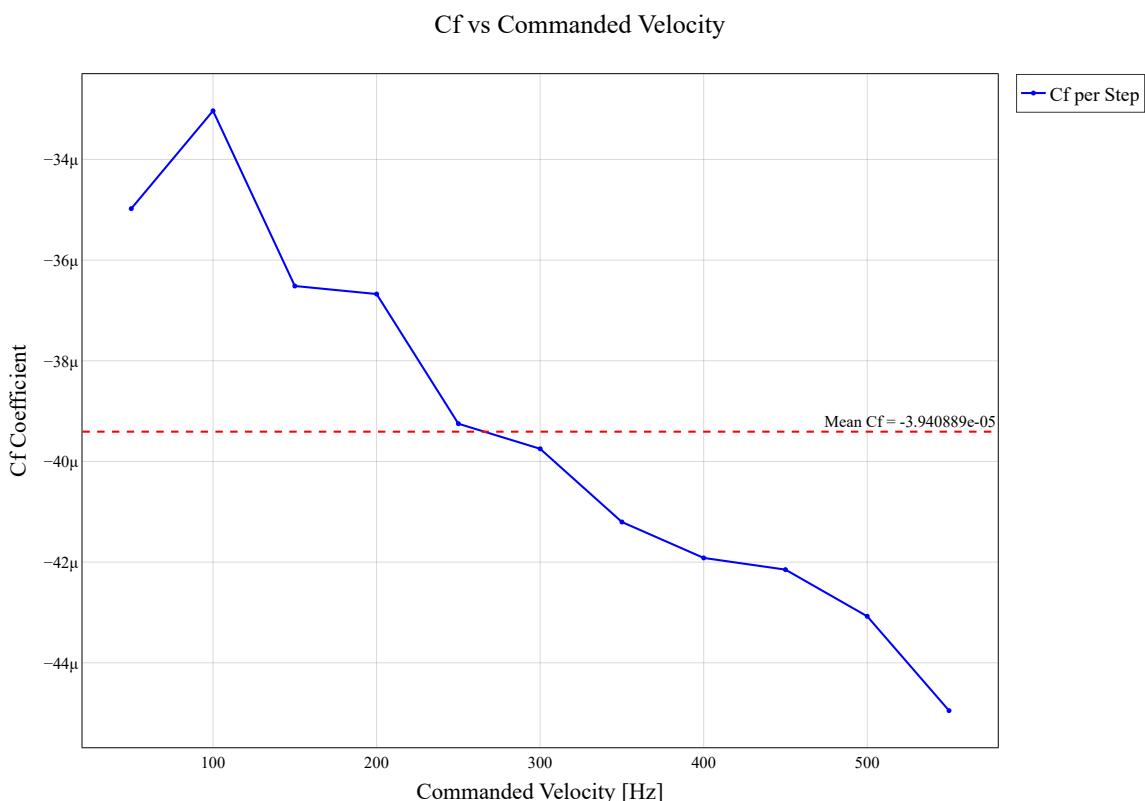


Figure 4.10: Estimated thrust coefficient C_f vs. commanded motor speed.

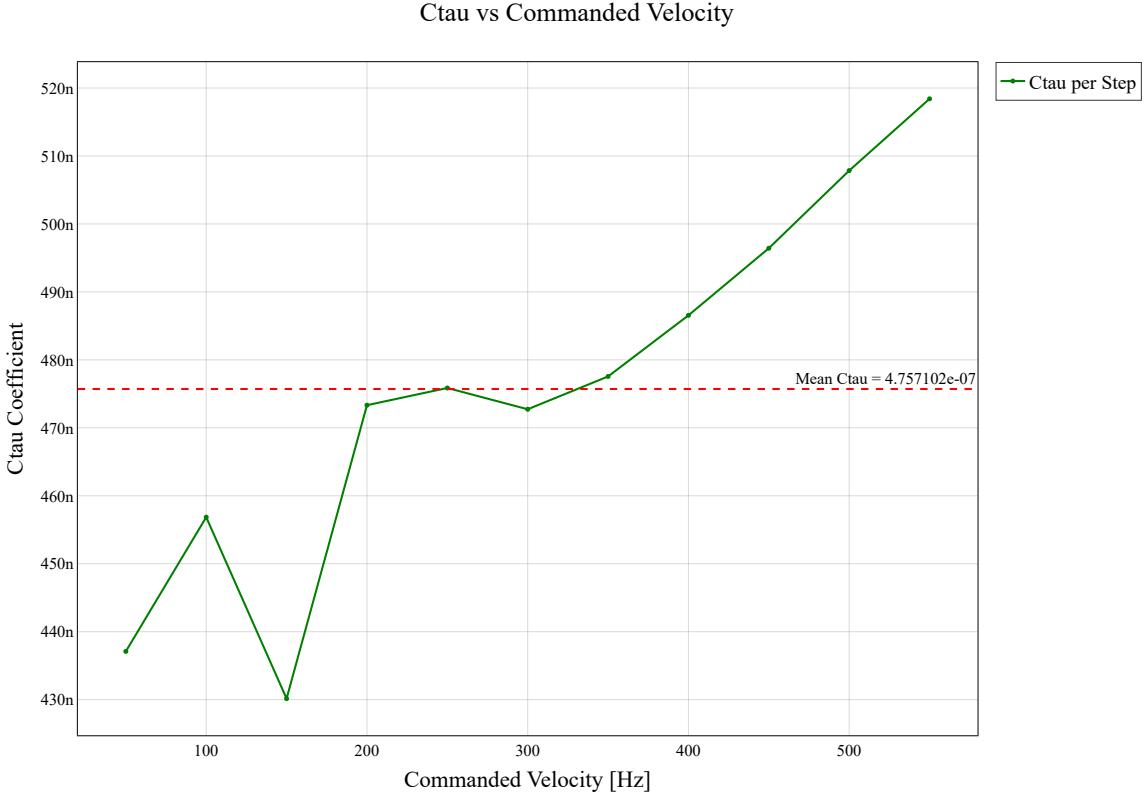


Figure 4.11: Estimated torque coefficient C_τ vs. commanded motor speed.

Extended Aerodynamic Modeling: Quadratic + Linear Fit

To enhance the accuracy of thrust and torque estimation, an extended aerodynamic model incorporating both quadratic and linear terms was adopted. While the classical formulation assumes that thrust and torque scale solely with the square of the angular velocity, empirical results revealed systematic deviations, particularly at lower frequencies, that justified the inclusion of a first-order term.

The proposed model expresses force and torque as:

$$f_i = a\omega^2 + b\omega, \quad \tau_i = a'\omega^2 + b'\omega \quad (4.2)$$

where a, a' represent the quadratic contributions and b, b' the linear corrections for thrust and torque, respectively.

Fitting this model to the experimental data yielded the following parameters:

- **Thrust model:** $a = 4.87 \times 10^{-5}$, $b = -2.58 \times 10^{-3}$, with a mean squared error (MSE) of 0.0147.
- **Torque model:** $a' = 5.53 \times 10^{-7}$, $b' = -2.27 \times 10^{-5}$, with a corresponding MSE of 1.36×10^{-6} .

These results were benchmarked against a purely quadratic formulation:

- **Quadratic-only thrust MSE:** 0.3753
- **Quadratic-only torque MSE:** 2.25×10^{-5}

The considerable reduction in MSE highlights the superior predictive performance of the extended model. Figures 4.12 and 4.13 show the fitting results, confirming improved alignment with the measured data, particularly at medium-to-high motor speeds.

Fitting a Quadratic Curve to Data

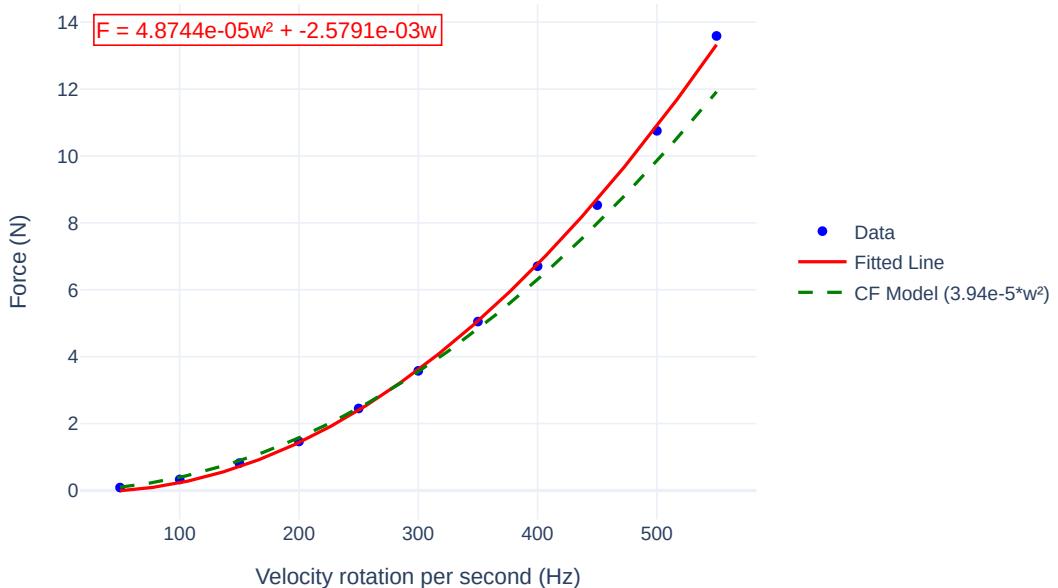


Figure 4.12: Fitting result for thrust data: the linear-plus-quadratic model improves prediction accuracy across the measured range.

Fitting a Quadratic Curve to Torque Data

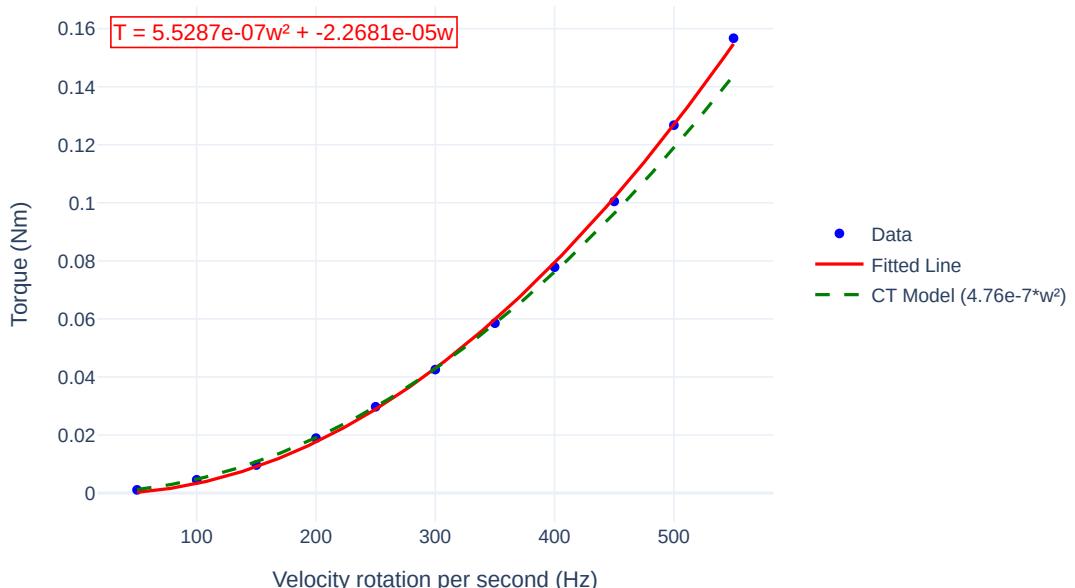


Figure 4.13: Fitting result for torque data: the extended model captures deviations from ideal quadratic behavior, particularly at higher velocities.

In conclusion, while the classical quadratic model remains suitable for control allocation tasks, the linear-plus-quadratic formulation offers improved accuracy and should be considered in future iterations of the configurator, particularly for use cases involving high-fidelity simulation or precise dynamic control.

4.3.3 Comparison with Manufacturer Datasheet

To assess the accuracy of the Test-bench measurements and to validate the configurator's predictive capabilities, the results obtained experimentally were compared with the thrust values reported in the motor manufacturer's datasheet.

Figure 4.14 presents a direct comparison between the measured thrust data, the corresponding quadratic fit, and the datasheet curve provided by the manufacturer. While the general trend is preserved, a consistent discrepancy is observed across the full velocity range.

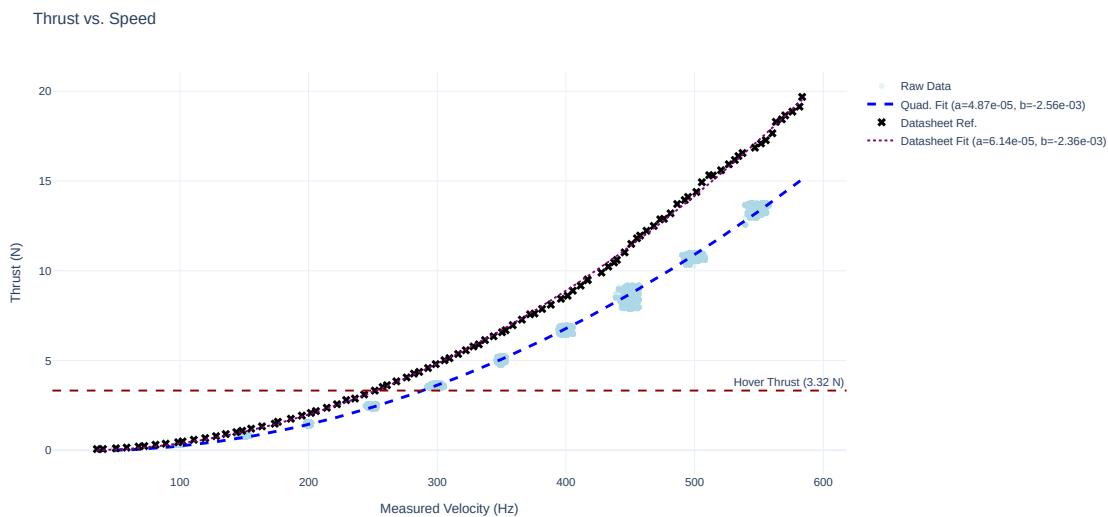


Figure 4.14: Comparison of thrust vs. motor speed: measured data, fitted model, and manufacturer datasheet reference.

This discrepancy is more clearly visualized in Figure 4.15, which shows the ratio between the datasheet and measured thrust values. The ratio stabilizes around a constant value (~ 1.22) after 200 Hz, indicating that the relative deviation becomes steady at higher speeds.

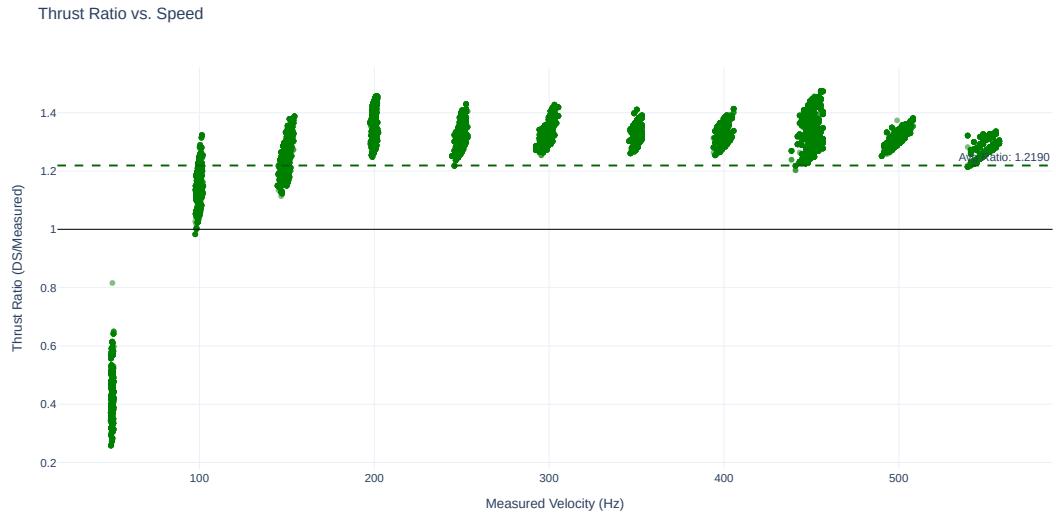


Figure 4.15: Ratio of datasheet to measured thrust values as a function of speed.

Discussion and Hypothesis

The discrepancy observed between the thrust measurements obtained in this work and those reported in the manufacturer's datasheet can likely be attributed to differing experimental setups. A key factor under consideration is the aerodynamic ground effect, which is known to increase apparent thrust when the propeller operates close to a reflective surface, such as a Test-bench platform. Interestingly, the ratio between the datasheet and measured thrust values remains roughly constant at higher rotational speeds. This trend supports the hypothesis that the deviation stems not from scaling errors in dynamic modeling, but from a static offset introduced by test environment differences, most notably, ground effect.

To investigate this hypothesis, an additional test was conducted with the propeller positioned approximately 10 cm above a flat surface. The maximum thrust increased from 15 N (figure 4.14) to approximately 17.2 N (figure 4.16). This significant rise confirms that ground proximity can artificially boost thrust readings.

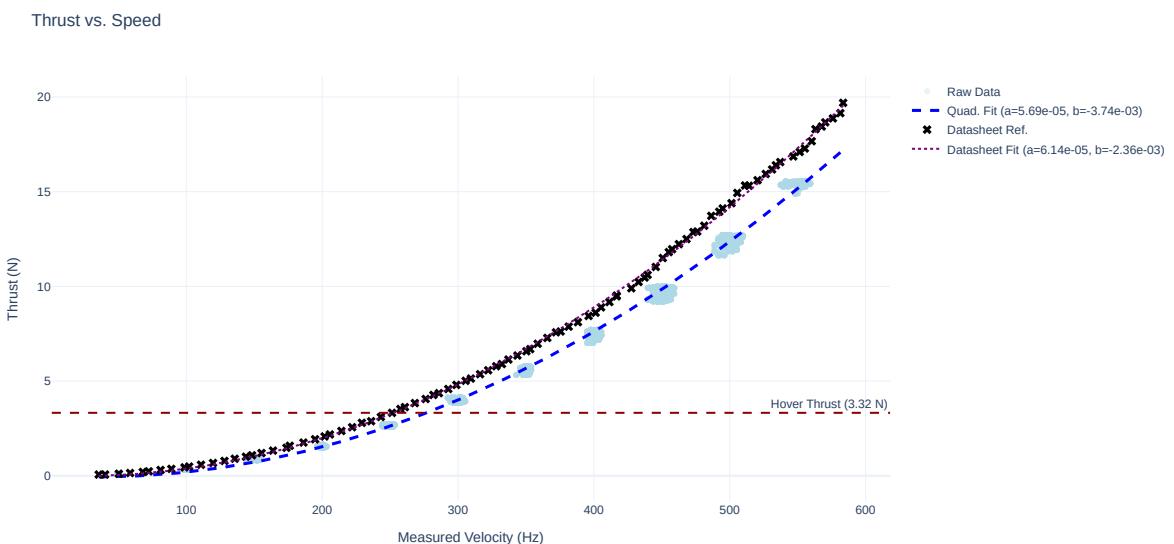


Figure 4.16: Measured thrust vs. motor speed with ground effect at 10 cm.

In contrast, the Test-bench setup used throughout this thesis was explicitly designed to minimize such aerodynamic bias. The motor-propeller unit was mounted in an elevated "push" configuration, with thrust directed downward and away from any surface that could cause flow recirculation. This geometry reduces interference and provides a more isolated measurement of motor-propeller performance.

These findings reinforce the critical role of hardware-in-the-loop validation in UAV design. Overreliance on idealized or vendor-supplied thrust values, without considering test setup influence, may lead to overestimation of system capabilities, controller misconfiguration, or even actuator saturation during demanding maneuvers. Conversely, although more conservative, the measurements obtained using the custom-built Test-bench provide a physically grounded foundation for accurate system identification and control tuning.

Methodology and Design

5.1 Configurator

This Section is fundamental in defining the key metrics and criteria that guided the redesign of the new quadrotor platform. The primary requests from the lab were to reduce the physical dimensions, and improve the TWR.

5.1.1 Design Requirements and Initial Analysis

The first step was to analyze the existing MKQuad (MKQ) platform. This drone has an inter-axis dimension of 46 cm, a total weight of 1386 g, and a TWR of approximately 2.2. Its size makes it unsuitable for multi-robot scenarios within the confined indoor flight space, which is limited due to the camera field-of-view of the motion capture system. Moreover, the current drone exhibits limited actuation capabilities in both force and torque, making it vulnerable to external disturbances such as wind during outdoor flight. Finally, its large dimensions reduce maneuverability and prevent the execution of agile trajectories.

5.1.2 Component Selection Tool

Armed with this background knowledge, the next step was to build a configurator to facilitate the component selection process. The configurator allows the user to select and combine:

- Motors
- Propellers
- FC
- ESC
- Frame
- Onboard computer
- Battery
- Optional sensors

Each component is listed with its mass, allowing an initial estimate of the total system weight:

$$m = n \cdot m_{\text{motor}} + n \cdot m_{\text{propeller}} + m_{\text{onPC}} + m_{\text{battery}} + m_{\text{frame}} + m_{\text{FC}} + m_{\text{ESC}} + m_{\text{sensors}} \quad [\text{g}] \quad (5.1)$$

where:

- n is the number of motors (and associated propellers),
- $m_{component}$ represents the mass of relative component.

During this phase, all data was taken directly from manufacturers' datasheets. In particular, for each motor-propeller combination, the datasheets typically report:

- Propeller dimensions (inches)
- Operating voltage (V)
- Throttle percentage
- Current draw (A)
- Thrust (g)
- Power consumption (W)
- Efficiency (g/W)
- Temperature under load

These performance indicators were critical for identifying optimal configurations in terms of efficiency, thrust generation, and power consumption, ensuring that the final design meets the target performance metrics established during the requirements analysis.

5.1.3 From Configuration to Performance Estimation

The configurator also enables preliminary estimation of flight-relevant metrics.

Weight and Gravitational Force

The gravitational force acting on the UAV is:

$$F_g = m \cdot g \quad [\text{N}] \quad (5.2)$$

with:

- $g = 9.81 \text{ m/s}^2$: gravitational acceleration,
- m : total UAV mass in kilograms.

Total Thrust and TWR

The total thrust generated by all motors is:

$$T = n \cdot F_{\text{motor}} \quad [\text{N}] \quad (5.3)$$

The TWR is:

$$\text{TWR} = \frac{T}{F_g} \quad (5.4)$$

as described in [32]. A TWR above 3 is generally desirable for agile and responsive flight.

Expected Vertical Acceleration

Two methods were considered for estimating vertical acceleration:

Newton's Second Law:

$$a = \frac{T - F_g}{m} \quad [\text{m/s}^2] \quad (5.5)$$

TWR-Based Estimate:

$$a = g \cdot (\text{TWR} - 1) \quad [\text{m/s}^2] \quad (5.6)$$

Lateral Acceleration

To evaluate lateral dynamics, two approaches were used:

Tilt-Angle Based Model:

$$a_{\text{lat}} = \frac{T \cdot \sin(\theta) - F_g}{m} \quad (5.7)$$

where θ is the tilt angle of the drone.

Alternatively, the maximum sustainable tilt angle under different acceleration conditions was evaluated:

$$\theta_{1G} = \arccos\left(\frac{F_g}{T}\right) \cdot \frac{180}{\pi} \quad (5.8)$$

$$\theta_{2G} = \arccos\left(\frac{2F_g}{T}\right) \cdot \frac{180}{\pi} \quad (5.9)$$

$$\theta_{3G} = \arccos\left(\frac{3F_g}{T}\right) \cdot \frac{180}{\pi} \quad (5.10)$$

These expressions provide bounds for maximum tilt without loss of vertical lift.

Rotational Acceleration from Motor Placement

To evaluate the rotational agility of the platform and determine the impact of motor placement, we compute the total moment of inertia and corresponding angular accelerations about the center of mass.

Total Inertia The main frame is modeled as a rigid rectangular body of mass m_{frame} , height h , and width w . Its moment of inertia is given by:

$$J_{\text{frame}} = \frac{1}{12}m_{\text{frame}}(h^2 + w^2) \quad (5.11)$$

Each motor-propeller assembly is modeled as a point mass located at a distance ℓ from the center. The total system inertia becomes:

$$J = J_{\text{frame}} + 4 \cdot (m_{\text{motor}} + m_{\text{propeller}}) \cdot \ell^2 \quad (5.12)$$

Thrust and Torque Allocation In an X-configuration, each arm lies along a diagonal and forms an angle α with the body axes. This angle defines the projection of thrust contributions onto the roll and pitch axes. Assuming symmetric arm angles and thrust distributions, the torque allocation can be modeled using:

$$\begin{bmatrix} T \\ \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \mathbf{A} \cdot \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} \quad (5.13)$$

with the allocation matrix as in Section 3.2.3:

$$\mathbf{A} = \begin{bmatrix} C_f & C_f & C_f & C_f \\ C_f\ell \sin \alpha & C_f\ell \sin \alpha & -C_f\ell \sin \alpha & -C_f\ell \sin \alpha \\ -C_f\ell \cos \alpha & C_f\ell \cos \alpha & C_f\ell \cos \alpha & -C_f\ell \cos \alpha \\ -C_\tau & C_\tau & -C_\tau & C_\tau \end{bmatrix} \quad (5.14)$$

Where:

- ℓ is the distance from the center to each motor,
- C_f is the thrust coefficient,
- C_τ is the yaw torque coefficient,
- α is the angle between the arm and the body axis.

Angular Acceleration Given a torque vector $\boldsymbol{\tau}$, the angular accelerations about the principal axes are:

$$\dot{\boldsymbol{\omega}} = \mathbf{J}^{-1} \cdot \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} \quad (5.15)$$

Assuming decoupled dynamics with a diagonal inertia tensor:

$$\dot{\omega}_x = \frac{\tau_\phi}{J_{xx}}, \quad \dot{\omega}_y = \frac{\tau_\theta}{J_{yy}}, \quad \dot{\omega}_z = \frac{\tau_\psi}{J_{zz}} \quad (5.16)$$

5.1.4 Envelope Analysis

Using the corrected aerodynamic coefficients obtained from experimental data (see Section 4.3.2), the configurator can compute the full actuation envelopes of each UAV platform, describing their feasible torque and linear acceleration domains. These envelopes, visualized through both 2D and 3D plots, serve as graphical tools to assess flight capability limits, controller feasibility, and structural design robustness.

Three configurations were analyzed:

- **MKQuad (MKQ):** The quadrotor present in the laboratory used as the baseline reference.
- **Bumblebee Limited Power (BumbleBee Limited Power (BBLP)):** The custom-designed BBLP quadrotor, operated under thrust-limited conditions. This constraint was applied to approximately match the performance envelope of the MKQ, allowing an isolated evaluation of geometric layout and inertial contributions to maneuverability.

- **Bumblebee Full Power (BumbleBee Full Power (BBFP)):** The same platform as BBLP, but operated without thrust limitations to fully exploit its propulsion system and structural configuration. This variant reveals the upper-bound capabilities of the new design.

By comparing these configurations, the envelopes provide insight into how design choices, such as motor power, frame geometry, and inertia, affect the actuation authority and dynamic responsiveness of the platform.

The thrust-torque and thrust-angular acceleration envelopes, in particular, highlight the drone's maximum achievable wrench along each axis. They are crucial for evaluating whether aggressive flight maneuvers can be safely executed without saturating the actuators or exceeding structural limits.

3D Attainable ACCELERATION Envelopes with Hover Planes

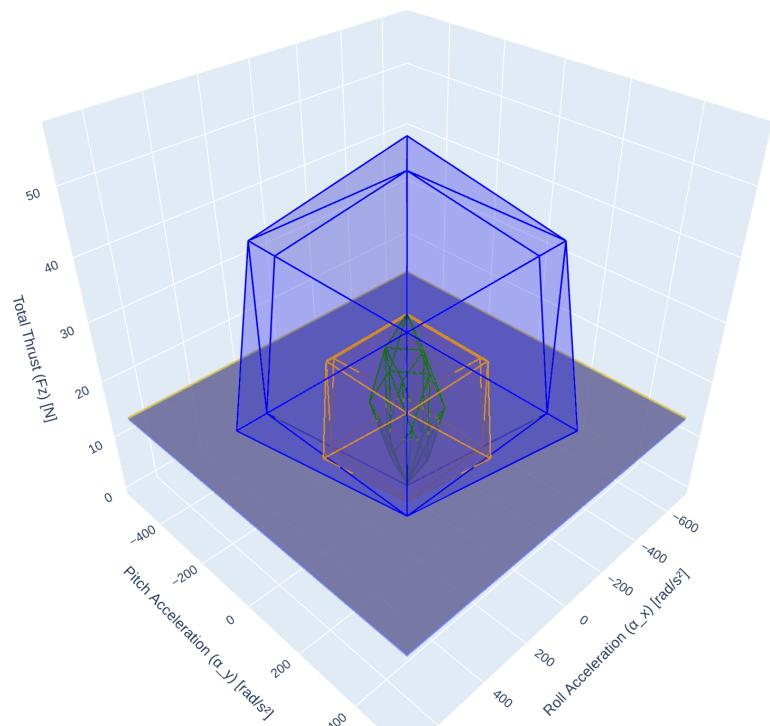


Figure 5.1: 3D attainable acceleration envelope with hover planes.

3D Attainable TORQUE Envelopes with Hover Planes

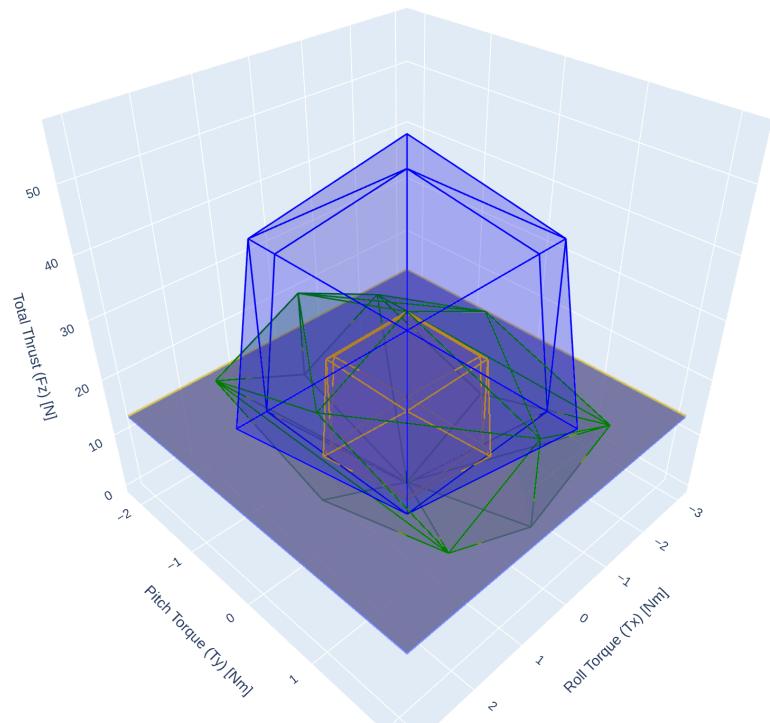


Figure 5.2: 3D attainable torque envelope with hover planes.

Thrust vs. Yaw Angular Acceleration

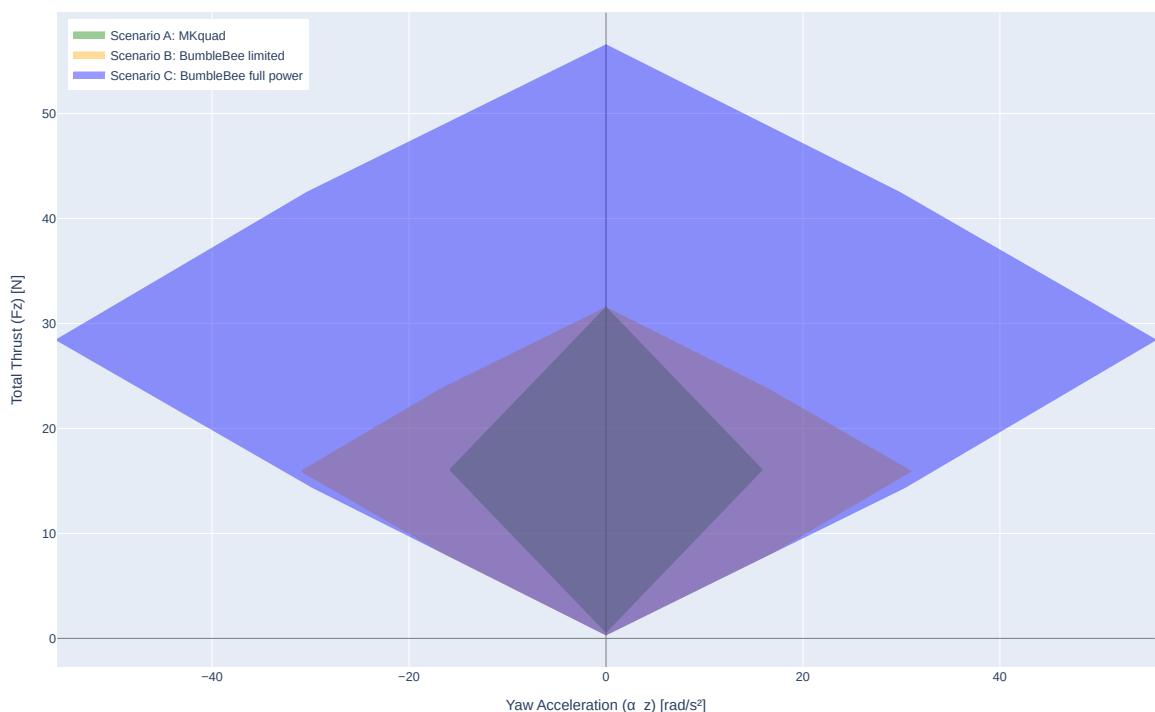


Figure 5.3: 2D cross-section of the attainable acceleration envelope.

Thrust vs. Yaw Torque

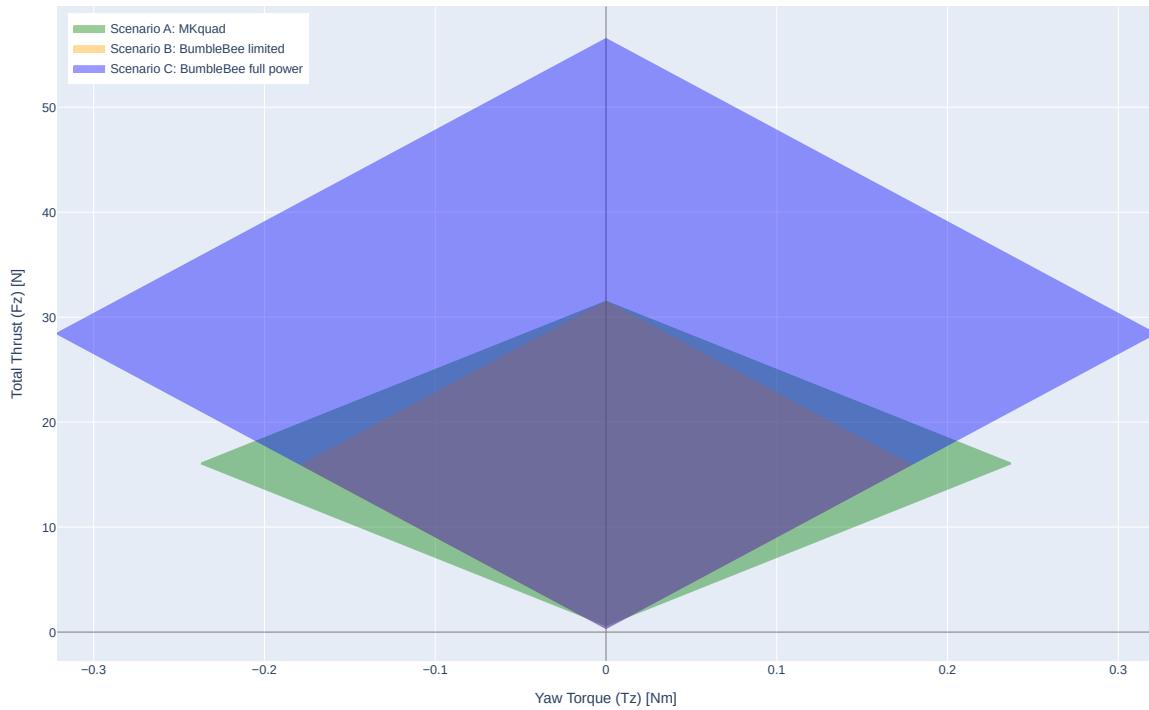


Figure 5.4: 2D cross-section of the attainable torque envelope.

Linear vs. Angular Acceleration Capability

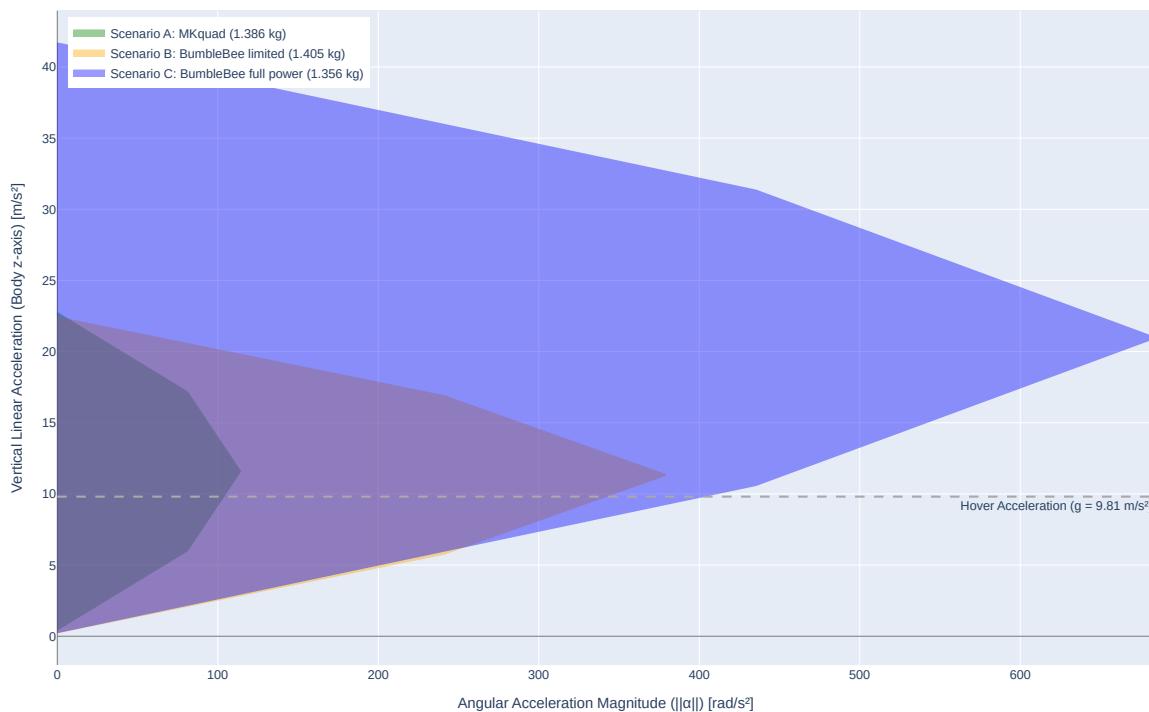


Figure 5.5: Comparison of linear and angular acceleration across platforms.

Overall, the envelope analysis provides a physics-informed benchmark for each platform's capabilities, supporting informed decisions on controller aggressiveness, safety margins, and mechanical stress tolerance across the three configurations.

5.2 Quadrotor Design

After selecting the propulsion system that meets the TWR requirements via the configurator, the next step was to design the physical structure of the quadrotor. This phase focused not only on reducing the overall dimensions but also on optimizing the internal layout to maintain a low Center of Mass (COM). A low COM is particularly critical in applications such as flying cranes, where it helps reduce destabilizing wrenches and improves flight stability.

5.2.1 Initial Design Considerations

Several design concepts were evaluated to achieve the goal of compactness while ensuring compatibility with the chosen electronic components. Before defining the structural layout, it was necessary to identify the components that must be housed within the UAV and to estimate their spatial requirements. The following key components were considered:

- **Onboard computer:** Two options are supported—Jetson Xavier NX and Intel NUC 12. Since both must be accommodated, the larger footprint of the two is considered in layout planning.
- **ESC:** A standard off-the-shelf board with dimensions of 30.5 mm × 30.5 mm is selected.
- **FC:** Two potential candidates are the Chimera v1.0 (larger) and the Tawaki v1.10 (smaller but not yet available). To ensure future compatibility, the larger Chimera dimensions are used.
- **Battery:** The current setup uses a 4S battery, but the frame must support an upgrade to 6S for extended endurance and increased thrust.
- **Motors and Propellers:** Propeller size is fixed by the selected model, Gemfan GF51466R, which defines the minimum spacing required to avoid interference.

5.2.2 Minimum Motor Distance Calculation

A key challenge in the quadrotor design was determining the minimum allowable distance between the motors and the COM, which directly affects the arm span. This decision impacts the drone's overall footprint, agility, and ability to operate in constrained spaces such as motion-capture arenas.

The central idea was to minimize the span while ensuring that all internal components—battery, onboard computer (PC), ESC, and FC—could be safely housed without mechanical interference.

Battery vs. PC Placement

Two possible stacking configurations were evaluated:

- **PC at motor plane height:** Due to its square footprint, placing the PC at the same height as the arms significantly increased the required clearance.

- **Battery at motor plane height:** The rectangular footprint of the battery allowed for a more compact design along the lateral axes, enabling a tighter fit without compromising internal space.

Conclusion: Placing the battery in the motor plane (and the PC above or below) minimizes the required inter-motor span, thereby reducing the quadrotor's footprint.

Geometric Estimation

To ensure all components fit safely within the body of the quadrotor and propeller interference is avoided, we compute the minimum lateral distance between adjacent motors in the *X-configuration*. This distance must accommodate:

- The full propeller radius from each motor,
- The maximum body width,
- A margin for structural clearances and safety.

The total lateral span between two adjacent motors, denoted as d_{lat} , is estimated as:

$$d_{\text{lat}} = 2 \cdot r_{\text{prop}} + d_{\text{battery}} + 2 \cdot \Delta_{\text{margin}} \quad [\text{mm}] \quad (5.17)$$

Where:

- r_{prop} is the radius of one propeller,
- d_{battery} is the width of the battery,
- Δ_{margin} is a small safety margin (typically 5–10 mm).

This formulation guarantees:

1. The propellers on adjacent motors do not overlap or touch the battery casing,
2. All internal components are enclosed within the central body,
3. Structural compactness is preserved while ensuring mechanical safety.

Given the lateral span d_{lat} and the longitudinal battery dimension l_{battery} , the true distance from the center of mass to each motor (denoted r_{motor}) in an *X-configuration* is given by the Euclidean norm:

$$r_{\text{motor}} = \sqrt{\left(\frac{d_{\text{lat}}}{2}\right)^2 + \left(\frac{l_{\text{battery}}}{2}\right)^2} \quad [\text{mm}] \quad (5.18)$$

Where:

- l_{battery} is the longer dimension (length) of the battery,
- The factor $\frac{1}{2}$ accounts for the projection from the COM to one corner of the rectangular body layout.

This formula ensures precise arm length computation for an X-frame layout, minimizing size while maintaining clearance for all components.

5.2.3 Lower Center of Mass

Once the lateral layout and minimum motor distance were defined, the next crucial design question was how to minimize the overall COM of the system. This is particularly important for tasks involving aerial manipulation or flying cranes, where external wrenches act on the body. A lower COM improves stability, reduces control effort, and enhances safety.

To address this, we evaluated four different component stacking configurations in CAD. These setups varied in the vertical placement of the motors, battery, and onboard PC. The primary goal was to assess how the placement of the rectangular battery influences the vertical COM.

Design Variants Analyzed

- **Design A:** Battery and motors on top, PC underneath.
- **Design B:** Battery and motors on top, PC underneath, battery rotated to place the shorter side vertically.
- **Design C:** Battery and motors on the bottom, PC above.
- **Design D:** Battery and motors on the bottom, PC above, battery rotated to place the shorter side vertically.

The battery used in all configurations had dimensions $131 \times 43.5 \times 29$ mm, approximated as a rectangular parallelepiped. For each design, the COM along the vertical axis was extracted from the CAD assembly and recorded in the table below:

Design	Battery Orientation	Vertical COM [mm]	Battery Base [mm]	COM–Base Distance [mm]	Diagonal Span [mm]
A	Long side vertical (top)	-6.5	-43	36.5	260.7
B	Short side vertical (top)	7	-28	35	260.7
C	Short side vertical (bottom)	4.7	-28	32.7	272.7
D	Long side vertical (bottom)	-5.0	-31	26.0	272.7

Table 5.1: Comparison of COM for Different Layouts

Outcome and Final Design Choice

The design with the lowest COM was configuration D, which places both the motors and battery beneath the main frame and orients the battery with its longest dimension horizontal, minimizing vertical height. This design effectively lowers the vertical position of the heaviest component and achieves the best stability without compromising footprint.

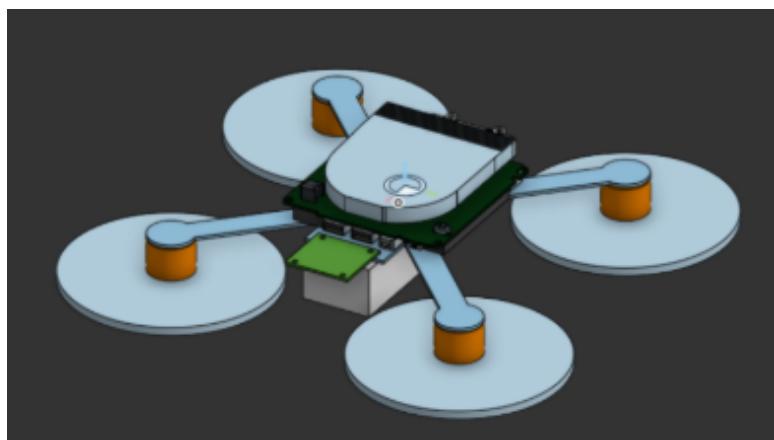


Figure 5.6: Final CAD Configuration (Design D): Motors and Battery Mounted Below

This setup was selected as the final configuration for the prototype due to its optimal mass distribution, mechanical compactness, and improved controllability.

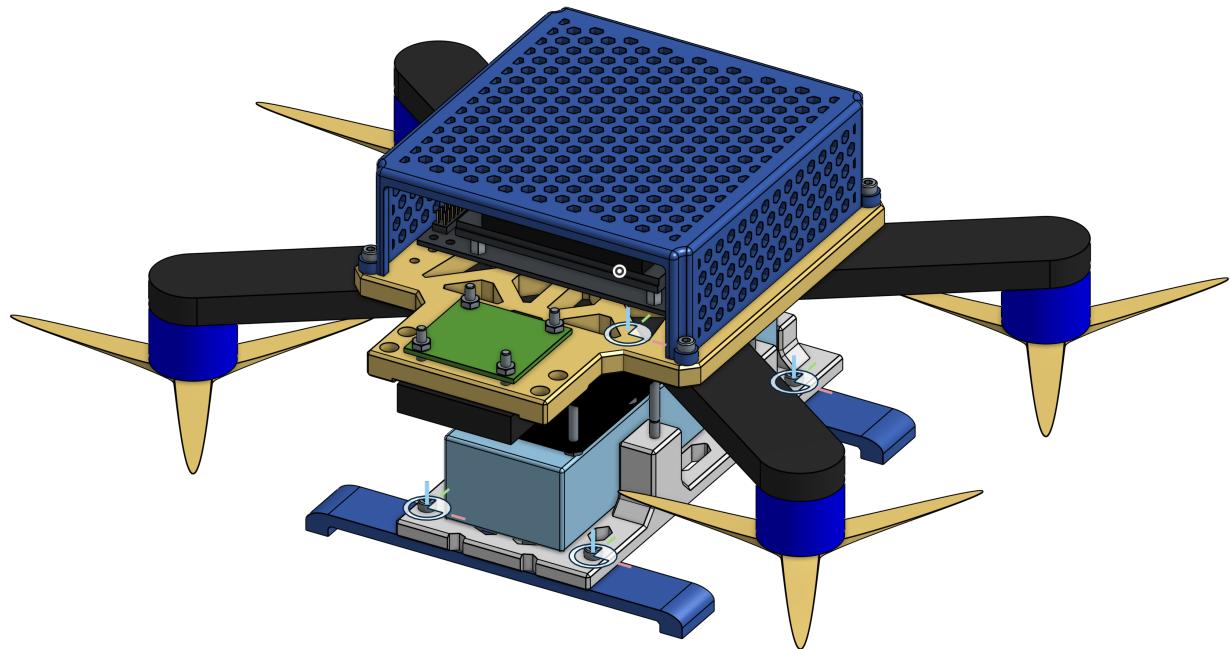


Figure 5.7: Final CAD Render of the BB Platform

Implementation

6.1 Software Stack: telekyb3

The aerial robotic platform is controlled using the `telekyb3` [33] software framework, a modular and open-source collection of real-time components developed at LAAS–CNRS, a research laboratory located in Toulouse, France. Originally conceived in 2015 as a successor to the earlier `TeleKyb` stack, `telekyb3` (`tk3`) is part of the broader `OpenRobots` ecosystem and is actively maintained by the robotics team at LAAS–CNRS [34].

6.1.1 Framework Design Principles

`telekyb3` is built around three fundamental principles: **modularity**, **reusability**, and **interoperability**. The architecture is based on the `GenoM3` formalism, a component description tool enabling the development of middleware-independent modules with clearly defined interfaces. This ensures seamless integration with different robotics backends such as `pocolibs`, ROS, or YARP.

Each functional capability—such as estimation, control, or sensing—is implemented as a separate `GenoM3` component, allowing fine-grained configuration and testing. The components are compiled and distributed via the `robotpkg` infrastructure, ensuring consistency and reproducibility across platforms.

6.1.2 Core Components

The control stack used for flight experiments is composed of the following core `GenoM3` components:

- `rotorcraft-genom3`: Interface layer to low-level flight controllers, including motor PWM and RPM control.
- `nhfc-genom3`: Near-hovering flight controller for UA quadrotors, implementing cascade PID logic.
- `uavatt-genom3` and `uavpos-genom3`: Attitude and position controllers for fully-actuated multirotors.
- `maneuver-genom3`: Trajectory generation and waypoint execution using polynomial interpolations.

6.1.3 Custom Modifications

As part of this work, a minor yet impactful contribution was made to the `nhfc-genom3` component, which implements a near-hovering cascade PID controller for UA multirotors. In its original design, the controller enforced symmetric PID gain tuning across the lateral (X and Y) translational axes, as well as for roll and pitch, limiting the ability to address axis-specific dynamics or asymmetries in the platform.

To overcome this limitation, the component was modified to support asymmetric gain tuning along the X and Y axes. This enhancement allows independent configuration of the proportional, integral, and derivative gains for each direction, enabling more precise adaptation to vehicle-specific behavior, environmental disturbances, or task requirements. Such flexibility is particularly beneficial in scenarios involving unbalanced mass distribution, directional drag, or anisotropic control authority. These modifications were carefully integrated to preserve the component's original interface and compatibility with the existing control pipeline.

6.1.4 Perception and Estimation Modules

To support closed-loop control and interaction, several perception and estimation modules are included:

- `pom-genom3`: Unscented Kalman filter for sensor fusion (IMU, Global Positioning System (GPS), motion capture).
- `optitrack-genom3` and `realsense-genom3`: Interfaces to Optitrack and Intel Realsense devices, respectively.
- `gps-genom3`: Interfaces with U-blox¹, Novatel² and Tersus³ GPS receivers.

6.1.5 Middleware and Execution Environment

While GenoM3 components can operate over multiple middleware systems, `telekyb3` is typically deployed using the `pocolibs` runtime. This choice provides deterministic timing and low-latency communication, both of which are essential for aggressive maneuvering and physical interaction scenarios.

¹<https://www.u-blox.com/en>

²<https://novatel.com/>

³<https://www.tersus-gnss.com/>

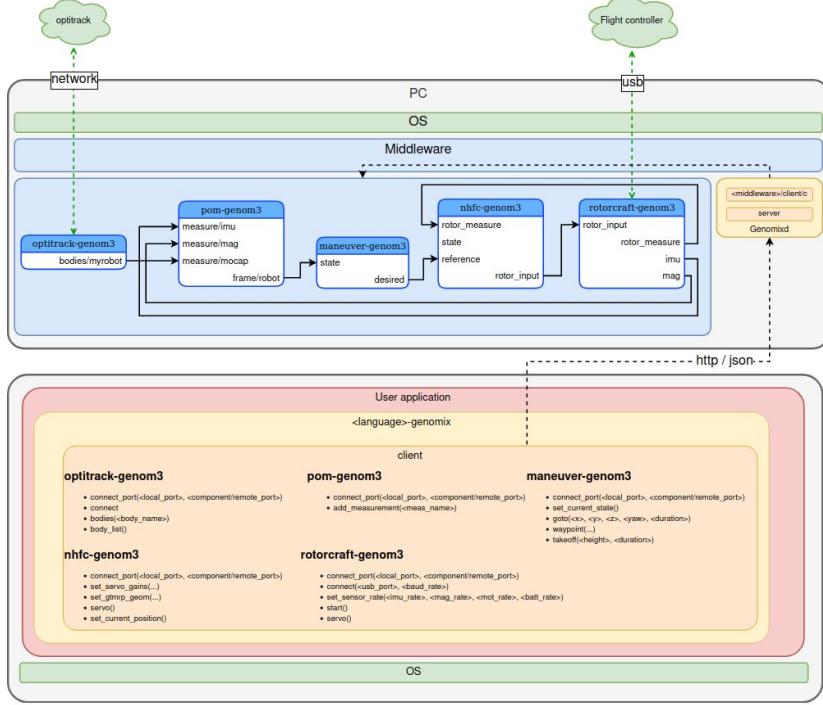


Figure 6.1: Quadrotor software architecture implemented using `telekyb3`. Components include state estimation, control, trajectory generation, and hardware interfacing. Courtesy of Gianluca Corsini [1].

6.1.6 Summary

The `telekyb3` stack offers a robust and extensible infrastructure for the development and deployment of aerial robotic systems. Its modular design, real-time guarantees, and integration with open-source tools make it suitable for advanced research in aerial physical interaction and human-robot collaboration.

- Project repository: <https://git.openrobots.org/projects/telekyb3>
- Documentation: <https://git.openrobots.org/projects/telekyb3/pages/index>
- Package manager: <https://git.openrobots.org/projects/robotpkg>

6.2 Quadrotor Platform

This section describes the full implementation of the quadrotor platform used for trajectory tracking, control benchmarking. The platform integrates a high-performance hardware stack with a simulation-consistent software pipeline, allowing seamless transitions between virtual and real-world testing. All components are natively supported by the `telekyb3` architecture, enabling agile development and rigorous evaluation of aerial control strategies.

6.2.1 Mechanical and Electronic Setup

The quadrotor platform is built upon a freestyle T700 carbon frame, selected to meet the structural and spatial constraints identified in Section 5.2. To accommodate the full complement of electronic subsystems, including compute modules, flight controller, power electronics, and batteries, custom 3D-printed mounts were designed and integrated into the frame. These

additions enable compact internal organization, vibration isolation, and ease of replacement or upgrade. A summary of the complete hardware configuration is provided in Table 6.1.

Subsystem	Components
Frame and Propulsion	Volador II VX6 O3 FPV Frame Kit (T700 carbon) 4× AOS Supernova 2207 1980KV brushless motors 4× Gemfan GF51466R propellers
Electronics	nVidia Jetson Xavier NX (onboard compute) Paparazzi Chimera v1.00 flight controller Skystars KO45A 4-in-1 ESC
3D-Printed Components	Custom PC housing, battery mount, and FC bracket FC protection shell and optional landing legs

Table 6.1: Bill of materials for the quadrotor platform.



Figure 6.2: Assembled quadrotor platform used in experiments. Modular layout with onboard compute, flight controller, and protective shell visible.

6.2.2 Software and Control Integration

The quadrotor control pipeline builds directly on `telekyb3`'s real-time modules. Control commands are issued through `nhfc-genom3`, with sensor fusion handled by `pom-genom3`. State updates, emergency behavior, and trajectory planning are integrated through a Python control interface.

The `control.py` script is responsible for end-to-end management of flight operations. It supports multiple mission types including:

- Initialization and motor arming

- Takeoff, landing, and return-to-origin
- Complex trajectories using differential flatness (e.g., powerloop, circling, flip)
- PID gain tuning through a GUI interface
- Logging of IMU, motor, and pose data for post-flight analysis

This script ensures synchronized startup of `rotorcraft`, `nhfc`, `pom`, `maneuver`, and `optitrack` modules. Based on the mode (simulation or real), it adjusts connection ports, PID gains, and saturation constraints.

6.2.3 Simulation Implementation

Before conducting real-world flight tests, the quadrotor model was imported into the Gazebo simulation environment. This step ensures early-stage validation of the control architecture, dynamic behavior, and sensor integration in a safe, reproducible setting.

Drone Model Specification

The quadrotor was described using the SDF (Simulation Description Format). The main structure includes:

- A central `<link>` named `drone` with full inertial parameters (mass, center of mass offset, and inertia tensor).
- A mesh-based visual and collision representation using STL files.
- Four rotors, each defined using a modular `mrsim-rotor` model.
- Revolute joints (`motor-1` to `motor-4`) connecting each rotor to the central body.

The complete model is embedded within a top-level `<model>` tag named `BumbleBee_model`. The physical parameters (mass, inertia) reflect those measured or estimated from the real system. Each rotor is positioned according to the frame layout and connected via revolute joints for motor actuation.

Rotor Modeling Plugin

Each rotor is implemented as a separate Gazebo model (`mrsim-rotor`) using SDF. The propeller is defined as a rigid body with associated inertial properties, a cylindrical collision geometry, and visual elements including motor housing, axle, and blade meshes.

A custom Gazebo plugin named `mrsim` is used to model the aerodynamic and electrical behavior of the motors. This plugin supports:

- Rotor force and torque coefficients ($C_f = 3.94 \times 10^{-5}$, $C_\tau = 4.76 \times 10^{-7}$),
- Electrical parameters such as torque constant ($K = 0.0048$ Nm/A), resistance, inductance, and max current,
- Internal rotor dynamics via inertia ($J = 5 \times 10^{-6}$ kg · m²),
- Noise modeling and damping terms to improve realism.

Each rotor is configured with a spin direction (clockwise or counterclockwise), and optionally, an explicit allocation matrix can be specified to override default motor mapping.

Parameter Tuning and Validation

The parameters for inertia, torque constants, and electrical dynamics were derived from manufacturer specifications and validated against experimental data from the Test-bench. In particular, the aerodynamic coefficients C_f and C_r used in the simulation match those extracted from real-world experiments.

The Gazebo environment enables closed-loop testing of the full control pipeline using the same `telekyb3` components deployed on the physical system. This allows PID gains, controller structure, and sensor integration to be evaluated under repeatable conditions before hardware deployment.

Simulation results are visualized using Gazebo's built-in tools as well as custom plotting scripts. This visualization assists in tuning controllers and validating dynamic behavior (e.g., overshoot, steady-state error, response time) before committing to real flights.

6.2.4 Advanced Control Execution

All advanced trajectory execution routines implemented in the flight script leverage the principle of differential flatness, enabling the generation of dynamically feasible trajectories with smooth position, velocity, and acceleration profiles. This framework allows for real-time trajectory planning that inherently satisfies the quadrotor's dynamic constraints and simplifies the control law derivation.

The following high-level trajectory primitives are currently supported:

- `follow_leminscate()`: Executes a lemniscate (figure-eight) trajectory with coordinated yaw tracking, used for lateral agility benchmarking.
- `follow_cirp()`: Applies a frequency-swept linear path to evaluate control bandwidth and system resonance behavior.
- `air_flip()`: Generates a time-optimal 360-degree flip trajectory that respects motor thrust limits and moment of inertia, validated in simulation and flight.
- `powerloop()`: Generates a time-optimal powerloop trajectory that respects motor thrust limits and moment of inertia.

Each routine generates a dynamically consistent reference trajectory using the flat outputs and their derivatives up to snap. This formulation enables the computation of feedforward acceleration, orientation, and angular velocity, ensuring compatibility with the cascade control architecture of `telekyb3`.

The quadrotor implementation tightly couples hardware, simulation, and control infrastructure. The unified control script provides consistent flight behavior across environments, while Test-bench derived parameters ensure realistic simulation performance.

Experiments

7.1 Overview

The experimental phase of this thesis was designed to validate the accuracy of the configurator and to empirically demonstrate the improved capabilities of the newly developed quadrotor platform. The experiments range from controlled bench tests to simulations and real-world flight trials, concluding with comparative analyses and advanced flight demonstrations. The methodology is organized into the following four main Sections:

- **Simulation-Based Controller Tuning:** This Section details the tuning of the quadrotor's PID controller within a simulation environment. A combination of classical tuning strategies and empirical refinement was employed to achieve stable and responsive flight behavior. The controller's performance is validated through a square trajectory tracking task, where the drone performs a 90° yaw rotation at each corner. The quality of the control is assessed by analyzing the trajectory tracking error over time.
- **Real-World Validation:** This Section reports flight tests of the new quadrotor platform following the same square trajectory used in simulation. By comparing real-world results with simulation data, the accuracy of the motor models and the behavior of the controller under real dynamic conditions are assessed.
- **Comparative Evaluation:** The BumbleBee (BB) platform is evaluated against the legacy MKQ system through a sensitivity-based analysis. Two operating configurations of BB are considered: BBLP, in which motor thrust is artificially capped to match the motor performance of the MKQ; and BBFP, which leverages the platform's maximum thrust capabilities. The Constant Increasing Ramp Path (CIRP) maneuver is used as the basis for this comparative assessment, providing a controlled framework to quantify performance differences across platforms and configurations.
- **Agile Flight Demonstrations:** To validate the enhanced maneuverability of the BB platform, this section presents two advanced flight experiments: a complete 360-degree flip and a powerloop. These maneuvers are designed to assess the drone's control authority and confirm its capability for highly dynamic and research-oriented flight operations.

This structured and multi-layered experimental framework provides both detailed and system-level insights into the performance of the new quadrotor architecture. It also serves to validate the configurator as a reliable design and prediction tool for UAV development.

7.2 Simulation-Based Controller Tuning

This Section outlines the methodology adopted to tune the quadrotor’s flight controller within a simulation environment. The objective was to achieve accurate and stable trajectory tracking using a cascaded control architecture composed of an outer-loop position PID controller and an inner-loop attitude controller, as introduced in Chapter 3.

7.2.1 Tuning Strategy

Given the UA nature of the quadrotor, a hierarchical tuning procedure was adopted. The inner attitude control loop, governing the orientation and angular rate tracking, was tuned first in isolation, with the outer position loop kept weakly active to limit drift during hover. The rotational gain k_R was initially increased to improve responsiveness to attitude errors, followed by adjustments to the angular rate gain k_ω to damp oscillations and shorten the settling time.

Subsequently, the position controller was tuned following a similar logic. The proportional and derivative gains were tuned to minimize steady-state error and improve transient performance. The integral terms K_i were introduced to account for slow drift and residual steady-state errors arising from unmodeled disturbances and finite simulation accuracy.

After iterative refinement, the following controller parameters were selected as a balanced configuration that ensured robustness and precision across nominal flight scenarios:

- **Position Controller (PID):**

$$K_{p_x} = 35, \quad K_{p_y} = 35, \quad K_{p_z} = 16.23$$

$$K_{d_x} = 9, \quad K_{d_y} = 9, \quad K_{d_z} = 10$$

$$K_{i_x} = 0.04, \quad K_{i_y} = 0.04, \quad K_{i_z} = 0.04$$

- **Attitude Controller:**

$$K_{R_x} = 1, \quad K_{R_y} = 1.5, \quad K_{R_z} = 0.2$$

$$K_{\omega_x} = 0.12, \quad K_{\omega_y} = 0.13, \quad K_{\omega_z} = 0.1$$

7.2.2 Validation Using a Square Trajectory

To validate the performance of the tuned controller, a trajectory tracking test was conducted in simulation using a square flight path. At each corner of the square, the drone executed a 90° yaw rotation, introducing coupled position–attitude dynamics that challenge the controller across all degrees of freedom.

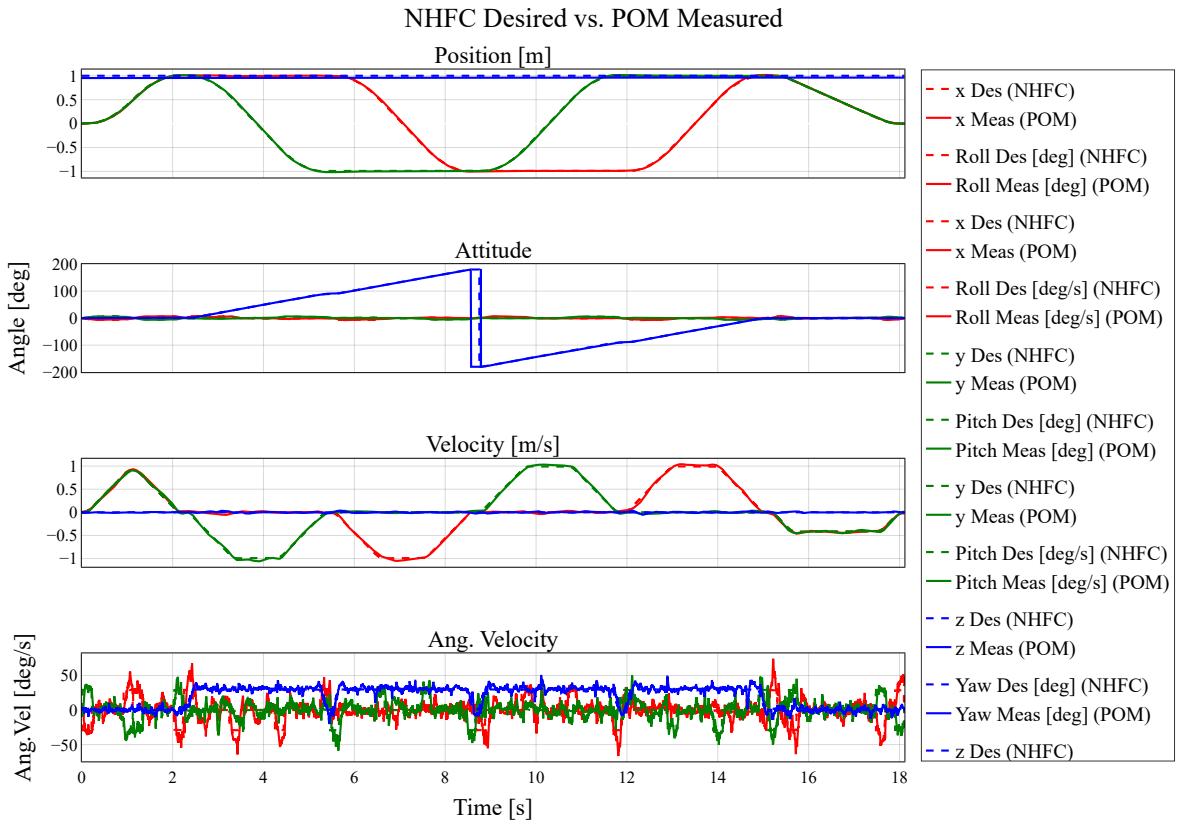


Figure 7.1: Comparison of desired vs. measured state along the square trajectory: NHFC (controller) vs. POM (state estimator).

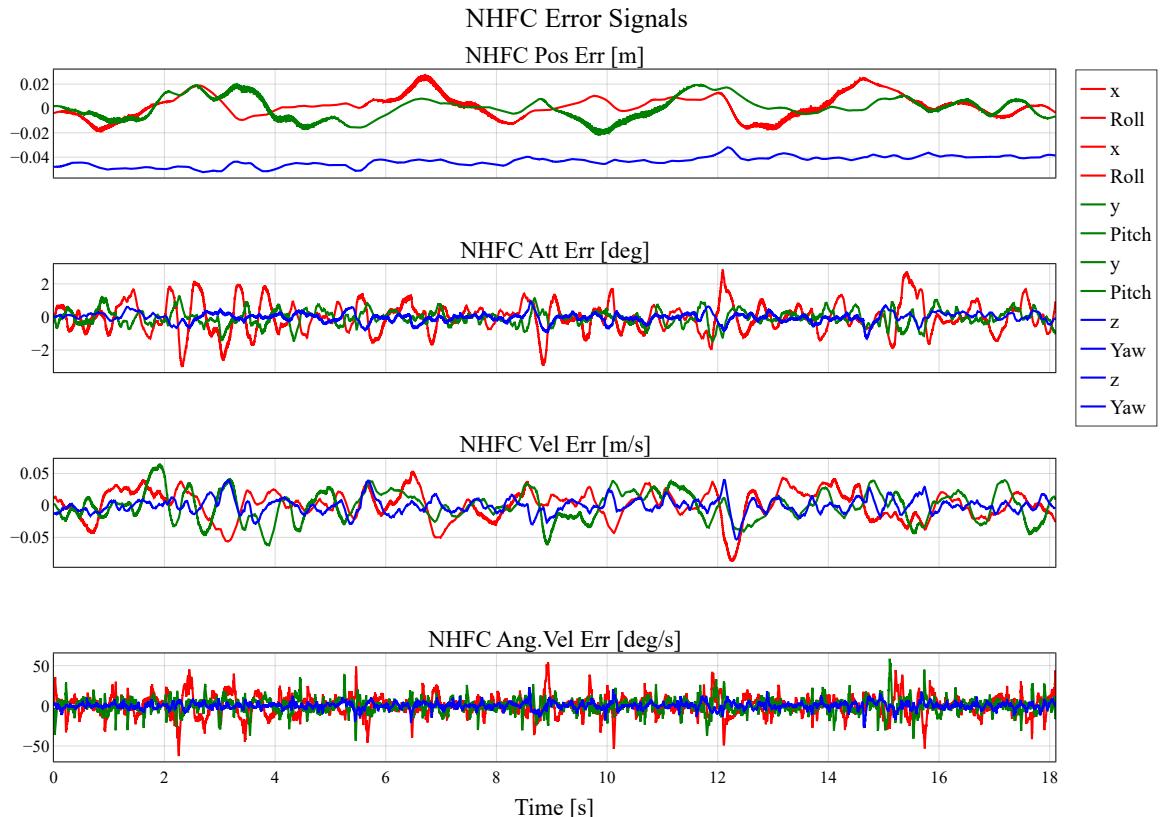


Figure 7.2: Time evolution of tracking errors for position and attitude components.

The tracking plots in Figures 7.1 and 7.2 demonstrate that the controller ensures stable flight and bounded tracking errors across the entire trajectory. Attitude transitions, particularly during yaw rotations, are completed rapidly without inducing significant positional deviation.

To evaluate the effectiveness of the controller, a set of key quantitative metrics was extracted from the square trajectory simulation. Table 7.1 summarizes the Root Mean Square Error (RMSE), mean error, and percentage overshoot for each controlled degree of freedom, encompassing both translational and rotational axes.

Axis	RMSE	Mean Error	Overshoot (%)
x	0.012 m	0.009 m	1.35%
y	0.009 m	0.007 m	1.44%
z	0.040 m	0.039 m	—
Roll (ϕ)	0.015 deg	0.012 deg	20.79%
Pitch (θ)	0.008 deg	0.006 deg	14.26%
Yaw (ψ)	0.006 deg	0.004 deg	5.54%

Table 7.1: RMSE, mean error, and percentage overshoot for position and attitude tracking during the square trajectory simulation.

The results demonstrate that the controller maintained accurate and consistent tracking performance. Positional deviations were minimal, particularly along the lateral axes (x, y), while slightly elevated errors in the vertical direction (z) are attributed to increased sensitivity to thrust variations. Attitude regulation was similarly precise, with roll and pitch errors exhibiting rapid convergence. A moderate overshoot was observed in yaw during 90° transitions, yet this effect was transient and did not compromise stability.

7.3 Real-World Validation

To assess the transferability of the simulated controller performance to physical hardware, the same square trajectory presented in Section 7.2 was executed on the real drone. This test aimed to validate the controller’s robustness under real-world conditions, including unmodeled dynamics, sensor noise, and actuation delays.

7.3.1 Performance Metrics

Table 7.2 reports the RMSE, mean tracking error, and percentage overshoot for each degree of freedom. The position is expressed in meters, while attitude angles are reported in degrees.

Axis	RMSE	Mean Error	Overshoot (%)
x	0.022 m	0.016 m	4.87%
y	0.015 m	0.011 m	3.54%
z	0.074 m	0.073 m	—
Roll (ϕ)	0.092 deg	0.070 deg	22.88%
Pitch (θ)	0.073 deg	0.052 deg	24.40%
Yaw (ψ)	0.022 deg	0.013 deg	9.96%

Table 7.2: RMSE, mean error, and percentage overshoot for position and attitude tracking during real-world square trajectory.

The tracking performance in real-world conditions closely mirrored that observed in simulation, thereby validating the robustness of the controller design. The RMSE for the x and y axes

increased by 0.010 m and 0.006 m, respectively, while the z axis exhibited a larger deviation of 0.0339 m, attributable to unmodeled vertical thrust dynamics and the absence of aerodynamic disturbances in the simulation.

Attitude tracking remained reliable in real-world conditions, with only moderate increases in error relative to simulation. The roll and pitch RMSE rose by 0.077 deg and 0.065 deg, respectively, suggesting a slight degradation due to external disturbances and hardware-induced latency. The yaw axis exhibited the most significant variation, with RMSE increasing by 0.016 deg and overshoot nearly doubling from 5.54% in simulation to 9.96% in reality. These discrepancies are consistent with the impact of inertial sensor noise, actuator delay, and minor asymmetries in real hardware during rotational maneuvers.

Despite these differences, the final position and orientation errors remained well within acceptable thresholds for indoor autonomous navigation. These findings confirm that the control architecture, when tuned in simulation, generalizes effectively to real-world conditions with minimal performance loss.

7.4 Comparative Evaluation

This section presents a comparative analysis between the legacy MKQ and the newly developed BB platform, focusing on two key aspects: the sensitivity of flight performance to geometric properties, and the influence of maximum motor thrust. The comparison is based on a standardized dynamic maneuver known as the CIRP.

7.4.1 CIRP Test: Constant Increasing Ramp Path

The CIRP test is designed to evaluate the dynamic limits of the quadrotor by applying a progressively demanding trajectory. Specifically, it allows us to assess the system's ability to maintain control authority under increasing actuation demands, thereby exposing limitations related to acceleration, tracking performance, and inertia.

The CIRP trajectory consists of repeated linear segments, each with a duration that decreases by 0.1 s per iteration. This results in an increasing excitation frequency over time, which challenges the dynamic responsiveness of the platform.

Three experiments were conducted along the x -axis, chosen as the critical direction due to the non-square geometry of BB platform, which makes it the limiting factor in directional acceleration:

- **MKQ:** Used as the baseline platform.
- **BBLP:** Operated under thrust-limited conditions to approximately match the MKQ performance envelope, isolating geometric and inertial contributions.
- **BBFP:** Executed with no thrust limitations to reveal the full capabilities of the new platform.

Figures 7.3–7.5 show the controller reference (NHFC) and measured state (POM) for the three platforms during the CIRP maneuver, in which the trajectory generated is the same for the three platforms¹.

¹<https://www.youtube.com/watch?v=tMDycno8ZVo>

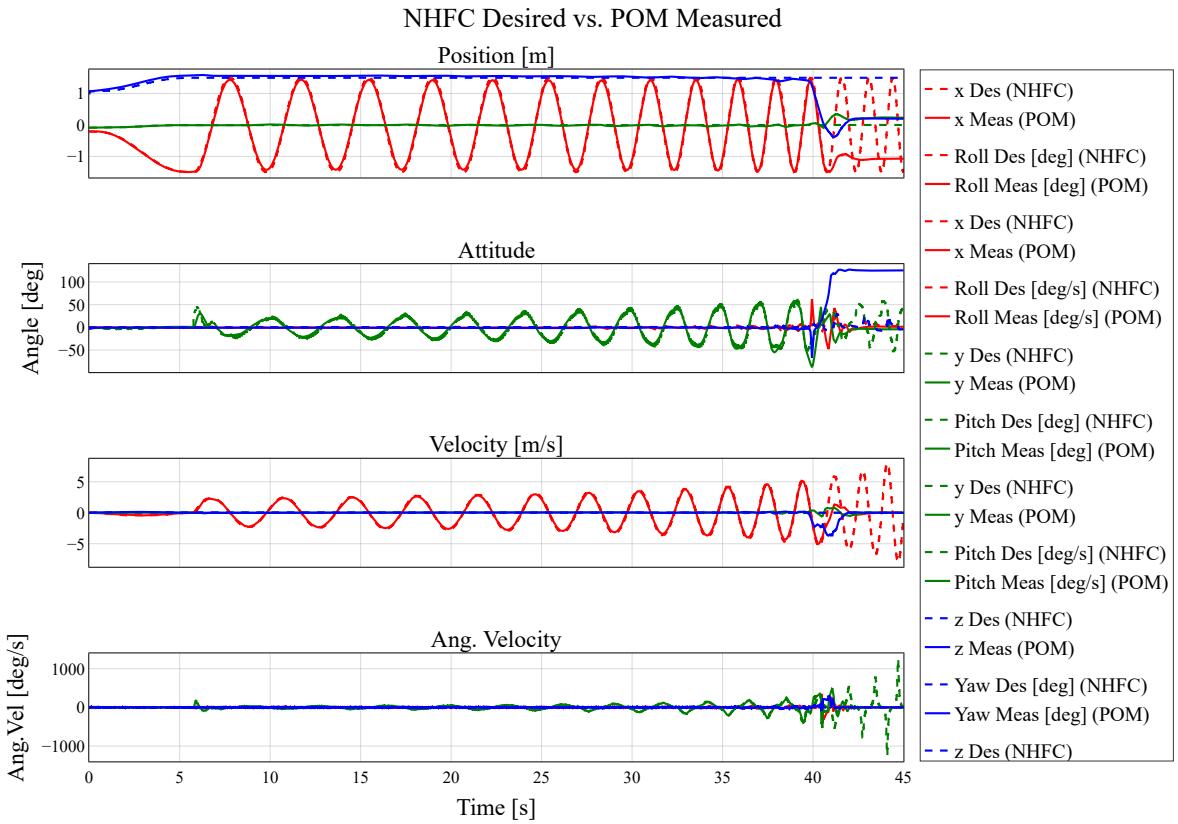


Figure 7.3: MKQuad – Comparison between controller reference (NHFC) and measured state (POM) during CIRP execution.

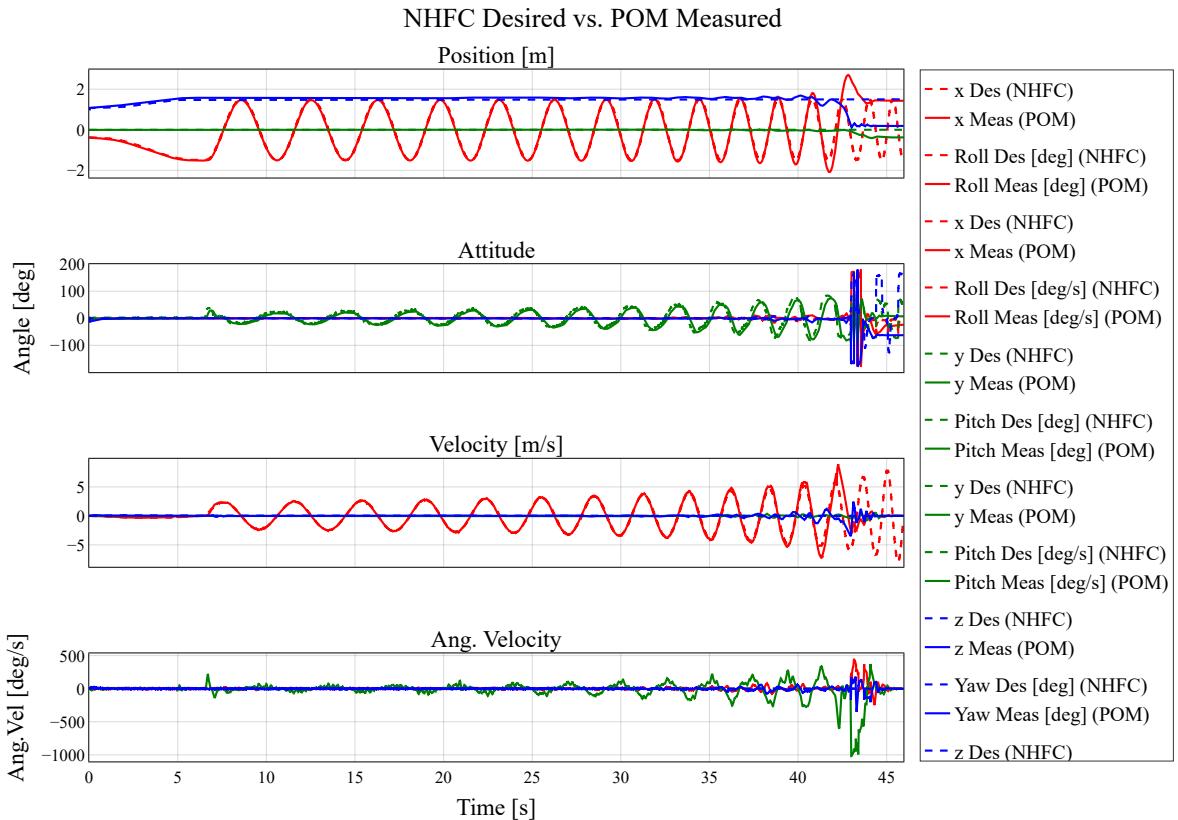


Figure 7.4: BumbleBee (Limited Power) – NHFC vs. POM state tracking during CIRP.

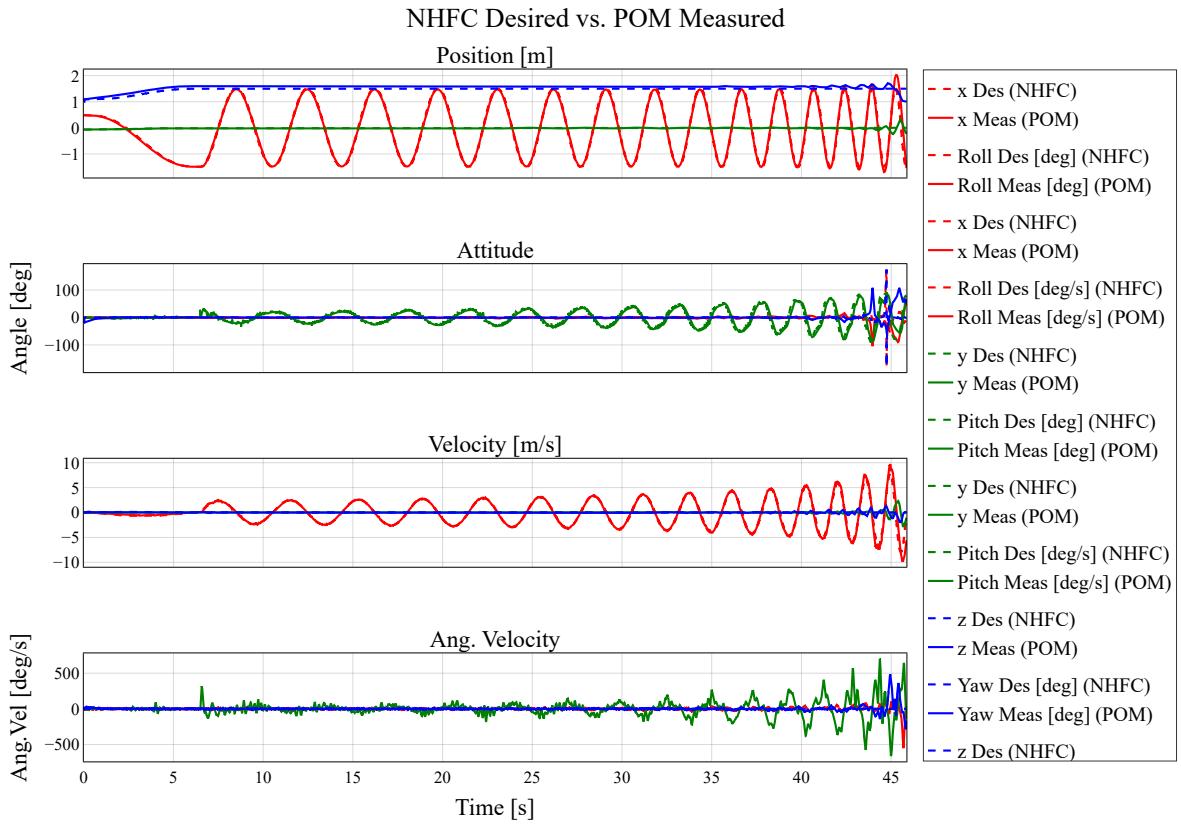


Figure 7.5: BumbleBee (Full Power) – NHFC vs. POM state tracking during CIRP.

As expected, the BBLP configuration outperformed the MKQ, sustaining the increasing-frequency trajectory for a longer time, 42.7 s compared to 40.1 s, though it ultimately failed to complete the entire sequence. In contrast, the BBFP configuration successfully completed the full trajectory, reaching 45.9 s and confirming the performance potential of the new platform when unconstrained.

7.4.2 CIRP Analysis

To further understand these results, Figure 7.6 presents the real measured linear and angular accelerations for each configuration. These are compared with the theoretical actuator envelopes previously introduced in Section 5.1.4.

Real Data: Linear vs. Angular Acceleration Capability

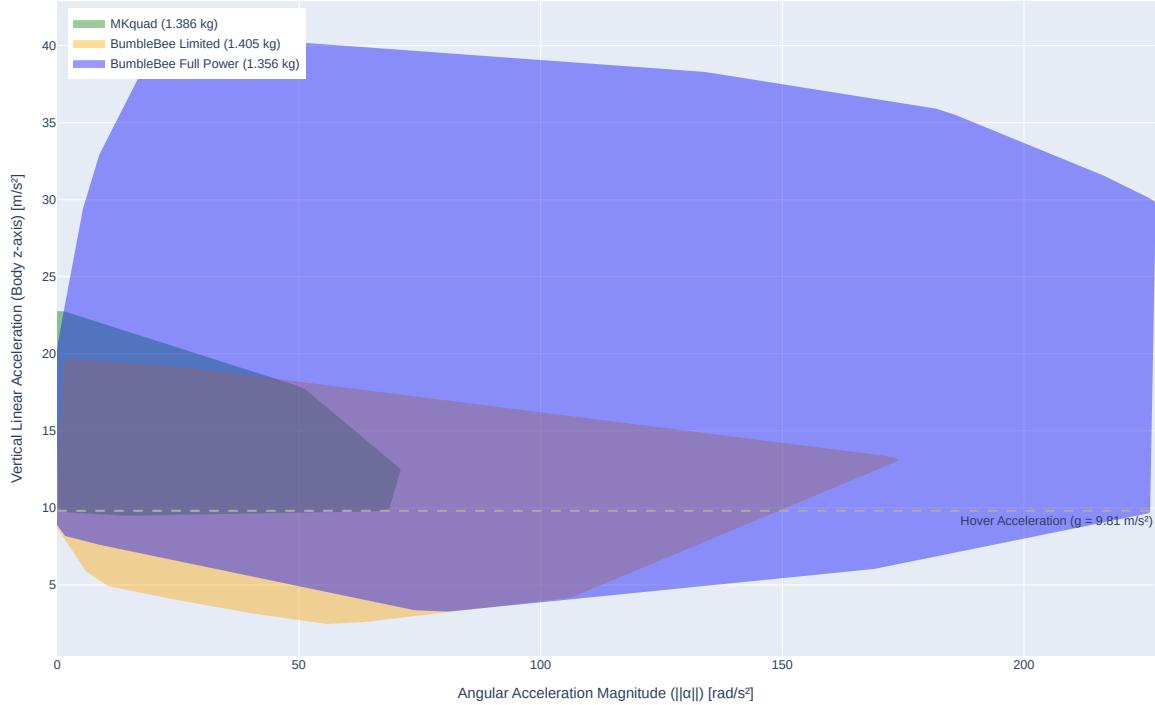


Figure 7.6: Real linear vs. angular acceleration data during CIRP for MKQuad, BB (Limited Power), and BB (Full Power).

The principal difference between the MKQ and the BBLP configurations lies in their inertial properties. The MKQ, in comparison with the BBLP platform, has:

- Mass ratio: $\frac{m_{MKQ}}{m_{BBLP}} = 0.986$
- Thrust ratio: $\frac{T_{MKQ}}{T_{BBLP}} = 1.143$
- Angular acceleration ratio: $\frac{\dot{\omega}_{MKQ}}{\dot{\omega}_{BBLP}} = 0.407$
- TWR ratio: $\frac{TWR_{MKQ}}{TWR_{BBLP}} = 1.158$
- Mean moment of inertia ratio: $\frac{J_{MKQ}}{J_{BBLP}} = 4.125$

The moment of inertia plays a critical role in determining the platform's angular acceleration, as analytically described in Section 5.1.3 through the relation $\dot{\omega} = \tau/J$. Experimental results corroborate this theoretical model: despite comparable thrust and mass across configurations, the reduced inertia of BBLP with respect to MKQ leads to a markedly increased angular responsiveness. Specifically, although the BBLP platform exhibits slightly lower torques ($\tau_{BBLP} < \tau_{MKQ}$), its significantly lower moment of inertia ($J_{BBLP} \ll J_{MKQ}$) results in a higher angular acceleration, consistent with the theoretical prediction.

Similarly, the principal difference between the BBLP and the BBFP configurations lies in their maximum motor velocity. The BBLP, in comparison with the BBFP platform, has:

- Mass ratio: $\frac{m_{BBLP}}{m_{BBFP}} = 1.036$
- Thrust ratio: $\frac{T_{BBLP}}{T_{BBFP}} = 0.498$

- Angular acceleration ratio: $\frac{\dot{\omega}_{BBLP}}{\dot{\omega}_{BBFP}} = 0.766$
- TWR ratio: $\frac{TWR_{BBLP}}{TWR_{BBFP}} = 0.481$
- Mean moment of inertia ratio: $\frac{J_{BBLP}}{J_{BBFP}} = 1.000$

In this case, motor velocity emerges as the dominant factor influencing angular acceleration, as modeled in Section 5.1.3 via both $\dot{\omega} = \tau/J$ and the actuator mapping:

$$\begin{bmatrix} f_z \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = \mathbf{A} \cdot \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix}.$$

Experimental results reinforce this prediction: although BBLP and BBFP have nearly identical inertia and similar mass, the higher maximum motor velocities of BBFP allow for a broader control authority. Specifically, the increased range of available motor speed variations ($\Delta\omega_{i_{BBLP}}^2 < \Delta\omega_{i_{BBFP}}^2$ where Δ represents the difference between maximum and minimum velocities of the $i - th$ motor) results in greater achievable torque and thus higher angular accelerations, again aligning with theoretical expectations.

While discrepancies persist between the predicted and observed angular acceleration limits, these deviations are likely attributable to uncertainties in inertia estimation and unmodeled experimental factors such as structural flexibilities, sensor delays, or aerodynamic disturbances. Nonetheless, the CIRP test effectively highlights the influence of geometric and inertial properties on the platform’s dynamic behavior. Moreover, it validates the use of aggressive reference trajectories as a reliable means for performance benchmarking and comparative evaluation across UAV configurations.

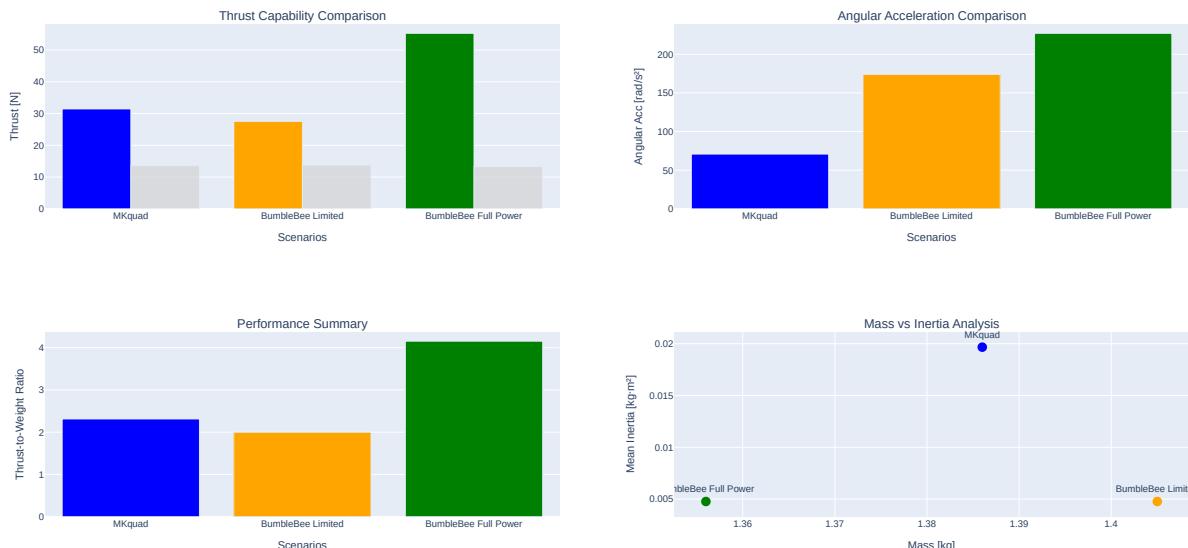


Figure 7.7: Quantitative comparison across the three configurations MKQ, BBLP, and BBFP—based on thrust capability, angular acceleration, thrust-to-weight ratio, and mean inertia.

Figure 7.7 provides a compact yet informative summary of the key performance metrics across the three UAV configurations. The first plot illustrates the thrust capability, with grey

bars indicating the respective hovering thrust required for each platform. As shown, the BBFP configuration demonstrates a substantial advantage over both BBLP and MKQ, consistent with its higher motor velocity capacity. Specifically, BBFP delivers approximately 43% more maximum thrust than BBLP, while MKQ and BBLP exhibit comparable performance levels.

The second plot compares angular acceleration: despite having similar mass and identical inertia, BBFP achieves significantly higher values than BBLP, driven by increased available torque. Compared to MKQ, both BBLP and BBFP show considerable improvements, with BBFP reaching more than twice the angular acceleration of MKQ.

The third subplot reports the TWR, reinforcing the trend: while MKQ slightly outperforms BBLP, BBFP far exceeds both, offering a TWR more than double that of its limited-power counterpart. Lastly, the inertia analysis shows that MKQ possesses a mean moment of inertia over four times greater than BBLP and BBFP, explaining its limited angular responsiveness despite strong motors.

This visual comparison solidifies the conclusions drawn throughout the analysis: improved angular dynamics can be achieved either by reducing inertia (as in BBLP vs. MKQ) or by increasing available motor torque (as in BBFP vs. BBLP). The BBFP configuration thus emerges as the most dynamically responsive platform among the three, demonstrating how tailored design choices in both structure and actuation can yield significantly enhanced performance.

7.5 Agile Flight Demonstrations

Executing agile flight maneuvers represents a significant challenge for multirotor platforms, as it requires sustaining high linear and angular accelerations. After validating the dynamic capabilities of the BB platform, we conducted experimental trials to assess its performance in executing aggressive maneuvers, specifically a powerloop and a flip (barrel roll).

In addition to demonstrating the platform's agility, these experiments served a second purpose: to verify the feasibility of performing such trajectories using the `telekyb3` framework. To the best of our knowledge, this represents the first successful execution of these maneuvers with this framework.

7.5.1 Trajectory Generation for Agile Maneuvers

Both agile maneuvers are generated using a **differential flatness**-based framework, which is consistent with the control approach described in Section 3.4. The flat outputs of the quadrotor are the position $\mathbf{p}_d = [x_d, y_d, z_d]^\top$ and the yaw angle ψ_d , while all other states (attitude \mathbf{R}_d , angular velocity $\boldsymbol{\omega}_d$, and thrust) are obtained as algebraic functions of their derivatives up to jerk.

Instead of directly commanding angular rates, we first define a smooth geometric trajectory in 3D space and compute the complete feedforward reference set:

$$\left\{ \mathbf{p}_d, \mathbf{v}_d, \mathbf{a}_d, \mathbf{j}_d, \mathbf{R}_d, \boldsymbol{\omega}_d \right\},$$

where $\mathbf{v}_d = \dot{\mathbf{p}}_d$, $\mathbf{a}_d = \ddot{\mathbf{p}}_d$, $\mathbf{j}_d = \dddot{\mathbf{p}}_d$. These references are streamed to the NHFC controller using `nhfc.set_state` at 50 Hz.

The trajectory progression along the spatial path is parameterized by a normalized time variable $s(t) \in [0, 1]$, which smoothly evolves from 0 to 1 over the total maneuver duration κ :

$$\hat{t} = \frac{t}{\kappa}, \quad s(\hat{t}) = a_0 + a_1 \hat{t} + a_2 \hat{t}^2 + a_3 \hat{t}^3 + a_4 \hat{t}^4 + a_5 \hat{t}^5. \quad (7.1)$$

To achieve a *rest-to-rest* trajectory, the polynomial must satisfy six boundary conditions that enforce zero velocity and acceleration at the start and end:

$$s(0) = 0, \quad \dot{s}(0) = 0, \quad \ddot{s}(0) = 0, \quad (7.2)$$

$$s(1) = 1, \quad \dot{s}(1) = 0, \quad \ddot{s}(1) = 0. \quad (7.3)$$

From the initial conditions, we immediately obtain:

$$a_0 = 0, \quad a_1 = 0, \quad a_2 = 0, \quad (7.4)$$

leaving the reduced form:

$$s(\hat{t}) = a_3 \hat{t}^3 + a_4 \hat{t}^4 + a_5 \hat{t}^5. \quad (7.5)$$

The final conditions generate the linear system:

$$a_3 + a_4 + a_5 = 1 \quad (7.6)$$

$$3a_3 + 4a_4 + 5a_5 = 0 \quad (7.7)$$

$$6a_3 + 12a_4 + 20a_5 = 0 \quad (7.8)$$

whose solution is

$$a_3 = 10, \quad a_4 = -15, \quad a_5 = 6. \quad (7.9)$$

Therefore, the quintic polynomial for the normalized trajectory is:

$$s(\hat{t}) = 10\hat{t}^3 - 15\hat{t}^4 + 6\hat{t}^5 \quad (7.10)$$

which ensures smooth motion with zero initial and final velocity and acceleration.

The position along the spatial path $\mathbf{p}_d(s)$ is expressed as a function of $s(t)$, so its time derivatives are obtained via the chain rule:

$$\mathbf{v}_d(t) = \frac{d\mathbf{p}_d}{ds} \dot{s}(t), \quad (7.11)$$

$$\mathbf{a}_d(t) = \frac{d^2\mathbf{p}_d}{ds^2} \dot{s}^2(t) + \frac{d\mathbf{p}_d}{ds} \ddot{s}(t), \quad (7.12)$$

$$\mathbf{j}_d(t) = \frac{d^3\mathbf{p}_d}{ds^3} \dot{s}^3(t) + 3\frac{d^2\mathbf{p}_d}{ds^2} \dot{s}(t)\ddot{s}(t) + \frac{d\mathbf{p}_d}{ds} \ddot{\ddot{s}}(t). \quad (7.13)$$

From these derivatives, the flatness mapping described in Section 3.4 yields the desired thrust direction, orientation \mathbf{R}_d , and angular velocity $\boldsymbol{\omega}_d$. This approach ensures smooth, dynamically feasible motion for highly agile maneuvers such as the powerloop and flip.

Mapping to Attitude and Angular Velocity Following the flatness-based mapping in Section 3.4:

1. The body z -axis \mathbf{z}_b is aligned with $\mathbf{a} + g\mathbf{k}_z$.
2. The full rotation matrix \mathbf{R}_d is constructed using \mathbf{z}_b and the desired yaw ψ .
3. The desired angular velocity $\boldsymbol{\omega}_d$ and acceleration $\dot{\boldsymbol{\omega}}_d$ are computed from $\dot{\mathbf{R}}_d$ and $\ddot{\mathbf{R}}_d$.

The resulting feedforward states are streamed to the NHFC controller via `nhfc.set_state` at 50 Hz, providing dynamically consistent position, velocity, acceleration, attitude, and angular velocity references. This approach allows the quadrotor to follow complex 3D paths while performing fast flips or loops without violating its dynamic limits.

7.5.2 Powerloop Maneuver

The powerloop maneuver consists of an aggressive climb followed by a complete 360° inverted dive, forming a vertical loop. In our experiment, the maneuver was executed **twice consecutively** (forward and backward) without intermediate reorientation. The final position of the first loop trajectory coincides with the starting point of the second loop, resulting in a smooth out-and-back motion.

Each individual powerloop is structured as a composite trajectory consisting of three segments:

1. **Approach line:** a straight segment from the initial hover point to the loop entry.
2. **Vertical loop:** a circular trajectory in a vertical plane aligned with the current flight direction.
3. **Departure line:** a straight segment from the loop exit to the next trajectory waypoint (which, for the first loop, serves as the entry point for the reverse loop).

Figure 7.8 illustrates a chronophotographic sequence of the full maneuver. The quadrotor starts at point **A**, performs the first forward loop, and reaches point **B**, which becomes the starting position for the return loop. This structure ensures geometric continuity and avoids the need for yaw realignment, enabling a seamless bidirectional execution.²

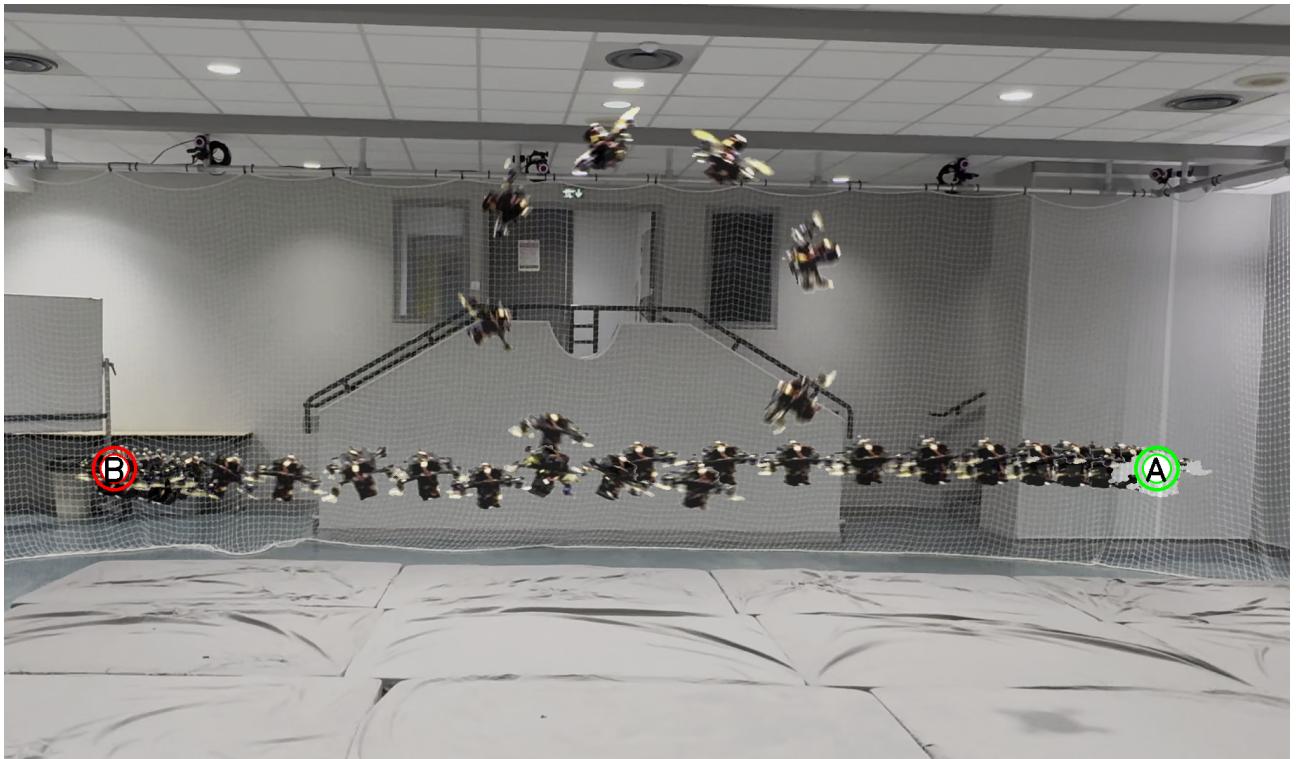


Figure 7.8: Chronophotography of the powerloop maneuver. The drone starts from point **A**, completes the first vertical loop reaching point **B**, and then performs a second loop in the opposite direction.

²https://www.youtube.com/watch?v=QgQt_DPFcEA

Experimental Results

Figure 7.9 shows the 3D trajectory of the maneuver, highlighting the agreement between the reference trajectory (NHFC Desired) and the estimated position (POM Measured). The vertical loop is clearly visible and demonstrates stable spatial tracking throughout the maneuver.

Figure 7.10 shows the evolution of position, attitude, velocity and angular velocity. The roll angle smoothly surpasses 180° , confirming a complete inversion, with controlled recovery to hover.

It is worth noting that the roll angle in the plot exhibits abrupt transitions around 6 s and 12 s. These discontinuities are not physical but are due to the standard $[-180^\circ, 180^\circ]$ Euler-angle normalization: when the true roll passes smoothly through $+180^\circ$, it is wrapped to -180° , resulting in a sudden jump in the plot.

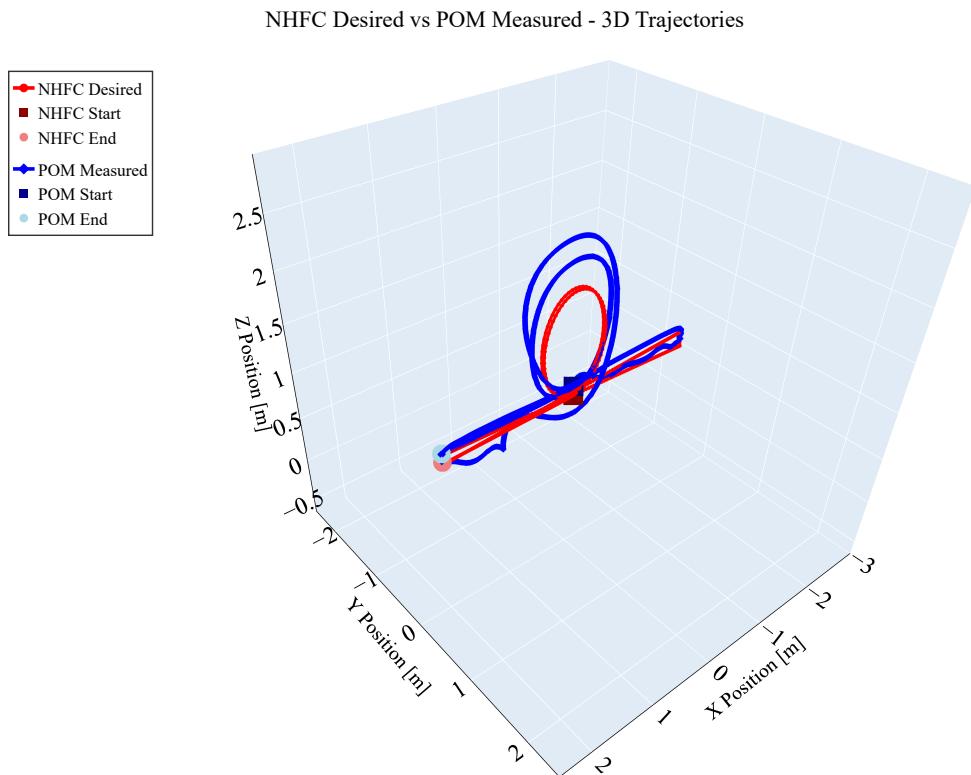


Figure 7.9: 3D trajectory of the power-loop maneuver: NHFC desired vs. POM measured.

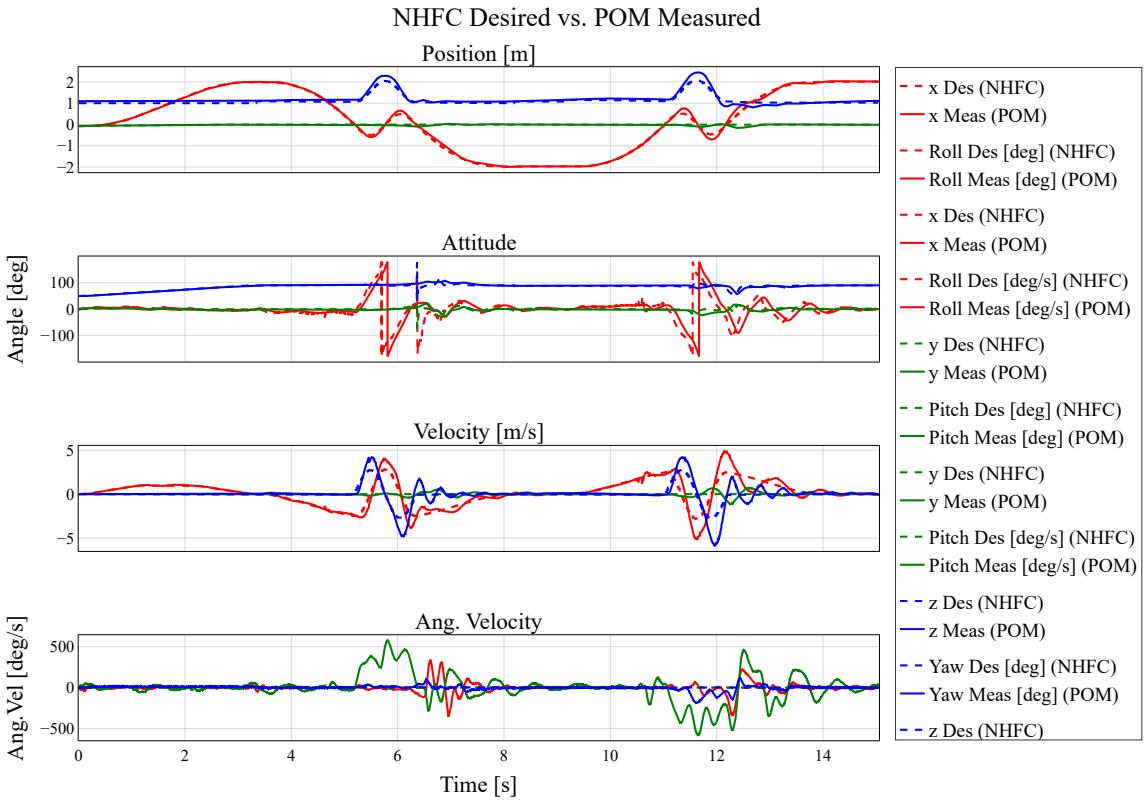


Figure 7.10: Position, attitude, and angular velocity evolution during the power-loop maneuver.

7.5.3 Flip Maneuver

The flip maneuver consists of a full 360° rotation about a single body axis (either roll or pitch). In this maneuver, the quadrotor performs a rotational flip around its own center of mass with only minimal lateral displacement. Unlike the *powerloop*, which traces a large vertical circular trajectory in space, the flip is essentially an in-place maneuver. Its primary purpose is to test the platform's ability to generate high angular velocities and recover from a temporary loss of vertical thrust, while maintaining overall position and altitude within a small spatial envelope.

Experimental Results

Figure 7.11 shows the position, attitude, and angular velocity evolution during the flip. The primary rotation axis completes a full 360° rotation while the other two Euler angles remain bounded, confirming a clean single-axis flip. Figure 7.12 illustrates the commanded and measured rotor speeds, showing the transient thrust demand at the start and end of the maneuver.

These results confirm that the BB platform, combined with the `telekyb3` + NHFC control architecture, is capable of executing smooth, dynamically feasible flips.

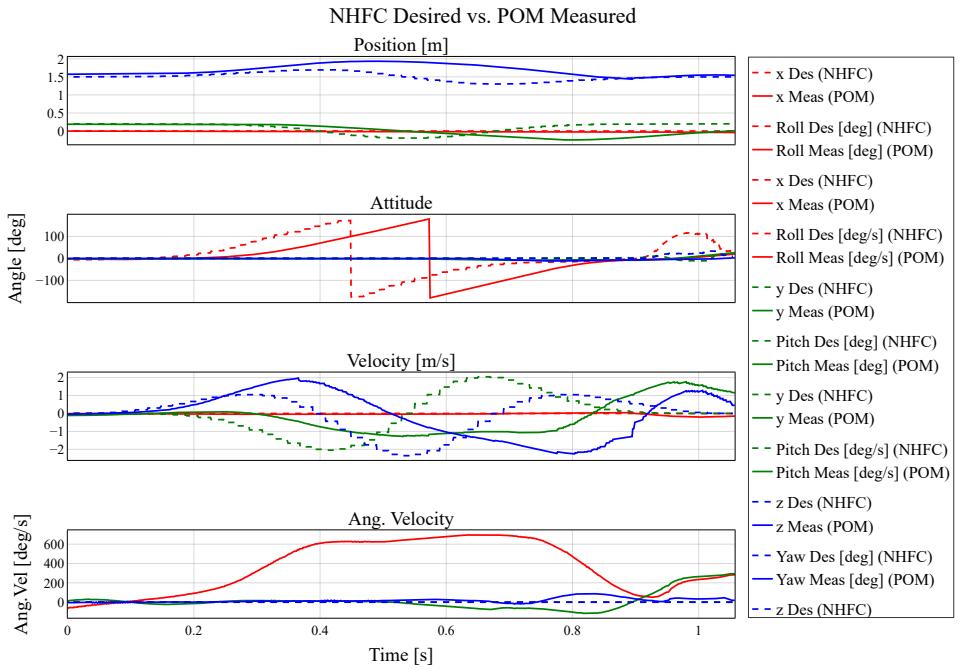


Figure 7.11: Position, attitude, and angular velocity evolution during the flip maneuver.

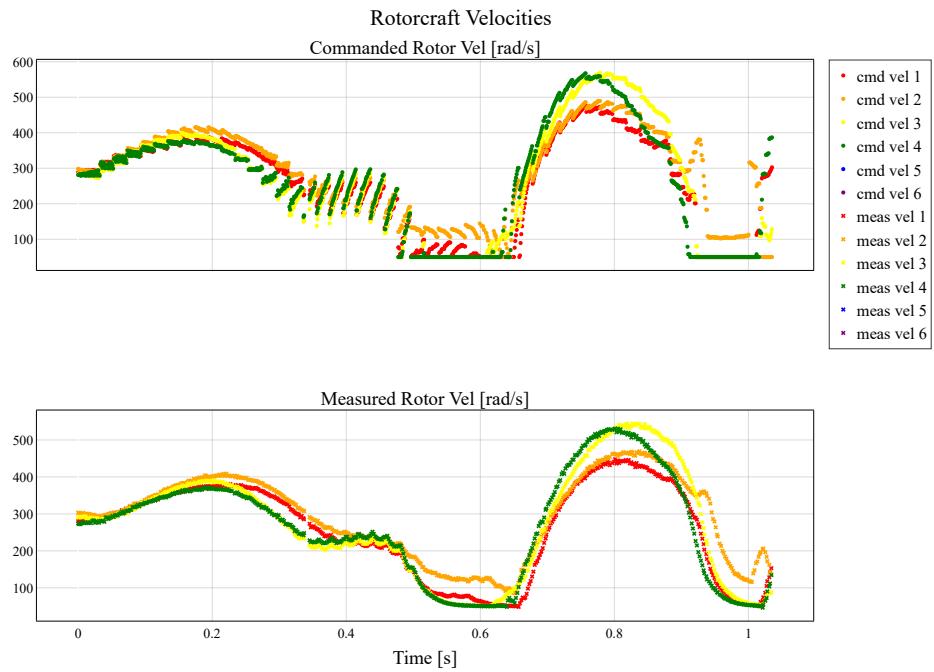


Figure 7.12: Commanded and measured rotor velocities during the flip maneuver.

CHAPTER 8

Conclusion

This thesis presented a complete design, modeling, and experimental validation pipeline for a novel quadrotor UAV platform, *BumbleBee*, optimized for agile flight and collaborative aerial transportation. Motivated by the limitations of legacy platforms, particularly the MKQuad system, the work addressed critical challenges in maneuverability, TWR, modularity, and actuation authority, providing a rigorous methodology that bridges mechatronic design, simulation, and real-world implementation.

The contributions of this thesis can be summarized as follows:

- A custom-built, hardware-in-the-loop test bench was developed to enable high-fidelity identification of thrust and torque coefficients, surpassing the accuracy of vendor datasheets. The system supported closed-loop motor PID tuning and aerodynamic parameter extraction, essential for dynamic modeling and controller synthesis.
- A UAV configurator tool was introduced to evaluate design trade-offs under realistic constraints. By integrating empirical actuator models, it allowed quantitative estimation of performance envelopes, including linear and angular accelerations, control authority, and system inertia.
- A novel quadrotor platform was designed with a compact footprint and low center of mass. Component layout was optimized via CAD-based analyses, yielding a structurally efficient and dynamically responsive frame.
- The software component implementing the flight controller within the *telekyb3* architecture was extended to support axis-asymmetric PID tuning. This enhancement enabled independent tuning of the lateral and longitudinal control gains, increasing flexibility in handling platform-specific asymmetries. Simulation-tuned parameters were successfully transferred to real-world experiments with minimal performance degradation, demonstrating robust behavior across both nominal and aggressive maneuvers.
- Experimental validation included comparative CIRP maneuvers across three platforms (MKQ, BBLP, BBFP), highlighting the importance of inertia reduction and increased motor authority. The BBFP configuration achieved over twice the angular acceleration of MKQ, confirming theoretical predictions based on torque allocation and moment of inertia.
- Finally, the platform successfully performed agile maneuvers—namely, the powerloop and 360-degree flip, generated via differential flatness-based planning. These represent a milestone in the *telekyb3* framework, demonstrating its capability for highly dynamic flight with feedforward-consistent control inputs.

In summary, this thesis delivers a rigorous and modular UAV development pipeline grounded in physical validation and real-time control. The proposed BumbleBee platform not only exceeds the performance of its predecessor but also establishes a scalable foundation for further research in agile robotics, cooperative manipulation, and high-speed navigation. The integration of accurate motor modeling, advanced control, and a compact mechanical layout marks a significant advancement toward deployable, high-performance multirotor systems for both academic and applied aerial robotics.

CHAPTER 9

Future Works

This thesis established a complete framework for the mechatronic design, dynamic modeling, and experimental validation of agile quadrotor platforms. While the developed BumbleBee system demonstrated superior performance in TWR, maneuverability, and agility, several avenues remain open for future research and improvement.

Advanced Control and Planning

Future efforts could explore the integration of more sophisticated control strategies to fully exploit the platform's high dynamic capabilities. In particular:

- **Nonlinear and predictive control:** Implementation of NMPC or adaptive control laws could improve performance in aggressive maneuvers and under external disturbances.
- **Fault-tolerant control:** Developing controllers capable of handling actuator degradation or motor failures would enhance the platform's reliability in high-risk missions.
- **Online trajectory generation:** Coupling differential flatness with real-time optimization could enable autonomous agile navigation in cluttered or dynamic environments.

Enhanced System Modeling

Although the current modeling approach achieved high fidelity for design and control, further refinements can expand its predictive accuracy:

- **Aerodynamic modeling:** Incorporating rotor-airframe interaction effects, blade flapping, and induced drag would improve high-speed flight prediction.
- **Environmental effects:** Modeling wind fields and turbulence is critical for extending the platform to outdoor and cooperative missions.
- **Flexible-body dynamics:** Including structural flexibility could better explain small deviations observed in high-frequency maneuvers and inform future lightweight designs.

Hardware Optimization and Scalability

The BumbleBee platform offers a robust foundation for laboratory research, yet hardware improvements can further increase its efficiency and applicability:

- **Lightweight structures:** Optimizing frame geometry to minimize inertia without sacrificing structural integrity will enhance angular accelerations.
- **Advanced propulsion:** Exploring variable-pitch or ducted-fan propellers could expand the thrust envelope and improve efficiency in aggressive flight.
- **Scalability to swarms:** Producing multiple identical BumbleBee platforms will enable multi-robot experiments for collaborative aerial manipulation and transportation.

Autonomous Operation and Applications

Finally, future research may focus on leveraging the platform for advanced applications:

- **Full autonomy in GPS-denied environments:** Integration of Visual-Inertial Odometry (VIO) and LiDAR-Inertial Odometry (LIO) pipelines will enable reliable indoor and outdoor navigation without relying on GNSS.
- **Collaborative aerial transportation:** Extending experiments to multi-UAV payload sharing and cooperative manipulation will address scenarios relevant to aerial construction and logistics.
- **Agile mission execution:** The platform can be applied to high-speed inspection, search-and-rescue, and human-robot collaboration tasks requiring both agility and precision.

By addressing these directions, the BumbleBee platform can evolve into a highly versatile research tool, enabling the next generation of aerial robotics studies in agile flight, physical interaction, and cooperative multi-agent systems.

Bibliography

- [1] G. Corsini, “telekyb3 – a research framework for aerial physical interaction and manipulation,” Presentation, Journée Drones 2024, 2024, accessed from internal documentation.
- [2] R. Mahony, V. Kumar, and P. Corke, “Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor,” *IEEE Robotics & Automation Magazine*, vol. 19, no. 3, p. 20–32, sept 2012.
- [3] S. Bouabdallah, P. Murrieri, and R. Siegwart, “Design and control of an indoor micro quadrotor,” in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, vol. 5, Apr. 2004, pp. 4393–4398 Vol.5. [Online]. Available: <https://ieeexplore.ieee.org/document/1302409>
- [4] D. Mellinger, N. Michael, and V. Kumar, *Trajectory Generation and Control for Precise Aggressive Maneuvers with Quadrotors*. Springer Berlin, Heidelberg, 2014.
- [5] M. Hamandi, F. Usai, Q. Sablé, N. Staub, M. Tognon, and A. Franchi, “Design of multirotor aerial vehicles: A taxonomy based on input allocation,” *The International Journal of Robotics Research*, vol. 40, no. 8–9, p. 1015–1044, Aug. 2021.
- [6] M. Ryll, H. H. Bülthoff, and P. R. Giordano, “A novel overactuated quadrotor unmanned aerial vehicle: Modeling, control, and experimental validation,” *IEEE Transactions on Control Systems Technology*, vol. 23, no. 2, p. 540–556, Mar. 2015.
- [7] G. Michieletto, M. Ryll, and A. Franchi, “Fundamental actuation properties of multirotors: Force–moment decoupling and fail-safe robustness,” *IEEE Transactions on Robotics*, vol. 34, no. 3, p. 702–715, Jun. 2018.
- [8] M. Ryll, H. H. Bülthoff, and P. R. Giordano, “Modeling and control of a quadrotor uav with tilting propellers,” in *2012 IEEE International Conference on Robotics and Automation*, May 2012, p. 4606–4613. [Online]. Available: <https://ieeexplore.ieee.org/document/6225129>
- [9] M. Kamel, S. Verling, O. Elkhatib, C. Sprecher, P. Wulkop, Z. Taylor, R. Siegwart, and I. Gilitschenski, “The voliro omniorientational hexacopter: An agile and maneuverable tilttable-rotor aerial vehicle,” *IEEE Robotics & Automation Magazine*, vol. 25, no. 4, p. 34–44, Dec. 2018.
- [10] D. Brescianini and R. D’Andrea, “Design, modeling and control of an omnidirectional aerial vehicle,” in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, p. 3261–3266. [Online]. Available: <https://ieeexplore.ieee.org/document/7487497>

- [11] P. Foehn, E. Kaufmann, A. Romero, R. Penicka, S. Sun, L. Bauersfeld, T. Laengle, G. Cioffi, Y. Song, A. Loquercio, and D. Scaramuzza, “Agilicious: Open-source and open-hardware agile quadrotor for vision-based flight,” *Science Robotics*, vol. 7, no. 67, p. eabl6259, Jun. 2022.
- [12] K. Mohta, M. Watterson, Y. Mulgaonkar, S. Liu, C. Qu, A. Makineni, K. Saulnier, K. Sun, A. Zhu, J. Delmerico, K. Karydis, N. Atanasov, G. Loianno, D. Scaramuzza, K. Daniilidis, C. J. Taylor, and V. Kumar, “Fast, autonomous flight in gps-denied and cluttered environments,” *Journal of Field Robotics*, vol. 35, no. 1, p. 101–120, 2018.
- [13] T. Baca, M. Petrlik, M. Vrba, V. Spurny, R. Penicka, D. Hert, and M. Saska, “The mrs uav system: Pushing the frontiers of reproducible research, real-world deployment, and education with autonomous unmanned aerial vehicles,” *Journal of Intelligent & Robotic Systems*, vol. 102, no. 1, p. 26, Apr. 2021.
- [14] J. Martí-Saumell, H. Duarte, P. Gorsch, J. Andrade-Cetto, A. Santamaria-Navarro, and J. Solà, “Borinot: an open thrust-torque-controlled robot for research on agile aerial-contact motion,” *arXiv preprint*, no. arXiv:2307.14686, Jul. 2023, arXiv:2307.14686 [cs]. [Online]. Available: <http://arxiv.org/abs/2307.14686>
- [15] S. Sun, X. Wang, D. Sanalitro, A. Franchi, M. Tognon, and J. Alonso-Mora, “Agile and cooperative aerial manipulation of a cable-suspended load,” *arXiv preprint*, no. arXiv:2501.18802, Jan. 2025, arXiv:2501.18802 [cs]. [Online]. Available: <http://arxiv.org/abs/2501.18802>
- [16] Y. Ren, F. Zhu, G. Lu, Y. Cai, L. Yin, F. Kong, J. Lin, N. Chen, and F. Zhang, “Safety-assured high-speed navigation for mavs,” *Science Robotics*, vol. 10, no. 98, p. eado6187, 2025. [Online]. Available: <https://www.science.org/doi/abs/10.1126/scirobotics.ado6187>
- [17] P. Liu, C. Feng, Y. Xu, Y. Ning, H. Xu, and S. Shen, “Omninxt: A fully open-source and compact aerial robot with omnidirectional visual perception,” *arXiv preprint*, no. arXiv:2403.20085, Mar. 2024, arXiv:2403.20085 [cs]. [Online]. Available: <http://arxiv.org/abs/2403.20085>
- [18] S. Oğuz, M. K. Heinrich, M. Allwright, W. Zhu, M. Wahby, E. Garone, and M. Dorigo, “An open-source uav platform for swarm robotics research: Using cooperative sensor fusion for inter-robot tracking,” *IEEE Access*, vol. 12, p. 43378–43395, 2024.
- [19] I. Sa, M. Kamel, M. Burri, M. Bloesch, R. Khanna, M. Popović, J. Nieto, and R. Siegwart, “Build your own visual-inertial drone: A cost-effective and open-source autonomous drone,” *IEEE Robotics & Automation Magazine*, vol. 25, no. 1, p. 89–103, Mar. 2018.
- [20] M. Jacquet and A. Franchi, “Motor and perception constrained nmpc for torque-controlled generic aerial vehicles,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, p. 518–525, Apr. 2021.
- [21] Oscar, “Building a \$150 7-inch fpv drone for long range - cheapest 7” quad in 2024?” August 2024. [Online]. Available: <https://oscarliang.com/150-dollar-7inch-fpv-drone/>
- [22] A. Saba Afshaan, G. Priyanka, N. S. Manoj, N. Venu, L. Ansar Hussain, and H. G. Rajashekharareddy, “Design and analysis of uav test bench for engine/motor characterization,” *ACS Journal for Science and Engineering*, vol. 3, no. 1, p. 21–31, Mar. 2023.

- [23] A. Gong, R. MacNeill, and D. Verstraete, *Performance Testing and Modeling of a Brushless DC Motor, Electronic Speed Controller and Propeller for a Small UAV Application*, ser. AIAA Propulsion and Energy Forum. American Institute of Aeronautics and Astronautics, Jul. 2018. [Online]. Available: <https://arc.aiaa.org/doi/10.2514/6.2018-4584>
- [24] Z. S. Islami and F. Hartono, “Development of small propeller test bench system,” *IOP Conference Series: Materials Science and Engineering*, vol. 645, no. 1, p. 012017, Oct. 2019.
- [25] P. Piljek, D. Kotarski, and M. Krznar, “Method for characterization of a multirotor uav electric propulsion system,” *Applied Sciences*, vol. 10, no. 2222, p. 8229, Jan. 2020.
- [26] M. Jdiobe, K. Rouser, R. Paul, and A. Rouser, “Validation of a wind tunnel propeller dynamometer for group 2 unmanned aircraft,” *Applied Sciences*, vol. 12, no. 1717, p. 8908, Jan. 2022.
- [27] D. Bicego, J. Mazzetto, R. Carli, M. Farina, and A. Franchi, “Nonlinear model predictive control with enhanced actuator model for multi-rotor aerial vehicles with generic designs,” *Journal of Intelligent & Robotic Systems*, vol. 100, no. 3, p. 1213–1247, Dec. 2020.
- [28] M. Faessler, A. Franchi, and D. Scaramuzza, “Differential flatness of quadrotor dynamics subject to rotor drag for accurate, high - speed trajectory tracking,” *IEEE Robot . Autom . Lett .*, vol. 3, no. 2, pp. 620–626, Apr. 2018.
- [29] T. Lee, M. Leok, and N. H. McClamroch, “Geometric tracking control of a quadrotor uav on $\text{se}(3)$,” in *49th IEEE Conference on Decision and Control (CDC)*, Dec. 2010, p. 5420–5425. [Online]. Available: <https://ieeexplore.ieee.org/document/5717652>
- [30] D. Mellinger and V. Kumar, “Minimum snap trajectory generation and control for quadrotors,” in *2011 IEEE International Conference on Robotics and Automation*, May 2011, p. 2520–2525. [Online]. Available: <https://ieeexplore.ieee.org/document/5980409>
- [31] A. D. M. Africa, J. O. Q. Chua, and J. L. H. Solis, “Pid tuning of speed controller using ziegler-nichols and manual method dc motor,” in *2023 IEEE 15th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, Nov. 2023, p. 1–6. [Online]. Available: <https://ieeexplore.ieee.org/document/10589041>
- [32] L. M. Ariffin, A. H. Rostam, and W. M. E. Shibani, “Study of aircraft thrust-to-weight ratio,” *Journal of Aviation and Aerospace Technology*, vol. 1, no. 22, 2019. [Online]. Available: <https://jaat.fazpublishing.com/index.php/jaat/article/view/10>
- [33] G. Corsini, M. Jacquet, H. Das, A. Afifi, D. Sidobre, and A. Franchi, “Nonlinear model predictive control for human-robot handover with application to the aerial case,” in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2022, p. 7597–7604. [Online]. Available: <https://ieeexplore.ieee.org/document/9981045>
- [34] LAAS-CNRS, “telekyb3 Project Page,” <https://git.openrobots.org/projects/telekyb3>, 2024, accessed: 2025-07-24.