Doctoral Dissertation

TOWARD COLLISION-ACCOMMODATED DRONE WITH TOMBO PROPELLERS: MODELING, SENSING, AND CONTROL

Pham Tien Hung

Supervisor **Ho Anh Van**

Graduate School of Advanced Science and Technology

Japan Advanced Institute of Science and Technology

Material Science

September 2025

Abstract

The growing demand for vertical take-off and landing (VTOL) aerial vehicles, including unmanned aerial vehicle (UAV), emphasizes the importance of ensuring operational safety and the ability to accommodate collisions during flight. Such vehicles often operate in environments where contact with surrounding obstacles is likely, posing a significant risk of damage that can compromise mission success. To enhance safety in the event of collisions—particularly those involving the propellers—prior studies have explored the use of soft materials in propeller design to absorb impact forces and prevent mechanical failure, allowing the drone to remain airworthy after impact. One such development is the 9-inch deformable propeller (Tombo propeller), designed in our laboratory, which combines a rigid base with soft silicone tips to provide structural flexibility. Despite these advances, no unified control framework has been proposed to detect random collisions at the propeller level and respond accordingly. This thesis addresses this gap by integrating rotary encoders into each motor to enable real-time measurement of propeller speed, allowing for reliable collision detection and the activation of a reactive flight mode to avoid the obstacle and continue the predefined trajectory. Furthermore, while Tombo propellers offer mechanical robustness, their inherent deformability introduces flight instabilities due to shape fluctuations during sudden acceleration, deceleration, or minor impacts. These disturbances can significantly affect flight dynamics and trajectory tracking. To address this issue, the thesis implements the $\mathcal{L}_1$Quad control strategy, which enhances geometric control with $\mathcal{L}_1$ adaptive augmentation, enabling rapid compensation for modeling uncertainties and external disturbances. This ensures that the UAV can maintain stability and recover its planned path even after unforeseen collisions. Together, this work presents a comprehensive solution that integrates novel sensing and

control strategies for collision detection and recovery, representing a significant step toward realizing a UAV capable of safe and reliable operation in complex and dynamic environments.

Keywords: Quadrotor collision detection, deformable propeller, collision reaction strategy, drone safety, $\mathcal{L}_1$ adaptive control.

Acknowledgment

First, I would like to express my deepest gratitude and sincere appreciation to my supervisor *Prof. Ho Anh Van*, for his unwavering support throughout my doctoral journey. When I first joined the Ho Lab in 2022, I lacked confidence, skills, and knowledge in robotics research. However, with his passion, profound expertise, and dedication, Prof. Ho has always been there for me: listening patiently, offering encouragement, and providing invaluable advice for both my studies and research. The knowledge and experiences I have gained under his guidance are immense and will undoubtedly shape my future career as a researcher.

I am also very grateful to my second supervisor *Prof. Yonghoon Ji*, for his attentive listening, patience, and guidance to keep my research on the right track. In addition, I sincerely appreciate the insightful advice, support and mentoring of *Prof. Nguyen Le Minh* and *Prof. Naira Hovakimyan*, whose contributions were instrumental in the success of my minor research project.

I would like to extend my heartfelt thanks to my lab mates, particularly *Dr. Nguyen Huu Nhan*, *Dr. Nguyen Quang Dinh*, *Dr. Bui Tien Son*, *Dr. Luu Khanh Quan*, and *Mr. Pham Ngoc Quang*, whose support and collaboration were crucial to completing this dissertation. I also wish to acknowledge the assistance of *Dr. Sheng Cheng*, *Mr. Chengyu Yang*, and *Mr. Ziyin Han*, who provided invaluable help during my minor research off-campus at the University of Illinois at Urbana-Champaign.

To my beloved wife *Mrs. Nguyen Thi Kim Ly*, I owe an immense debt of gratitude and a heartfelt apology for the sacrifices you have made. For more than three years, you have single-handedly taken care of our children and family while I was away, all while being a constant source of companionship and support through the challenges I faced during this journey. To my little angels, *Dang Khoi* and *Ha*

My, I promise to do my best to make up for the time I have been absent once I return home.

Last but certainly not least, I wish to express my deep gratitude to my father, *Mr. Pham Dinh Anh*, and my mother, *Mrs. Nguyen Thi Thanh*, for their unconditional love and unwavering support. I fully understand the silent sacrifices they have made and their steadfast companionship throughout this journey. I wish them eternal health and happiness.

Contents

Abstract I

Acknowledgment III

Contents V

List of Figures IX

List of Tables XV

List of Abbreviations XV

Chapter 1 Introduction 1

1.1 Trends and challenges in UAV safety 1

1.2 Research problems 2

1.2.1 Existing Approaches to UAV Collision Safety 2

1.2.2 Progress and Limitations in Our Prior Works 3

1.2.3 Research Objectives 6

1.2.4 Contributions 7

1.3 Publications 9

1.4 Structure of thesis 10

Chapter 2 Background and related work 13

2.1 Collision Avoidance strategies 13

2.2 Advanced Control for Collision Resilience 17

2.3 Sensor-Based Collision Detection 22

Chapter 3 Sensorless Collision Handling for Tombo Quadrotor 27

3.1 Quadrotor Specifications 27

3.1.1 Quadrotor Hardware 27

3.1.2 Measurement of the moment of inertia of quadrotor 29

3.1.3 Thrust and moment measurement of propeller and motor . . 30

3.2 Quadrotor dynamics and $\mathcal{L}_1$Quad controller 32

3.2.1 Quadrotor dynamics 32

3.3 $\mathcal{L}_1$Quad architecture 35

3.3.1 Geometric tracking control 35

3.3.2 Uncertainty modelling 36

3.3.3 Uncertainty and disturbance analysis for a Tombo quadrotor 38

3.3.4 $\mathcal{L}_1$ adaptive augmentation 41

3.4 Experimental results 47

3.4.1 Trajectory tracking with $\mathcal{L}_1$Quad in collision-free flights . . . 48

3.4.2 Collision with a foam bar 52

3.4.3 Collision with a carbon fiber bar 55

3.4.4 Severe collision during hover flight 56

3.5 Summary and Discussion 57

Chapter 4 Sensor-Based Collision Handling for Tombo Quadrotor 59

4.1 Collision dynamics of motorized deformable propellers 60

4.1.1 System Modeling 60

4.1.2 Experimental Validation of Collision Dynamics 74

4.2 Encoder-Based Collision Detection System 79

4.2.1 System Overview 79

4.2.2 Quadrotor Hardware Setup 79

4.2.3 Propeller Speed Variation During Collision 81

4.2.4 Propeller speed variation during collision in real flight 82

4.2.5 Encoder-Based Collision Detection Method 84

4.3 Collision reaction strategy 88
4.3.1 Collision reaction strategy during hovering 89
4.3.2 Collision reaction strategy along the planned trajectory. . . . 92
4.4 Summary and Discussion 97

Chapter 5 Benchmarking Framework for Collision Recovery Evaluation **103**

5.1 Motivation and Purpose 103
5.2 Benchmarking Experimental Setups 104
5.3 Performance Metrics for Collision Recovery Evaluation 107
5.4 Summary and Future Utility 110

Chapter 6 Conclusion **113**

6.1 Discussion 113
6.1.1 Sensorless Collision Handling for Tombo Quadrotor 113
6.1.2 Sensor-Based Collision Handling for Tombo Quadrotor . . . 114
6.2 Conclusions 115
6.3 Future Works 117

References **121**

List of Figures

1.1 Numerous fields have leveraged drones to enhance human well-being. 1

1.2 The hardware designs that protect drones and mitigate the impact during collisions. 2

1.3 The propellers are made of soft material to protect the drone when a collision occurs at the propeller. 3

1.4 The Tombo propeller deforms upon collision. 4

1.5 Drone recovers after collision when drone moves to position of collision that we set up at specific point. 5

1.6 Enhancing fast recovery from brief collisions by combining the Tombo quadrotor hardware with the intelligent $\mathcal{L}_1$Quad control law. 6

1.7 Integrating encoders onto the motors to measure propeller speed for detecting collisions occurring on the propellers. 7

2.1 Vision-based sensors integrated on drones for obstacle avoidance. . . 15

2.2 Other sensors mounted on drones for obstacle avoidance. 16

2.3 A quadrotor unmanned aerial vehicle (UAV) with a damaged propeller operating near irregular ground surface; the box is placed to simulate a sudden change in ground elevation. 17

2.4 Controlling a quadrotor using the $\mathcal{L}_1$Quad strategy to track a geometric flight trajectory under complex flight conditions. 18

2.5 The quadrotor collides with a soft object to collect data from the drone's inertial measurement unit (inertial measurement unit (IMU)) for collision classification. 23

2.6 Quadrotor with foldable arms and optical flow sensor for collision detection and post-impact recovery. 24

3.1 The Tombo quadrotor. 28

3.2 Experimental setup of the bifilar pendulum for J_{zz} moment of inertia estimation. 29

3.3 Experimental setup for measuring propeller thrust force and moment as a function of pulse width modulation (PWM) percentage from the Ardupilot controller. 31

3.4 The $\mathcal{L}_1$Quad architecture used with the Tombo quadrotor. 33

3.5 The body-fixed frame and inertial frame. 34

3.6 The $\mathcal{L}_1$Quad architecture used with the Tombo quadrotor. 41

3.7 Flight experiment setup for the Tombo quadrotor: The quadrotor's position is tracked using Vicon cameras and processed by a ground computer, which transmits the data to the onboard Pixhawk 6X flight controller via the micro air vehicle proxy (MAVProxy) protocol. The Pixhawk 6X then computes the position error between the desired and actual locations and generates the corresponding PWM percentage commands to the motors. 48

3.8 Comparison of position tracking error and motor's PWM percentage between $\mathcal{L}_1$ on and off when the quadrotor is equipped with Tombo propellers, follows a circular trajectory. The switching time from $\mathcal{L}_1$ off to $\mathcal{L}_1$ on was set as 0 s. 49

3.9 Comparison of position tracking error and motor's PWM percentage between $\mathcal{L}_1$ on and off when the quadrotor is equipped with Tombo propellers, follows a lemniscate trajectory. The switching time from $\mathcal{L}_1$ off to $\mathcal{L}_1$ on was set as 0 s. 50

3.10 Illustration of the collision experiments. 51

3.11 Altitude error before and after collision with the foam bar. The collision point is marked at 0 s on the graph. The "pre-collision mean error" refers to the average positional error before the collision. The "full recovery threshold" denotes the threshold for "Full recovery," which is offset by a distance of 0.01 m (0.02 m) for $\mathcal{L}_1$ on (off) from the pre-collision mean error. 52

3.12 Statistics of recovery time and maximum deviation distance after collisions with (a) a foam bar and (b) a carbon fiber bar. The "×" symbol represents the mean value, and "I" represents the standard deviation for each case. 53

3.13 The severe collision setup uses a carbon fiber rod that intrudes into the propeller's path. 56

3.14 Quadrotor and propellers after severe collision. The rigid are broken in the area marked by the blue area. 56

3.15 Trajectory of the quadrotor in a severe collision with Tombo and rigid propellers. (a) The Tombo propeller withstands the second collision (among three collisions), where the steady state after the collision is defined as the tracking error $\leq$0.1 m after the collision. (b) The quadrotor with a rigid propeller crashes into the ground after the second collision. 58

4.1 The deformation of the propeller when collisions happen with top view. The propeller folds into three segments at the locations of the nodus, similar to a three-link mechanism with the hubs acting as joints. 61

4.2 Rotor motor's moment of inertia about the rotational axis (SolidWorks) 61

4.3 Timoshenko beam theory. 63

4.4 Cosserat rod model for continuous deformation analysis. 63

4.5 Illustration of a fluid-structure interaction modeling approach using FEM for a fully deformable PET propeller. 64
4.6 Modeling of deformable propeller 65
4.7 Inertia moment J_1 and haft-length l_1 of link hub (SolidWorks) . . . 65
4.8 Mass m_2 and inertia moment J_2 of link wing (SolidWorks) 66
4.9 The experiment setup for stiffness of joint k. 66
4.10 Contact between propeller and external object. In which, the effective mass (m_{eff}) represents mass of propeller in impact model. 67
4.11 Maxwell contact force model. 68
4.12 Modeling of brushless direct current (BLDC) motor-deformable propeller system. 72
4.13 Devices for collision experiment. 75
4.14 Deformation of the propeller during collisions at different initial rotational speeds, shown over time (ms). In scenarios 3, 4, and 5, the propeller experiences two successive impacts on opposite sides. . 76
4.15 Experimental results at 1500 revolutions per minute (rpm): a) Impact force measurement, b) Angular velocity of the propeller (link 1). A single collision event is observed in this scenario. 77
4.16 Experimental results at 3000 rpm: a) Measured collision force, b) Angular velocity of the propeller (link 1). Two successive impacts are observed on opposite sides of the propeller. 78
4.17 The system architecture designed for conducting the experiments. 80
4.18 Illustration of the encoder pulse counting cycle during collision. . . 81
4.19 Simulation results showing changes in Tombo propeller velocity due to collisions occurring at various points within the velocity measurement interval. 82
4.20 The real-flight collision experiment is performed by freely dropping an impact object into the propeller's rotation zone while the quadrotor is flight. 83

4.21 Speed variation of the Tombo propeller during collisions occurring on that propeller while the drone is in flight. 84

4.22 The speed changes in non-colliding propellers when a collision occurs. 87

4.23 The flight experiment area utilizes the OptiTrack motion capture system to determine and record the quadrotor's position. 88

4.24 Control strategy for drone hovering response when a propeller collision occurs. 90

4.25 Collision experiment during drone hovering. 90

4.26 Experimental results when Tombo quadrotor hovers with random collisions on the propeller. 91

4.27 Collision reaction strategy of the Tombo quadrotor while moving to the target point. The quadrotor's avoidance trajectory depends on the position of the impacted propeller. The numbers in each frame illustrate the position of the quadrotor's impacted propeller. . 93

4.28 The quadrotor fitted with rigid propellers crashed following a collision with the obstacle, indicated by the blue shaded region. Additionally, part of Propeller 1 was damaged, highlighted in the red area. 95

4.29 Experimental results of collision detection and response for a quadrotor equipped with rigid propellers while flying toward the target point. 98

4.30 Experimental collision and collision reaction result when Tombo quadrotor is en route to the target point. 99

4.31 Collision response results in the case of a collision at propeller 2 during the journey to the target point. 100

4.32 Collision response results in the case of a collision at propeller 4 during the journey to the target point. 101

5.1 Standardized experimental setup for drone collisions with soft (foam bar) and rigid (carbon fiber bar) objects along predefined trajectories. The foam bar and carbon fiber bar setups align with those validated in the experiments in Sections 3.4.2 and 3.4.3. 105

5.2 Illustration of Drone Trajectory Deviation During Collision. Point **C** indicates the collision location, while point **R** marks the recovery point. The green line represents the maximum deviation from the reference trajectory. 109

6.1 Flight testing of the Tombo quadrotor using the $\mathcal{L}_1$Quad controller and the quadrotor's collision detection and response strategy while navigating to a target destination in an outdoor environment. . . . 118

List of Tables

3.1 Experimentally measured moments of inertia of the quadrotor with respect to the body-fixed rotational axes. 30

3.2 The thrust force and moment generated by varying the throttle rate with a 4-cell lithium polymer (LiPo) battery were experimentally measured. 32

3.3 The values of the thrust function parameters were obtained through fitting. 32

3.4 Parameters of system and controller 42

3.5 Tracking root-mean-squared error (RMSE) of the quadrotor with $\mathcal{L}_1$ on and off in collision-free flight. Unit: [m] 48

4.1 Parameters of the motor 62

4.2 Parameters of propeller 65

4.3 Behavior characteristics of deformable during collision process . . . 102

List of Abbreviations

BLDC	brushless direct current
CoM	center of mass
DC	direct current
DIP	dual in-line package
DNN	deep neural network
DOBC	Disturbance observer-based control
ESC	electronic speed controllers
FCU	flight control unit
IMU	inertial measurement unit
INDI	incremental nonlinear dynamic inversion
LIDAR	Light detection and ranging
LiPo	lithium polymer
LPF	low-pass filter
LQR	linear quadratic regulator
MAV	micro aerial vehicles
MAVProxy	micro air vehicle proxy
MAVROS	micro air vehicle link router for ROS
Mocap	motion capture
MPC	model predictive control
PET	polyethylene terephthalate
PID	proportional integral derivative
POC	point of collision
PWM	pulse width modulation

RMSE	root-mean-squared error
ROS	robot operating system
rpm	revolutions per minute
SE(3)	special euclidean group
SLAM	simultaneous localization and mapping
SO(3)	special orthogonal group
TPU	thermoplastic polyurethane
UART	universal asynchronous receiver/transmitter
UAV	unmanned aerial vehicle
USB	universal serial bus
VIO	visual-inertial odometry
VTOL	vertical take-off and landing

Chapter 1

Introduction

1.1 Trends and challenges in UAV safety

Figure 1.1: Numerous fields have leveraged drones to enhance human well-being.

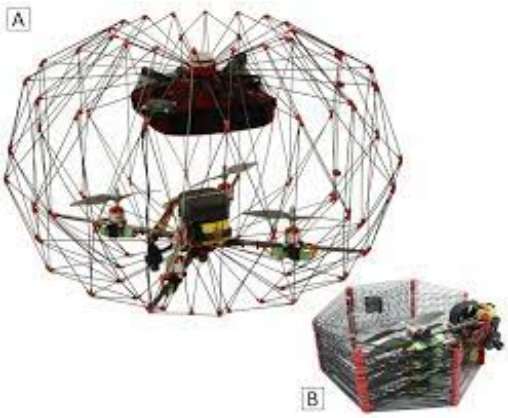
The global commercial UAV market has experienced rapid growth. According to 2022 data from [1], this market is substantial and continues to expand. The market share distribution among the three main categories of drones is as follows: fixed-wing drones account for 20.41 %, rotary-wing drones (multirotors) dominate with 79.30 %, and hybrid drones represent 10.29 %. This expansion trend is expected to be accompanied by an increasing number of active units, particularly rotary-wing drones, which currently dominate the market.

These drones are used in various sectors, including defense and security, smart agriculture, transportation and logistics, film and media production, healthcare

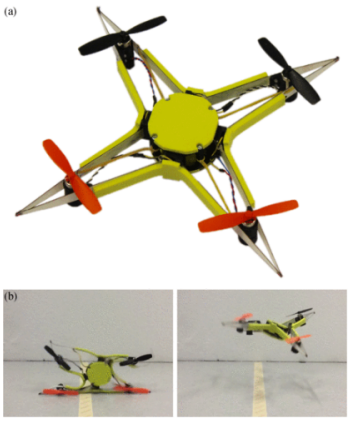
and emergency response, and infrastructure support (see Figure 1.1). In most of these applications, drones typically operate close to humans and surrounding objects. Consequently, the risk of causing damage to people, nearby structures, or the drone itself in the event of a collision is significant. Therefore, ensuring the safety of rotary-wing unmanned aerial vehicles is crucial. In addition to software-based solutions such as vision-based collision avoidance, hardware-based approaches are also essential to improve drone safety in the event of a collision.

1.2 Research problems

1.2.1 Existing Approaches to UAV Collision Safety



(a) A foldable and safety-oriented UAV for logistics applications. [2]

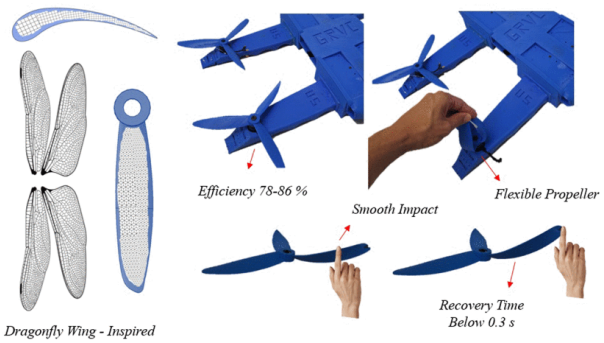


(b) A quadcopter with a crash resilient frame inspired by the wings of insects. [3]

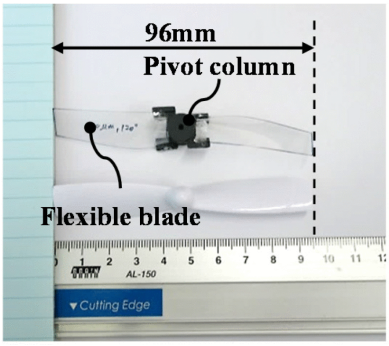
Figure 1.2: The hardware designs that protect drones and mitigate the impact during collisions.

Recent studies have proposed various approaches to enhance UAV safety during collisions. One category involves protective frames—ranging from full-body enclosures to structures specifically shielding the propellers, which typically operate at high speeds of several thousand rpm (see Figure 1.2(a)). Some approaches draw inspiration from bio-mimetic morphologies and soft robotics (see Figure 1.2(b)), aiming to increase structural compliance and the ability to absorb

impact forces, thereby reducing damage to the drone.



(a) Propellers made from soft TPU. [4]



(b) Propellers made from soft PET film. [5]

Figure 1.3: The propellers are made of soft material to protect the drone when a collision occurs at the propeller.

Protective cage designs focus on isolating the drone from the external environment but increase both size and weight, thereby reducing the flexibility of the drone in practical operations. Meanwhile, designs centered on soft frames effectively absorb impact forces but are only suitable for drones with low overall mass, making them challenging to apply in real-world scenarios. Moreover, such structures cannot fully enclose the surrounding airspace of the drone, particularly the propeller area. To address this limitation, several studies have focused specifically on improving propeller safety using soft materials such as TPU and PET, as explored in Figure 1.3. These designs focus solely on improving the structure of the propellers, the most sensitive and potentially destructive component during collisions, without increasing the weight of the drone, thus ensuring safety for both the drone and the objects with which it may collide in hazardous situations.

1.2.2 Progress and Limitations in Our Prior Works

To further advance this line of research, our earlier work [6] introduced a bio-inspired deformable propeller known as the Tombo propeller. Designed with a rigid hub and rigid wing combined with soft edge and soft nodus made from Dragon Skin 30 silicone (Smooth-On), Tombo propeller preserves the structural

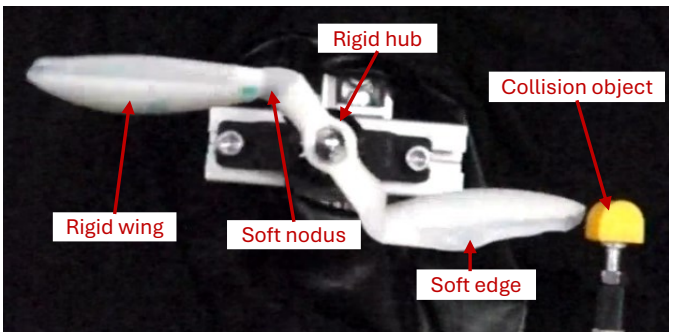


Figure 1.4: The Tombo propeller deforms upon collision.

strength of a conventional 9-inch rigid propeller at high rotational speeds (5000-6000 rpm), while flexing upon impact to dissipate collision energy (see Figure 1.4). Unlike propellers made entirely from soft materials such as TPU or PET, which tend to deform uniformly during rotation and struggle to balance between thrust generation and impact absorption, Tombo propeller employs targeted flexibility. Only two specific regions of the propeller are intentionally designed to be flexible: the edge, which is the primary point of collision, and the hinge, which is made from viscoelastic silicone to enable controlled folding and prolong the collision duration. This extended impact time helps reduce the force transmitted to the propeller structure. Meanwhile, the main body of the propeller remains rigid to maintain aerodynamic efficiency and stable thrust. This design successfully overcomes the typical trade-off between softness and thrust performance, enabling both effective collision mitigation and reliable flight stability.

In that study, we completed the design, aerodynamic optimization, and fabrication of the Tombo propeller, and experimentally verified its ability to absorb impact energy and enable the drone to remain airborne after collisions at the propeller level. Based on this design, we mounted the Tombo propellers on a quadrotor and conducted controlled collision experiments [7]. The results confirmed that the drone could survive and recover from mid-flight propeller collisions using predefined flight sequences (see Figure 1.5).

However, several key limitations remained unaddressed: First, the recovery strategy employed in previous experiments resulted from deliberate user program-

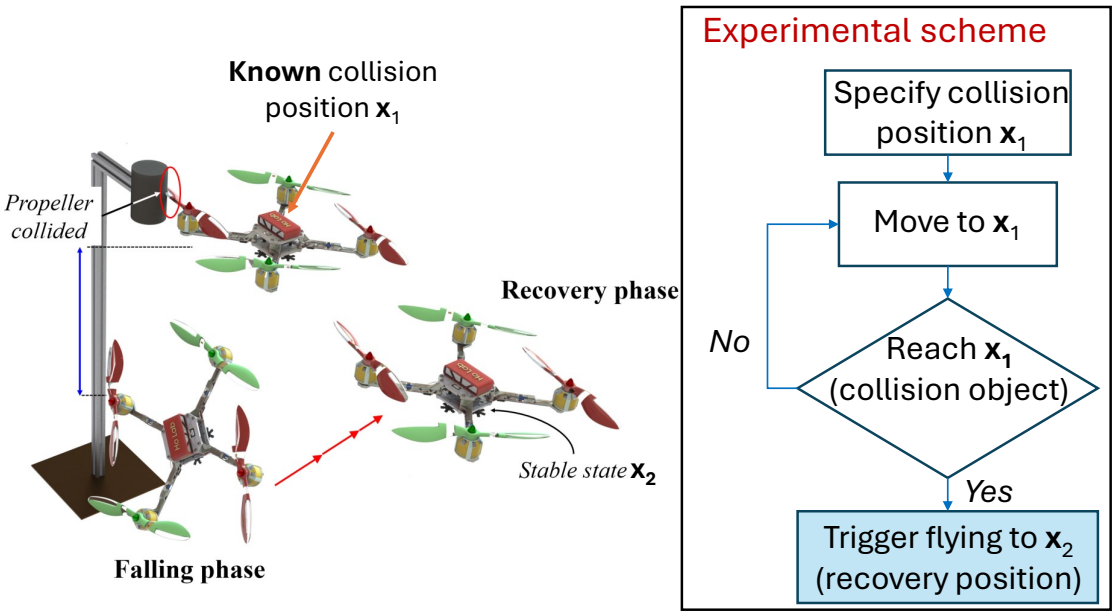


Figure 1.5: Drone recovers after collision when drone moves to position of collision that we set up at specific point.

ming, in which the location of the obstacle was known in advance. The drone was configured to navigate toward a predefined collision point, intentionally make contact, and then track a predefined trajectory toward a designated secondary waypoint. This setup assumed that the collision location and collision trajectory were predefined and known in advance, which do not reflect real-world conditions where obstacles may appear unexpectedly and collisions occur without prior knowledge. As a result, the system lacked autonomous awareness of collision events and was unable to respond in real time to sudden disturbances.

Second, while the Tombo propellers prevented serious damage during impact, the system remained passive: the drone could continue flying, but it had no mechanism for detecting a collision or responding adaptively to avoid further contact with the obstacle. In effect, the benefit of the propeller was limited to mechanical resilience, without enabling a closed-loop behavioral response to environmental hazards.

These limitations motivate the development of active collision-awareness and adaptive response strategies, which form the core focus of this dissertation.

1.2.3 Research Objectives

This dissertation proposes two complementary strategies for handling collisions encountered by quadrotors equipped with Tombo propellers—one that enables real-time recovery without additional sensors, and another that incorporates sensors for precise collision detection and future avoidance:

Strategy 1: Sensorless handling of light or brief collisions.

For minor or short-duration collisions that induce only brief disturbances, explicit collision detection may be unnecessary and could even introduce undesirable delay. Instead, rapid stabilization is more effective. To this end, the $\mathcal{L}_1$Quad control framework [8]—which integrates geometric control with $\mathcal{L}_1$ adaptive augmentation—is employed (see Figure 1.6). This controller not only enables fast, sensorless recovery from light collisions by responding directly to disturbance-induced deviations but also compensates for broader modeling uncertainties. These include aerodynamic variability, environmental disturbances, and the inherent instability introduced by deformable Tombo propellers compared to rigid ones. By adaptively adjusting the control response in real time, the $\mathcal{L}_1$Quad framework enhances robustness and ensures stable flight performance under diverse operating conditions.

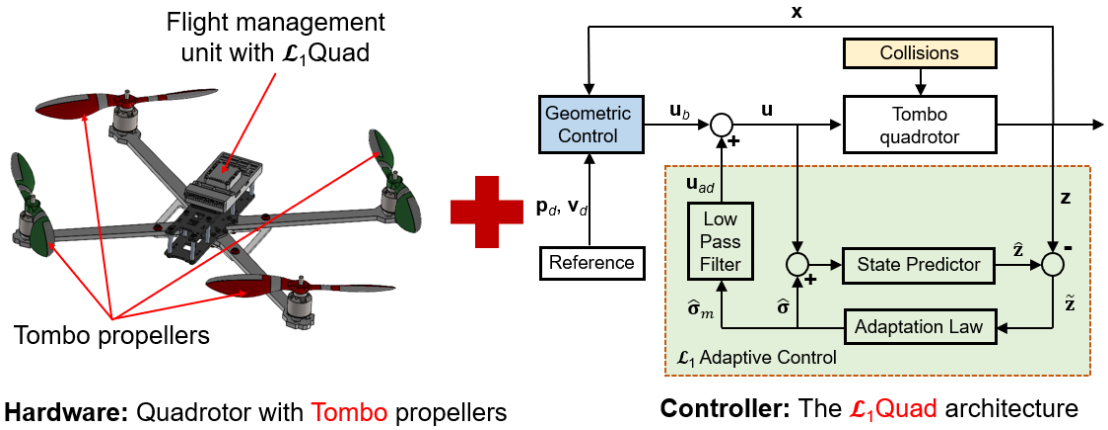


Figure 1.6: Enhancing fast recovery from brief collisions by combining the Tombo quadrotor hardware with the intelligent $\mathcal{L}_1$Quad control law.

Strategy 2: Sensor-based handling of strong or sustained collisions.

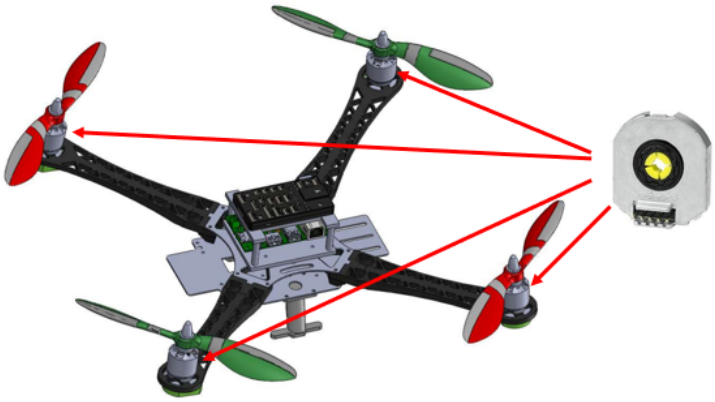


Figure 1.7: Integrating encoders onto the motors to measure propeller speed for detecting collisions occurring on the propellers.

In scenarios involving significant collisions—such as those with large velocity fluctuations or prolonged contact—accurate and timely detection is critical for effective recovery. Existing fault detection methods, which rely on indirect inference [9–11], are frequently subject to false positives and cannot localize the source of the collision. To address these limitations, encoders are integrated into Tombo propeller design to directly measure propeller speed variations (see Figure 1.7). This enables real-time, localized detection of collision events, not only facilitating immediate recovery maneuvers but also supporting proactive avoidance in future flights by identifying the impact's location.

Together, these two strategies form a comprehensive solution for achieving resilient flight with Tombo propellers. They enable recovery from both undetectable (sensorless) and detectable (sensor-based) collisions, while mitigating disturbances caused by the soft structure of Tombo propellers and environmental uncertainties.

1.2.4 Contributions

The main contributions of this dissertation are summarized as follows:

1. A physically consistent modeling framework is introduced to simulate collisions on deformable propellers. This model integrates the dynamics of the BLDC motor with the Tombo propeller, which is represented as a three-link

mechanical structure, while the collision object is simplified as an external contact force. This approach ensures physical consistency, computational simplicity, and easy integration into the overall mechanical system, allowing for efficient solution using Lagrangian methods.

2. The $\mathcal{L}_1$Quad control architecture is applied to quadrotors equipped with Tombo propellers to enable sensorless post-collision recovery. This control logic is integrated as a dedicated flight mode—Tombo-$\mathcal{L}_1$—within ArduPilot, an open-source autopilot software widely used for controlling various types of UAV, such as fixed-wing aircraft, helicopters, and multirotors. The integration allows the vehicle to autonomously stabilize after minor impacts without requiring explicit collision detection, while also compensating for disturbances caused by blade deformation and other modeling uncertainties.
3. An encoder-based collision detection system is implemented for quadrotors using Tombo propellers, representing, to the best of our knowledge, the first implementation of encoder-based direct propeller collision sensing in UAV. Real-time detection is achieved through analysis of velocity deviations, offering a lightweight and practical solution for drones equipped with damage-tolerant or flexible propellers. The system not only supports immediate recovery but also enables future obstacle avoidance by providing localized impact information.
4. A standardized and flexible benchmarking framework is proposed to evaluate the collision recovery performance of UAVs systematically. This framework provides clearly defined experimental setups and objective performance metrics, enabling future researchers to quantitatively assess the effectiveness of both hardware solutions (such as deformable propellers and protective structures) and control strategies (including adaptive or collision-resilient controllers) in mitigating the effects of collisions. It establishes an essential foundation for comparative studies and facilitates the development of safer, more resilient drones by offering a reproducible methodology that can be

adapted across different UAV platforms, materials, and control architectures.

Collectively, these contributions advance the understanding and practical implementation of collision-aware quadrotor systems employing deformable propellers, and provide transferable methodologies for future research on resilient aerial platforms and soft-robotic actuation in UAVs.

1.3 Publications

Journal:

[1] **H. T. Pham**, Q. N. Pham, Q. K. Luu, G. Loianno and V. A. Ho, "Collision Detection and Reaction for Quadrotors Using Encoder-Integrated Tombo Propellers" in *IEEE Access*, 2025.
(Video: https://www.youtube.com/watch?v=czliIvss5_I), doi: https://doi.org/10.1109/ACCESS.2025.3572243.

IEEE Access: Impact Factor: 3.4, Eigenfactor: 0.35701, Article Influence Score: 0.698, CiteScore: 9.8; Ranking: Q1 in Computer Science (miscellaneous), Q1 in Engineering (miscellaneous), Q1 in Material Science (miscellaneous); #8 journal in top publications of Engineering & Computer Science by Google scholar.

Conference proceeding:

[1] **H. T. Pham**, D. Q. Nguyen, S. T. Bui, G. Loianno and V. A. Ho, "Collision Dynamics of Motorized Deformable Propellers for Drones" in *2024 21st International Conference on Ubiquitous Robots (UR)*, New York, NY, USA, 2024, pp. 176-183, doi: https://doi.org/10.1109/UR61395.2024.10597535.

Patent:

[1] S. T. Bui, **H. T. Pham**, and V. A. Ho, "Impact-absorbing and collision detection mechanism for quadrotor propellers," Vietnam Patent No. 49382, granted 2025.

1.4 Structure of thesis

This dissertation is organized into five chapters, each addressing a key aspect of enabling collision-aware and resilient quadrotor operation using deformable propellers. The structure is as follows:

- Chapter 1 introduces the motivation for enhancing UAV safety during propeller collisions, identifies research challenges, and presents the originality and contributions of the proposed dual-mode approach combining encoder-based collision detection with adaptive sensorless control.
- Chapter 2 reviews the state of the art in drone collision safety, covering both frame-level and propeller-level protection strategies. It highlights the limitations of existing methods in accurately detecting and handling propeller collisions, motivating the need for more robust real-time solutions.
- Chapter 3 focuses on achieving fast and robust stabilization in the presence of light or transient collisions using a sensorless approach. The chapter describes the integration of Tombo propellers with the $\mathcal{L}_1$Quad control architecture, which enables adaptive compensation for aerodynamic and actuation uncertainties induced by propeller deformation. Experimental results demonstrate improved post-collision flight stability and trajectory tracking without relying on explicit collision detection.
- Chapter 4 develops an analytical model of the BLDC motor and deformable propeller system under collision dynamics, based on the Lagrangian method. It further presents the design and implementation of an encoder-based collision detection framework. Experimental validation demonstrates the ability to detect and localize collision events in real time and to support effective recovery maneuvers in challenging scenarios.
- Chapter 5 introduces a benchmarking framework for evaluating post-collision recovery performance in UAVs. It defines standardized experimental setups

and two key performance metrics—recovery time and maximum deviation—to objectively compare the effectiveness of different drone hardware and control strategies. This framework provides a reference methodology that supports future research on UAV collision resilience.

- Chapter 6 summarizes the research contributions, evaluates the effectiveness of the proposed encoder-based collision detection and adaptive control framework, and outlines future research directions, including intelligent collision-aware navigation and further integration of learning-based approaches.

Chapter 2

Background and related work

This chapter presents a structured review of key techniques that enhance quadrotor safety and resilience during flight. It is organized into three main sections. The first section introduces collision avoidance strategies based on vision and perception, focusing on how drones detect and react to nearby obstacles in advance. The second section explores advanced control methods, with an emphasis on the $\mathcal{L}_1$Quad approach, which enables the quadrotor to maintain stability and adapt to external disturbances such as collisions or deformation of Tombo propellers. The final section discusses sensor-based collision detection, highlighting methods that utilize onboard sensors and passive mechanical responses to identify and localize collision events in real time.

2.1 Collision Avoidance strategies

Numerous collision avoidance strategies have been developed for UAVs to ensure safe navigation, especially in cluttered or dynamic environments. These methods typically fall into categories such as geometric planning, force-field approaches, optimization-based schemes, and sense-and-avoid systems [12,13]. Central to these strategies is integrating onboard sensors that perceive the environment and guide the UAV's response.

Vision-based sensors are widely adopted due to their rich spatial sensing capabilities and low payload weight. Among these, monocular vision offers the simplest hardware setup, relying on a single camera to extract visual features

from sequential images. While machine learning methods such as YOLO-based detectors or neural networks [14] can improve detection accuracy, they often require significant onboard computational power. Simpler alternatives include image processing approaches that identify color or shape-defined objects [15], but they struggle with unknown or unstructured obstacles.

In [16], monocular vision was used for tasks such as landing on moving platforms. However, it remains insufficient for detecting large or static obstacles like buildings. To address this, stereo vision systems are employed to generate disparity maps for depth estimation. For instance, [12] developed a biologically inspired stereo system mounted on a servo, allowing owl-like head movements to track dynamic obstacles. Another approach, "pushbroom stereo," combines stereo imagery with forward motion and lightweight processing for fast obstacle detection [17].

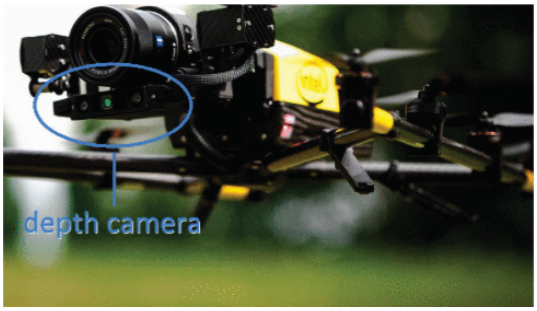
Depth cameras, particularly RGB-D sensors, provide rich 3D information and are frequently used in indoor UAV navigation. In [18], depth and IMU data were fused to detect static and dynamic obstacles and compute evasive trajectories. Methods like U-map conversion [19] offer efficient processing on embedded platforms, making them ideal for onboard use. While effective in cluttered environments, depth cameras can struggle with high-speed objects or those beyond their limited range.

Light detection and ranging (LIDAR) is another vision-adjacent technology commonly used for obstacle detection. Though heavier and costlier, LIDAR provides high accuracy and robustness under varied lighting and weather conditions. 2D LIDAR suffers from limitations due to UAV attitude changes, whereas 3D LIDAR offers full spatial coverage at the expense of processing and power demands [20].

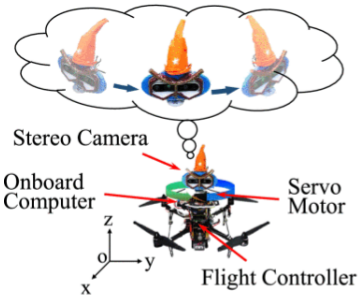
Recently, event cameras have emerged as a promising solution to overcome motion blur and latency issues in fast-moving scenarios. Unlike traditional cameras, event cameras asynchronously capture pixel changes, enabling high

temporal resolution and low power operation. In [21], a stereo event camera system combined with IMU data is proposed to perform motion compensation, reconstruct event streams into frames, and accurately distinguish dynamic from static obstacles with low latency. Despite their advantages, the practical adoption of event cameras is still limited due to high cost, sensitivity to noise, and a restricted sensing range (typically under 1.5 m).

Despite their advantages, all vision-based systems remain susceptible to environmental conditions such as poor lighting, reflections, fog, or smoke. They also tend to perform poorly when detecting objects moving faster than 35 km/h [22], posing challenges for reliable deployment in dynamic outdoor environments.



(a) Depth camera. [12]



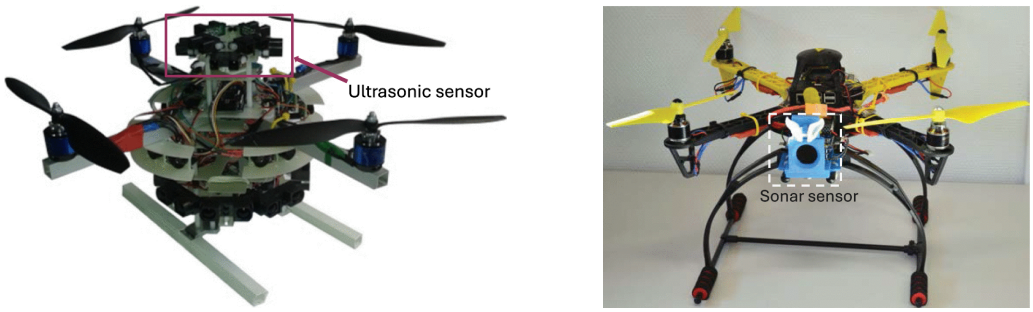
(b) Stereo camera. [21]

Figure 2.1: Vision-based sensors integrated on drones for obstacle avoidance.

While vision-based sensors, such as monocular, stereo, depth, and event cameras, have enabled significant progress in obstacle perception for UAVs, they still face limitations related to lighting conditions, computational cost, and sensing range. To overcome these challenges, researchers have explored alternative contactless sensing modalities that do not rely on visible light or detailed image processing.

Among them, ultrasonic and infrared sensors are attractive due to their low cost, compact size, and minimal power requirements [23]. However, their relatively slow response times make them less suitable for high-speed obstacle detection during agile drone maneuvers. Infrared (IR) cameras, which detect thermal radiation rather than reflected visible light, offer reliable performance in low-light

and nighttime conditions [24]. However, they often produce images with lower spatial resolution and contrast, reducing their effectiveness in distinguishing fine obstacle details [25]. To mitigate this, IR sensors are frequently fused with visual or depth-based sensors to enhance perception.



(a) Ultrasonic sensor. [23]

(b) Sonar sensor. [26]

Figure 2.2: Other sensors mounted on drones for obstacle avoidance.

Sonar sensors have also been studied for UAV obstacle detection [26], but their use remains largely experimental due to limited range and directional accuracy in open-air flight. In contrast, radar sensors offer better robustness against lighting variability, fog, and precipitation. Their ability to detect objects through certain materials or in cluttered environments makes them valuable, albeit at the expense of added weight and power consumption.

In specialized applications, magnetometers and electromagnetic sensors have been mounted on UAV to detect metallic underground targets, such as buried pipelines or unexploded ordnance [27, 28]. However, these sensors are not viable for general-purpose obstacle avoidance due to their specificity to metallic objects and vulnerability to electromagnetic interference.

In summary, no single sensor modality currently offers a comprehensive solution for UAV collision avoidance across all environmental and operational conditions. Vision-based methods can provide rich spatial information but are often sensitive to lighting, motion blur, and require high processing power. Contactless alternatives such as ultrasonic, infrared, or radar sensors provide robustness under certain conditions but typically suffer from lower resolution, slower response, or

limited applicability. Moreover, many of these sensors are either too specialized (e.g., magnetometers) or underexplored (e.g., sonar) for practical deployment in dynamic flight scenarios.

To achieve reliable and real-time obstacle avoidance, multi-sensor fusion has emerged as the most promising approach. By combining complementary sensing technologies—such as fusing RGB-D data with infrared or radar input—UAVs can compensate for individual sensor weaknesses [29, 30]. However, this results in higher system complexity, payload, and power consumption, which are critical constraints for small-scale drones. The design of an effective obstacle avoidance system must, therefore, carefully balance sensing fidelity with platform limitations in weight, energy, and processing resources.

2.2 Advanced Control for Collision Resilience

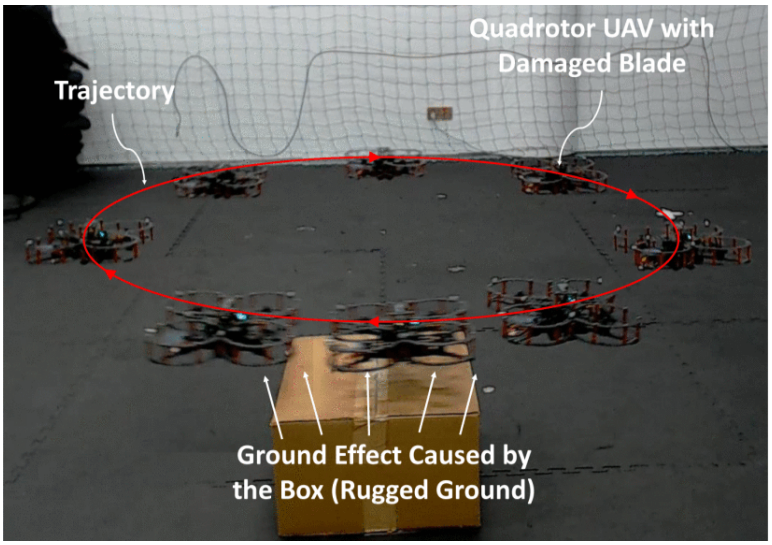
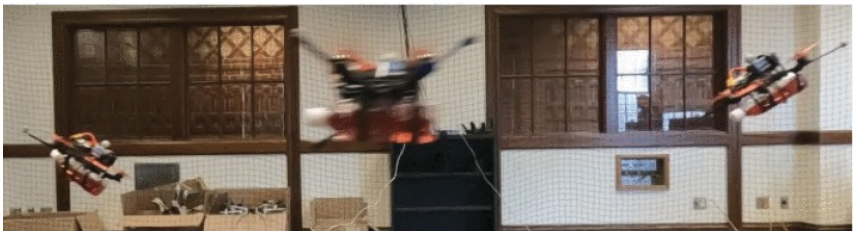
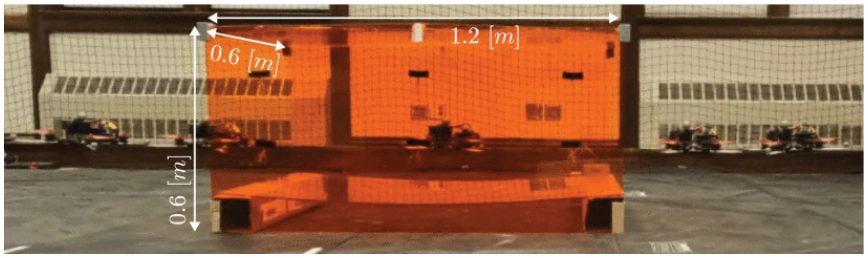


Figure 2.3: A quadrotor UAV with a damaged propeller operating near irregular ground surface; the box is placed to simulate a sudden change in ground elevation.

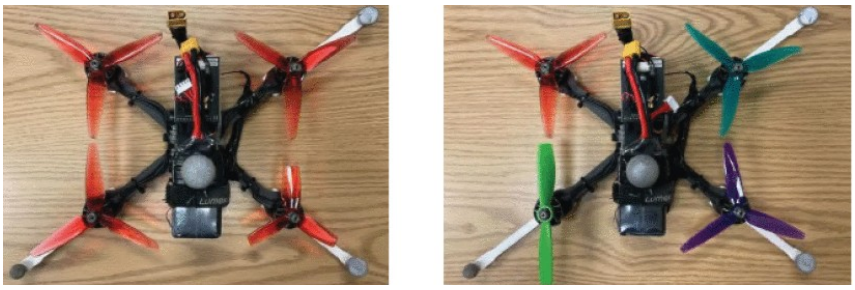
Precise trajectory tracking of quadrotor UAVs in real-world environments remains a challenging task due to various sources of disturbances and uncertainties. These include aerodynamic effects, actuator nonlinearities, structural flexibility (such as hand-made or deformable propellers), battery voltage variations, and



(a) Sloshing payload experiment.



(b) Tunnel experiment.



(c) Quadrotors with chipped propeller and mixed propellers.

Figure 2.4: Controlling a quadrotor using the $\mathcal{L}_1$Quad strategy to track a geometric flight trajectory under complex flight conditions.

transient forces arising from collisions or environmental interactions.

While numerous advanced control strategies have been developed to compensate for disturbances and model uncertainties, none have specifically addressed trajectory deviations caused by propeller collisions. In particular, when using deformable propellers designed to absorb impacts, the resulting dynamic changes introduce additional uncertainties into the system. These impact-induced variations can be viewed as a subset of broader modeling uncertainties. Therefore, the following section reviews existing advanced control approaches, emphasizing their capabilities and limitations in handling such uncertainties, and motivates the adoption of $\mathcal{L}_1$ adaptive control in the context of this dissertation.

Early works on quadrotor control relied on linearizing the dynamics around a

hover state to employ linear controllers like proportional integral derivative (PID) and linear quadratic regulator (LQR) control [31], [32]. While these methods are simple and computationally efficient, they suffer from poor performance when the quadrotor operates far from the linearization point, especially during aggressive maneuvers. To address these challenges, nonlinear control methods such as backstepping [33, 34] and feedback linearization [35, 36] have been proposed. These techniques allow better handling of the inherent nonlinearities in quadrotor dynamics. However, most of these methods rely on Euler angles for attitude representation, which introduces the risk of gimbal lock [37] and limits the ability to perform aggressive maneuvers.

Geometric control techniques address the limitations of Euler-angle-based approaches by directly controlling the quadrotor's attitude on the nonlinear manifold special euclidean group (SE(3)), using rotation matrices or unit quaternions [38–40]. This allows for globally defined and singularity-free attitude representations, enabling aggressive flight. In particular, the work by Lee et al. [41] established exponential stability guarantees for geometric control of quadrotors. Geometric controllers are now widely adopted in research and practical UAV systems due to their solid theoretical properties and excellent performance in agile flight. However, these controllers are typically designed assuming nominal dynamics and may not fully compensate for modeling uncertainties or external disturbances.

To address uncertainties such as parameter variations or unmodeled dynamics, robust and adaptive control methods have been integrated with geometric controllers [42–44]. These approaches can handle certain classes of uncertainties and disturbances and offer provable stability guarantees. However, many of these methods rely on parameterizations of uncertainties using known basis functions, which limits their applicability to more complex, nonlinear, or time-varying disturbances. Moreover, adaptation laws may require slow update rates to maintain system robustness, reducing their effectiveness against fast transients

such as impulsive forces due to collisions.

Disturbance observer-based control (DOBC) approaches provide another way to compensate for external disturbances [45, 46]. By estimating unknown disturbance signals online and feeding this information into the control loop, DOBC can improve tracking performance. A recent work [47] proposed a safety control scheme that integrates a disturbance observer with sliding mode control and a fixed-time sliding mode observer to handle complex disturbances such as blade damage and ground effect (see Figure 2.3). The study quantitatively analyzed the impacts of these factors and demonstrated improved robustness in experimental flights close to rugged ground surfaces under blade damage conditions. However, many DOBC designs assume that disturbances are generated by an exogenous system with known dynamics [48], which is a limiting assumption. Additionally, theoretical analyses of DOBC often neglect the full state-dependence of uncertainties, and practical implementations face challenges related to the numerical differentiation of noisy signals, which can degrade estimation accuracy. Furthermore, while effective for specific disturbance types such as ground effect and blade damage, such approaches do not provide formal transient performance guarantees, nor do they easily generalize to rapidly time-varying disturbances such as those arising from deformable propeller collisions.

incremental nonlinear dynamic inversion (INDI) has been proposed as a robust control strategy to compensates for aerodynamic drag and other fast disturbances, especially in high-speed flights [49]. INDI uses sensor-based measurements to provide fast feedback on acceleration errors and adjust control inputs accordingly. While effective, this method often requires additional sensors, such as optical flow or IMUs with high update rates, which may not be available on standard quadrotor platforms. Moreover, it mainly targets high-speed flight drag compensation and may not generalize to a broad class of uncertainties.

Recent works have explored the use of tools, deep neural network (DNN)s, to model and compensate for quadrotor uncertainties [50–53]. These models can

capture complex, nonlinear behaviors that are difficult to represent analytically. ML-based approaches are typically integrated with conventional controllers, such as LQR or model predictive control (MPC), to enhance their robustness. However, several practical challenges limit the deployment of -based controllers: data collection for training is time-consuming and difficult in unknown environments, and theoretical guarantees on closed-loop stability are difficult to establish. Furthermore, models often require substantial computational resources that exceed the capabilities of typical onboard flight controllers.

Unlike the previously described methods, $\mathcal{L}_1$ adaptive control offers a promising architecture that overcomes many of these limitations [54–56]. The key feature of $\mathcal{L}_1$ adaptive control is that it isolates the adaptation mechanism from the control loop through the use of a low-pass filter. This allows for arbitrarily fast adaptation rates while maintaining robustness of the overall system. As a result, $\mathcal{L}_1$ adaptive control can simultaneously compensate for both slow-varying uncertainties (e.g., aerodynamic changes due to battery voltage drop) and fast transients (e.g., impulsive forces from collisions). The $\mathcal{L}_1$ adaptive control framework has been successfully demonstrated in various aerospace applications, including NASA's AirStar [57], Calspan's Learjet [58, 59] and UAV platforms [60, 61].

Recent work in $\mathcal{L}_1$Quad [8] augments a geometric controller with an adaptive control layer $\mathcal{L}_1$ to compensate for translational and rotational uncertainties on SE(3). The architecture guarantees transient performance by bounding the tracking errors within well-characterized tubes centered around the nominal trajectory. $\mathcal{L}_1$Quad has demonstrated superior performance compared to baseline geometric and adaptive controllers across a wide range of practical uncertainties, including slosh payloads, asymmetric propellers, aerodynamic interactions (ground effect, downwash), battery voltage variations, and even physical damage such as chipped or mixed propellers [8] (see Figure 2.4). Importantly, $\mathcal{L}_1$Quad achieves this level of robustness without requiring sensor augmentation or re-tuning between different test scenarios.

This dissertation extends the application of the $\mathcal{L}_1$Quad control architecture to quadrotors equipped with deformable propellers—specifically, the Tombo quadrotor platform. Deformable propellers, often handcrafted or constructed from flexible materials, inherently introduce significant variability in thrust generation and are susceptible to structural deformation during flight and mid-air collisions. The uncertainties associated with such propellers include persistent thrust asymmetries due to fabrication tolerances, state-dependent thrust variations as blade shapes deform under aerodynamic loading, and fast transients resulting from minor or moderate impacts. In addition, fluctuations in battery voltage may further exacerbate these effects.

The decoupled adaptation structure of the $\mathcal{L}_1$Quad controller is well suited to address these challenges: fast, collision-induced disturbances are rapidly rejected, while longer-term thrust asymmetries and deformation-induced variations are adaptively compensated without compromising stability. Using this control strategy, the practical robustness of deformable propeller quadrotors is enhanced, particularly in cluttered or collision-prone environments where conventional controllers often experience performance degradation.

Experimental validation demonstrates that the integration of Tombo propellers with the $\mathcal{L}_1$Quad architecture significantly improves post-collision recovery and trajectory tracking performance.

2.3 Sensor-Based Collision Detection

Prolonged collisions with quadrotor propellers can lead to significant thrust loss, making it difficult or impossible for control algorithms to compensate for resulting trajectory deviations. Without timely collision detection, the system lacks the awareness necessary to initiate recovery actions. This motivates the integration of sensor-based collision detection strategies, which enable real-time impact awareness and support corrective control in response to unexpected contacts.

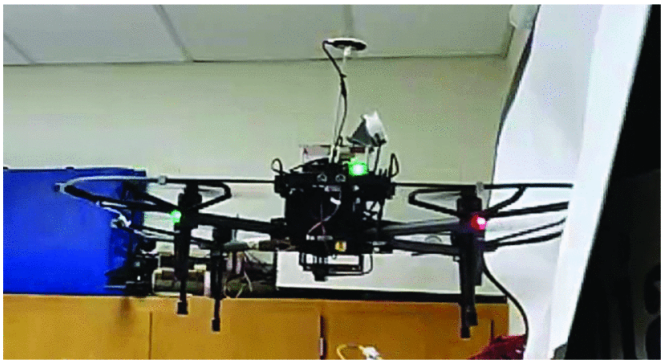


Figure 2.5: The quadrotor collides with a soft object to collect data from the drone's inertial measurement unit (IMU) for collision classification.

The problem of enabling quadrotors to detect and recover from collisions has been addressed through a variety of strategies that combine mechanical design, sensor integration, and post-impact control. A primary direction involves the use of only onboard inertial sensors. For example, [62] (see Figure 2.5) presents a lightweight solution that uses time-series acceleration data from the IMU to detect and classify collisions into soft and hard contact types via a deep neural network, achieving 98.7 % classification accuracy. This method supports real-time collision prediction without requiring additional hardware. Similarly, [63] and [64] propose reactive frameworks where the drone detects collisions based on spikes in horizontal acceleration. Upon detection, the drone estimates the point of collision (POC) and replans its path. These systems rely solely on IMU data and incorporate dynamic replanning modules to ensure the mission continues post-collision.

To enhance robustness against unexpected contacts, several studies explore the use of rigid or protected frames. In [65], the drone is equipped with propeller guards and collision detection is implemented by monitoring the inertial acceleration in the horizontal plane. A threshold-based method detects impacts when acceleration exceeds 1 g. Upon detection, fuzzy logic is employed to classify the scenario and trigger an aggressive attitude recovery controller to regain stable flight. This pipeline was validated both in simulation and on a custom-built experimental platform, demonstrating reliable real-time recovery from various initial flight states. In a similar direction, [66] proposes a method that compares expected

drone dynamics with real-time IMU measurements to detect collisions. Unlike simple threshold-based techniques, it employs a gain-scheduling recovery controller that adapts control parameters based on the estimated collision characteristics and drone attitude. The simulation results demonstrated superior detection accuracy compared to conventional approaches. Furthermore, experimental validation using a custom-built quadrotor confirmed the system's capability to reliably distinguish between propeller collisions and aggressive flight maneuvers, achieving stable post-collision recovery within 0.8 seconds.

Another class of solutions enhances physical resilience through compliant or foldable mechanical structures that absorb impact forces while integrating sensing capabilities. In [67], a foldable quadrotor uses integrated torsional springs in its arms, which fold inward on impact to reduce rebound and minimize structural damage. The drone also includes a lightweight LIDAR and optical flow module to assist in post-collision stabilization and hovering recovery. Similarly, [68] (see Figure 2.6) introduces SCoReR, a collision-resilient aerial robot with soft coil-spring bumpers embedded with resistive force sensors. These deform upon contact, allowing for passive detection of collisions based on resistance change. The drone achieves 100 % detection and stable recovery in indoor environments when it crashes into rigid obstacles at speeds of up to 1 m/s.

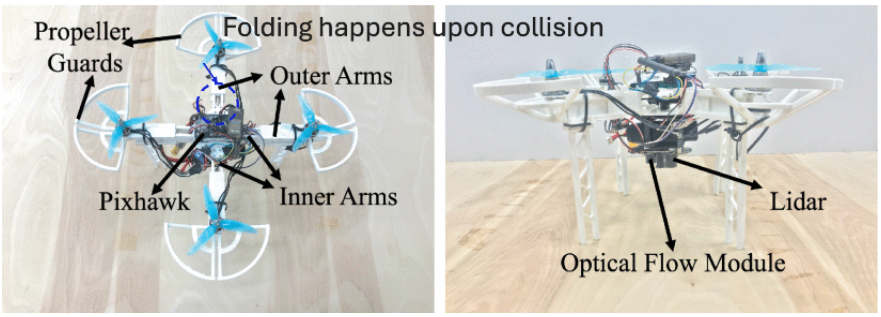


Figure 2.6: Quadrotor with foldable arms and optical flow sensor for collision detection and post-impact recovery.

A more sensor-rich approach is proposed in [69], which integrates hall sensors into a deformable arm structure. These sensors detect deformation during

collision, enabling differentiation between aggressive intentional maneuvers and unplanned impacts. Although effective, this method requires frame redesign and additional sensor wiring, increasing the system's complexity.

In addition to detection and recovery, [70] proposes a novel use of collisions as an information source. The Tiercel micro aerial vehicles (MAV), a small and agile drone with a lightweight frame, intentionally takes advantage of contact with obstacles to infer environmental boundaries. It combines collision cues with visual-inertial odometry (VIO) obtained from a downward-oriented fisheye camera and IMU to perform simultaneous localization and mapping (SLAM) in environments where transparent or reflective surfaces pose challenges to vision-only systems. A swarm of Tiercel MAV demonstrated safe navigation and mapping through such challenging spaces, using collisions as a sensing modality rather than simply as a failure event.

Overall, existing studies demonstrate a variety of sensor-based collision detection and recovery strategies, ranging from software-only solutions using onboard inertial sensors to advanced designs incorporating protective frames with embedded sensors. These approaches have significantly improved the robustness of drones against unexpected collisions and supported rapid recovery during flight. However, most existing solutions are either based on detecting drone-body collisions or require additional hardware that increases system complexity and weight.

Notably, no prior research has directly addressed collision detection at the propeller level. This is a critical gap, as propellers are often the most vulnerable components during flight, especially in cluttered or dynamic environments. In this context, the key contribution of this study is the integration of Tombo propellers with encoder-based collision detection, which enables real-time awareness of impacts that occur specifically on the propellers. This unique combination not only enhances flight safety but also allows for timely collision detection and response without significantly increasing the structural complexity or weight of the drone.

Chapter 3

Sensorless Collision Handling for Tombo Quadrotor

This chapter focuses on the application of the $\mathcal{L}_1$Quad controller to the Tombo quadrotor, enabling effective handling of mild and short-duration collisions occurring while the drone follows a predefined trajectory towards a target point. The $\mathcal{L}_1$Quad architecture, grounded in geometric control combined with an $\mathcal{L}_1$ adaptive control strategy, compensates for uncertainties such as collision-induced disturbances that cause trajectory tracking errors at each time step. This compensation is achieved through a piecewise constant adaptation law, which allows fast and reliable error correction during flight. Experimental results demonstrate that the integration of the Tombo quadrotor with $\mathcal{L}_1$Quad effectively mitigates collision impacts, especially those occurring at the propeller, by rapidly compensating tracking errors and thereby reducing collision recovery time.

3.1 Quadrotor Specifications

3.1.1 Quadrotor Hardware

A custom-built quadrotor platform, shown in Figure 3.1, is constructed using an FPV QAV-RXL 2 – 10" frame (Lumenier). The propulsion system consists of four SURPASS 2216 1120KV BLDC (Goolsky), each individually driven by a 30A electronic speed controllers (ESC) (RC Electric Parts). Power is supplied by a

1500 mAh 4-cell (4S) 95C LiPo battery (Lumenier), which is distributed to the ESCs for simultaneous motor control. The flight control unit (FCU) is a Pixhawk 6X (Holybro), which handles low-level stabilization and communicates with the onboard controller for trajectory tracking and collision handling. The total mass of the quadrotor, including all components, is approximately 1.18 kg. The distance between the two front motors is l=0.345 m, while the distance between the front and rear motors is d=0.26 m.

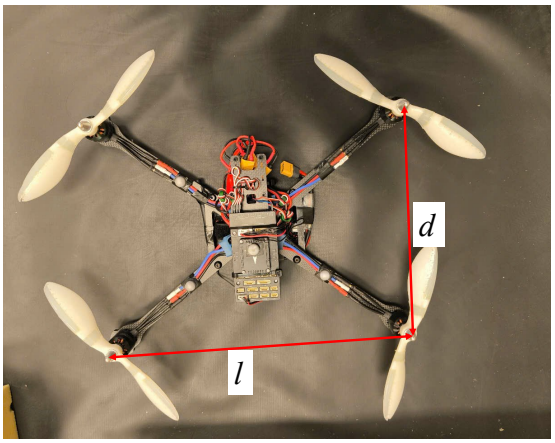


Figure 3.1: The Tombo quadrotor.

The evaluation of the proposed $\mathcal{L}_1$Quad controller in collision scenarios is conducted using two types of propellers: (1) commercially available rigid propellers and (2) Tombo propellers. These propellers are interchangeable on the same quadrotor platform to ensure consistency in experimental comparison. The rigid propellers used for comparison are 9x3.8SF-B4 (APC), which features a length of 9 inches and a pitch of 3.8 inches. To ensure a fair evaluation, Tombo propellers are designed to match the same length as rigid propellers, while differing in structural design to allow for deformation upon impact. In particular, the rigid propeller is thinner and lighter than the Tombo propeller, weighing 9 g compared to 18 g. This mass difference reflects the added material and structure in the Tombo design to enable its deformability.

3.1.2 Measurement of the moment of inertia of quadrotor

To experimentally determine the moment of inertia of the quadrotor along each principal axis, the bifilar pendulum method was employed. In this approach, the drone is suspended by two parallel strings of equal length attached symmetrically about the axis of interest. As shown in Figure 3.2, the experimental setup is designed to measure the moment of inertia J_{zz} about the rotational axis b_3 in the body frame of the quadrotor. When the drone is rotated slightly about this axis and released, it undergoes torsional oscillations. By recording the oscillation period and knowing the mass of the drone, the distance between the suspension strings, and the length of the strings, the moment of inertia J can be calculated using the following relation:

$$J = \frac{mgD^2T^2}{4\pi^2 L}, \tag{3.1}$$

where m is the mass of the drone, g is the gravitational acceleration, D is the distance from the quadrotor's center of mass to each suspension string, L is the length of the strings, and T is the oscillation period. This procedure was repeated for each of the three principal axes to obtain the corresponding inertia values J_{xx}, J_{yy}, and J_{zz} as shown in Table 3.1.

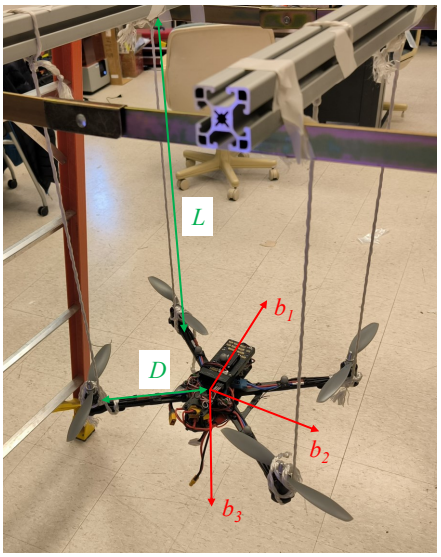


Figure 3.2: Experimental setup of the bifilar pendulum for J_{zz} moment of inertia estimation.

Table 3.1: Experimentally measured moments of inertia of the quadrotor with respect to the body-fixed rotational axes.

Axis	Inertia moment	Value (kgm^2)
b_1	J_{xx}	0.0126
b_2	J_{yy}	0.0081
b_3	J_{zz}	0.0181

3.1.3 Thrust and moment measurement of propeller and motor

In quadrotor control, regulating the thrust generated by the four propellers is essential to achieve and maintain stable flight. Each propeller produces thrust and generates torque, directly affecting the attitude and yaw control of the vehicle. Therefore, precise control requires accurate modeling of both thrust and torque contributions from all four motors. Since the thrust and torque produced by each motor-propeller pair depend on the PWM duty cycle supplied through the ESC, it is crucial to experimentally determine the relationship between PWM input and the resulting forces and moments. This characterization enables more accurate force allocation and attitude control, which are fundamental for achieving stability, responsiveness, and robustness in flight performance.

An experimental setup, illustrated in Figure 3.3, was designed to determine the relationship between the PWM percentage supplied to the motor and the resulting thrust and moment, based on the specific motor and propeller hardware. The experiment consists of a motor-propeller system powered via ESCs, with PWM signals generated by the ArduPilot flight controller installed on Pixhawk 6x hardware. Different PWM levels were supplied from a 4-cell LiPo battery, producing corresponding thrust and moment outputs. These forces were simultaneously measured using a force-torque sensor. Due to their differing mechanical characteristics, measurements were performed separately for the commercial rigid

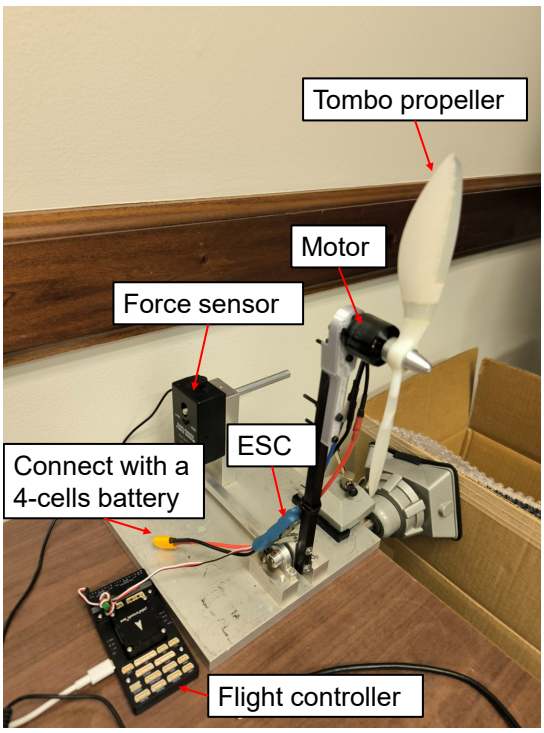


Figure 3.3: Experimental setup for measuring propeller thrust force and moment as a function of PWM percentage from the Ardupilot controller.

propellers and the Tombo propellers. The measurement results are presented in Table 3.2.

Since the measured values are discrete while the required motor PWM percentage must be computed continuously, a mathematical function was constructed based on the measurement data to determine the PWM percentage corresponding to a given thrust. A second-degree polynomial was employed for this fitting, with the general form expressed as follows:

$$f(x) = ax^2 + bx + c, \tag{3.2}$$

where $f(x)$ represents the thrust, and x denotes the throttle percentage (% PWM). Here, the constant $c = 0$ since the generated thrust is zero when the throttle is zero. The objective is to determine the coefficients a and b corresponding to each propeller type. To improve the accuracy of the fitting function for the experimental data, the throttle range was divided into two intervals: one from 0 %

Table 3.2: The thrust force and moment generated by varying the throttle rate with a 4-cell LiPo battery were experimentally measured.

Throttle rate (% PWM)	Rigid propeller		Tombo propeller	
	Thrust force (N)	Moment (Nm)	Thrust force (N)	Moment (Nm)
11.25	0.64	0.087	0.59	0.01885
15	1.21	0.0174	0.93	0.06815
18.75	1.73	0.02755	1.24	0.0464
22.5	2.38	0.03625	1.72	0.0609
26.25	2.82	0.0464	2.02	0.0754
30	3.46	0.0551	2.45	0.0899
33.75	3.93	0.0638	2.86	0.10295
37.5	4.53	0.07105	3.23	0.1189
41.25	4.96	0.0812	3.74	
45	5.52	0.0899	4.17	
48.75	5.97	0.09715		
52.5	6.68	0.10875		

to 22.5 % and another from 22.5 % to 100 %, with a separate function fitted to each range. The corresponding coefficients a and b for these functions, for both types of propellers, are shown in Figure 3.4, with detailed values listed in Table 3.3.

Table 3.3: The values of the thrust function parameters were obtained through fitting.

Interval for fitting	Rigid propeller		Tombo propeller	
	a	b	a	b
0-22.5 %	0.003969	0.01729	0.00206	0.02944
22.5-100 %	0.0006009	0.09529	0.0007729	0.05808

3.2 Quadrotor dynamics and $\mathcal{L}_1$Quad controller

3.2.1 Quadrotor dynamics

Two coordinate frames are used to represent the dynamics of the system. As illustrated in Figure 3.5, the inertial frame $\{i_1, i_2, i_3\}$ and the body-fixed frame $\{b_1, b_2, b_3\}$ are defined within the North-East-Down world coordinate system. The origin of the body-fixed frame is located at the center of mass (CoM), which is assumed to coincide with the geometric center of the vehicle due to its symmetric

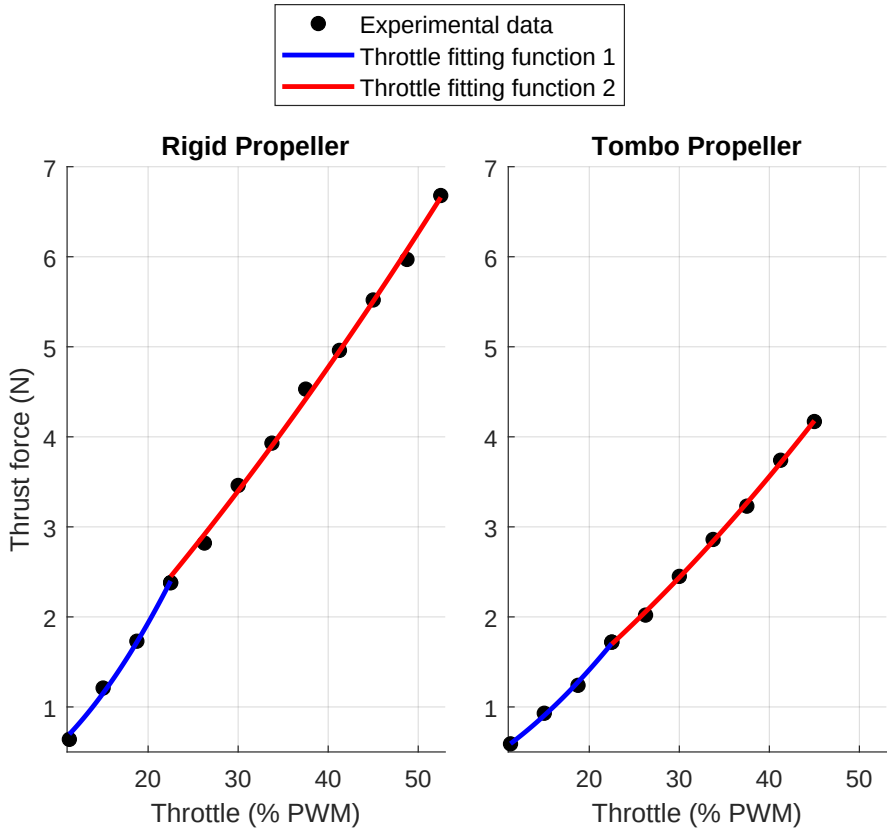


Figure 3.4: The $\mathcal{L}_1$Quad architecture used with the Tombo quadrotor.

mechanical configuration. The first axis, b_1, aligns with the forward direction, while the third axis, b_3, points opposite to the vehicle's thrust vector. The equations of motion of the quadrotor can be written as

$$\dot{\mathbf{p}} = \mathbf{v}, \tag{3.3}$$

$$\dot{\mathbf{v}} = g\mathbf{e}_3 - \frac{T_c}{m}\mathbf{R}\mathbf{e}_3, \tag{3.4}$$

$$\dot{\mathbf{R}} = \mathbf{R}\boldsymbol{\Omega}^{\wedge}, \tag{3.5}$$

$$\dot{\boldsymbol{\Omega}} = \mathbf{J}^{-1}(\mathbf{M}_c - \boldsymbol{\Omega} \times \mathbf{J}\boldsymbol{\Omega}), \tag{3.6}$$

where $\mathbf{p}, \mathbf{v} \in \mathbb{R}^3$ represent the position and velocity, respectively, of the CoM of the quadrotor in the inertial frame. The rotation matrix $\mathbf{R} \in \mathrm{SO}(3)$ transforms coordinates from the body-fixed frame to the inertial frame. The direction of the fixed axis b_i in the inertial frame is given by $\mathbf{R}\mathbf{e}_i$, where $\mathbf{e}_1 = [1\ 0\ 0]^\top$, $\mathbf{e}_2 = [0\ 1\ 0]^\top$

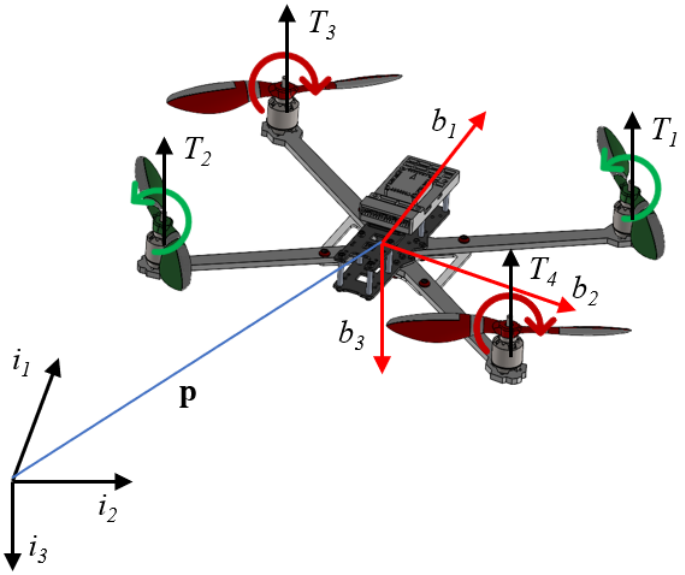


Figure 3.5: The body-fixed frame and inertial frame.

$\mathbf{e}_3 = [0\ 0\ 1]^\top$. $\boldsymbol{\Omega} \in \mathbb{R}^3$ is the angular velocity expressed in the fixed body frame. Here, $\boldsymbol{\Omega}^\wedge$ denotes the skew-symmetric matrix corresponding to $\boldsymbol{\Omega} = [\Omega_1\ \Omega_2\ \Omega_3]^\top$, defined as:

$$\boldsymbol{\Omega}^\wedge = \begin{bmatrix} 0 & -\Omega_3 & \Omega_2 \\ \Omega_3 & 0 & -\Omega_1 \\ -\Omega_2 & \Omega_1 & 0 \end{bmatrix}, \tag{3.7}$$

g is the gravitational acceleration, m is the total mass of the quadrotor, and $\mathbf{J} \in \mathbb{R}^{3\times3}$ denotes the moment of inertia matrix expressed in the body-fixed frame. The collective positive thrust along the body-fixed axis b_3 is denoted by T_c, while the moments about the three body frame axes are represented by the vector $\mathbf{M}_c = [M_1\ M_2\ M_3]^\top$. It is also assumed that the torque generated by each propeller is directly proportional to its thrust. Accordingly, the total thrust T_c and moment $\mathbf{M}_c$ are modeled as linear functions of the squared motor speeds [39]. In this setup, the first and second propellers are designated to rotate in the counterclockwise direction. In contrast, the third and fourth propellers rotate clockwise when the i-th propeller generates a thrust force T_i along $-b_3$. From these assumptions, the desired thrust T_c and desired moment $\mathbf{M}_c$ can be determined from the thrust

produced by each propeller and will be determined using the motor mixing method in [39], as follows.

$$\begin{bmatrix} T_c \\ M_1 \\ M_2 \\ M_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -l/2 & l/2 & l/2 & -l/2 \\ d/2 & -d/2 & d/2 & -d/2 \\ c_{Mf} & c_{Mf} & -c_{Mf} & -c_{Mf} \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix}, \tag{3.8}$$

where l is the distance between the first and third rotor, d is the distance between the first and fourth rotor, and c_{Mf} represents the ratio of torque moment to the thrust produced by the motor, $c_{Mf} = 0.015711$ with rigid propeller and 0.035923 with Tombo propeller.

3.3 $\mathcal{L}_1$Quad architecture

3.3.1 Geometric tracking control

The objective is to guide the quadrotor to follow a specified trajectory $\mathbf{p}_d(t) \in \mathbb{R}^3$ and yaw $\psi_d(t)$ over the time interval $[0,t_f]$. To achieve this, the transverse thrust T_c.

$$T_c = -\mathbf{F}_d \cdot (\mathbf{R}\mathbf{e}_3) \tag{3.9}$$

which is derived by projecting the desired force $\mathbf{F}_d$ onto the body-fixed z-axis b_3. The desired force $\mathbf{F}_d \in \mathbb{R}^3$ is computed as

$$\mathbf{F}_d = m\ddot{\mathbf{p}}_d - mg\mathbf{e}_3 - \mathbf{K}_p\mathbf{e}_p - \mathbf{K}_v\mathbf{e}_v, \tag{3.10}$$

where the errors in tracking the position and velocity are defined as

$$\mathbf{e}_p = \mathbf{p} - \mathbf{p}_d, \tag{3.11}$$

$$\mathbf{e}_v = \dot{\mathbf{p}} - \dot{\mathbf{p}}_d, \tag{3.12}$$

and $\mathbf{K}_p, \mathbf{K}_v \in \mathbb{R}^3$ are the gains for the respective errors. The $\mathbf{p}_d$ represents the desired location of the quadrotor's CoM. In a traditional geometric controller [40], the attitude tracking error $\mathbf{e}_R$ and the angular velocity tracking error $\mathbf{e}_\Omega$ are calculated as follows:

$$\mathbf{e}_R = \frac{1}{2}(\mathbf{R}_d^\top \mathbf{R} - \mathbf{R}^\top \mathbf{R}_d)^\vee, \tag{3.13}$$

$$\mathbf{e}_\Omega = \boldsymbol{\Omega} - \mathbf{R}^\top \mathbf{R}_d \boldsymbol{\Omega}_d, \tag{3.14}$$

where the $^\vee$ operator maps a skew-symmetric matrix back to its corresponding vector form, acting as the inverse of the $^\wedge$ operator defined in Equation (3.7). $\mathbf{R}_d$, $\boldsymbol{\Omega}_d$ are the desired rotation matrix and desired angular velocity, respectively. The moments for the rotational motions are calculated using

$$\mathbf{M}_c = \boldsymbol{\Omega} \times \mathbf{J}\boldsymbol{\Omega} - \mathbf{J}(\boldsymbol{\Omega}^\wedge \mathbf{R}^\top \mathbf{R}_d \boldsymbol{\Omega}_d - \mathbf{R}^\top \mathbf{R}_d \dot{\boldsymbol{\Omega}}_d) - \mathbf{K}_R \mathbf{e}_R - \mathbf{K}_\Omega \mathbf{e}_\Omega, \tag{3.15}$$

where $\mathbf{K}_R, \mathbf{K}_\Omega \in \mathbb{R}^3$ are rotation and angular velocity gains.

3.3.2 Uncertainty modelling

The nominal dynamics described in Equations (3.3)–(3.6) characterize the ideal motion of a quadrotor. In practical scenarios, however, the motion is influenced by various uncertainties and external disturbances, including aerodynamic effects of the propellers, ground effect, and wind. Developing precise models for such phenomena is often costly and offers limited practical advantages.

To address this, the proposed method aggregates these uncertainties into equivalent unknown force and moment terms. The subsequent analysis incorporates these aggregated disturbances into the state-space formulation of the system. Let the state vector be defined as $\mathbf{x} = (\mathbf{p}, \mathbf{v}, \mathbf{R}, \boldsymbol{\Omega})$, where $\mathbf{x} \in \mathbb{R}^3 \times \mathbb{R}^3 \times$ special orthogonal group (SO(3)) $\times \mathbb{R}^3$. The state-space representation of Equations (3.3)–

(3.6) then becomes:

$$\dot{\mathbf{x}}(t) = \mathbf{f}_F(\mathbf{x}(t)) + \mathbf{B}_F(\mathbf{R}(t))\mathbf{u}_c(t), \tag{3.16}$$

where

$$\mathbf{f}_F(\mathbf{x}) = \begin{bmatrix} \mathbf{v} \\ g\mathbf{e}_3 \\ \text{vec}(\mathbf{R}\boldsymbol{\Omega}^\wedge) \\ -\mathbf{J}^{-1}\boldsymbol{\Omega} \times \mathbf{J}\boldsymbol{\Omega} \end{bmatrix}, \quad \mathbf{B}_F(\mathbf{R}) = \begin{bmatrix} 0_{3\times1} & 0_{3\times3} \\ -m^{-1}\mathbf{R}\mathbf{e}_3 & 0_{3\times3} \\ 0_{9\times1} & 0_{9\times3} \\ 0_{3\times1} & \mathbf{J}^{-1} \end{bmatrix},$$

and $\mathbf{u}_c^\top = [T_c \ \mathbf{M}_c^\top]$ denotes the nominal input generated by the geometric controller. It is important to note that when the state $\mathbf{x}$ is represented in the state-space formulation, such as in Equation (3.16), it is expressed in vectorized form as $\mathbf{x}^\top = [\mathbf{p}^\top \ \mathbf{v}^\top \ \text{vec}(\mathbf{R})^\top \ \boldsymbol{\Omega}^\top]$. The operator $\text{vec}(\mathbf{R})^\top$ denotes the vectorization of the rotation matrix $\mathbf{R} \in \text{SO}(3)$, followed by its transposition. Specifically, $\text{vec}(\mathbf{R})$ stacks the columns of $\mathbf{R} \in \mathbb{R}^{3\times3}$ into a column vector $\in \mathbb{R}^9$, and the transposition $(\cdot)^\top$ converts this into a row vector $\in \mathbb{R}^{1\times9}$. For a rotation matrix $\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$, this operation is expressed as:

$$\text{vec}(\mathbf{R})^\top = \begin{bmatrix} r_{11} & r_{21} & r_{31} & r_{12} & r_{22} & r_{32} & r_{13} & r_{23} & r_{33} \end{bmatrix}. \tag{3.17}$$

This vectorized form is often used in optimization problems or gradient computations involving the quadrotor's orientation.

The uncertainties enter the system (3.3)- (3.6) through the dynamics via (3.4) and (3.6) (the kinematics (3.3) and (3.5) are treated as ideal integrators and assumed to be unaffected by uncertainties), which results in the following uncertain dynamics:

$$\dot{\mathbf{x}}(t) = \mathbf{f}_F(\mathbf{x}(t)) + \mathbf{B}_F(\mathbf{R}(t))\left(\mathbf{u}_c(t) + \boldsymbol{\sigma}_m(t, \mathbf{x}(t))\right) + \mathbf{B}_F^\perp(\mathbf{R}(t))\boldsymbol{\sigma}_{um}(t, \mathbf{x}(t)), \tag{3.18}$$

where $\boldsymbol{\sigma}_m \in \mathbb{R}^4$ corresponds to the matched uncertainties, while $\boldsymbol{\sigma}_{um} \in \mathbb{R}^2$ corresponds to the unmatched ones, and

$$\mathbf{B}_F^{\perp}(\mathbf{R}) = \begin{bmatrix} \mathbf{0}_{3\times 1} & \mathbf{0}_{3\times 1} \\ m^{-1}\mathbf{R}\mathbf{e}_1 & m^{-1}\mathbf{R}\mathbf{e}_2 \\ \mathbf{0}_{12\times 1} & \mathbf{0}_{12\times 1} \end{bmatrix}.$$

The matched uncertainty $\boldsymbol{\sigma}_m$ affects the system through the same input channel as the nominal control input $\mathbf{u}_c$, i.e., via $\mathbf{B}_F(\mathbf{R})$, and can therefore be directly compensated. In contrast, the unmatched uncertainty $\boldsymbol{\sigma}_{um}$ acts through $\mathbf{B}_F^{\perp}(\mathbf{R})$, whose columns are orthogonal to those of $\mathbf{B}_F(\mathbf{R})$.

3.3.3 Uncertainty and disturbance analysis for a Tombo quadrotor

The Tombo propellers are designed to deform and recover from collisions, offering enhanced safety and resilience. However, this innovative design also introduces complexities in controller design. These complexities are analyzed in this section, and it is demonstrated how they can be effectively addressed by $\mathcal{L}_1$Quad.

Variable thrust and torque coefficients: The Tombo propellers have a nodus made of soft material that allows them to fold upon impact, which, however, leads to a changing thrust force and reaction torque generated by the Tombo propeller compared to a conventional (rigid) propeller. can model this phenomenon

$$T^{\text{Tombo}} = k_F^{\text{Tombo}}(\omega)\omega^2, \tag{3.19}$$

where ω is the rotation speed of the propeller, and $k_F^{\text{Tombo}}(\omega)$ is the speed-dependent thrust coefficient. Likewise, one can write the reaction torque of the Tombo propeller with a speed-dependent torque coefficient as $M^{\text{Tombo}} = k_M^{\text{Tombo}}(\omega)\omega^2$. Detailed mathematical models can be found in our previous papers

[7, 71]. For conciseness, the discussion in the sequel focuses on thrust modeling, as the modeling of the reaction torque follows the same principles.

In addition, the Tombo propellers are individually hand-crafted, leading to low uniformity among the four propellers mounted on the quadrotor. In other words, for the four rotors $i \in \{1, 2, 3, 4\}$, the individual thrust is

$$T_i^{\text{Tombo}} = k_{F,i}^{\text{Tombo}}(\omega_i)\omega_i^2. \tag{3.20}$$

In practice, for four *Tombo* propellers mounted on a quadrotor, the thrust coefficient can be decomposed into constant (speed-independent) and variable (speed-dependent) components, i.e., $\mathrm{k}_{F,i}^{\text{Tombo}}(\omega_i) = \bar{k}_F^{\text{Tombo}} + \Delta k_{F,i}^{\text{Tombo}}(\omega_i)$, where $\bar{k}_F^{\text{Tombo}} = \frac{1}{4}\sum_{i=1}^{4} \bar{k}_{F,i}^{\text{Tombo}}$ is the averaged coefficient of the individual *Tombo* propeller's constant coefficient $\bar{k}_{F,i}^{\text{Tombo}}$, and $\Delta k_{F,i}^{\text{Tombo}}(\cdot)$ refers to the variable component. Note that $\bar{k}_{F,i}^{\text{Tombo}}$ can be fitted from the thrust-speed curve with steady-state measurements, whereas $\Delta k_{F,i}^{\text{Tombo}}(\cdot)$ is more complicated to fit and is treated as an unknown mapping in this case. The torque coefficients can be decomposed similarly, with the averaged nominal coefficient denoted by $\bar{k}_M^{\text{Tombo}}$.

The nominal coefficients $\bar{k}_F^{\text{Tombo}}$ and $\bar{k}_M^{\text{Tombo}}$ are used in motor control to determine the rotational speed of each rotor $\omega_{c,i}$ from the command thrust $T_{c,i}$ (which can be uniquely determined from the total command thrust and moment $(T_c, \mathbf{M}_c)$ by motor mixing (3.8)), so that $\omega_{c,i} = \sqrt{T_{c,i}/\bar{k}_F^{\text{Tombo}}}$. However, the presence of the variable coefficients $\Delta k_{F,i}^{\text{Tombo}}$ results in extra thrusts generated, leading to an actual thrust $T_{a,i}$ of motor i being

$$T_{a,i} = T_{c,i} + \Delta k_{F,i}^{\text{Tombo}}(\omega_{c,i})\omega_{c,i}^2. \tag{3.21}$$

Likewise, the extra reaction torque will be generated due to the variable torque coefficient $\Delta k_{M,i}^{\text{Tombo}}(\cdot)$, which impacts the z-axis moment in the body frame. The

actual moment is then

$$\mathbf{M}_a = \mathbf{M}_c + \Lambda(\Delta k_{F,i}^{\text{Tombo}}, \Delta k_{M,i}^{\text{Tombo}})\boldsymbol{\omega}^2, \tag{3.22}$$

where $\Lambda \in \mathbb{R}^{3\times 4}$ denotes the linear mapping from the squared motor speeds to the moments due to the variable coefficients and $\boldsymbol{\omega}^2 = [\omega_1^2\ \omega_2^2\ \omega_3^2\ \omega_4^2]^\top$.

Voltage fluctuation: Another factor affecting the Tombo quadrotor's flight performance is the significant current drain due to the Tombo propeller's higher drag compared with a conventional rigid propeller (of the same size). Considering that the battery can be modeled with internal resistance [72], the current drain leads to voltage fluctuation. Denote the constant value of voltage in the thrust/torque coefficients identification as V_{nom} and the actual voltage driving the Tombo-equipped motors by V_a. Since the propeller's thrust (and reaction torque) are proportional to squared motor speed, and the motor speed is proportional to the voltage V_a, the actual thrust generated is scaled by $(\frac{V_a}{V_{\text{nom}}})^2$, which results in $(\frac{V_a}{V_{\text{nom}}})^2 T_{a,i}$ and $(\frac{V_a}{V_{\text{nom}}})^2 M_a$.

Disturbances from collisions: One of the standout features of Tombo propellers is their ability to absorb and recover from collisions. However, these collisions introduce impact forces and moments that can potentially destabilize the quadrotor. The collision-induced force and moment, denoted by $\mathbf{F}_{\text{collision}}$ and $\mathbf{M}_{\text{collision}}$, respectively, act as external disturbances to the translational and rotational dynamics described in (3.4) and (3.6).

Based on the analysis above, the uncertainties experienced by the Tombo propeller can be considered actuation-induced uncertainties, and the collision force $\mathbf{F}_{\text{collision}}$ and moment $\mathbf{M}_{\text{collision}}$ belong to external disturbances. These factors can be concluded as matched uncertainties $\boldsymbol{\sigma}_m$ in (3.25), i.e.,

$$\boldsymbol{\sigma}_m = \begin{bmatrix} (\frac{V_a}{V_{\text{nom}}})^2 \sum_{i=4}^{4} T_{a,i} - T_c + \Delta T + \mathbf{F}_{\text{collision}}\mathbf{e}_3 \\ (\frac{V_a}{V_{\text{nom}}})^2 \sum_{i=4}^{4} \mathbf{M}_a - \mathbf{M}_c + \mathbf{M}_{\text{collision}} + \Delta \mathbf{M} \end{bmatrix}, \tag{3.23}$$

where ΔT and $\Delta \mathbf{M}$ are the lumped thrust and moment introduced by the other factors that are not included in this analysis. Note that the collision moment $\mathbf{M}_{\text{collision}}$ and the projection of $\mathbf{F}_{\text{collision}}$ onto the body z-axis, $\mathbf{F}_{\text{collision}}\mathbf{e}_3$, are considered as matched uncertainties, whereas the horizontal components $\mathbf{F}_{\text{collision}}\mathbf{e}_1$ and $\mathbf{F}_{\text{collision}}\mathbf{e}_2$ belong to the unmatched uncertainty, which cannot be directly compensated for due to lack of actuation in the body xy-plane. In other words, among a total of six dimensions of uncertainties, four are classified as matched uncertainties, while the remaining two are unmatched uncertainties. Following (3.23), $\mathcal{L}_1$Quad is well fitted to compensate for the induced uncertainties and external disturbances. Experimental validations are detailed in the following section.

3.3.4 $\mathcal{L}_1$ adaptive augmentation

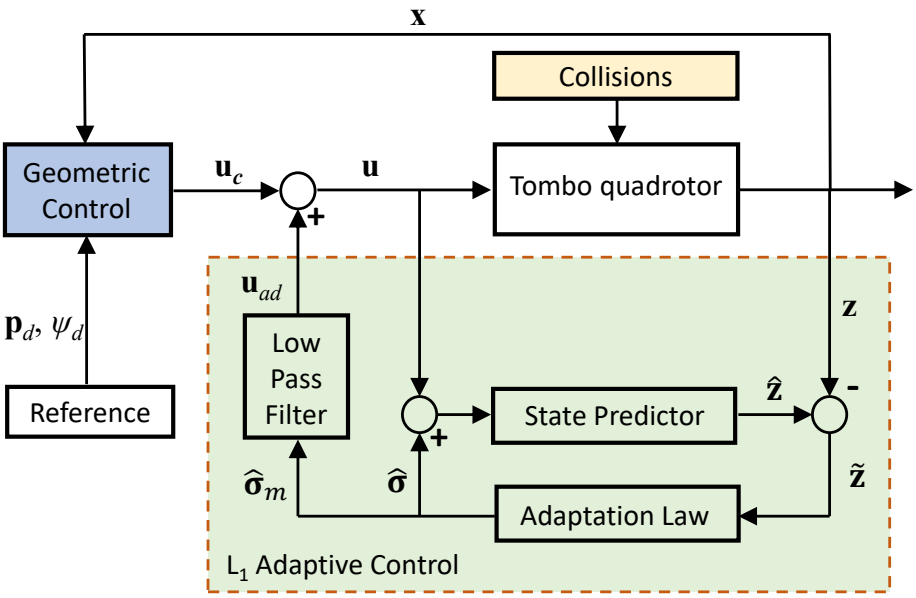


Figure 3.6: The $\mathcal{L}_1$Quad architecture used with the Tombo quadrotor.

This section introduces the application of $\mathcal{L}_1$ adaptive augmentation to the quadrotor control framework. The $\mathcal{L}_1$Quad control input, defined as $\mathbf{u}_{ad} = [T_{ad}\ \mathbf{M}_{ad}^\top]^\top \in \mathbb{R}^4$, is injected into the system through same input channel as the

Table 3.4: Parameters of system and controller

Symbols	Definitions	Values
m	Total mass of quadrotor	$1.18\,\text{kg}$
g	Gravitational acceleration	$9.81\,\text{m/s}^2$
$\mathbf{J}$	Matrix moment of inertia	$10^{-3}\text{diag}(12.6, 8.1, 18.1)\,\text{kgm}^2$
T_s	Sampling time	0.0025 s
$\mathbf{K}_p$	Translational P-gains of the geometric control	$\text{diag}(18.5, 15, 16)$
$\mathbf{K}_v$	Translational D-gains of the geometric control	$\text{diag}(2.5, 1.3, 3.2)$
$\mathbf{K}_R$	Rotational P-gains of the geometric control	$10^{-1}\text{diag}(8.8, 1.6, 10)$
$\mathbf{K}_\Omega$	Rotational D-gains of the geometric control	$10^{-2}\text{diag}(45, 39, 3.28)$
$\mathbf{A}_s$	Hurwitz matrix in (3.26)	$-\text{diag}(20, 20, 20, 20, 20, 20)$
β_T	Cut-off frequencies of the low-pass filter (LPF) for thrust channel	$30\,\text{rad/s}$
β_M	Cut-off frequencies of the LPF for moment channel	$2,\ 4\,\text{rad/s}$

nominal control input $\mathbf{u}_c$. As a result, the system dynamics can be expressed as:

$$\begin{aligned}\dot{\mathbf{x}}(t) =&\mathbf{f}_F(\mathbf{x}(t)) + \mathbf{B}_F(\mathbf{R}(t))\left(\mathbf{u}_c(t) + \mathbf{u}_{ad}(t) + \boldsymbol{\sigma}_m(t, \mathbf{x}(t))\right)\\ &+ \mathbf{B}_F^{\perp}(\mathbf{R}(t))\boldsymbol{\sigma}_{um}(t, \mathbf{x}(t)).\end{aligned} \tag{3.24}$$

The $\mathcal{L}_1$Quad [8,73] architecture is an application of the $\mathcal{L}_1$ adaptive control [54] to quadrotors. $\mathcal{L}_1$Quad applies to the dynamics (3.4) (3.6) of the quadrotor equation of motion, since the kinematics (3.3) (3.5) are integrators and hence considered as uncertainty-free. Denote the full state as $\mathbf{x} = [\mathbf{p}^\top\ \mathbf{v}^\top\ \text{vec}(\mathbf{R})^\top\ \boldsymbol{\Omega}^\top]^\top \in \mathbb{R}^{18}$ and the partial state as $\mathbf{z} = [\mathbf{v}^\top\ \boldsymbol{\Omega}^\top]^\top \in \mathbb{R}^6$. The dynamics in (3.4) and (3.6) are expressed in a state-space form using the partial state vector $\mathbf{z}$, as follows:

$$\begin{aligned}\dot{\mathbf{z}}(t) =&\mathbf{f}(\mathbf{z}(t)) + \mathbf{B}(\mathbf{R}(t))\left(\mathbf{u}_c(t) + \mathbf{u}_{ad}(t) + \boldsymbol{\sigma}_m(t, \mathbf{x}(t))\right)\\ &+ \mathbf{B}^{\perp}(\mathbf{R}(t))\boldsymbol{\sigma}_{um}(t, \mathbf{x}(t)),\end{aligned} \tag{3.25}$$

where

$$\mathbf{f}(\mathbf{z}) = \begin{bmatrix} g\mathbf{e}_3 \\ -J^{-1}\boldsymbol{\Omega} \times \mathbf{J}\boldsymbol{\Omega} \end{bmatrix},$$

$$\mathbf{B}(\mathbf{R}) = \begin{bmatrix} -m^{-1}\mathbf{R}\mathbf{e}_3 & 0_{3\times 3} \\ 0_{3\times 1} & \mathbf{J}^{-1} \end{bmatrix},$$

$$\mathbf{B}^{\perp}(\mathbf{R}) = \begin{bmatrix} m^{-1}\mathbf{R}\mathbf{e}_1 & m^{-1}\mathbf{R}\mathbf{e}_2 \\ 0_{3\times 1} & 0_{3\times 1} \end{bmatrix}.$$

The control $\mathbf{u}_c = [T_c \ \mathbf{M}_c^{\top}]^{\top} \in \mathbb{R}^4$ denotes the geometric control actions, and $\mathbf{u}_{ad} \in \mathbb{R}^4$ denotes the $\mathcal{L}_1$ adaptive control action. The matched and unmatched uncertainties are denoted by $\boldsymbol{\sigma}_m \in \mathbb{R}^4$ and $\boldsymbol{\sigma}_{um} \in \mathbb{R}^2$, respectively.

$$\boldsymbol{\sigma}_m = \begin{bmatrix} \sigma_{T_3} \\ \sigma_{\mathbf{M}} \end{bmatrix} = \begin{bmatrix} \sigma_{T_3} \\ \sigma_{M_1} \\ \sigma_{M_2} \\ \sigma_{M_3} \end{bmatrix}, \boldsymbol{\sigma}_{um} = \begin{bmatrix} \sigma_{T_1} \\ \sigma_{T_2} \end{bmatrix}.$$

The matched uncertainty $\boldsymbol{\sigma}_m$ influences the system via the same input channel as the geometric control input $\mathbf{u}_c$. Therefore, control actions can compensate directly $\boldsymbol{\sigma}_m$. Specifically, in the body frame of the quadrotor, $\boldsymbol{\sigma}_m$ includes σ_{T_3}, which represents the uncertainty associated with the total thrust along the body z axis (since the thrust acts along $\mathbf{b}_3 = [0 \ 0 \ 1]^{\top}$), and $\sigma_{M_1}, \sigma_{M_2}, \sigma_{M_3}$, which corresponds to the uncertainties in the moment forces affecting the angular velocities $\Omega_x, \Omega_y, \Omega_z$ (roll, pitch, and yaw) around the body x, y, and z axes, respectively. In contrast, the unmatched uncertainty $\boldsymbol{\sigma}_{um}$ contains forces along any direction in the body-xy plane, which enters the dynamics via $\mathbf{B}^{\perp}(\mathbf{R})$ (whose columns are orthogonal to those of $\mathbf{B}(\mathbf{R})$). Specifically, σ_{T_1} and σ_{T_2} represent uncertainties along the body x- and y-axes, respectively, as they are associated with the directions $\mathbf{e}_1 = [1 \ 0 \ 0]^{\top}$ and $\mathbf{e}_2 = [0 \ 1 \ 0]^{\top}$ in the body frame, affecting the linear velocities v_x and v_y in the inertial frame after being transformed by $\mathbf{R}(t)$.

The $\mathcal{L}_1$ adaptive controller consists of a state predictor, an adaptation mechanism, and a LPF, as depicted in Figure 3.6. The state predictor mimics the structure of the system described in Equation (3.25), substituting the unknown uncertainties $\boldsymbol{\sigma}_m$ and $\boldsymbol{\sigma}_{um}$ with their corresponding estimates $\hat{\boldsymbol{\sigma}}_m$ and $\hat{\boldsymbol{\sigma}}_{um}$, respectively.

$$\dot{\hat{\mathbf{z}}}(t) = \mathbf{f}(\mathbf{z}(t)) + \mathbf{B}(\mathbf{R}(t))(\mathbf{u}_c(t) + \mathbf{u}_{ad}(t)) + \mathbf{h}(t) + \mathbf{A}_s\tilde{\mathbf{z}}(t), \tag{3.26}$$

where $\tilde{\mathbf{z}} = \hat{\mathbf{z}} - \mathbf{z}$ denotes the prediction error, and $\mathbf{A}_s \in \mathbb{R}^{6\times6}$ is a Hurwitz matrix chosen by the user, $h(t) \in \mathbb{R}^{6\times6}$ represents the estimated uncertainties is chosen as follow

$$\mathbf{h}(t) = \bar{\mathbf{B}}(\mathbf{R}(t))\hat{\boldsymbol{\sigma}}(t, \boldsymbol{x}(t)) = -\boldsymbol{\Phi}^{-1}\boldsymbol{\mu}(t), \tag{3.27}$$

where $\bar{\mathbf{B}}(\mathbf{R}(t)) = [\mathbf{B}(\mathbf{R}(t))\ \mathbf{B}^{\perp}(\mathbf{R}(t))]$.

The estimated uncertainties are computed using a piecewise-constant adaptation law, which holds for $t \in [kT_s, (k+1)T_s)$ as follows:

$$\mathbf{h}(t) = \mathbf{h}(kT_s) = -\boldsymbol{\Phi}^{-1}\boldsymbol{\mu}(kT_s), \tag{3.28}$$

where T_s is the sampling time, $\bar{\mathbf{B}}(kT_s) = [\mathbf{B}(\mathbf{R}(kT_s))\ \mathbf{B}^{\perp}(\mathbf{R}(kT_s))]$, $\boldsymbol{\Phi} = \mathbf{A}_s^{-1}(\exp(\mathbf{A}_sT_s) - \mathbf{I})$, and $\boldsymbol{\mu}(kT_s) = \exp(\mathbf{A}_sT_s)\tilde{\mathbf{z}}(kT_s)$ for $k \in \mathbb{N}$.With the piecewise-constant adaptation law, calculating the estimated uncertainty over the interval $t \in [kT_s, (k+1)T_s)$ becomes straightforward.

$$\hat{\boldsymbol{\sigma}}(t) = \hat{\boldsymbol{\sigma}}(kT_s) = -\bar{\mathbf{B}}(kT_s)^{-1}\Phi^{-1}\boldsymbol{\mu}(kT_s), \tag{3.29}$$

where $\hat{\boldsymbol{\sigma}} = [\hat{\boldsymbol{\sigma}}_m^{\top}\ \hat{\boldsymbol{\sigma}}_{um}^{\top}]^{\top}$. It is important to note that the square matrix $\bar{\mathbf{B}} \in \mathbb{R}^{6\times6}$ is invertible. Furthermore, its inverse, $\bar{\mathbf{B}}^{-1}$, admits an explicit closed-form expression, allowing for efficient computation without the need for numerical inversion methods. To illustrate the functioning of the piecewise-constant adaptation law, consider the state prediction error obtained by subtracting Equation (3.25) from

Equation (3.26):

$$\dot{\tilde{\mathbf{z}}}(t) = \mathbf{A}_s\tilde{\mathbf{z}} + \bar{\mathbf{B}}(t)(\hat{\boldsymbol{\sigma}}(t) - \boldsymbol{\sigma}(t)), \tag{3.30}$$

where $\boldsymbol{\sigma}^\top = [\boldsymbol{\sigma}_m^\top \; \boldsymbol{\sigma}_{um}^\top]$. Without loss of generality, it is assumed that the prediction error at the start of the sampling interval is nonzero, i.e., $\tilde{\mathbf{z}}(kT_s) \neq 0$. The closed-form expression for Equation (10) at the subsequent sampling instant $(k+1)T_s$ is given by

$$\begin{aligned}\tilde{\mathbf{z}}((k+1)T_s) =& \exp(\mathbf{A}_sT_s)\tilde{\mathbf{z}}(kT_s) + (\exp(\mathbf{A}_sT_s) - \mathbf{I})\mathbf{A}_s^{-1}\bar{\mathbf{B}}(kT_s)\boldsymbol{\sigma}(kT_s) \\ &- \int_{kT_s}^{(k+1)T_s} \exp(\mathbf{A}_s((k+1)T_s - t))\bar{\mathbf{B}}(t)\boldsymbol{\sigma}(t)\mathrm{d}t\end{aligned} \tag{3.31}$$

As the state predictor (3.30) replicates the system's structure in (3.25), with the terms related to the uncertainties replaced by the estimation $\mathbf{h}(t)$, a small prediction error implies an accurate estimation. Consequently, the design philosophy of the piecewise-constant adaptation law is to drive the prediction error (3.31) as small as possible. When the piecewise-constant adaptation law is executed at $t = kT_s$, insufficient information about $\sigma(t)$ over the interval $t \in [kT_s, (k+1)T_s)$ prevents direct computation of the integral term on the right-hand side of (3.31). Consequently, the adaptation law is designed to cancel the term $\exp(\mathbf{A}_sT_s)\tilde{\mathbf{z}}(kT_s)$ which arises due to the initial condition at the beginning of the sampling period. Using the adaptation law (3.28) and plugging $\mathbf{h}(kT_s)$ into (3.31), the state prediction error at $t = (k+1)T_s$ takes the form

$$\tilde{\mathbf{z}}((k+1)T_s) = -\int_{iT_s}^{(k+1)T_s} \exp(\mathbf{A}_s((k+1)T_s - t))\bar{\mathbf{B}}(t)\boldsymbol{\sigma}(t)\mathrm{d}t \tag{3.32}$$

where the initial error $\tilde{\mathbf{z}}(kT_s)$ is no longer present. Furthermore, the error at time $\tilde{\mathbf{z}}((k+1)T_s)$ will be eliminated at $\tilde{\mathbf{z}}((k+2)T_s)$ by applying the same reasoning.

By selecting a sufficiently small sampling period T_s (limited by hardware constraints), the norm of the prediction error $\|\tilde{\mathbf{z}}\|$ can be maintained at a low level, enabling an arbitrarily precise estimation of the uncertainties. It should be noted

Algorithm 3.1 $\mathcal{L}_1$ Adaptive Control: Discrete-Time Implementation Notation with $k-1$ and k Denoting the Previous and Current Time Steps, Respectively. Note that the User designs Parameters, and Input Includes Terms Either From the State Estimation or Computation at the Previous Time Step

Parameters: Sampling time T_s; Hurwitz matrix $\mathbf{A}_s$; cutoff frequency β_T, β_M of first-order LPF

Input: $\mathbf{u}_c(k-1)$, $\mathbf{u}_{ad}^{pre}(k-1)$, $\mathbf{z}(k-1)$, $\mathbf{R}(k-1)$, $\hat{\mathbf{z}}(k-1)$, $\hat{\boldsymbol{\sigma}}_m(k-1)$, $\hat{\boldsymbol{\sigma}}_{um}(k-1)$, $\mathbf{R}(k)$, $\mathbf{z}(k)$

Output: $\mathcal{L}_1$ adaptive control $\mathbf{u}_{ad}(k)$

1: State predictor update:
$\hat{\mathbf{z}}(k) = \hat{\mathbf{z}}(k-1) + T_s\dot{\hat{\mathbf{z}}}(k-1)$

2: State prediction error update:
$\tilde{\mathbf{z}}(k) = \hat{\mathbf{z}}(k) - \mathbf{z}(k)$

3: Compute estimated uncertainty $\mathbf{h}(k)$:
$\mathbf{h}(k) = (\exp(\mathbf{A}_s T_s) - \mathbf{I})^{-1}\mathbf{A}_s \exp(\mathbf{A}_s T_s)\tilde{\mathbf{z}}(k)$ ((3.28))

4: Estimate uncertainty:
$\begin{bmatrix}\hat{\boldsymbol{\sigma}}_m(k)\\ \hat{\boldsymbol{\sigma}}_{um}(k)\end{bmatrix} = \hat{\boldsymbol{\sigma}}(k) = -\bar{\mathbf{B}}(\mathbf{R}(k))^{-1}\mathbf{h}(k)$ ((3.29))

5: Filter matched uncertainty (First Stage - LPF1):
For the total thrust component:
$u_{ad,T}^{pre}(k) = e^{-\beta_f T_s}u_{ad,T}^{pre}(k-1) + (1-e^{-\beta_f T_s})\sigma_{T_3}(k)$
For the moment force components:
$\mathbf{u}_{ad,M,1}^{pre}(k) = e^{-\beta_{M,1} T_s}\mathbf{u}_{ad,M,1}^{pre}(k-1) + (1-e^{-\beta_{M,1} T_s})\hat{\boldsymbol{\sigma}}_M(k)$

6: Filter matched uncertainty (Second Stage - LPF2) for the moment force components:
$\mathbf{u}_{ad,M}^{pre}(k) = e^{-\beta_{M,2} T_s}\mathbf{u}_{ad,M}^{pre}(k-1) + (1-e^{-\beta_{M,2} T_s})\mathbf{u}_{ad,M,1}^{pre}(k)$ ((3.33))

7: Compute $\mathcal{L}_1$ adaptive control:
$\mathbf{u}_{ad}(k) = \begin{bmatrix}-u_{ad,T}^{pre}(k)\\ -\mathbf{u}_{ad,M}^{pre}(k)\end{bmatrix}$

8: **return** $\mathbf{u}_{ad}(k)$

that a smaller T_s results in a higher adaptation gain $\boldsymbol{\Phi}^{-1}\exp(\mathbf{A}_s T_s)$ as defined in Equation (3.29). However, an increased adaptation gain may introduce high-frequency components into the estimated signals. To prevent these components from affecting the control input channel—which will be elaborated upon in the next section—the uncertainty estimates undergo filtering. This filtering mechanism effectively decouples the rapid estimation dynamics from the control input, thus improving the overall robustness of the system. The $\mathcal{L}_1$ control input, $\mathbf{u}_{ad}$, compensates exclusively for the matched uncertainty $\boldsymbol{\sigma}_m$ within the bandwidth

ω of the LPF characterized by the transfer function $C(s)$:

$$\mathbf{u}_{ad}(s) = -C(s)\hat{\boldsymbol{\sigma}}_m(s), \tag{3.33}$$

where the control input $\mathbf{u}_{ad}$ compensates solely for the matched uncertainty $\boldsymbol{\sigma}_m$ within the bandwidth β of the LPF characterized by the transfer function $C(s)$. A comprehensive implementation of the $\mathcal{L}_1$Quad controller is presented in Algorithm 3.1. Conceptually, the $\mathcal{L}_1$ adaptive controller employs the state predictor and adaptation law to estimate both matched and unmatched uncertainties, represented by $\hat{\boldsymbol{\sigma}}_m$ and $\hat{\boldsymbol{\sigma}}_{um}$, respectively. The LPF is then used to attenuate high-frequency noise in $\hat{\boldsymbol{\sigma}}_m$, thereby generating the adaptive control input $\mathbf{u}_{ad}$.

3.4 Experimental results

The performance of a Tombo quadrotor and $\mathcal{L}_1$Quad was experimentally validated in both collision-free and collision-involved flights. For collision-involved flights, the performance of the same quadrotor equipped with rigid and Tombo propellers was compared to assess the efficacy of collision compliance resulting from the combination of Tombo propellers and $\mathcal{L}_1$Quad.

Three collision scenarios were considered: impact with a freely suspended foam bar, impact with a carbon fiber bar, and a severe collision involving the intrusion of a carbon fiber rod into the center of the propeller. They represent gradually increasing levels of collision severity for a comprehensive assessment of the quadrotor's resilience to collisions.

The flight controller is a Pixhawk 6X with customized Ardupilot firmware[1]. The quadrotor's actual position is measured using a setup of nine Vicon V16 cameras (see Figure 3.7). The $\mathcal{L}_1$Quad updates the control commands at 400 Hz. The system specifications and control parameters used in the experiments are

[1] https://github.com/sigma-pi/L1Quad

Figure 3.7: Flight experiment setup for the Tombo quadrotor: The quadrotor's position is tracked using Vicon cameras and processed by a ground computer, which transmits the data to the onboard Pixhawk 6X flight controller via the MAVProxy protocol. The Pixhawk 6X then computes the position error between the desired and actual locations and generates the corresponding PWM percentage commands to the motors.

summarized in Table 3.4.

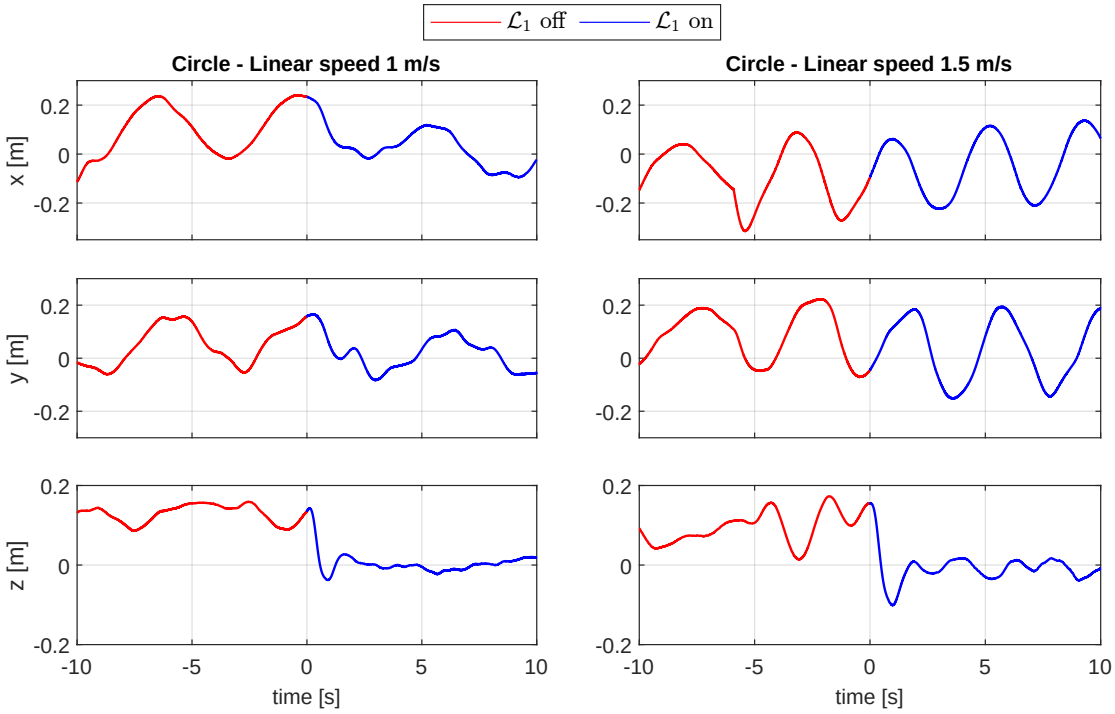
The values of the geometric control parameters $\mathbf{K}_p, \mathbf{K}_v, \mathbf{K}_R, \mathbf{K}_\Omega$ are tuned using differential tuning (DiffTune) [74] to provide satisfactory tracking performance.

Table 3.5: Tracking RMSE of the quadrotor with $\mathcal{L}_1$ on and off in collision-free flight. Unit: [m]

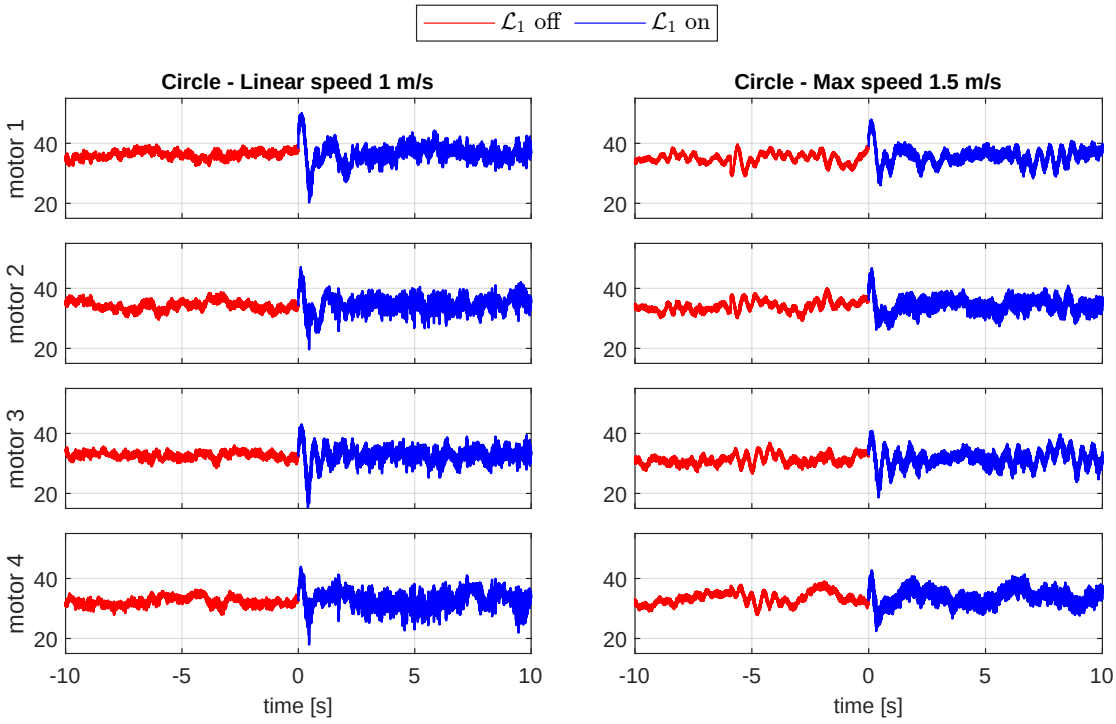
Trajectory	Circle				Lemniscate			
Speed	1 m/s		1.5 m/s		1 m/s		1.5 m/s	
$\mathcal{L}_1$	off	on	off	on	off	on	off	on
x	0.14	**0.07**	0.15	**0.13**	0.13	**0.08**	0.10	**0.06**
y	0.10	**0.06**	0.14	**0.12**	0.14	**0.11**	0.11	**0.06**
z	0.14	**0.01**	0.12	**0.02**	0.11	**0.02**	0.10	**0.01**

3.4.1 Trajectory tracking with $\mathcal{L}_1$Quad in collision-free flights

This section presents experiments conducted on the quadrotor equipped with Tombo propellers, both with $\mathcal{L}_1$ enabled and disabled. The quadrotor tracks

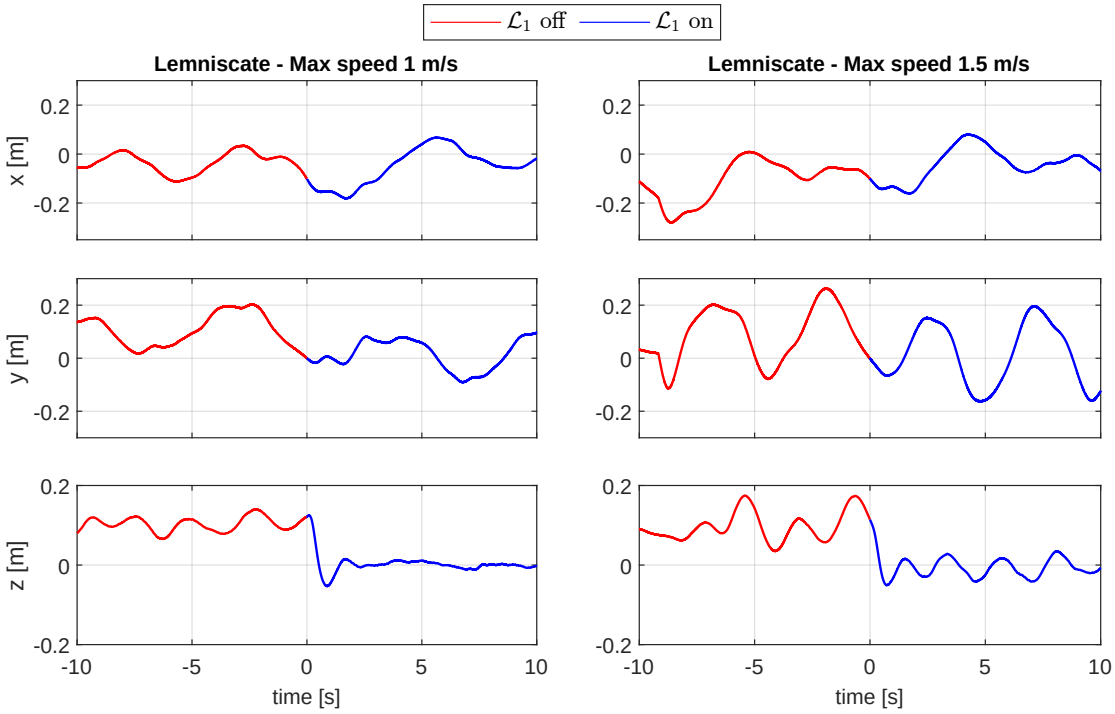


(a) Position tracking error.

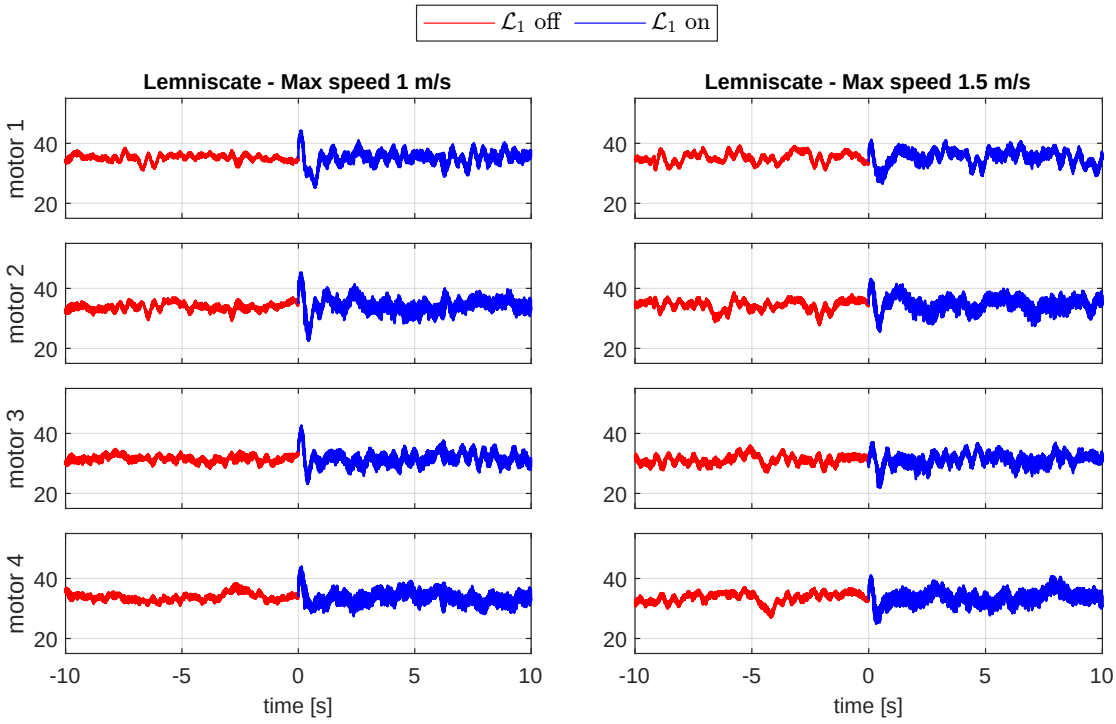


(b) PWM percentage provided to each motor (unit: % PWM).

Figure 3.8: Comparison of position tracking error and motor's PWM percentage between $\mathcal{L}_1$ on and off when the quadrotor is equipped with Tombo propellers, follows a circular trajectory. The switching time from $\mathcal{L}_1$ off to $\mathcal{L}_1$ on was set as 0 s.



(a) Position tracking error.



(b) PWM percentage provided to each motor (unit: % PWM).

Figure 3.9: Comparison of position tracking error and motor's PWM percentage between $\mathcal{L}_1$ on and off when the quadrotor is equipped with Tombo propellers, follows a lemniscate trajectory. The switching time from $\mathcal{L}_1$ off to $\mathcal{L}_1$ on was set as 0 s.

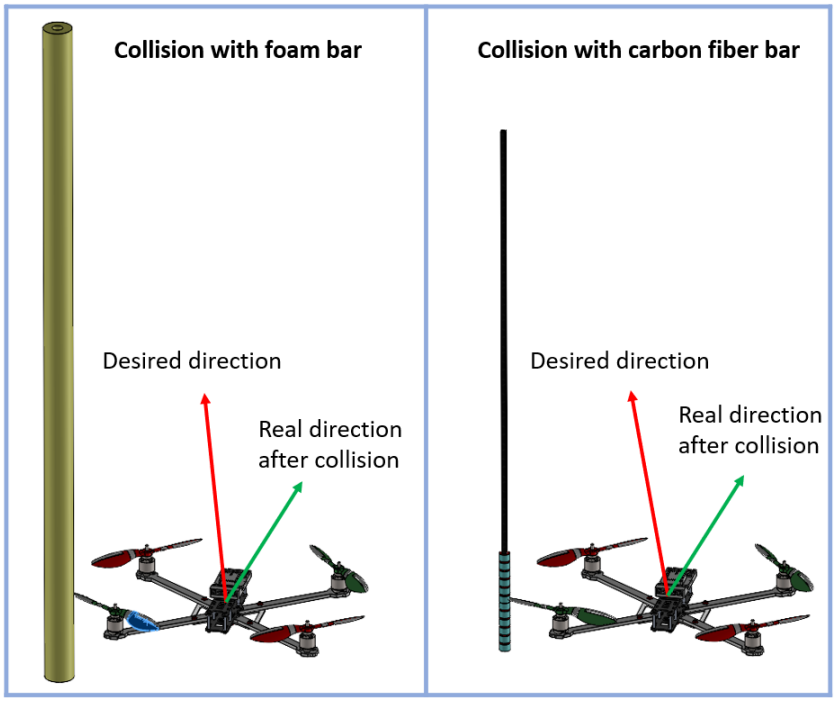


Figure 3.10: Illustration of the collision experiments.

circular and lemniscate trajectories with maximum linear speeds of 1 m/s and 1.5 m/s, respectively. The position tracking is evaluated by the RMSE, when $\mathcal{L}_1$ is on or off, shown in Table 3.5. It can be seen that with $\mathcal{L}_1$ on, the tracking error is significantly reduced in all four cases, with the most substantial reduction on the z-axis. Figures 3.8(a) and 3.9(a) show the transient response when $\mathcal{L}_1$ is turned on in the middle of flights, where the conclusion above still holds. This phenomenon is expected because the $\mathcal{L}_1$ adaptive control compensates for the matched uncertainties in the body z-axis induced by the Tombo propellers' uncertain thrust coefficients.

Additionally, Figures 3.8(b) and 3.9(b) illustrate the percentage of PWM supplied to the motor in the respective trajectory-following cases. It can be observed that when switching to $\mathcal{L}_1$ on mode, the PWM variation is more pronounced because, at this point, the adaptive compensation control force $\mathbf{u}_{ad}$ has been calculated and added to the trajectory-tracking control force $\mathbf{u}_c$ to produce the final control $\mathbf{u}$ force applied to the quadrotor, compensating for disturbances and uncertainties affecting the quadrotor's trajectory tracking error.

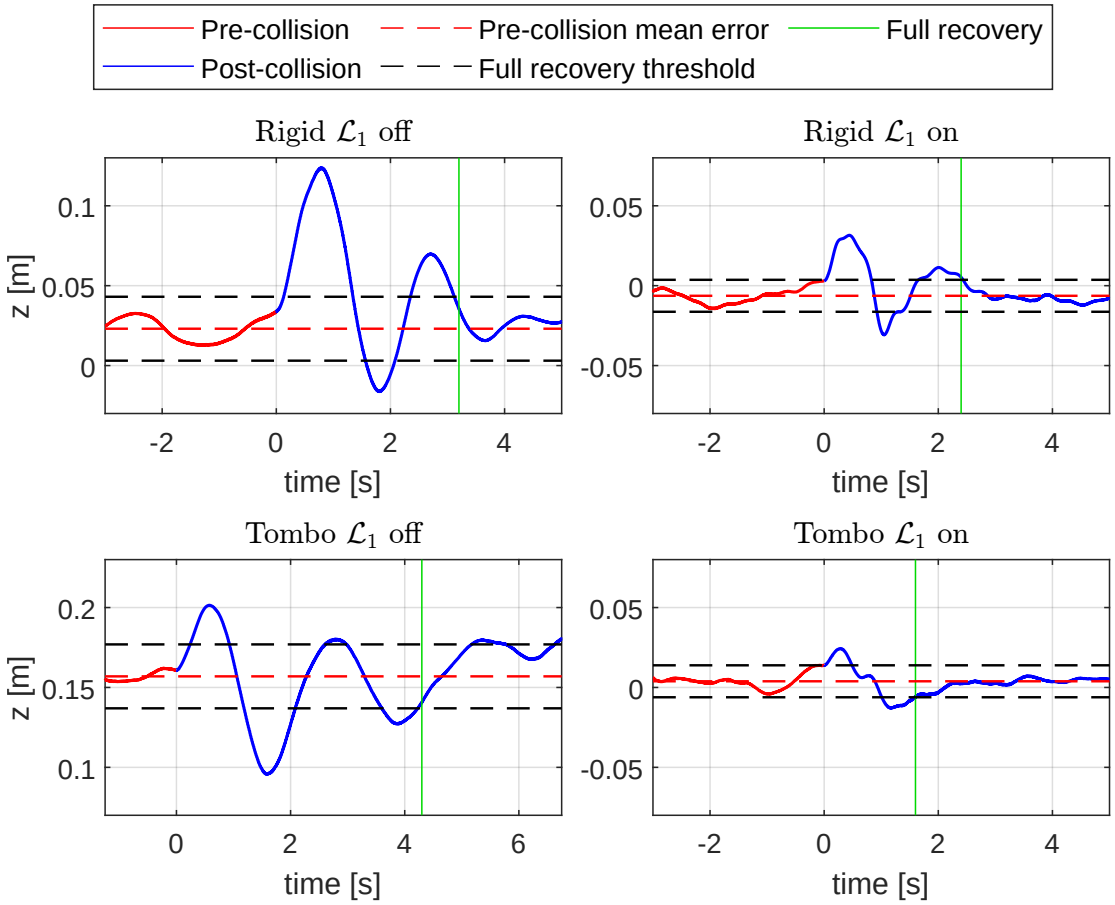
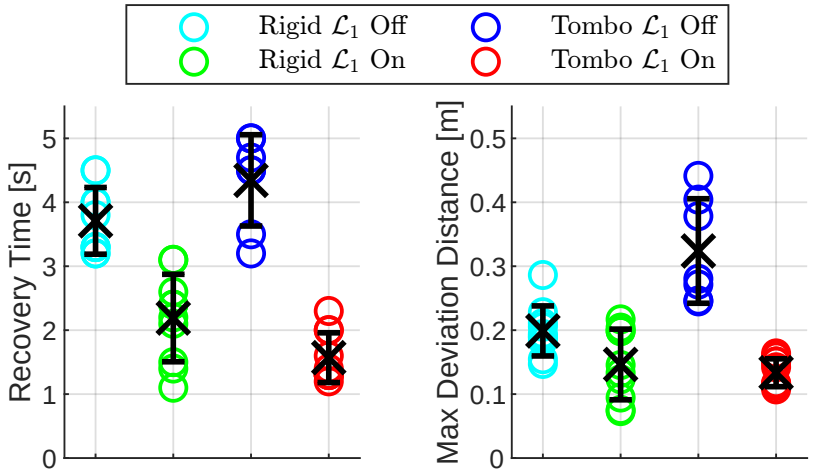


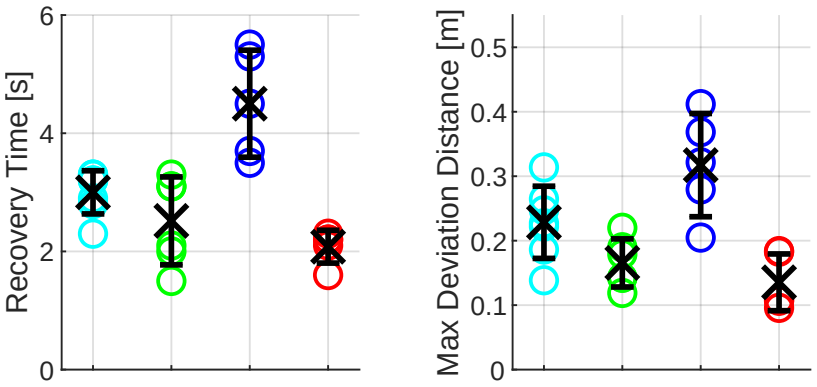
Figure 3.11: Altitude error before and after collision with the foam bar. The collision point is marked at 0 s on the graph. The "pre-collision mean error" refers to the average positional error before the collision. The "full recovery threshold" denotes the threshold for "Full recovery," which is offset by a distance of 0.01 m (0.02 m) for $\mathcal{L}_1$ on (off) from the pre-collision mean error.

3.4.2 Collision with a foam bar

After evaluating the performance in collision-free scenarios, the system's robustness under collision conditions is assessed, beginning with an impact involving a foam bar. The robustness of $\mathcal{L}_1$Quad is evaluated during random collisions between the quadrotor (equipped with Tombo and rigid propellers) and the foam bar. Four scenarios are tested: "Rigid $\mathcal{L}_1$ off", "Rigid $\mathcal{L}_1$ on", "Tombo $\mathcal{L}_1$ off", and "Tombo $\mathcal{L}_1$ on". These names indicate different combinations of equipped propellers and the state of $\mathcal{L}_1$ adaptive control. The quadrotor follows a 1-meter-radius horizontal circular trajectory with 1 m/s linear speed, which collides with a suspended foam bar (52 mm diameter, 1250 mm length) that intersects with



(a) Collision with a foam bar.



(b) Collision with a carbon fiber bar.

Figure 3.12: Statistics of recovery time and maximum deviation distance after collisions with (a) a foam bar and (b) a carbon fiber bar. The "×" symbol represents the mean value, and "I" represents the standard deviation for each case.

the circular trajectory (illustrated in Figure 3.10). Two metrics are used to evaluate the system's post-collision performance: recovery time and maximum deviation distance during collisions, defined as follows. Recovery time is defined as the duration from the instant of collision to the full recovery, as illustrated in Figure 3.11. The start of the collision is identified by the abrupt change in acceleration signals recorded by the onboard IMU. Full recovery is defined as the condition in which the quadrotor's altitude (position along the z-axis) falls within a threshold relative to the pre-collision steady-state; the specific threshold values are detailed in the caption of Figure 3.11. The recovery time is selected based solely on the z-axis tracking because experimental results indicate that, upon

collision, the largest tracking error occurs along the inertial z-axis and requires the longest time to return to the recovered state. Furthermore, the $\mathcal{L}_1$Quad controller directly compensates for uncertainties on the body z-axis, which makes it effective in suppressing tracking errors in the inertial z-axis when the quadrotor flies near the hover state. As a result, the most significant difference in recovery time occurs when evaluated by tracking errors on the inertial z-axis, whereas it remains comparable when evaluated by tracking errors on the inertial x- and y-axis, regardless of whether $\mathcal{L}_1$Quad is employed.

Meanwhile, the maximum deviation distance is defined as the maximum positional error of the quadrotor relative to the desired position during the recovery period defined above. Upon collision, Tombo propellers absorb impact forces, thereby reducing the overall force exerted on the quadrotor and mitigating trajectory deviation across all three axes. This effect is complemented by the $\mathcal{L}_1$Quad control law, which provides the most effective tracking error suppression along the inertial z-axis. Additionally, since collisions occur randomly, their effects on individual axes vary across different experiments. Consequently, the criterion for assessing deviation upon collision is based on the overall maximum deviation rather than deviations along an individual axis.

The results are statistically analyzed and shown in Figure 3.12(a). It is evident that the most favorable post-collision performance is observed in the case of "Tombo $\mathcal{L}_1$ on", followed by "Rigid $\mathcal{L}_1$ on", "Rigid $\mathcal{L}_1$ off", and the worst is "Tombo $\mathcal{L}_1$ off". This conclusion holds when it is examined by both metrics. The average collision recovery time in the four cases "Tombo $\mathcal{L}_1$ off", "Rigid $\mathcal{L}_1$ off", "Rigid $\mathcal{L}_1$ on", and "Tombo $\mathcal{L}_1$ on" is 4.34 s, 3.71 s, 2.19 s, and 1.57 s, respectively, while the average maximum deviation distance is 0.324 m, 0.199 m, 0.146 m, and 0.133 m, respectively.

The superior performance of "Tombo $\mathcal{L}_1$ on" is attributed to the combination of Tombo propellers and $\mathcal{L}_1$ adaptive control. First, the soft material on the edge and nodus of the Tombo propeller reduces the impact force on both the propeller

and the entire quadrotor. Second, $\mathcal{L}_1$ adaptive control's fast adaptation mitigates the impact with a rapid recovery. This is evident from the results showing better performance with both rigid and Tombo propellers when $\mathcal{L}_1$ is on. Additionally, the effectiveness of $\mathcal{L}_1$ adaptive control's compensation for the unique physical characteristics of the Tombo propellers is also clear from significantly different flight performance between "Tombo $\mathcal{L}_1$ on" (best) and "Tombo $\mathcal{L}_1$ off" (worst).

3.4.3 Collision with a carbon fiber bar

This experiment is set up similarly to the previous collision experiment, with the foam bar replaced by a carbon fiber bar, illustrated in Figure 3.10. The carbon fiber bar is a hollow rectangular tube with outer dimensions of 8×8 mm and a length of 1000 mm. A thin layer of foam is wrapped around the impact area of the carbon fiber bar to partially reduce the collision force, as neither the rigid propellers nor the Tombo propellers can survive a direct collision with the carbon fiber bar. The statistics of recovery time and the maximum deviation distance are summarized in Figure 3.12(b). The results in the four scenarios are similar to those observed in the foam-bar-collision experiments, where "Tombo $\mathcal{L}_1$ on" demonstrates the best performance evaluated by both metrics. The average collision recovery time for the four cases "Tombo $\mathcal{L}_1$ off", "Rigid $\mathcal{L}_1$ off", "Rigid $\mathcal{L}_1$ on" and "Tombo $\mathcal{L}_1$ on" are 4.5 s, 3.00 s, 2.52 s and 2.08 s, respectively. In contrast, the average maximum deviation distance was 0.317 m, 0.229 m, 0.166 m and 0.135 m, respectively.

Collisions with the carbon fiber bar often result in breakage at the tip of the rigid propellers. The severity depends on the position of the propeller at the moment of impact. However, due to the high rotational speed of the propellers, the breakage typically occurred very close to the tip, leading to insignificant impacts on the flight after the collision. In the case of "Rigid $\mathcal{L}_1$ on", the $\mathcal{L}_1$ adaptive control can effectively compensate for the minor thrust loss from the propeller tip

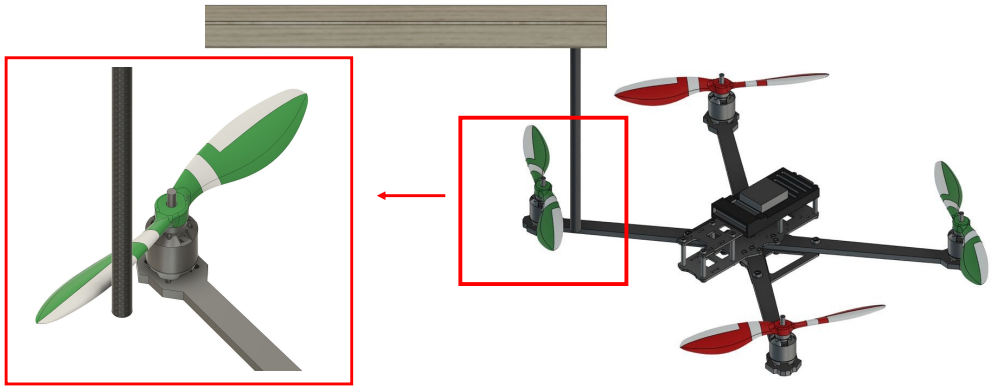


Figure 3.13: The severe collision setup uses a carbon fiber rod that intrudes into the propeller's path.

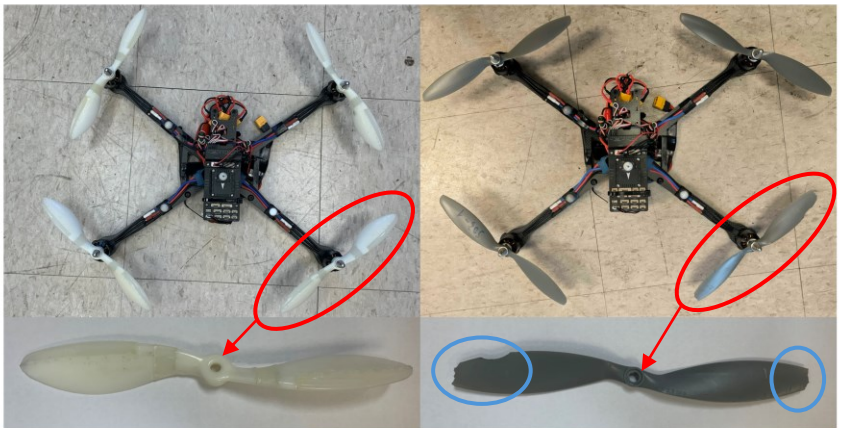


Figure 3.14: Quadrotor and propellers after severe collision. The rigid are broken in the area marked by the blue area.

damage. However, the impact forces in collision with the rigid propellers were greater than those with the Tombo propellers, resulting in the best response to be observed under "Tombo $\mathcal{L}_1$ on".

3.4.4 Severe collision during hover flight

After evaluating the effectiveness of $\mathcal{L}_1$ adaptive control and the Tombo propeller's capability to mitigate impacts from collisions with foam and carbon fiber bars, a more challenging scenario is designed and executed as follows. This experiment seeks to demonstrate further that the combination of $\mathcal{L}_1$ adaptive control with a Tombo propeller significantly enhances collision recovery capability, compared to the conventional usage of rigid propellers without the $\mathcal{L}_1$ adaptive control. This

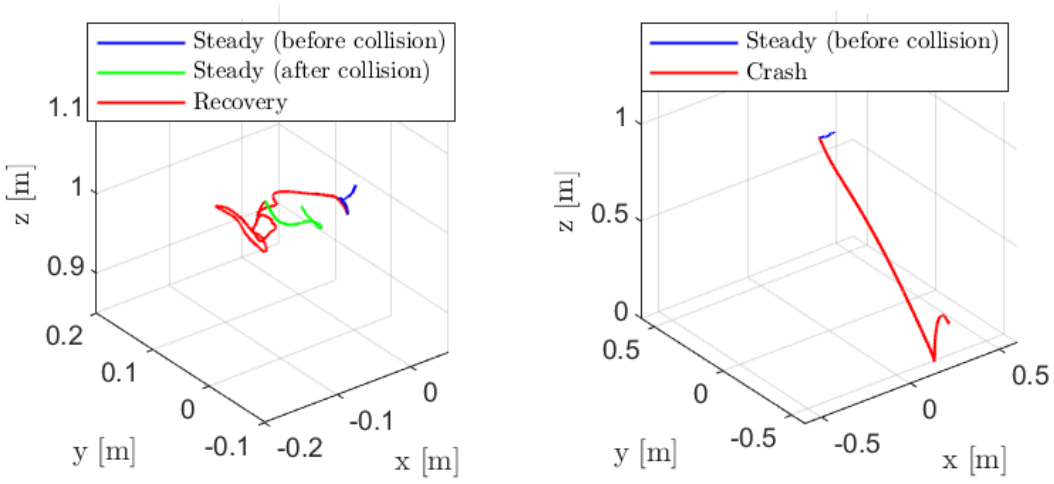
experiment is motivated by scenarios where a drone, during vertical flight, encounters obstacles such as branches or power lines that intrude into the propeller's path. The setup involved a 1058 mm × 28 mm × 28 mm wooden bar, vertically connected to a hollow carbon fiber rod of 238 mm in length and 5.47 mm in radius. The drone hovered 1 meter above its takeoff point, and the carbon fiber rod intruded into the propeller's path, which is illustrated in Figure 3.13.

The Tombo propeller, with its resilience to severe impacts, enabled the drone to withstand up to three severe collisions without breaking, ensuring that the propeller remained intact and capable of staying operational (shown in Figure 3.15(a)). This characteristic is particularly advantageous compared to rigid propellers, which fracture under the same conditions, leading to a loss of control (shown in Figure 3.15(b)). Conditions for both rigid and Tombo propellers after extreme collisions are shown in Figure 3.14. However, the same flexibility that allows the Tombo propeller to survive collisions also introduces unstable dynamics and perturbation thrusts into the system. The $\mathcal{L}_1$ adaptive control plays a crucial role in stabilizing the drone by effectively suppressing the oscillations that occur during sudden decelerations and accelerations of the propellers.

3.5 Summary and Discussion

This chapter introduced the use of the $\mathcal{L}_1$Quad control framework as a sensorless collision handling strategy for Tombo quadrotors. Through a series of controlled experiments, it was demonstrated that the combination of deformable Tombo propellers and $\mathcal{L}_1$ adaptive control effectively mitigates the effects of light or brief collisions. The $\mathcal{L}_1$Quad controller provided rapid recovery and significantly reduced trajectory deviations without the need for explicit collision detection.

However, it was also observed that in cases involving strong or prolonged collisions, sensorless control alone may be insufficient. When the drone experiences sustained thrust loss due to severe impacts, the controller's ability to compensate



(a) Tombo propeller.

(b) Rigid propeller.

Figure 3.15: Trajectory of the quadrotor in a severe collision with Tombo and rigid propellers. (a) The Tombo propeller withstands the second collision (among three collisions), where the steady state after the collision is defined as the tracking error $\leq$0.1 m after the collision. (b) The quadrotor with a rigid propeller crashes into the ground after the second collision.

becomes limited, increasing the risk of instability.

These findings highlight the need for a more comprehensive collision-handling strategy that can actively detect significant impacts and enable timely evasive actions. The next chapter will address this requirement by introducing a sensor-based collision detection and response framework using encoder feedback for enhanced collision awareness.

Chapter 4

Sensor-Based Collision Handling for Tombo Quadrotor

While the integration of $\mathcal{L}_1$Quad control with deformable Tombo propellers has shown promising results in mitigating the effects of minor collisions during trajectory-following flights, challenges remain when the quadrotor encounters strong or prolonged impacts. In such cases, significant and sustained reductions in propeller speed can prevent the controller from effectively compensating for thrust imbalances, leading to loss of stability or even crashes. This limitation highlights the need for a collision-aware mechanism that can detect severe impacts early and respond appropriately to prevent the vehicle from penetrating further into hazardous regions.

To address this challenge, this chapter presents a sensing and control framework that augments the existing $\mathcal{L}_1$Quad system with real-time collision detection based on encoder feedback. By continuously monitoring propeller speeds, the system can identify the occurrence and location of collisions, enabling timely evasive maneuvers to avoid deeper intrusion into obstacles and reduce the risk of failure.

The chapter is organized as follows: Section 4.1 develops a physically grounded dynamic model of the Tombo propeller under collision conditions, capturing the transient response of its hybrid rigid-soft structure. The model is validated through controlled experiments, confirming its ability to replicate observed deformation and velocity decay. Section 4.2 introduces an encoder-based collision

detection algorithm that leverages real-time rotational velocity data to identify impacted rotors with high accuracy. Section 4.3 proposes a post-collision control strategy that enables the quadrotor to recover from impact-induced disturbances. Initially demonstrated in hover, the method is extended to trajectory-tracking scenarios, enhancing flight resilience under unexpected collisions.

4.1 Collision dynamics of motorized deformable propellers

4.1.1 System Modeling

The electromechanical setup includes a deformable propeller, a BLDC motor, and an external object that interacts with the system during collisions, as depicted in Figure 4.1. During impact, the propeller is seen to bend into three separate parts—comprising the hub and two rigid plastic blades—with the bending occurring at the soft nodus region. Drawing from these observations, a physical model of the system is constructed, as illustrated in Figure 4.12. The dynamics are then derived using the Lagrangian method, allowing for the evaluation of the propeller's angular velocity throughout the collision process.

4.1.1.1 Modeling of BLDC Motor

The first dynamic model describes the BLDC motor using a structure similar to that of a regular direct current (DC) motor, without changing the motor's physical characteristics. This model includes important electrical components such as the armature resistance R_a, the winding inductance L_a, the back electromotive force, denoted as e_b, is directly associated with the rotor's angular velocity. The output torque, τ_m, is considered proportional to the armature current. Since the rotor also affects the motor's rotation during operation, its mechanical properties need to be included in the model. In particular, the moment of inertia of the rotor—a

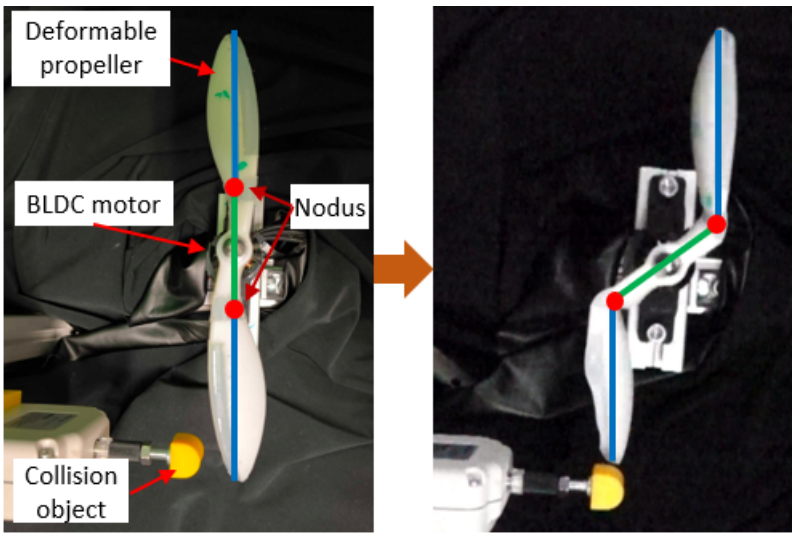


Figure 4.1: The deformation of the propeller when collisions happen with top view. The propeller folds into three segments at the locations of the nodus, similar to a three-link mechanism with the hubs acting as joints.

key mechanical value—is calculated using SolidWorks, as shown in Figure 4.2. The values used for modeling are based on the technical data of the SURPASS C2216-1120 Kv motor (Goolsky), presented in Table 4.1.

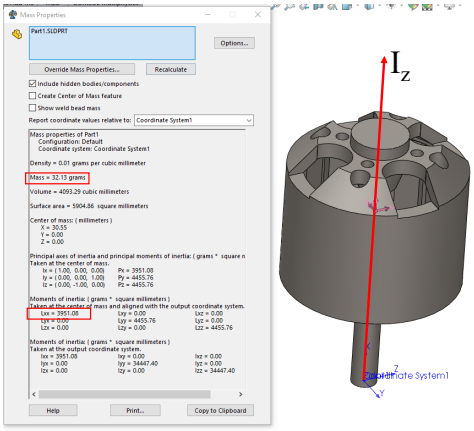


Figure 4.2: Rotor motor's moment of inertia about the rotational axis (SolidWorks)

4.1.1.2 Modeling of Deformable Propeller

As illustrated in Figure 4.1, Tombo propeller employed in this study features a tapered geometry with an overall length of approximately 230 mm. The propeller is designed with two localized soft hinges, each 12 mm wide, fabricated from

Table 4.1: Parameters of the motor

Symbols	Definitions	Values
K_t	Torque constant	0.00853 Nm/A
K_b	Back emf constant	0.00853 Vs/rad
R_a	Armature resistance	0.18 Ω
L_a	Armature Inductance	0.000085 H
b_m	Viscous damping factor of motor	0.000049 Nms/rad
J_r	Moment of inertia of the rotor motor	0.00000395108 kgm^2
m_r	Mass of rotor motor	0.03213 kg

Dragon Skin 30 silicone (Smooth-On). Under nominal operating conditions, the propeller behaves primarily as a rigid structure; however, during collision events, significant deformation is concentrated at these soft hinge regions, resulting in a multi-segment propeller geometry.

The soft hinge regions of Dragon Skin 30 exhibit a pronounced viscoelastic behavior, allowing the material to undergo large deformations without permanent deformation. From a materials science perspective, this silicone compound is known for its high elongation at break, low Shore hardness, and stable energy dissipation characteristics, making it well-suited for repetitive deformation cycles such as those induced by mid-flight collisions. By localizing the deformable zones to short (12 mm) hinge segments within an otherwise rigid blade, the design achieves both structural integrity and controlled compliance. This configuration requires a mechanical model that accurately reflects the behavior of the hinge-level material while remaining tractable within a full dynamic simulation. To that end, each hinge is modeled as a viscoelastic torsional joint with experimentally identified stiffness and damping coefficients, thus abstracting the nonlinear stress-strain behavior of the material into parameters suitable for incorporation within the Lagrangian dynamics framework.

Several modeling approaches were considered for capturing the mechanical behavior of the deformable propeller, with particular focus on the localized nature of the hinge deformation. Beam-based models such as Euler–Bernoulli or Timoshenko formulations [75] (see Figure 4.3) offer accurate representations

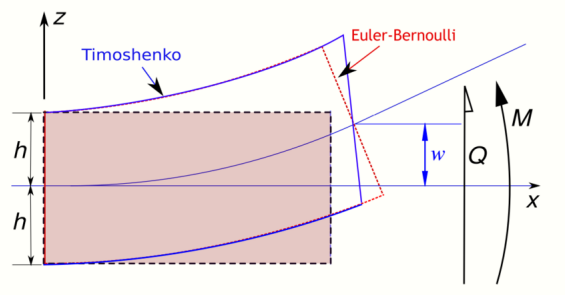


Figure 4.3: Timoshenko beam theory.

for slender structures undergoing distributed bending. However, in our case, deformation is not uniformly distributed but instead highly localized at two hinge regions. Furthermore, the propeller's tapered geometry and material heterogeneity (rigid blade body vs. soft silicone hinge) violate the assumptions of uniform cross-section and homogeneous elasticity underlying traditional beam theory.

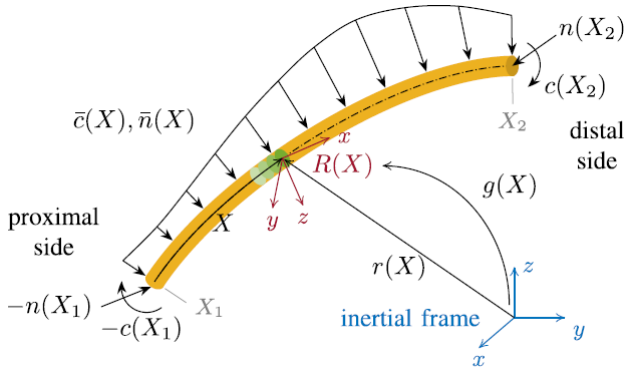


Figure 4.4: Cosserat rod model for continuous deformation analysis.

Cosserat rod models [76, 77] (see Figure 4.4) provide a more generalized description of large-deformation behavior in slender, compliant structures and can capture bending, torsion, and shear in geometrically exact form. However, the computational cost and formulation complexity of Cosserat rod theory — especially when coupled with motor dynamics and transient collision forces — render it impractical for our application, which requires real-time integration with a Lagrangian dynamics solver.

Finite Element Method (FEM)-based approaches offer full nonlinear material modeling capabilities and have been successfully applied in cases where the

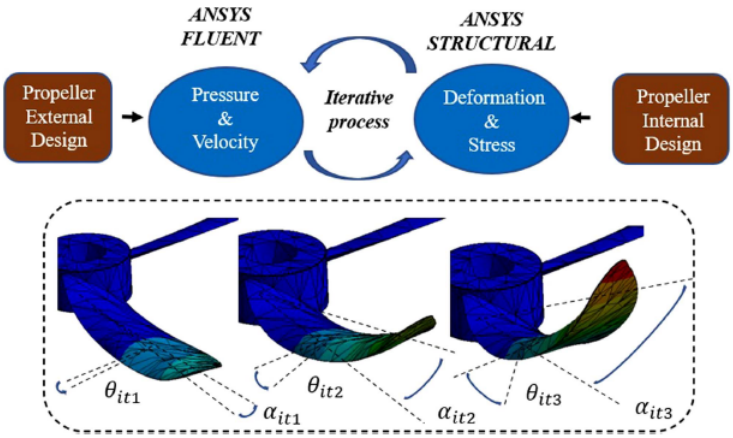


Figure 4.5: Illustration of a fluid-structure interaction modeling approach using FEM for a fully deformable PET propeller.

entire propeller is made of compliant materials (e.g., PET-based propellers [4], Figure 4.5). However, in our case, where deformation is concentrated at short, silicone-based hinge regions within an otherwise rigid blade, applying FEM to the full blade would introduce unnecessary complexity and is not suitable for efficient dynamics simulation or control integration.

To balance mechanical fidelity with modeling efficiency, a reduced-order multi-body dynamics approach is adopted, capturing the essential behavior of the propeller by representing it as a series of rigid links connected via rotational joints located at the soft hinge points. Each joint is characterized by experimentally determined torsional stiffness and damping, representing the viscoelastic response of the silicone hinge material. This modeling choice preserves the dominant physics observed in collision scenarios, namely localized, reversible deflection with energy dissipation, while enabling integration with the motor dynamics and external collision forces in a unified second-order Lagrangian framework.

The selected model enables real-time simulation and provides an effective foundation for subsequent control design and collision detection. In the following, the dynamic model of the deformable propeller is developed accordingly.

The Tombo propeller comprises a rigid hub, two rigid wings, two soft blades, and two nodus containing tendons [7]. Based on this structure, the propeller

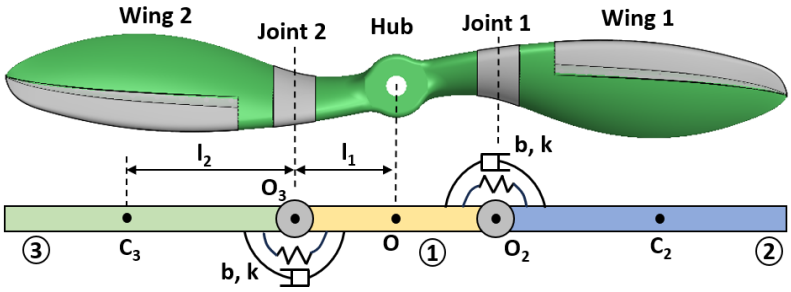


Figure 4.6: Modeling of deformable propeller

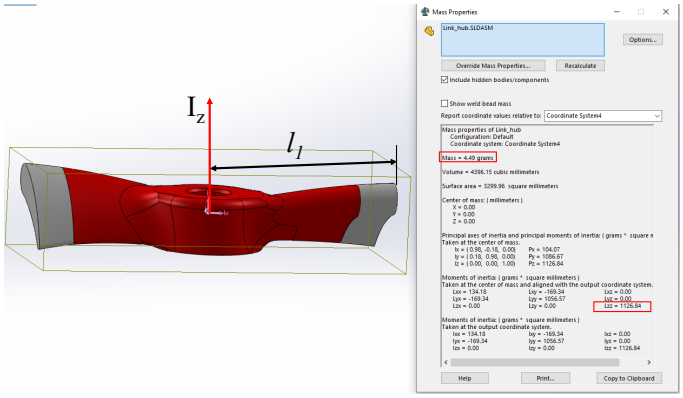


Figure 4.7: Inertia moment J_1 and haft-length l_1 of link hub (SolidWorks)

is modeled as a three-link mechanism, as illustrated in Figure 4.6. The hub is designated as Link 1, with its rotational axis centered at point O. The wings, comprising both rigid and soft parts, are modeled as Link 2 and Link 3, respectively. The centers of mass of Link 2 and Link 3 are denoted as C_2 and C_3. The two soft nodus serve as revolute joints located at O_2 and O_3, each modeled as a torsional spring (k) and damper (b).

Table 4.2: Parameters of propeller

Symbols	Definitions	Values
m_1	Mass of link 2, 3	$0.00449\,kg$
J_1	Moment of inertia of link 1	$0.00000112684\,kgm^2$
l_1	Half-length of link 1	0.029 m
m_2	Mass of link 2, 3	$0.00644\,kg$
J_2	Moment of inertia of link 2, 3	$0.00000275997\,kgm^2$
l_2	Length from the joints to CoM of links 2, 3	0.03537 m
b	Viscous damping factor of joint	0.005 Nms/rad
k	Stiffness factor of joint	0.1 Nm/rad

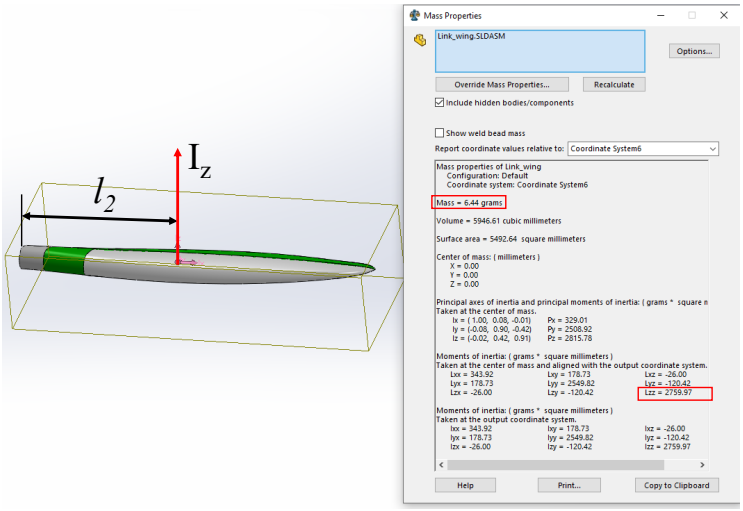


Figure 4.8: Mass m_2 and inertia moment J_2 of link wing (SolidWorks)

Figure 4.9: The experiment setup for stiffness of joint k.

This section employs the deformable propeller configuration proposed in [7], which incorporates soft components fabricated from DragonSkin 30. Five internal tendons, each with a diameter of 0.9 mm, are embedded within the hub. The propeller is designed to be deformable, with limited stiffness at the soft joints that allow the propeller to deflect upon impact. The mass, inertia and geometric properties of each link are determined using SolidWorks, as shown in Figures 4.7 and 4.8. The stiffness parameters of the joints are experimentally identified, as illustrated in Figure 4.9, and the damping coefficients are obtained by fitting the model to simulation data. All these characteristics of the propeller are summarized in Table 4.2.

4.1.1.3 Calculating drag force

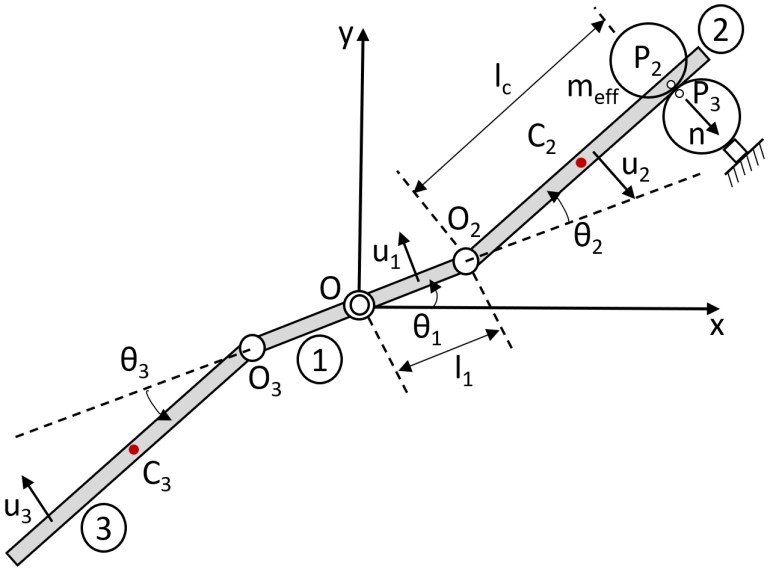


Figure 4.10: Contact between propeller and external object. In which, the effective mass (m_{eff}) represents mass of propeller in impact model.

Since the deformable propeller is analyzed using a two-dimensional model, it is subjected solely to drag forces, as illustrated in Figure 4.6. In the previous section, the deformable propeller was represented using a three-link model. Consequently, the drag force acting on the structure varies in both magnitude and application point depending on the velocity of each link. For link 1, with the pivot at the center, the drag force acting on link 1 is equivalently converted into a moment acting on link 1, denoted as τ_{d1}. Meanwhile, the drag forces acting on links 2 and 3 have magnitudes F_{d2} and F_{d3}, respectively, referring to formula 7 in [7], and are positioned at distances from the joints $l_{d2} = O_2D_2$ and $l_{d3} = O_3D_3$, respectively.

4.1.1.4 Calculating collision force

1) Effective mass: For mechanical systems with a single degree of freedom, the effective mass can be easily computed by analyzing the kinetic energy. However, this task becomes significantly more complex for deformable propellers, which possess multiple degrees of freedom. The complexity is further increased by the fact that the propeller forms part of a constrained multibody system. In such con-

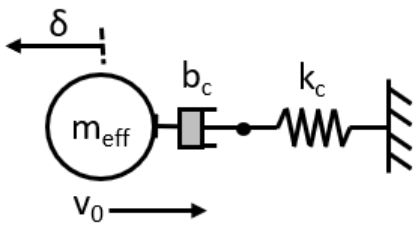


Figure 4.11: Maxwell contact force model.

figurations, evaluating the effective mass during a collision necessitates accounting for variations in the moment of inertia of the interconnected components due to joint constraints.

To address this, the impulse-momentum method is applied to compute the effective mass of the deformable propeller, which is modeled as a multibody system [78] (see Equation 39).

$$m_{eff} = \frac{1}{\mathbf{b}_{ij}^T \mathbf{M}_{dof}^{-1} \mathbf{b}_{ij}}, \tag{4.1}$$

where m_{eff} is effective mass of the multibody system, $\mathbf{M}_{dof}$ is inertia matrix of propellers as follows:

$$\mathbf{M}_{dof} = \mathbf{B}^T \mathbf{M} \mathbf{B}, \tag{4.2}$$

where $\mathbf{M}$ denotes mass matrix of deformable propeller and $\mathbf{B}$ depicts the null-space

of the constraint Jacobian $\mathbf{D}$, $\mathbf{DB} = \mathbf{0}$

$$\mathbf{M}_{(9\times9)} = \begin{bmatrix} m_1+m_r & 0 & & & & \cdots & & & 0 \\ 0 & m_1+m_r & & & & & & & \\ & & J_1+J_r & & & & & & \\ & & & & m_2 & & \ddots & & \vdots \\ & & & & & m_2 & & & \\ \vdots & & & & \ddots & & J_2 & & \\ & & & & & & & m_2 & \\ & & & & & & & m_2 & 0 \\ 0 & & & & & \cdots & & 0 & J_2 \end{bmatrix}, \tag{4.3}$$

$$\mathbf{B} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ -l_1\sin\theta_1 - l_2\sin\theta_{12} & -l_2\sin\theta_{12} & 0 \\ l_1\cos\theta_1 + l_2\cos\theta_{12} & l_2\cos\theta_{12} & 0 \\ 1 & 1 & 0 \\ l_1\sin\theta_1 + l_2\sin\theta_{13} & 0 & l_2\sin\theta_{13} \\ -l_1\cos\theta_1 - l_2\cos\theta_{13} & 0 & -l_2\cos\theta_{13} \\ 1 & 0 & 1 \end{bmatrix}, \tag{4.4}$$

where $\sin\theta_{12} = \sin(\theta_1+\theta_2)$; $\cos\theta_{12} = \cos(\theta_1+\theta_2)$ $\sin\theta_{13} = \sin(\theta_1+\theta_3)$; $\cos\theta_{13} = \cos(\theta_1+\theta_3)$.

The matrix $\mathbf{b}_{ij}$ is obtained by projecting the relative velocity between points P_2 and P_3 onto the direction of impact, represented by the unit vector $\mathbf{n}$ (see Figure 4.10), as shown below:

$$\mathbf{n}^T(-l_1\dot{\theta}_1\mathbf{u}_1 + l_c\dot{\theta}_2\mathbf{u}_2 - 0\mathbf{u}_3) = \dot{\delta}, \tag{4.5}$$

where l_c denotes the distance between the collision point and the joint of the associated link. In this analysis, the distance is set to $l_c = 0.071\,\text{m}$. The above expression can be reformulated in matrix form as:

$$\begin{bmatrix} -l_1 \cos\theta_2 - l_c & -l_c & 0 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} = \dot{\delta}, \tag{4.6}$$

where

$$\mathbf{b}_{ij}^T = \begin{bmatrix} -l_1 \cos\theta_2 - l_c & -l_c & 0 \end{bmatrix}. \tag{4.7}$$

2) Impact model: In the simulation section, the impact force is treated using numerical methods. Specifically, it is represented as an impulse applied at discrete time steps corresponding to collision events. The impact force F_c is determined using the Maxwell model (see Figure 4.11), which consists of a dashpot (b_c) and a spring (k_c) refer in [79]. The parameters are selected based on experimental observations and tuned to replicate the actual impact response of the propeller, with $b_c = 200\,Ns/m$ and $k_c = 18000\,N/m$. For an object with effective mass m_{eff} and an initial normal relative velocity v_0 (linear velocity of collision point P_2), the resulting impact force is computed using Equation 9.5 from Chapter 9 in [79] as follows:

$$F_c = (1-\zeta^2)^{-1/2} m_{eff}\omega_0 v_0 e^{-\zeta\omega_0 t} sin(\omega_d t), \tag{4.8}$$

where ζ denotes the damping ratio and ω_0 represents the natural angular frequency, which are calculated as:

$$\zeta = \frac{m_{eff}\omega_0}{2b_c}, \tag{4.9}$$

$$\omega_0^2 = \frac{k}{m_{eff}}. \tag{4.10}$$

4.1.1.5 Lagrange's Equation

In the Lagrangian approach, energy plays a key role in formulating the equations of motion for complex systems, particularly those involving both mechanical and electrical components.

The core concept is based on the Lagrangian function, which is defined as the difference between the kinetic energy T and the potential energy U, expressed as:

$$L = T - U. \tag{4.11}$$

Lagrange's equations are formulated using a set of n independent variables known as generalized coordinates, denoted as $\{q_1, ..., q_i, ..., q_n\}$. The second-order Lagrange equations incorporating the dissipation function are formulated as follows:

$$\frac{d}{dt}(\frac{\partial L}{\partial \dot{q}_i}) - \frac{\partial L}{\partial q_i} + \frac{\partial R}{\partial \dot{q}_i} = Q_i, \tag{4.12}$$

where R is a simplified form of Rayleigh's dissipation function, energy is usually dissipated in friction or damper for mechanical systems. In electrical systems, energy is dissipated in resistors. Q_i denotes the generalized external force acting on the system of q_i coordinates. If total virtual work done ∂W causes virtual displacement ∂q_i for q_i coordinate, generalized force Q_i is calculated as follow:

$$Q_i = \frac{\partial W}{\partial q_i}. \tag{4.13}$$

4.1.1.6 Lagrange's Equation of System

To describe the physical motion of the system, the generalized coordinates are selected as the charge q_1 and the angular positions of the links θ_1, θ_2, and θ_3. Specifically, these are defined as $q_1 = q_1$, $q_2 = \theta_1$, $q_3 = \theta_2$, and $q_4 = \theta_3$, as illustrated in Figure 4.12. The kinetic and potential energies of the electrical

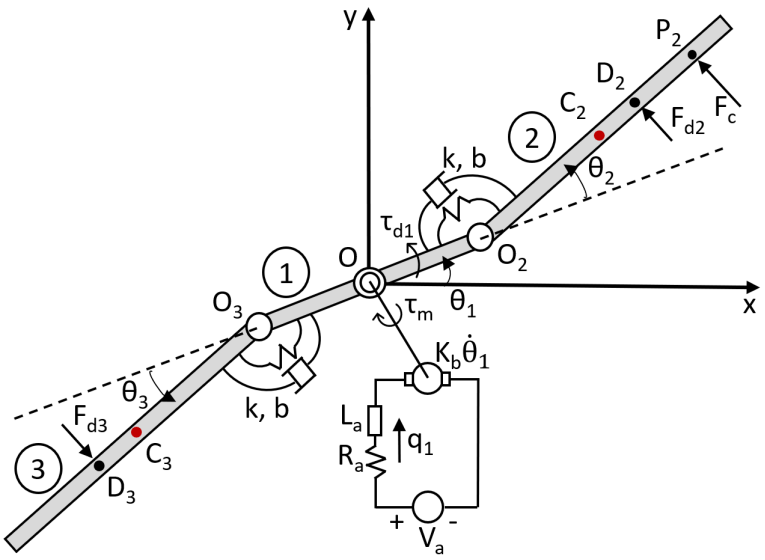


Figure 4.12: Modeling of BLDC motor-deformable propeller system.

components are given by:

$$T_e = \frac{1}{2} L_a \dot{q}_1^2, \tag{4.14}$$

$$U_e = -(V_a - e_b) q_1 = -(V_a + K_b \dot{\theta}_1) q_1, \tag{4.15}$$

where $e_b = -K_b \dot{\theta}_1$ because link 1 rotates counterclockwise. The kinetic and potential energies of the mechanical components are given by:

$$T_1 = \frac{1}{2}(J_1 + J_r)\dot{\theta}_1^2, \tag{4.16}$$

$$T_2 = \frac{1}{2} m_2 l_1^2 \dot{\theta}_1^2 + \frac{1}{2}(m_2 l_2^2 + J_2)(\dot{\theta}_1 + \dot{\theta}_2)^2 + m_2 l_1 l_2 \cos\theta_2 (\dot{\theta}_1^2 + \dot{\theta}_1 \dot{\theta}_2), \tag{4.17}$$

$$T_3 = \frac{1}{2} m_2 l_1^2 \dot{\theta}_1^2 + \frac{1}{2}(m_2 l_2^2 + J_2)(\dot{\theta}_1 + \dot{\theta}_3)^2 + m_2 l_1 l_2 \cos\theta_3 (\dot{\theta}_1^2 + \dot{\theta}_1 \dot{\theta}_3), \tag{4.18}$$

$$U_1 = \frac{1}{2} k \theta_2^2, \tag{4.19}$$

$$U_2 = \frac{1}{2} k \theta_3^2. \tag{4.20}$$

By replacing equations 4.14-4.20 into Lagrangian function, its form can be rewritten as follows:

$$\begin{aligned} L = & \frac{1}{2}L_a\dot{q}_1^2 + \left(\frac{1}{2}J_1 + \frac{1}{2}J_r + m_2l_1^2 + J_2 + m_2l_2^2\right)\dot{\theta}_1^2 \\ & + \frac{1}{2}(J_2 + m_2l_2^2)\left(\dot{\theta}_2^2 + \dot{\theta}_3^2 + 2\dot{\theta}_1\dot{\theta}_2 + 2\dot{\theta}_1\dot{\theta}_3\right) \\ & + m_2l_1l_2\cos\theta_2\left(\dot{\theta}_1^2 + \dot{\theta}_1\dot{\theta}_2\right) + m_2l_1l_2\cos\theta_3\left(\dot{\theta}_1^2 + \dot{\theta}_1\dot{\theta}_3\right) \\ & + (V_a + K_b\dot{\theta}_1)q_1 - \frac{1}{2}k\theta_2^2 - \frac{1}{2}k\theta_3^2. \end{aligned} \tag{4.21}$$

Here, Rayleigh dissipative function in the system can be formulated as follows:

$$R = R_e + R_1 + R_2 + R_3 = \frac{1}{2}R_a\dot{q}_1^2 + \frac{1}{2}b_m\dot{\theta}_1^2 + \frac{1}{2}b\dot{\theta}_2^2 + \frac{1}{2}b\dot{\theta}_3^2, \tag{4.22}$$

while the total virtual work done by the virtual displacements can be defined as:

$$\begin{aligned} \partial W = & [-K_t\dot{q}_1 + \tau_{d1} + (F_{d2} + F_c)l_1\cos\theta_2 + F_{d2}l_{d2} + F_cl_c + \\ & F_{d3}l_1\cos\theta_3 + F_{d3}l_{d3}]\partial\theta_1 + (F_{d2}l_{d2} + F_cl_c)\partial\theta_2 + F_{d3}l_{d3}\partial\theta_3, \end{aligned} \tag{4.23}$$

where the motor creates torque τ_m, which is replaced by $K_t\dot{q}_1$; τ_1 is the aerodynamic on link 1. The symbols F_{d2}, l_{d2}, F_{d3}, l_{d3} are drag force and its position, which are applied on link 2 and link 3, respectively. The symbols F_c, and l_c are collision force and its position. Each term of the Lagrange equation is derived for the generalized coordinates q_1, θ_1, θ_2, and θ_3. Substituting these into equation 4.12 yields:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} = \begin{bmatrix} t_{11} \\ t_{21} \\ t_{31} \\ t_{41} \end{bmatrix}, \tag{4.24}$$

where:

$$a_{11} = L_a, \tag{4.25}$$

$$a_{12} = a_{13} = a_{14} = a_{21} = a_{31} = a_{41} = a_{34} = a_{43} = 0, \tag{4.26}$$

$$a_{22} = J_1 + J_r + 2J_2 + 2m_2(l_1^2 + l_2^2 + l_1 l_2(\cos\theta_2 + \cos\theta_3)), \tag{4.27}$$

$$a_{23} = J_2 + m_2 l_2^2 + m_2 l_1 l_2 \cos\theta_2, \tag{4.28}$$

$$a_{24} = J_2 + m_2 l_2^2 + m_2 l_1 l_2 \cos\theta_3, \tag{4.29}$$

$$a_{32} = J_2 + m_2 l_2^2 + m_2 l_1 l_2 \cos\theta_2, \tag{4.30}$$

$$a_{33} = J_2 + m_2 l_2^2, \tag{4.31}$$

$$a_{42} = J_2 + m_2 l_2^2 + m_2 l_1 l_2 \cos\theta_3, \tag{4.32}$$

$$a_{44} = J_2 + m_2 l_2^2, \tag{4.33}$$

$$t_{11} = V_a + K_b\dot{\theta}_1 - R_a\dot{q}_1, \tag{4.34}$$

$$\begin{aligned} t_{21} = &- K_t\dot{q}_1 + \tau_{d1} + F_{d2}l_1\cos\theta_2 + F_{d2}l_{d2} + F_{d3}l_1\cos\theta_3 + F_{d3}l_{d3} + \\ &F_c l_1\cos\theta_2 + F_c l_c + 2m_2 l_1 l_2\sin\theta_2\dot{\theta}_1\dot{\theta}_2 + 2m_2 l_1 l_2\sin\theta_3\dot{\theta}_1\dot{\theta}_3 + \\ &m_2 l_1 l_2\sin\theta_2\dot{\theta}_2^2 + m_2 l_1 l_2\sin\theta_3\dot{\theta}_3^2, \end{aligned} \tag{4.35}$$

$$t_{31} = F_{d2}l_{d2} + F_c l_c - m_2 l_1 l_2\sin\theta_2\dot{\theta}_1^2 - k\theta_2 - b\dot{\theta}_2, \tag{4.36}$$

$$t_{41} = F_{d3}l_{d3} - m_2 l_1 l_2\sin\theta_3\dot{\theta}_1^2 - k\theta_3 - b\dot{\theta}_3. \tag{4.37}$$

4.1.2 Experimental Validation of Collision Dynamics

4.1.2.1 Experimental Setup

Figure 4.13 shows the experimental configuration used to investigate collisions involving the deformable propeller. The propeller is driven by a BLDC motor (SURPASS C2216-1120 Kv, Goolsky). Its angular velocity is measured by a rotary incremental encoder (AMT10E3, Same Sky), which supports 16 resolution settings via dual in-line package (DIP) switches, allowing for flexibility across various speed ranges. The collision target is rigidly attached to a force gauge

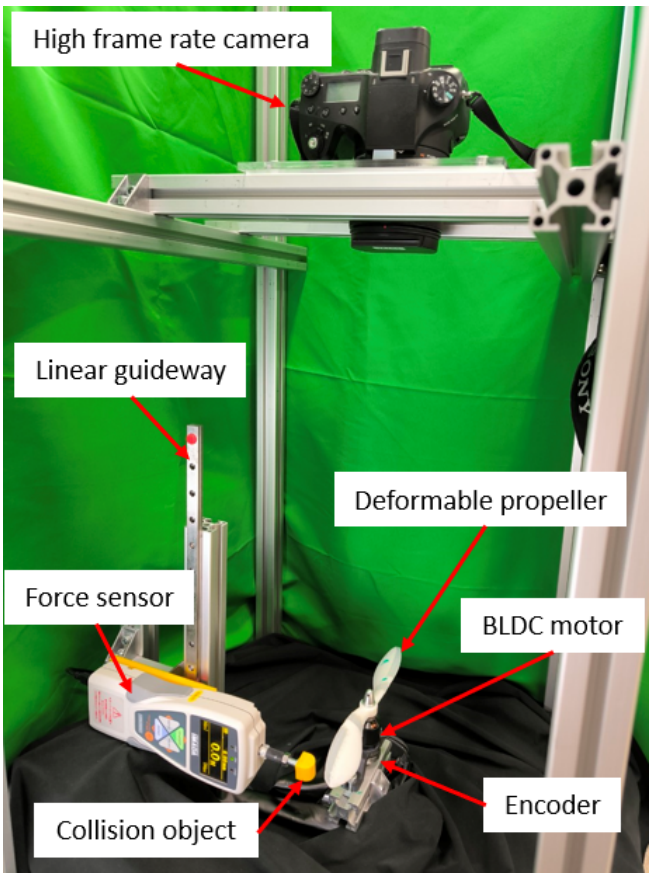


Figure 4.13: Devices for collision experiment.

(ZTS-500N, IMADA), capable of capturing data at a sampling rate of 2000 Hz with a resolution of 0.1 N.

During the experiment, the deformable propeller spins while the force gauge is allowed to fall freely along a guided linear rail, resulting in spontaneous collisions with the propeller blades. A high-speed camera (Sony RX10) capturing at 960 frames per second is employed to record both the collision event and the following recovery process of the propeller system.

4.1.2.2 Shape Deformation and Recovery

Figure 4.14 presents the deformation and subsequent recovery of the propeller following collision events, with key time points marked in red. Each experimental case begins at the moment of impact (0 ms) and concludes once the propeller has fully reverted to its original configuration. The five scenarios involve collision tests

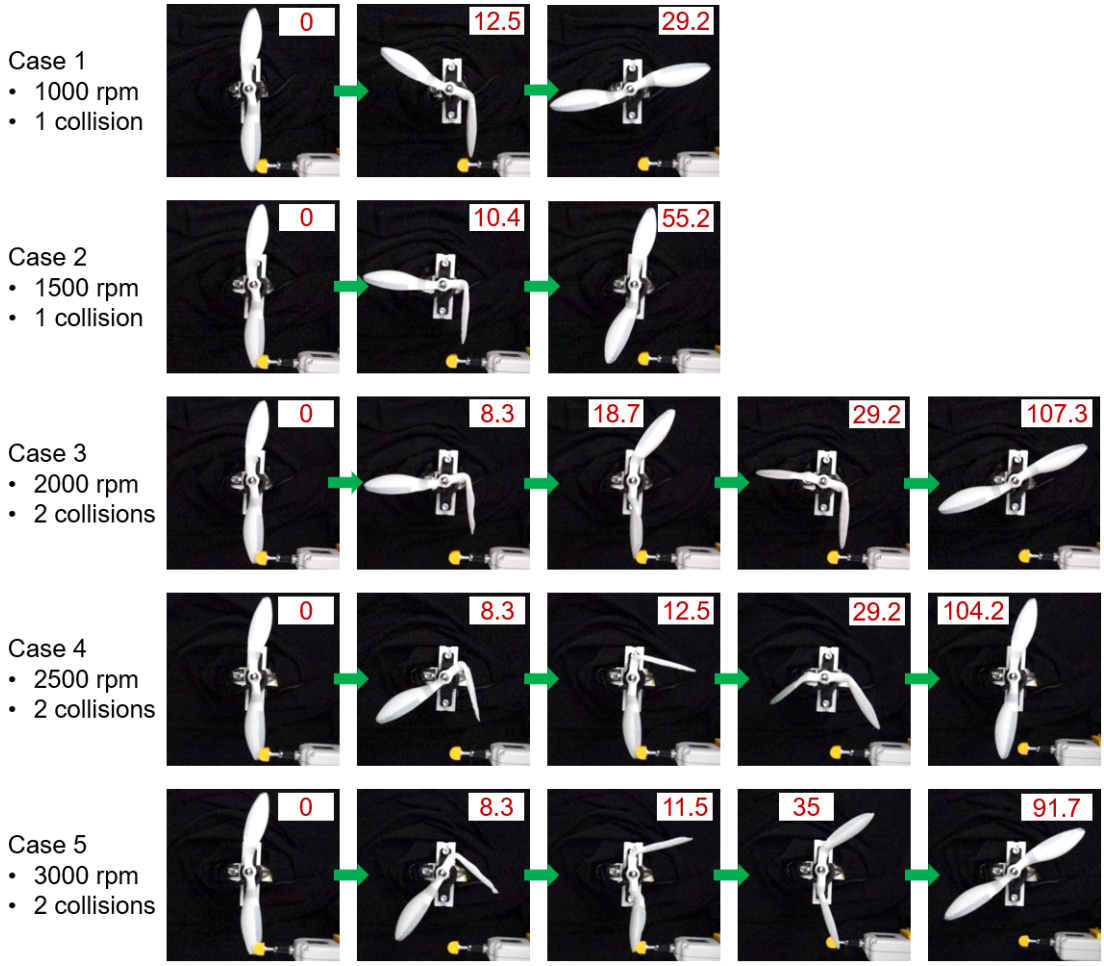


Figure 4.14: Deformation of the propeller during collisions at different initial rotational speeds, shown over time (ms). In scenarios 3, 4, and 5, the propeller experiences two successive impacts on opposite sides.

conducted at varying rotational speeds, from 1000 rpm to 3000 rpm. While only one impact is recorded in Cases 1 and 2, Cases 3 through 5 show dual impacts occurring on opposing sides of the propeller.

During the impact phase, the flexible regions of the blade typically bend by approximately $\pi/2$ radians. Depending on the impact conditions, the duration required for the blade to regain its original shape ranges from 29 ms to 107 ms. At elevated rotational speeds, the time gap between sequential impacts shortens, measured at 18.7 ms, 12.5 ms, and 11.5 ms for Cases 3, 4, and 5, respectively. These findings emphasize the propeller's capacity for rapid deformation and recovery, ensuring aerodynamic function even after multiple impacts.

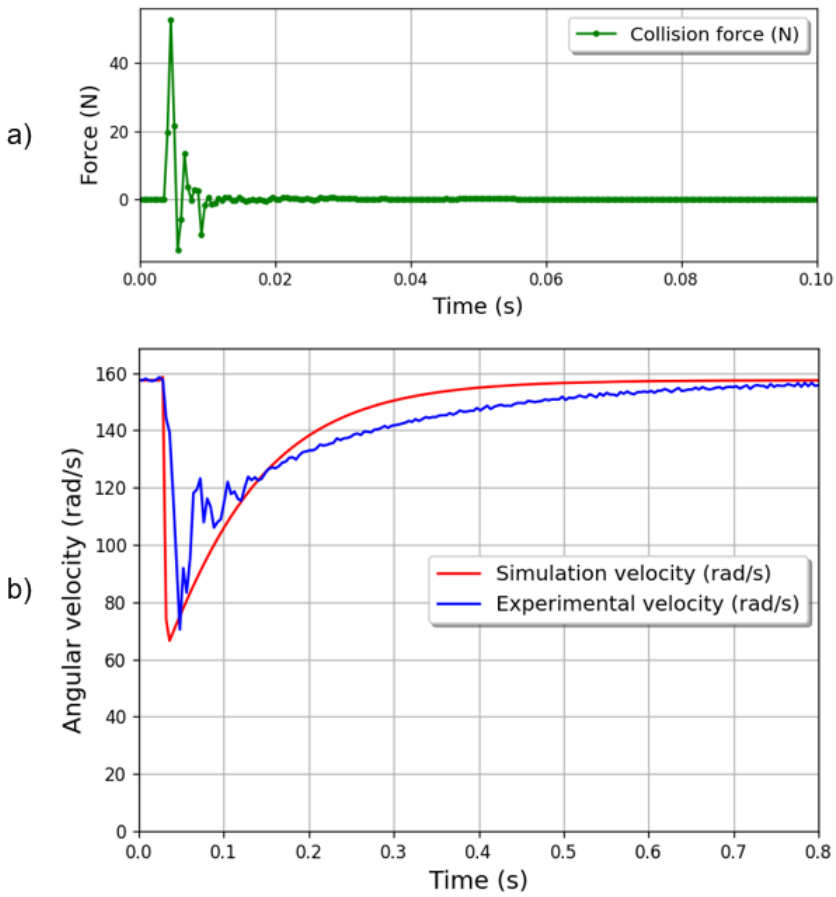


Figure 4.15: Experimental results at 1500 rpm: a) Impact force measurement, b) Angular velocity of the propeller (link 1). A single collision event is observed in this scenario.

4.1.2.3 Angular Velocity and Impact Force Analysis

Figures 4.15 and 4.16 present the angular velocity and collision force profiles for Cases 2 and 5, respectively. The angular velocity of link 1, represented as $\dot{\theta}_1$, is derived from simulations by numerically solving the system of equations described in 4.24. The applied input voltages are 1.83 V for Case 2 and 4.3 V for Case 5. The force gauge samples data at 2000 Hz, while the encoder operates at 250 Hz.

For Case 2, involving a single impact, $\dot{\theta}_1$ rapidly decreases from 157 rad/s to around 70 rad/s within 0.02 s and gradually returns to its original value after approximately 0.7 s. The peak force recorded is 52.7 N. In contrast, Case 5 includes two consecutive collisions, where angular velocity drops from 308 rad/s to a low of 88 rad/s in 0.012 s, then recovers to the initial speed in roughly 0.6 s. The two

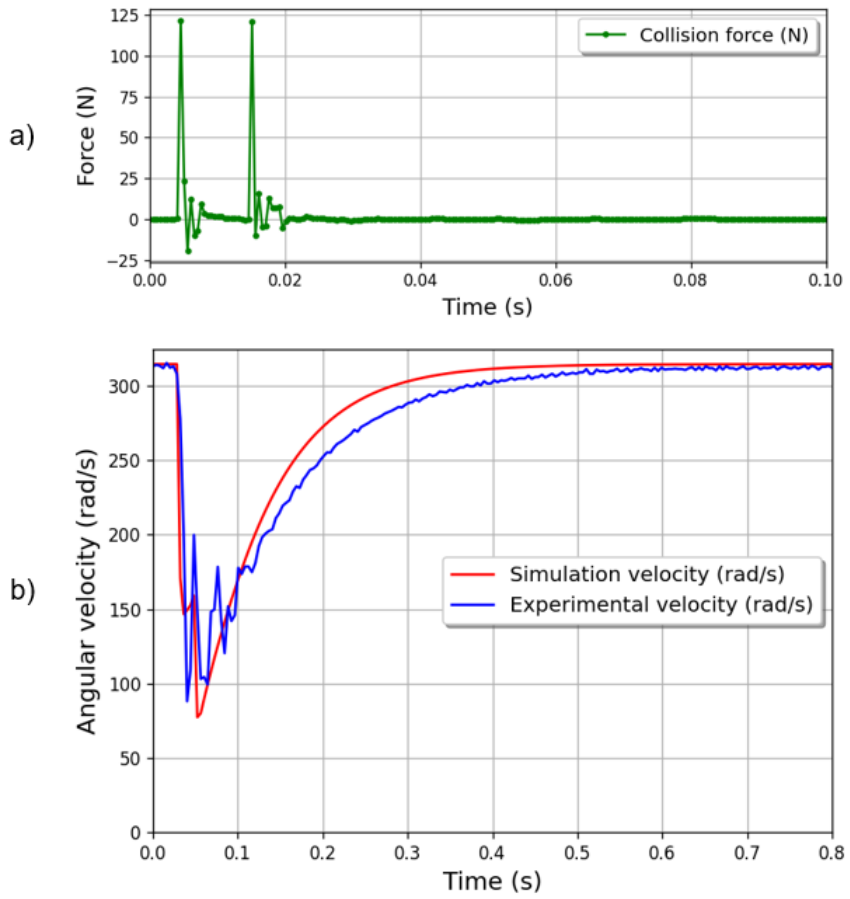


Figure 4.16: Experimental results at 3000 rpm: a) Measured collision force, b) Angular velocity of the propeller (link 1). Two successive impacts are observed on opposite sides of the propeller.

highest measured forces are 121.4 N and 120.6 N, respectively. Similar trends are evident in other test cases, where higher rotational speeds and repeated impacts result in greater reductions in angular velocity. These findings serve as the basis for further investigation into the system's dynamic response to collisions.

4.1.2.4 Summary of Simulation and Experimental Observations

The simulation and experimental findings consistently indicate that the propeller's angular velocity undergoes a rapid decline immediately following a collision. This trend is observed in all tested scenarios and becomes increasingly pronounced with higher initial rotational speeds and multiple impact events. The sharp drop in angular velocity, typically occurring within 12–20 ms, highlights its strong potential as a reliable metric for real-time collision detection.

Additionally, the results demonstrate that the time required for the propeller to regain its original shape is consistently shorter than the time needed for its angular velocity to recover. Specifically, the aerodynamic structure of the blade is generally restored within 30–100 ms, whereas the angular velocity may take up to 0.8 s to return to pre-collision levels. This temporal relationship suggests that the restoration of aerodynamic form precedes and supports the gradual recovery of thrust. As a result, this coordination between shape restitution and velocity ramp-up contributes to a smoother and more stable thrust regeneration process, thereby enhancing the drone's post-collision recovery performance.

4.2 Encoder-Based Collision Detection System

4.2.1 System Overview

This section describes the full hardware configuration mounted on the drone. The hardware setup consists of all essential components required for the off-board control of the quadrotor. To monitor the propeller speed and enable collision detection, an encoder system along with a signal processing unit is incorporated. The complete hardware assembly weighs 1,280 g and features an arm length of 233 mm. A detailed overview of the components is shown in Figure 4.17.

4.2.2 Quadrotor Hardware Setup

A custom-built quadrotor based on the S5 (Holybro) platform is utilized. Its propulsion system consists of four SURPASS 2216 1120KV BLDC motors (Goolsky), each equipped with a Tombo propeller (see Figure 1.4). These specially designed propellers enhance flight safety by absorbing impact forces during collisions, thereby minimizing structural damage. Motor speeds are controlled by four Swift 20A 2-4S BLHeli brushless ESC units (ARRIS), which interpret control signals from the flight management unit and adjust power delivery accordingly.

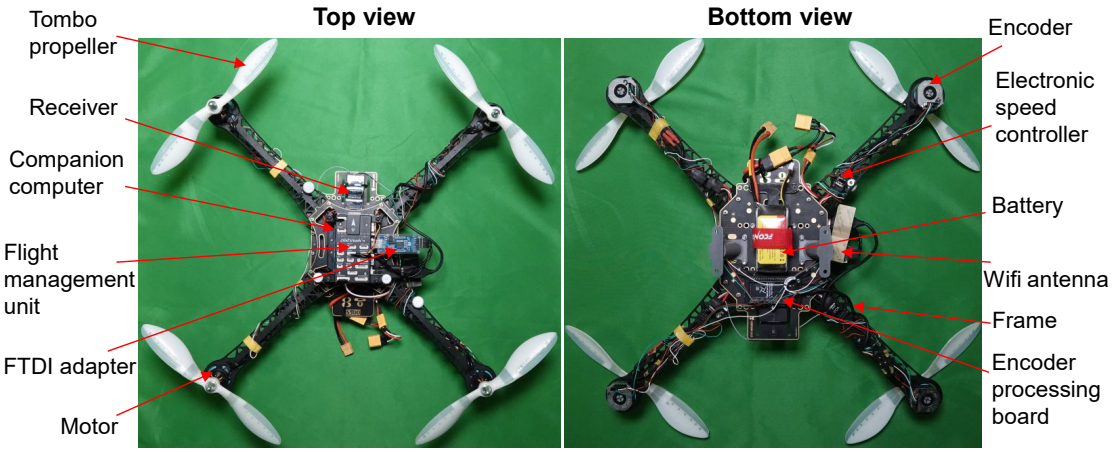


Figure 4.17: The system architecture designed for conducting the experiments.

The flight control architecture centers on the Pixhawk 4 flight management unit (Holybro), running the open-source PX4-Autopilot v1.15.0 firmware. For offboard control, a Jetson TX2 companion computer (NVIDIA) paired with an Orbitty Carrier board (Connect Tech) is mounted on the drone. The Jetson TX2 runs robot operating system (ROS) Melodic on an Ubuntu 18.04 platform. Communication between the Jetson TX2 and the Pixhawk 4 is facilitated through a universal serial bus (USB)-to-universal asynchronous receiver/transmitter (UART) (FTDI) adapter, which connects the companion computer's USB port to the Pixhawk's Telem2 UART interface. Additionally, quadrotor position and orientation data are transmitted wirelessly from a motion capture (Mocap) system to the Jetson TX2 through a router, which then forwards this information to the Pixhawk. Upon receiving both control commands and current pose data, the Pixhawk fuses this input with onboard sensor measurements to generate motor commands via the ESCs, ensuring stable flight and precise navigation.

Moreover, the Jetson TX2 interfaces with an ESP32-WROOM microcontroller (Freenove) to obtain velocity measurements from the four Tombo propellers. This data is processed onboard to detect collisions and send corrective flight commands to avoid impact zones.

All electronics—including motors, flight controller, and companion computer—are powered by a single 3-cell 1500mAh 80C LiPo battery (FCONEGY).

Power distribution and voltage regulation are handled by the PM06 V2.0 power module (Holybro), which maintains stable power supply and prevents electrical disturbances that could impair drone performance.

4.2.3 Propeller Speed Variation During Collision

4.2.3.1 Simulated results of propeller velocity during collision

Equation 4.24 is solved numerically to determine the change in the Tombo propeller velocity ($\dot{\theta}_1$) during a collision.

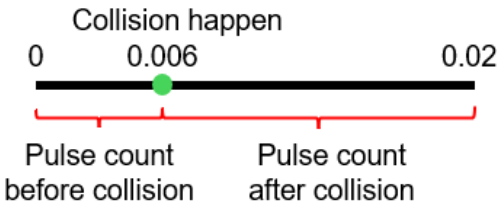


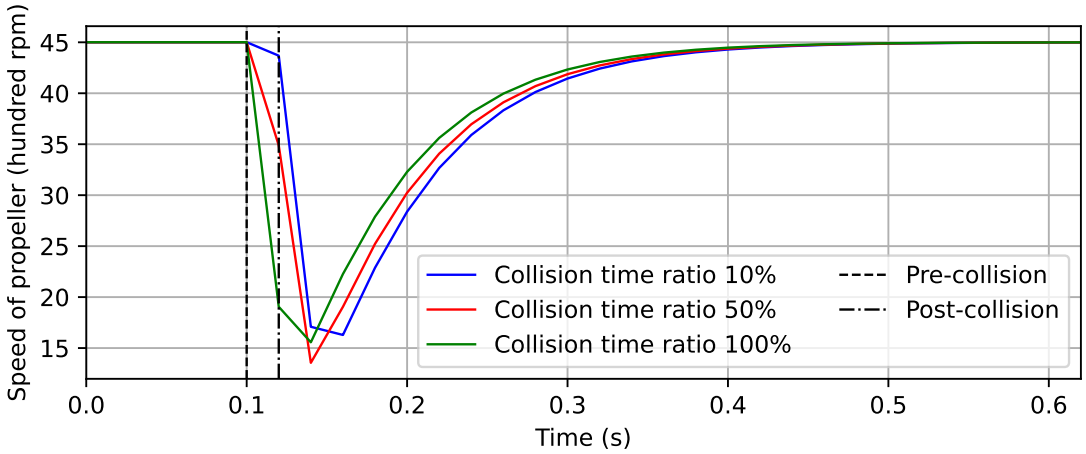
Figure 4.18: Illustration of the encoder pulse counting cycle during collision.

Figure 4.19 displays the simulation results of the Tombo propeller's velocity during a collision event, along with the corresponding velocity change rates. For instance, as shown in Figure 4.18, one encoder pulse counting interval spans 0.02 s, at the end of which the velocity is computed. Within this interval, a collision takes place at 0.006 s, meaning that pulses counted from 0 to 0.006 s represent the pre-collision phase, while those recorded from 0.006 to 0.02 s correspond to the post-collision phase. Thus, the collision duration covers 0.006 to 0.02 s, equating to 70 % of the total counting cycle.

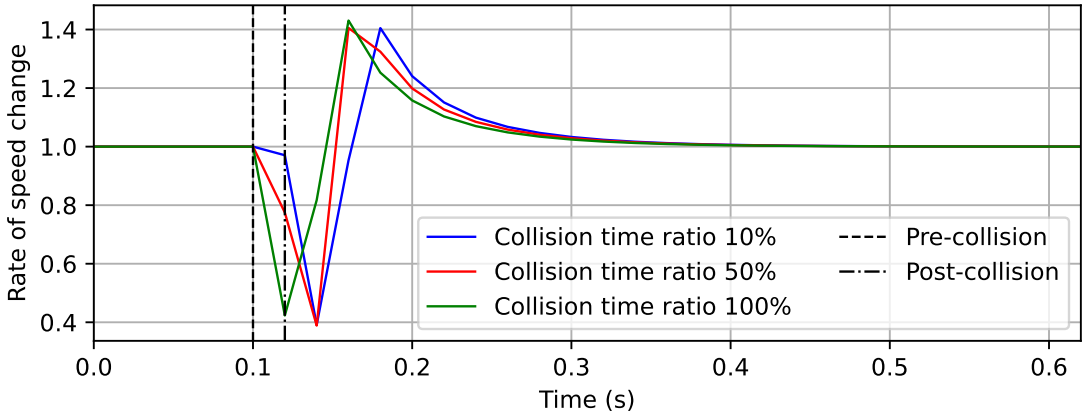
The simulation of velocity reduction during collision is performed for three scenarios as detailed below:

- Collision time ratio 10 %: Upon collision, the velocity sequentially decreases from 4500 to 4366, then to 1710, and finally to 1624 rpm. The corresponding rates of velocity change during this process are 0.97, 0.39, and 0.95, followed by a recovery phase.

- Collision time ratio 50 %: The velocity transitions from 4500 to 3487, then to 1357 rpm, with velocity change rates of 0.77 and 0.39 during the event, followed by recovery.
- Collision time ratio 100 %: The velocity shifts from 4500 to 1901, then to 1557 rpm, with velocity change rates of 0.42 and 0.82, followed by a recovery phase.



(a) Speed variation of Tombo propeller during collision



(b) Variation in the current-to-previous speed ratio

Figure 4.19: Simulation results showing changes in Tombo propeller velocity due to collisions occurring at various points within the velocity measurement interval.

4.2.4 Propeller speed variation during collision in real flight

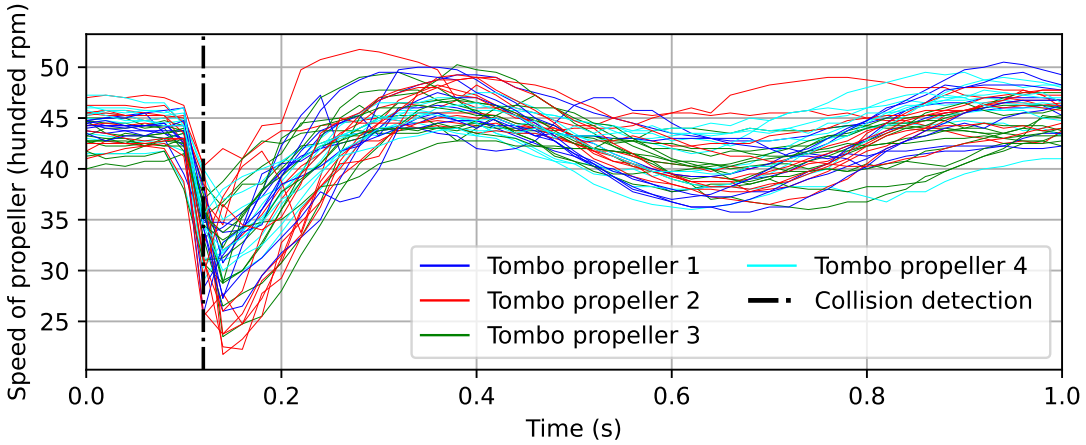
The simulation results above indicate that the Tombo propeller experiences 2 to 3 distinct deceleration phases during a collision before beginning recovery. However,

these findings are based on simulations of a single motor and do not account for the flight controller's dynamic response. Additionally, real-world collisions are inherently unpredictable, with variables such as the rigidity of the colliding object, collision duration, and number of impacts being uncontrollable. Consequently, actual in-flight collisions involving the drone are considerably more complex than the controlled conditions examined in the previous experiments.

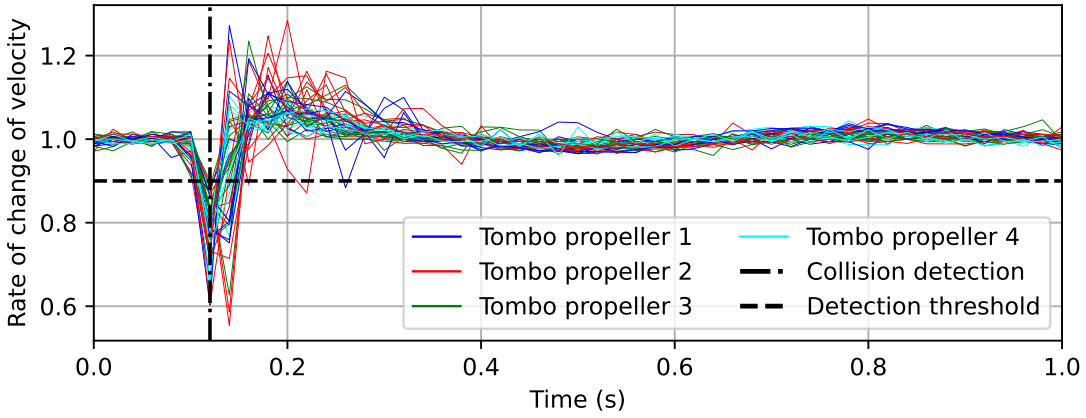
Figure 4.20: The real-flight collision experiment is performed by freely dropping an impact object into the propeller's rotation zone while the quadrotor is flight.

Given these factors, collision experiments in actual flight conditions were carried out, as depicted in Figure 4.20, to establish a reliable method for detecting propeller collisions. Collisions occurred unpredictably during manual control of the drone, with a total of 42 collision events recorded across all propellers. Figure 4.21 presents the velocity variations observed during these collisions. The ESP32-WROOM module transmits velocity data to the companion computer at a frequency of 50 Hz.

The results indicate that the deceleration pattern of the Tombo propeller aligns closely with the speed variations seen in both simulations and controlled experiments shown in Figure 4.21. However, due to the random nature of collisions and their timing within the encoder's measurement intervals, the minimum speed reduction ratio varies. Importantly, the propeller speed recovery process differs markedly from that observed in single-propeller experiments. When a collision happens during flight, the impacted propeller loses speed and thrust, destabilizing the drone. Consequently, the flight controller increases the speed of that propeller to restore balance, resulting in a more rapid recovery after collision. In contrast,



(a) Speed variation of Tombo propeller during collision



(b) Variation in the current-to-previous speed ratio

Figure 4.21: Speed variation of the Tombo propeller during collisions occurring on that propeller while the drone is in flight.

the single-propeller experiment shows recovery to pre-collision speed without any such compensation.

4.2.5 Encoder-Based Collision Detection Method

An algorithm for detecting collisions is developed based on the immediate deceleration of the propeller following an impact. The detection method compares the ratio of the current propeller speed to its previous speed. Let the current speeds of the four propellers be denoted by v_{1n}, v_{2n}, v_{3n}, v_{4n}, and their corresponding previous speeds v_{1p}, v_{2p}, v_{3p}, v_{4p}. The average speed and the individual speed

Algorithm 4.1 Collision Detection Based on Propeller Speed Ratios

Parameters: Minimum propeller speed threshold for collision detection $n_{QuadrotorWeight}$; collision detection threshold $r_{collision}$; post-detection delay Δt
Input: Current speeds $v_{1n}, v_{2n}, v_{3n}, v_{4n}$; previous speeds $v_{1p}, v_{2p}, v_{3p}, v_{4p}$
Output: Collision flag $col \in \{\text{True}, \text{False}\}$;
collision index $cnum \in \{1, 2, 3, 4, \text{None}\}$

Compute average speed $\bar{v}$ (Eq. (4.38))
Compute speed ratios r_1, r_2, r_3, r_4 (Eq. (4.39))
if $\bar{v} > n_{QuadrotorWeight}$(Eq. (4.40)) **then**
    Identify indices i where $r_i \leq r_{collision}$
    if any such r_i exists **then**
        $cnum \leftarrow \arg\min_{i \in \{1,2,3,4\}} r_i$
        $col \leftarrow$ True
        Wait for duration Δt before next evaluation
    else
        $col \leftarrow$ False, $cnum \leftarrow$ None
    end if
else
    $col \leftarrow$ False, $cnum \leftarrow$ None
end if
return $col, cnum$

ratios used for collision detection are then calculated as follows.

$$\bar{v} = \frac{v_{1n} + v_{2n} + v_{3n} + v_{4n}}{4}, \tag{4.38}$$

$$r_1 = \frac{v_{1n}}{v_{1p}}, \quad r_2 = \frac{v_{2n}}{v_{2p}}, \quad r_3 = \frac{v_{3n}}{v_{3p}}, \quad r_4 = \frac{v_{4n}}{v_{4p}}. \tag{4.39}$$

The collision detection process is executed as outlined in Algorithm 4.1, which outputs whether a collision occurred *col* and the index of the affected propeller *cnum*. Moreover, the criteria for detecting a collision and identifying the impacted propeller are detailed below:

4.2.5.1 Velocity condition before collision detection

During motor activation (arming) or deactivation (disarming), propeller speeds exhibit abrupt acceleration and deceleration, causing large fluctuations in velocity. Hence, collision detection is disabled during these intervals. To enforce this condition, the total thrust produced by all propellers at any given time must

be sufficient to counterbalance the drone's overall weight, as expressed by the following equation:

$$\bar{v} \geq n_{QuadrotorWeight}, \tag{4.40}$$

where $n_{QuadrotorWeight}$ represents the average rotational speed of each propeller required to generate a thrust equal to 0.8 times the quadrotor's weight. The factor 0.8 acts as a safety margin, ensuring that collision detection is only performed when the quadrotor operates near the takeoff threshold. Specifically, for the quadrotor used in this study, which has a mass of 1280 g, $n_{QuadrotorWeight}$ must produce a thrust of 2.56 N. According to experimental results, Tombo propellers need to spin at $n_{QuadrotorWeight} = 3800$ rpm to generate this thrust.

4.2.5.2 Collision detection threshold

Based on real-world experiments, as presented in Figure 4.20, the collision detection threshold is established at a speed reduction ratio of $r_{\text{collision}} = 0.9$. This threshold is sufficiently high to ensure reliable collision detection, as demonstrated in Figure 4.21, where the speed consistently drops below this value upon impact. Due to the 0.02 s velocity measurement interval, this reduction may be observed either in the first recorded velocity sample after collision or in the second. The exact timing of the collision within the seconds determines whether the propeller speed immediately falls below the $r_{\text{collision}}$ threshold or does so shortly afterward. In either case, the velocity measurements reliably reflect a sufficient decrease to trigger collision detection.

At this stage, the values r_1, r_2, r_3, and r_4 are compared to determine which ones are less than or equal to $r_{\text{collision}}$.

4.2.5.3 Determining the impacted propeller

When the propeller at the collision point loses thrust, its speed will drop below that of the other propellers. To compensate, the controller increases the thrust of the affected propeller while simultaneously decreasing the thrust of the other propellers, especially the one opposite it, to maintain drone stability. Consequently, there may be instances where the speed reduction ratio of the propeller opposite the collided one—or of the other two propellers—also falls below the threshold $r_{collision}$ as shown in Figure 4.22. However, this response will always occur with a delay relative to the collided propeller. Therefore, when multiple propellers exhibit a speed change ratio less than $r_{collision}$, the propeller currently experiencing the collision is identified by selecting the minimum ratio using $min(r_1, r_2, r_3, r_4)$.

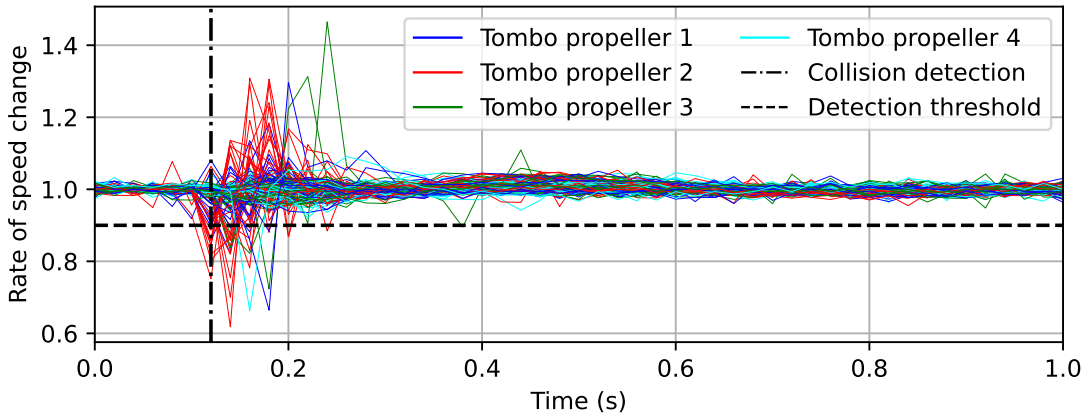


Figure 4.22: The speed changes in non-colliding propellers when a collision occurs.

4.2.5.4 Delay time after collision detection

When a collision occurs on a propeller, it causes the drone to lose balance abruptly. Consequently, the flight controller issues control commands to all four propellers to restore the drone's equilibrium. This recovery process typically takes about Δt = 1 s, during which the drone returns to its pre-collision position, as clearly shown in the recovery curve in Figure 4.22. Therefore, acceleration or deceleration of the propellers continues once a collision on a particular propeller has been detected.

To prevent multiple false detections for the same collision event, collision detection is suspended for a duration of Δt after the initial detection. This ensures that only one detection is registered per incident, avoiding repeated false alarms about timing, propeller identity, and collision location.

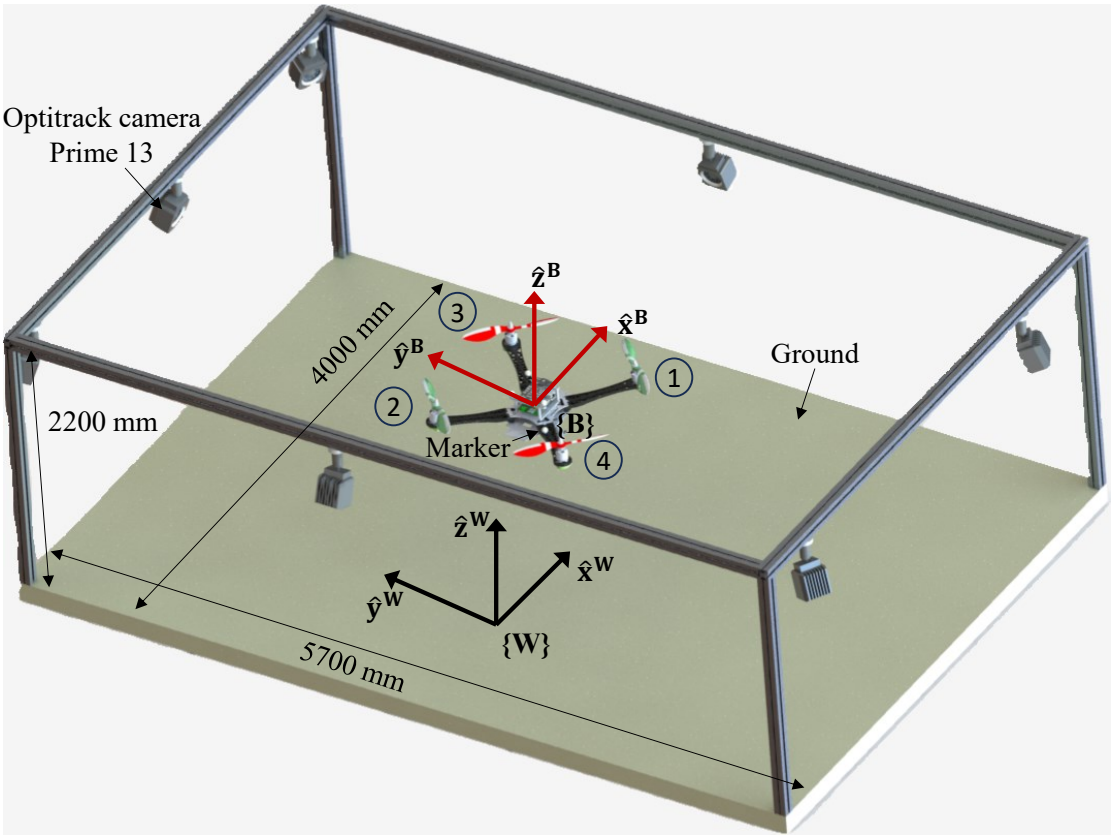


Figure 4.23: The flight experiment area utilizes the OptiTrack motion capture system to determine and record the quadrotor's position.

4.3 Collision reaction strategy

This section presents experiments evaluating the collision detection strategy during hovering and its application in trajectory-following flight. The test area is equipped with an OptiTrack Motive 2.0 tracking system, which includes seven Prime 13 cameras, as shown in Figure 4.23. The drone's position is measured and transmitted to its onboard computer at a frequency of 200 Hz. The flight area measures 5700 mm×4000 mm× 2200 mm. Both the global coordinate system

$\{\hat{x}^W, \hat{y}^W, \hat{z}^W\}$ and the body-fixed coordinate system $\{\hat{x}^B, \hat{y}^B, \hat{z}^B\}$ are defined within the East-North-Up (ENU) reference frame.

4.3.1 Collision reaction strategy during hovering

4.3.1.1 Collision-responsive hovering

This section describes the control strategy implemented to handle collisions during hovering, as depicted in Figure 4.24. The experiment involves collisions occurring randomly on any of the drone's propellers. The initial target position $\mathbf{x_i} \in \mathbb{R}^3$ is set within the global coordinate system {W} (refer to Figure 4.23), where the drone maintains a stable hover. Meanwhile, velocity measurements from all four propellers, represented by $\mathbf{v_p} = [v_{1n}\ v_{2n}\ v_{3n}\ v_{4n}]^\top$, are continuously sent to the onboard computer at a rate of 50 Hz. During the test, collisions are randomly introduced to the propellers.

When the collision detection algorithm identifies a collision, it triggers a detection signal that initiates two actions simultaneously. First, the drone's position at the collision instant, denoted as $\mathbf{x_c}$, is recorded. Second, the drone transitions from its initial target position to a recovery position $\mathbf{x_r}$, which is positioned at a safe distance d from the collision point. Here, the body-fixed coordinate system is aligned with the global reference frame. The recovery position $\mathbf{x_r}$ is calculated depending on the impacted propeller as described by the following equation:

$$\begin{aligned}
&\textbf{Propeller 1:} \quad \mathbf{x_r} = \mathbf{x_c} + [-d\ d\ 0]^\top \\
&\textbf{Propeller 2:} \quad \mathbf{x_r} = \mathbf{x_c} + [d\ -d\ 0]^\top \\
&\textbf{Propeller 3:} \quad \mathbf{x_r} = \mathbf{x_c} + [-d\ -d\ 0]^\top \\
&\textbf{Propeller 4:} \quad \mathbf{x_r} = \mathbf{x_c} + [d\ d\ 0]^\top
\end{aligned} \tag{4.41}$$

At this stage, the desired position $\mathbf{x_d}$ is set to $\mathbf{x_r}$ and is held steady until another collision is detected. All desired positions $\mathbf{x_d}$ are communicated via micro

air vehicle link router for ROS (MAVROS) from the Jetson TX2 onboard computer to the Pixhawk flight controller, which governs the Tombo quadrotor using the PX4 Autopilot's low-level PID control scheme.

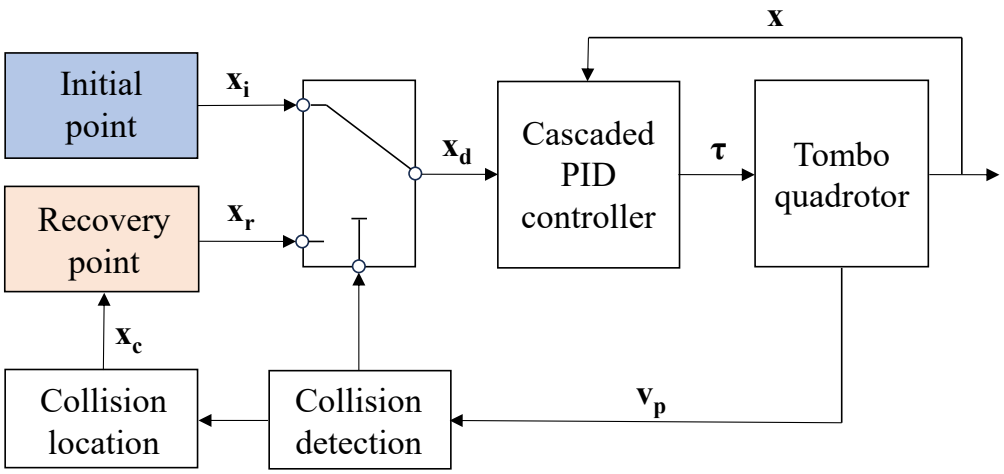


Figure 4.24: Control strategy for drone hovering response when a propeller collision occurs.

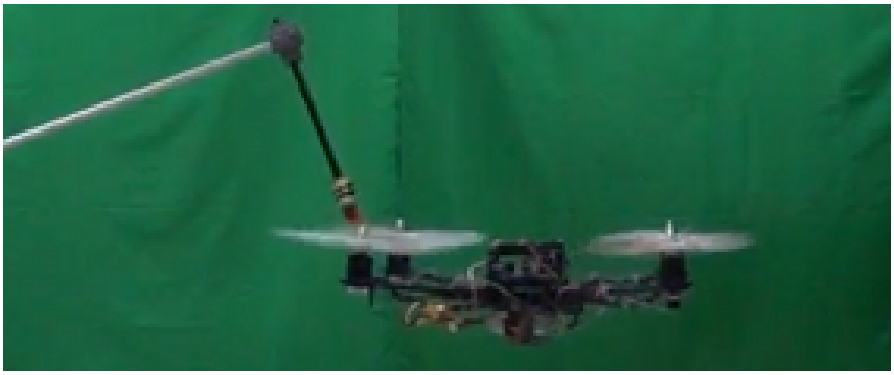


Figure 4.25: Collision experiment during drone hovering.

4.3.1.2 Experimental results

To assess the quadrotor's ability to detect and react to collisions during hovering, experiments were carried out where random impacts were applied while the drone maintained a hover. The initial position was set at $\mathbf{x_i} = [0\ 0\ 1]^\top$, and collisions were introduced to the propellers as shown in Figure 4.25. A safety margin of $d = 0.5\,\text{m}$ was used. In total, 12 collision events were conducted across multiple flights, with 8 of them being successfully detected because the impacted propeller

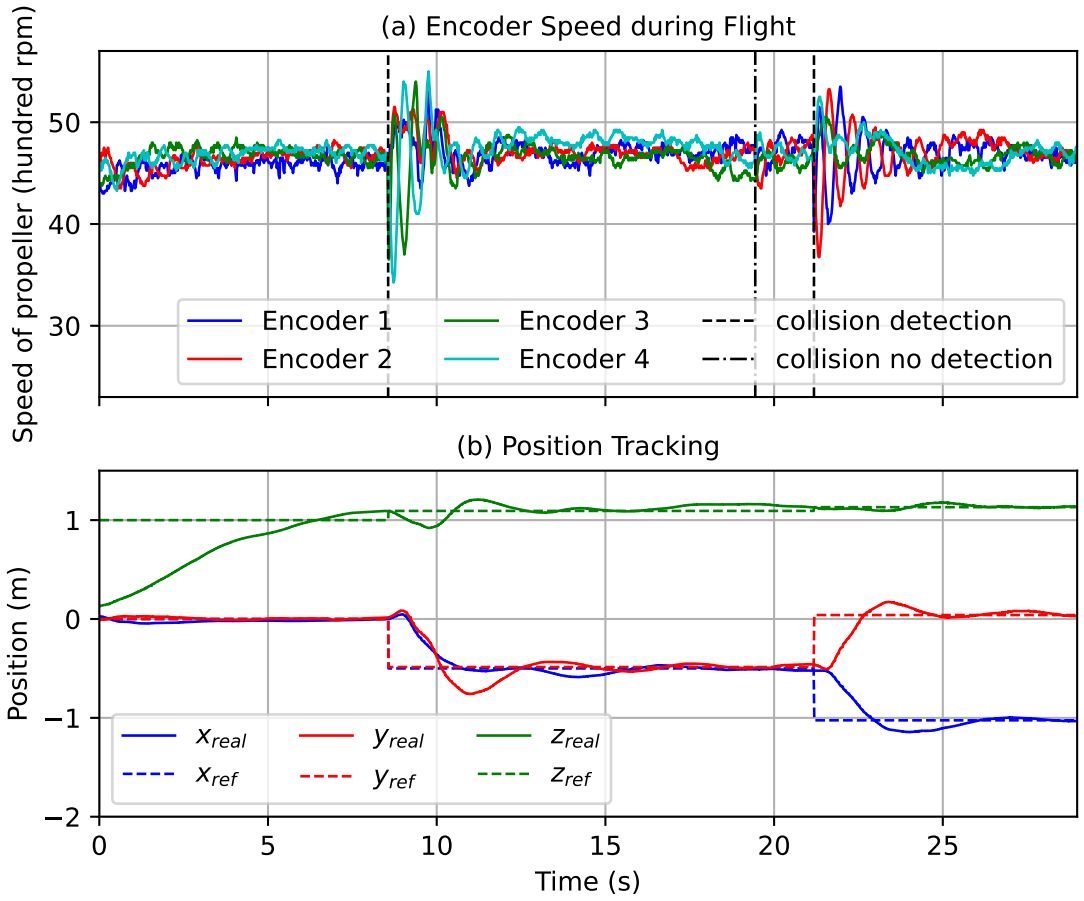


Figure 4.26: Experimental results when Tombo quadrotor hovers with random collisions on the propeller.

decelerated enough to meet the detection threshold of 0.9. The other 4 collisions went undetected due to their mild impact and short duration, causing decelerations that did not reach the threshold. Observations from these undetected cases showed that the propeller speeds barely dropped and quickly returned to their pre-collision values. Additionally, the quadrotor experienced minimal oscillations and stabilized its hover promptly. For the 8 detected collisions, the system accurately identified the affected propeller in 100 % of the cases. Among the 8 detected collisions, the impacted propeller was correctly identified in 100 % of the cases.

Figure 4.26 shows the results from a single flight during which three collisions occurred, with two of them successfully detected by the encoder.

During the first collision, the impact was identified on propeller 3. Its speed dropped from 4600 rpm to 3975 rpm, then further decreased to 3650 rpm

before eventually returning to its steady-state speed. The deceleration ratios corresponding to these two decreases were 0.86 and 0.92, respectively. Upon detecting the initial deceleration, the drone's position at the moment of collision was recorded as $\mathbf{x_c} = [0\ 0.014\ 1.093]^\top$. Following this, the desired position was updated to $\mathbf{x_r} = [-0.5\ -0.486\ 1.093]^\top$ to guide the drone's recovery.

During the second collision, the impact affected propeller 1, causing its speed to drop from 4750 rpm to 4575 rpm, then slightly to 4550 rpm before returning to its original velocity. Because the minimum deceleration ratio during this event was only 0.96, the collision went undetected. The drone remained stable throughout this period due to the mild impact and brief collision duration, which led to only minor changes in propeller speed and negligible shifts in position.

Shortly afterward, propeller 1 experienced a stronger impact, with its speed falling from 4775 rpm to 4475 rpm and then further decreasing to 3925 rpm before recovering. The two deceleration events corresponded to velocity change ratios of 0.94 and 0.88, respectively. The collision position was logged as $\mathbf{x_c} = [-0.525\ -0.046\ 1.131]^\top$, and the recovery position was updated accordingly to $\mathbf{x_r} = [-1.025\ 0.014\ 1.131]^\top$.

4.3.2 Collision reaction strategy along the planned trajectory.

4.3.2.1 Collision-responsive flight trajectory

The collision response approach while the drone is moving toward its target mirrors the method used during hovering. Upon detecting a collision, the algorithm determines which propeller was affected and records the collision location. This data is subsequently transmitted to the flight control system, enabling it to modify the flight trajectory accordingly. Figure 4.27 shows both the original flight trajectory and the modified trajectory after collision detection.

Once a collision is detected, the drone's position at the time of impact, denoted

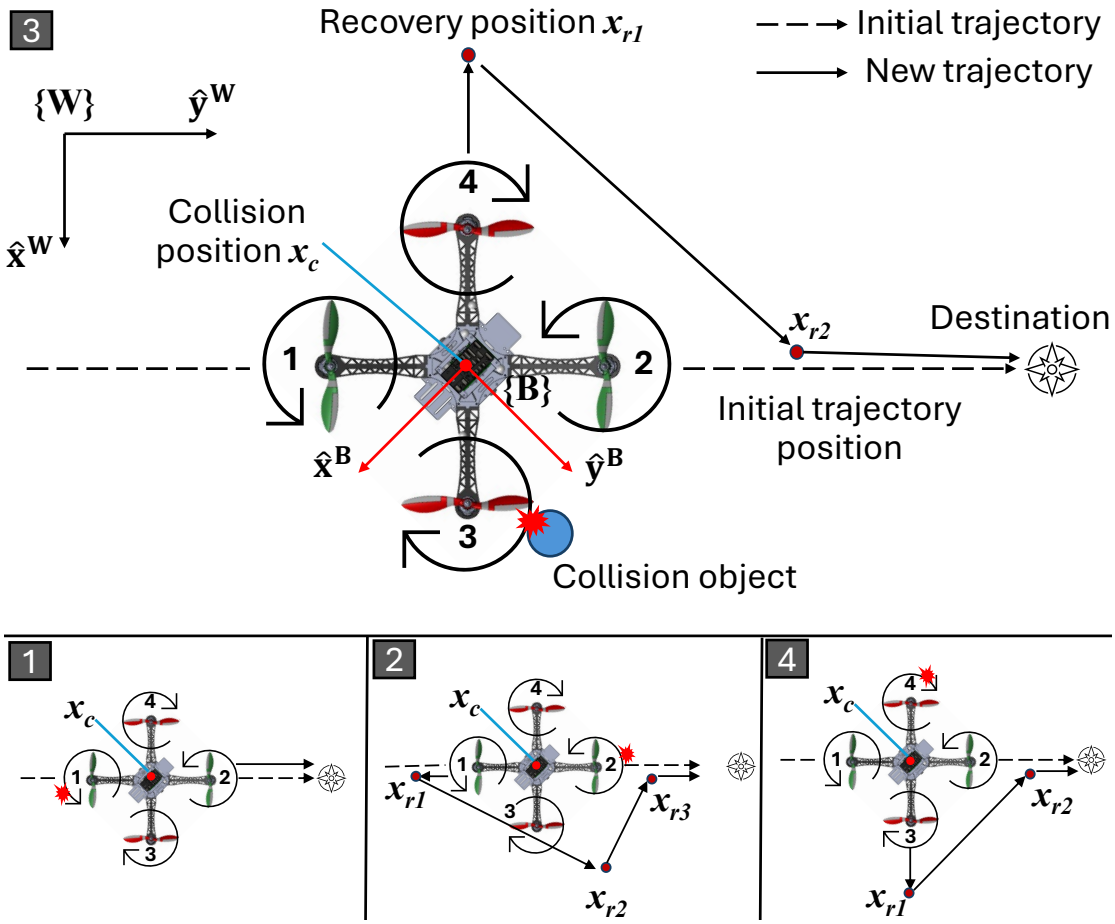


Figure 4.27: Collision reaction strategy of the Tombo quadrotor while moving to the target point. The quadrotor's avoidance trajectory depends on the position of the impacted propeller. The numbers in each frame illustrate the position of the quadrotor's impacted propeller.

as $\mathbf{x_c}$, is promptly recorded. In this study, the drone's flight path is planned along the y-axis, maintaining a specified safety distance d to prevent collisions. Consequently, the trajectory for collision response is defined according to the location of the affected propeller as follows:

Propeller 1: Because the collision happens at the rear relative to the drone's direction of flight, no additional parts of the drone are at risk of impact. Therefore, the drone can proceed along its original trajectory without any adjustments.

Propeller 2: If a collision occurs at this location, the estimated position of the obstacle is calculated as:

$$\mathbf{x_{co}} = \mathbf{x_c} + [0 \; l_a \; 0]^\top \tag{4.42}$$

where l_a denotes the distance from the drone's coordinate origin to the motor positions. Consequently, the drone will traverse the following waypoints to circumvent the obstacle before returning to its planned flight path:

$$\begin{aligned} \mathbf{x_{r1}} &= \mathbf{x_c} + [0 \ -d \ 0]^\top \\ \mathbf{x_{r2}} &= \mathbf{x_c} + [d \ l_a \ 0]^\top \\ \mathbf{x_{r3}} &= \mathbf{x_c} + [0 \ d + l_a \ 0]^\top \end{aligned} \tag{4.43}$$

Propeller 3: In this scenario, the estimated position of the obstacle is:

$$\mathbf{x_{co}} = \mathbf{x_c} + [l_a \ 0 \ 0]^\top \tag{4.44}$$

The points through which the drone maneuvers to avoid the obstacle are:

$$\begin{aligned} \mathbf{x_{r1}} &= \mathbf{x_c} + [-d \ 0 \ 0]^\top \\ \mathbf{x_{r2}} &= \mathbf{x_c} + [0 \ d \ 0]^\top \end{aligned} \tag{4.45}$$

Propeller 4: If a collision happens at this location, the estimated position of the obstacle is:

$$\mathbf{x_{co}} = \mathbf{x_c} + [-l_a \ 0 \ 0]^\top \tag{4.46}$$

The drone's maneuvering points to avoid the obstacle are:

$$\begin{aligned} \mathbf{x_{r1}} &= \mathbf{x_c} + [d \ 0 \ 0]^\top \\ \mathbf{x_{r2}} &= \mathbf{x_c} + [0 \ d \ 0]^\top \end{aligned} \tag{4.47}$$

4.3.2.2 Experimental results

Experiments were carried out to assess the quadrotor's ability to detect and respond to collisions while flying towards a designated target. The tests compared a quadrotor fitted with standard commercial rigid propellers against one equipped with Tombo propellers, which are engineered to better tolerate impacts. The rigid

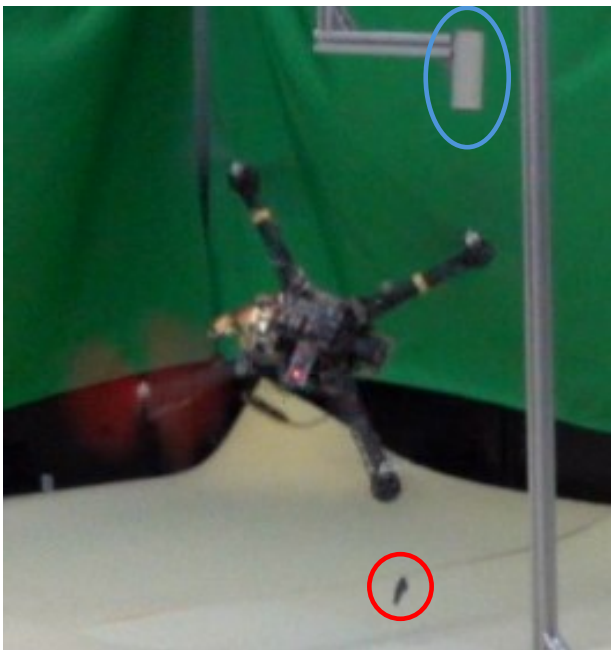


Figure 4.28: The quadrotor fitted with rigid propellers crashed following a collision with the obstacle, indicated by the blue shaded region. Additionally, part of Propeller 1 was damaged, highlighted in the red area.

propellers used in these trials measured 9 inches in diameter with a 4.7-inch pitch. The outcomes of the collision tests for both the rigid propeller quadrotor and the Tombo quadrotor are shown in Figures 4.29 and 4.30, respectively.

The collision event was detected on propeller 1 for the rigid propeller quadrotor. However, because of the propeller's rigid construction, it fractured upon impact, as illustrated in Figure 4.28. As a result, the quadrotor lost stability and crashed after the collisions. The recorded rotational speed data for Propeller 1 during the collision showed a sequence of values: 5550 – 5000 – 1425 – 0 – 0 – 350 – 350 – 925 – 1700 – 3075 – 4575 rpm, corresponding to velocity ratios of 0.9 – 0.29 – 0 – * – * – 1 – 2.64 – 1.83 – 1.81 – 1.49. After this sequence, the quadrotor flipped, preventing any further data collection. The collision detection algorithm accurately identified propeller 1 as the impacted propeller and recorded the quadrotor's position at the collision moment as $\mathbf{x_c} = [0.086 \quad -0.103 \quad 0.929]$. However, due to the excessive impact force on the rigid propeller, it fractured, causing a sharp drop in rotational speed from 5550 to 0 rpm within 0.04 s. This abrupt loss of thrust destabilized the

quadrotor, leading to an unrecoverable crash

For the Tombo quadrotor, the recorded speed changes of the impacted Tombo propeller 3 during the collision followed this sequence: 4450 – 3900 – 3300 – 2775 – 2675 – 2925 – 3325 – 3850 – 4425 rpm, with corresponding velocity ratios of 0.88 – 0.85 – 0.84 – 0.96 – 1.09 – 1.14 – 1.16 – 1.15. The collision detection algorithm successfully identified the collision at the first speed drop and logged the quadrotor's position at impact as $\mathbf{x_c} = [0.213 \ -0.337 \ 1.004]$. Because Tombo propellers are engineered to absorb impact forces, the speed variation was more gradual, and the minimum speed decrease was less severe compared to the rigid propeller scenario. The Tombo propeller took 0.16 s to recover to its pre-collision speed of 4450 rpm.

Upon impact, the quadrotor experienced a brief loss of balance and altitude caused by the reduced lift force. The altitude dropped by 0.11 m within 0.55 s following the collision. Regarding attitude response, the quadrotor showed noticeable oscillations, with the largest angular errors occurring around 0.52 s after impact. The maximum roll error reached 22 deg at 0.55 s, the peak pitch error was -14 deg at 0.86 s, and the maximum yaw error was 5 deg at 0.81 s post-collision.

Notably, the Tombo propeller recovered its rotational speed in just 0.16 s, which is considerably faster than the time needed for the positional and angular disturbances to diminish. Consequently, the Tombo quadrotor rapidly regained stability and moved to the recovery position. It stabilized without further oscillations within four seconds after the collision and took seven seconds to travel from the collision point to the recovery position at $\mathbf{x_{r1}} = [-0.387 \ -0.337 \ 1.004]$. The quadrotor then proceeded to follow the collision response trajectory to safely reach the target point.

Additionally, Figures 4.31 and 4.32 illustrate the outcomes of collision detection and avoidance when impacts occurred on propellers 2 and 4. The results demonstrate that the Tombo quadrotor successfully detected the collisions, correctly identified the affected propellers, and safely continued its flight to reach the target

point.

4.4 Summary and Discussion

This chapter presented a sensor-based collision detection and reaction strategy for Tombo quadrotors using rotary encoders integrated on each motor. The proposed method enables direct impact detection on the propellers, providing timely identification of the collision location. Two reactive strategies were developed to demonstrate post-collision recovery under both hovering and trajectory-tracking flight conditions.

The experimental results confirmed that the encoder-based detection system enhances the quadrotor's capability to recognize strong or sustained collisions and facilitates appropriate recovery maneuvers. Compared to sensorless handling, this approach provides an additional layer of safety by preventing prolonged contact with obstacles, which can otherwise lead to critical thrust loss and system failure.

To enable meaningful comparison and further development of collision resilience technologies, a standardized benchmarking framework is required. The next chapter introduces such a framework, focusing on experimental setups and objective performance metrics that can serve as reference tools for evaluating drone recovery performance across different designs and control strategies.

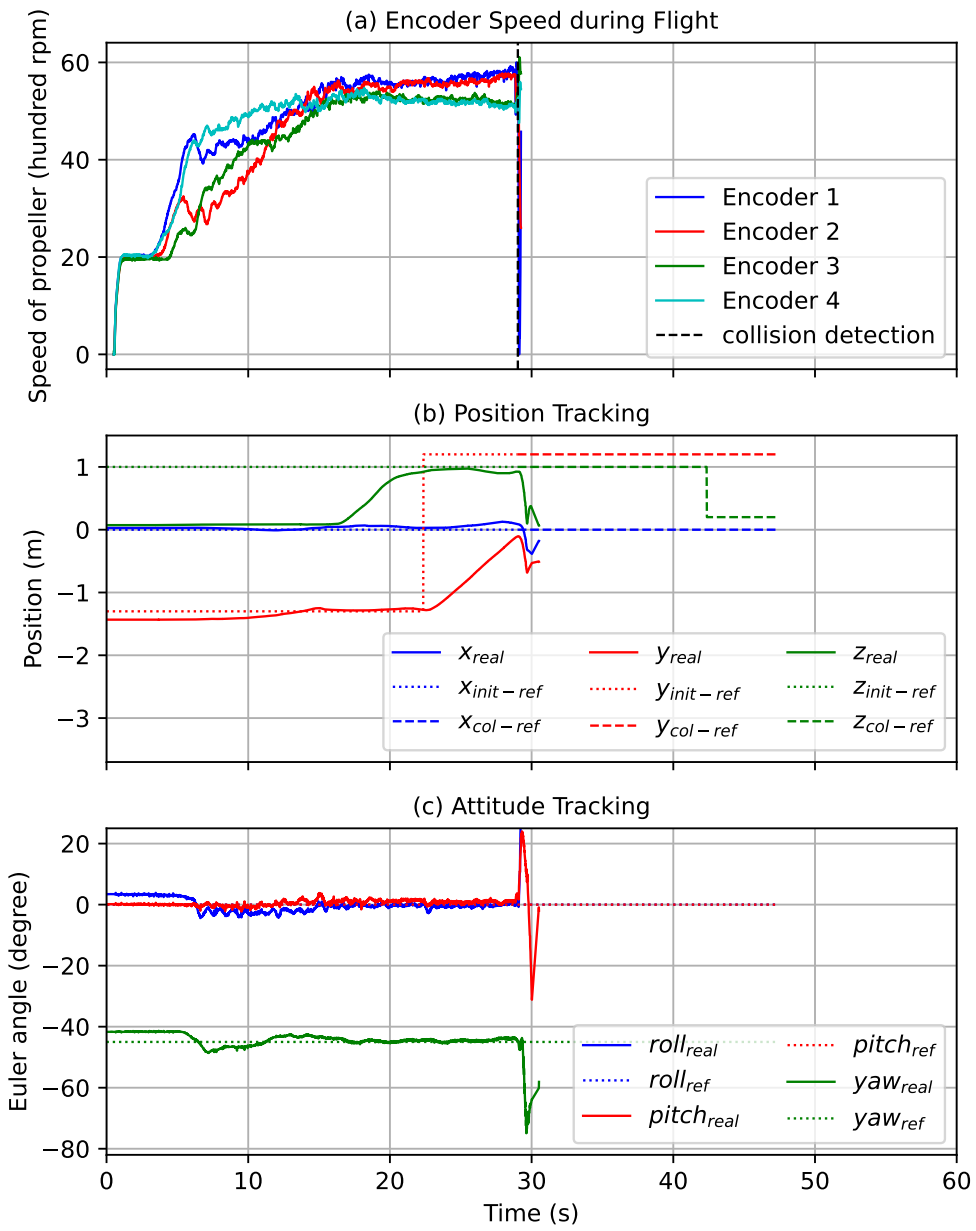


Figure 4.29: Experimental results of collision detection and response for a quadrotor equipped with rigid propellers while flying toward the target point.

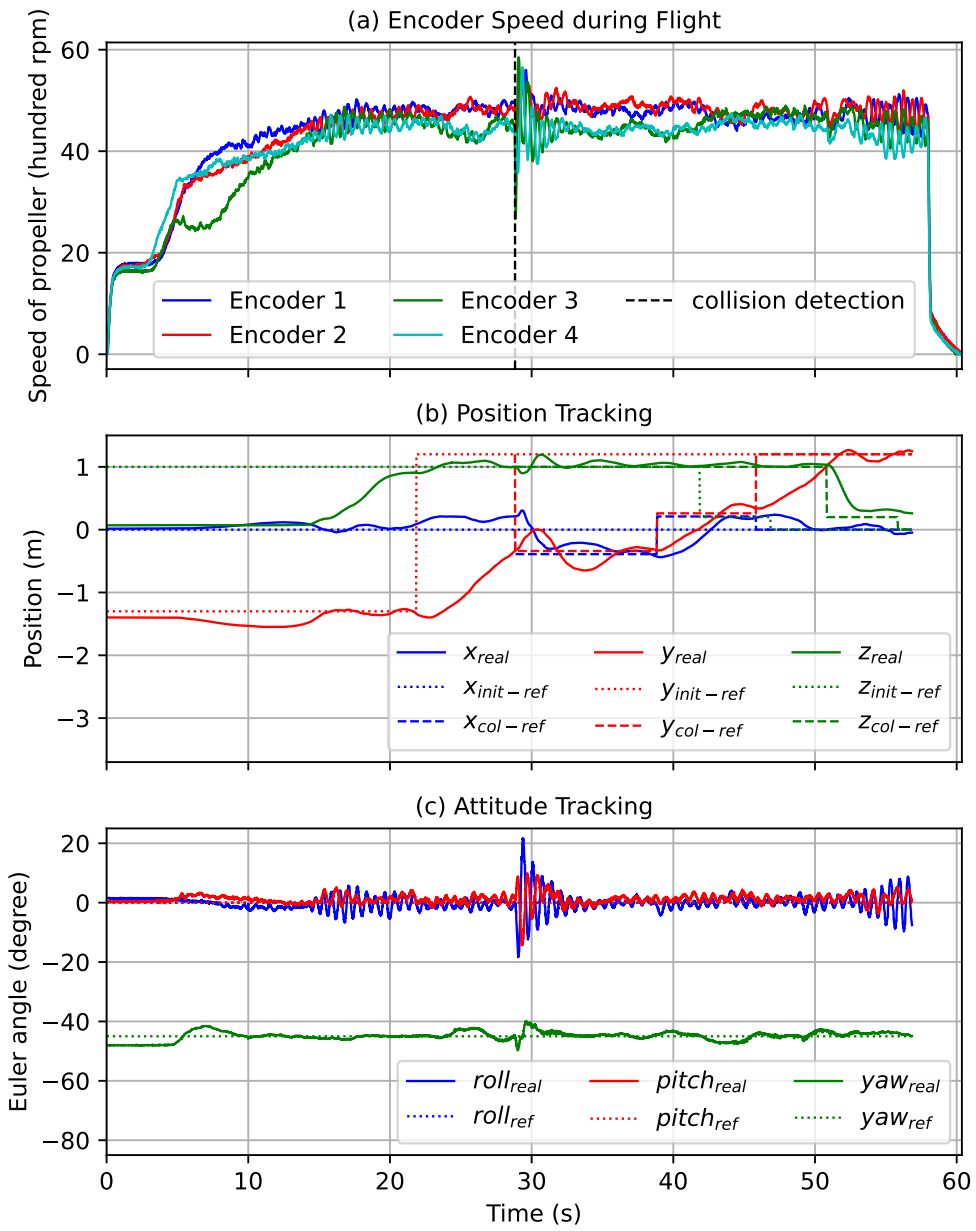


Figure 4.30: Experimental collision and collision reaction result when Tombo quadrotor is en route to the target point.

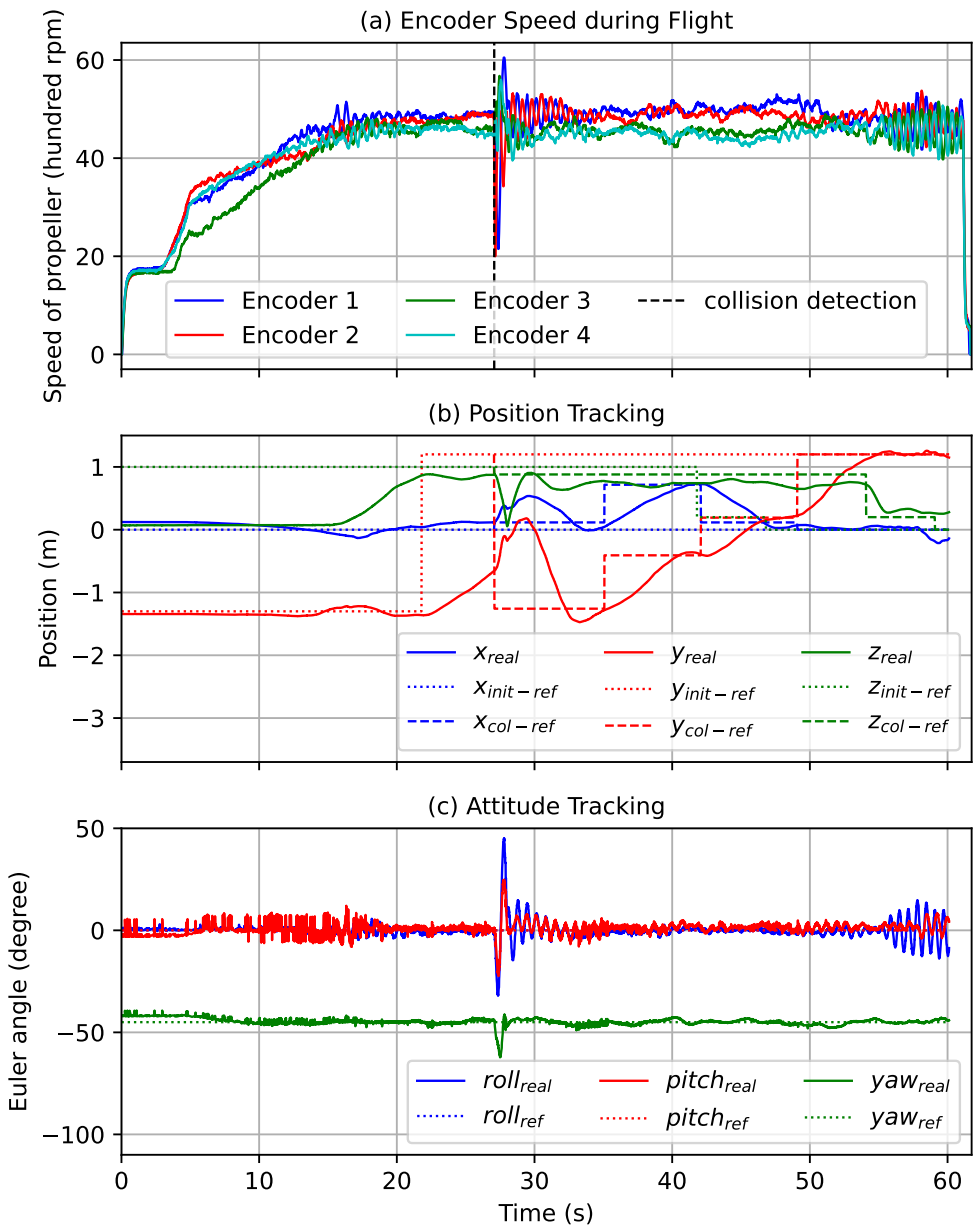


Figure 4.31: Collision response results in the case of a collision at propeller 2 during the journey to the target point.

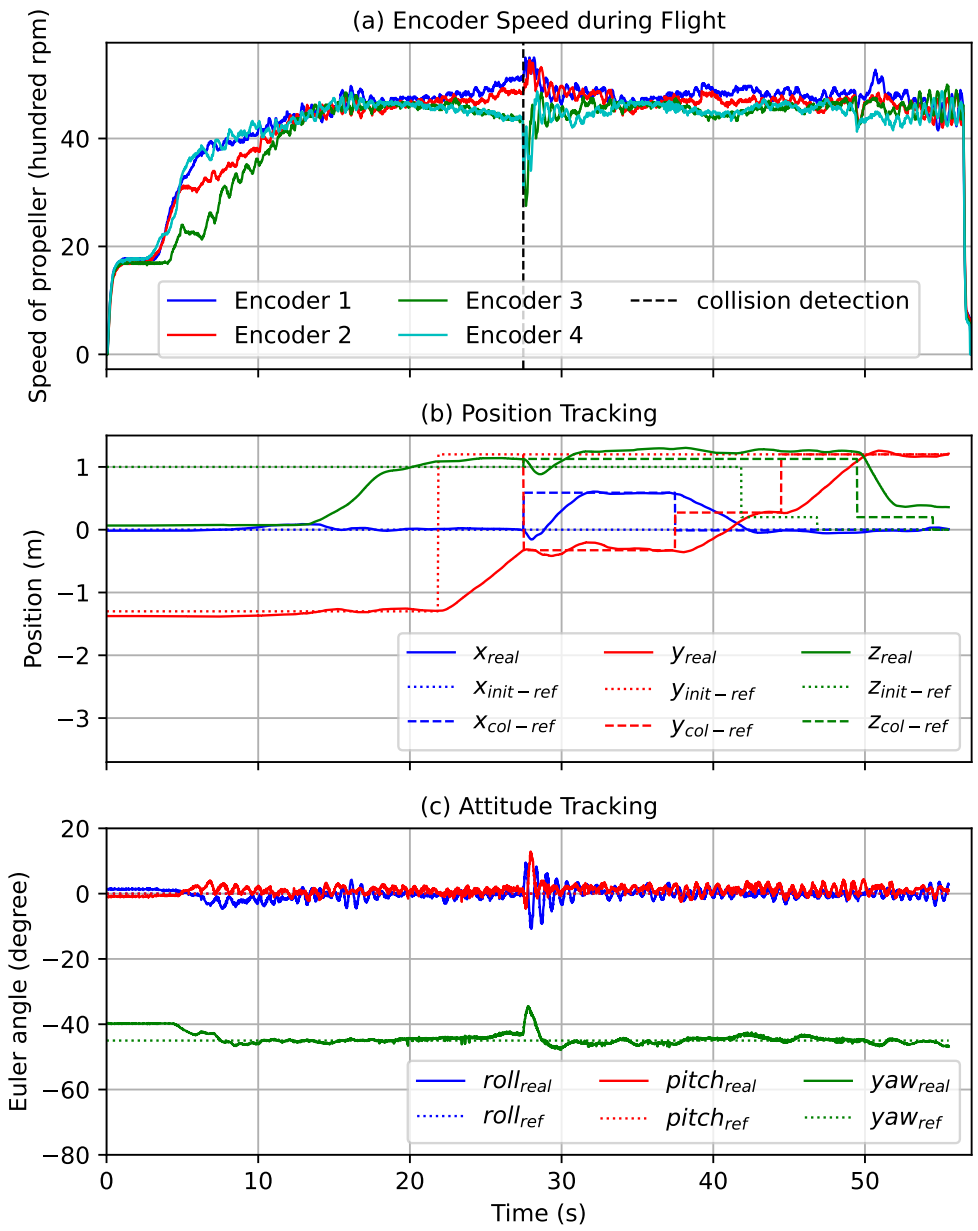


Figure 4.32: Collision response results in the case of a collision at propeller 4 during the journey to the target point.

Table 4.3: Behavior characteristics of deformable during collision process

Case	Initial angular velocity	Number of collisions	Collision first - Second force	Time between collisions	Time for shape recovery	Minimum angular velocity	Time to minimum angular velocity	Time to recover from minimum angular velocity
1	1000 rpm (105 rad/s)	1	42.6 N	-	29.2 ms	41 rad/s	20 ms	0.8 s
2	1500 rpm (157 rad/s)	1	52.7 N	-	61.5 ms	70 rad/s	20 ms	0.7 s
3	2000 rpm (209 rad/s)	2	87.2 N - 34 N	18.7 ms	107.3 ms	85 rad/s	16 ms	0.65 s
4	2500 rpm (262 rad/s)	2	128.7 N - 90.8 N	12.5 ms	104.2 ms	62 rad/s	16 ms	0.6 s
5	3000 rpm (314 rad/s)	2	121.4 N - 120.6 N	11.5 ms	91.7 ms	88 rad/s	12 ms	0.6 s

Chapter 5

Benchmarking Framework for Collision Recovery Evaluation

5.1 Motivation and Purpose

This chapter proposes a benchmarking framework specifically designed to evaluate the collision recovery performance of quadrotors, focusing on the ability of drones to stabilize and return to their reference trajectories after collision events. Unlike many previous studies that primarily assess whether drones survive a collision or crash, this framework emphasizes detailed, quantitative evaluations even in cases where the drone survives the collision. This is particularly important because, in real-world applications, quadrotors using conventional rigid propellers may experience collisions but remain airborne.

The need for a more objective and consistent evaluation tool is clear: current methods often fail to fully capture the differences in recovery effectiveness between hardware designs and control strategies. The proposed framework addresses this gap by introducing repeatable experimental setups and defining performance metrics to compare collision resilience across different systems, materials, and control laws.

The two main benchmarking criteria in this chapter—recovery time and maximum deviation after collision—offer valuable insights into how well drone hardware (such as deformable propellers) and control algorithms contribute to post-collision

stability. For example, propellers with better impact absorption characteristics can reduce the effect of collision forces on the drone body, minimizing trajectory deviations and enabling faster recovery. Similarly, this benchmark can be used to compare the effectiveness of various control algorithms, such as the $\mathcal{L}_1$1Quad controller versus alternative designs, in compensating for disturbances caused by collisions.

5.2 Benchmarking Experimental Setups

The following elements define the recommended experimental setups 3.10 for evaluating collision recovery performance in a standardized and reproducible manner. These setups are designed based on and generalize the experimental configurations that were practically implemented and validated in the collision handling studies in Chapter 3. By aligning directly with those established setups, this framework not only systematizes the experimental protocols but also provides a validated reference model that can be adapted for future UAV collision resilience evaluations.

Drone Configuration

- A general quadrotor platform capable of autonomous flight and trajectory tracking, applicable to a wide range of drone designs regardless of frame structure, propeller material, or onboard sensors.
- When comparing different drone configurations, such as drones equipped with rigid propellers, deformable propellers such as Tombo propellers, or other alternative impact mitigation designs, it is essential to ensure that the compared drones have similar physical characteristics, including comparable sizes, masses, and propulsion system specifications. This requirement ensures fairness in the comparison by maintaining consistent flight dynamics and thrust-to-weight ratios across configurations.

- This principle of comparable drone specifications was strictly followed in the experimental comparisons between drones using Tombo propellers and drones using commercial rigid propellers, as demonstrated in the foam bar collision experiments (Section 3.4.2) and carbon fiber bar collision experiments (Section 3.4.3) (see Figure 5.1). In these experiments, the drones shared the same frame, battery, and flight controller, with only the propellers and control strategies varied.

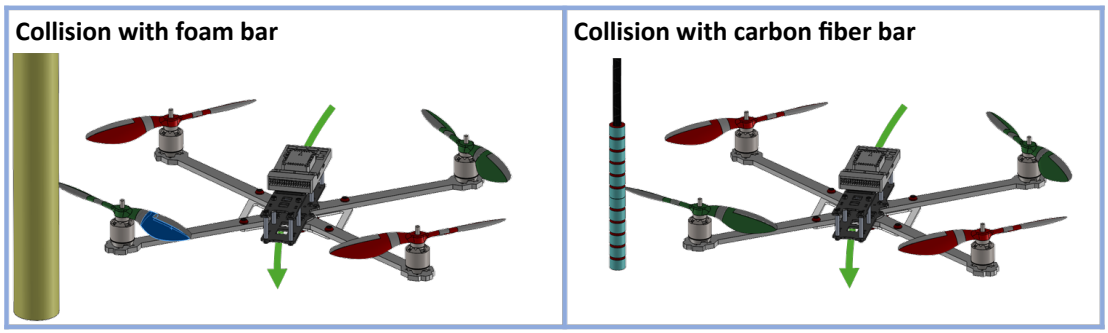


Figure 5.1: Standardized experimental setup for drone collisions with soft (foam bar) and rigid (carbon fiber bar) objects along predefined trajectories. The foam bar and carbon fiber bar setups align with those validated in the experiments in Sections 3.4.2 and 3.4.3.

Collision Objects and Setup

- Collision objects should be suspended in a free-hanging configuration to allow them to move away naturally after the collision occurs. This design ensures that each collision event is independent, prevents the drone from becoming stuck or trapped, and allows the number of collisions to remain consistent across repeated trials, even when using large or rigid objects along the predefined trajectory.
- Soft, deformable collision objects (e.g., foam bars) are recommended for simulating low-force collisions with minimal risk of drone damage. In this study, foam bars were chosen to reflect realistic scenarios such as soft obstacles like cables or foliage. In these cases, commercial rigid propellers tended to cut

through the foam, while Tombo propellers demonstrated superior energy absorption without structural damage, as validated in Section 3.4.2.

- Rigid collision objects (e.g., carbon fiber bars), optionally covered with protective foam layers, can be used to simulate higher-force collisions with harder environmental obstacles like poles or beams. In this study, carbon fiber bars were selected to replicate such conditions while preventing catastrophic damage by applying localized foam wrapping. Commercial rigid propellers exhibited minor tip fractures upon impact with these rigid objects, whereas Tombo propellers showed improved resilience and effective force absorption, as demonstrated in Section 3.4.3.
- The size, shape, and material of the collision objects should be flexibly selected based on the laboratory environment, available equipment, and safety considerations. This flexibility allows the framework to be adapted to different research facilities while maintaining comparability across studies.

Repetition

- Each collision scenario should be repeated multiple times to ensure statistically reliable and representative results. This practice is consistent with the repeated trials conducted in the experiments and summarized in the statistical evaluations presented in Figure 3.12.
- Repetition is particularly important because, in practical collision scenarios, several factors can vary between trials even under controlled initial conditions. These include variations in the drone's precise position and orientation at the moment of impact, as well as differences in propeller rotational speeds. These random variations can significantly influence the collision response and recovery behavior.
- By performing multiple trials and analyzing the resulting statistical distributions of recovery time and maximum deviation—as illustrated in Fig-

ure 3.12—researchers can mitigate the effects of these random fluctuations and ensure that the evaluation results are both reliable and representative. This statistically grounded approach provides a more objective and robust basis for comparing different drone configurations, propeller designs, and control strategies.

This experimental setup framework is directly inspired by and extends the practical collision handling experiments conducted in Chapter 3. By standardizing collision scenarios, flight conditions, and evaluation procedures, this framework enables reproducible and comparable assessments of drone collision recovery performance across diverse hardware designs and control algorithms. Its flexible and adaptable structure ensures that researchers can apply this framework to a wide variety of UAV platforms and experimental setups while maintaining scientific rigor and cross-study comparability.

5.3 Performance Metrics for Collision Recovery Evaluation

To objectively assess the effectiveness of collision recovery, this study defines two critical performance metrics: *recovery time* and *maximum deviation*. These metrics are specifically designed to quantitatively measure the quadrotor's ability to maintain flight stability and to return to its reference trajectory after a collision.

The rationale for selecting these two metrics is as follows: When a quadrotor experiences a collision, the immediate impact typically results in a significant disturbance to its current flight path. An effective collision recovery system should minimize both the extent of the trajectory deviation and the time required to return to stable flight along the original reference trajectory. By quantifying these two aspects, the framework offers a practical and measurable evaluation of the drone's resilience.

Recovery Time

Recovery time is defined as the elapsed time between the instant of collision and the moment when the quadrotor re-establishes stable flight within a predefined error threshold relative to the reference trajectory.

In this study, the evaluation specifically focuses on the quadrotor's altitude (z-axis) to determine the recovery point. This axis is selected because it is directly controlled by the thrust mechanism of the drone, whereas the x and y axes are indirectly influenced through tilt-induced motion due to the underactuated dynamics of quadrotors. Consequently, altitude provides a more direct and reliable indicator of the system's recovery status.

This criterion is illustrated in Figure 3.11, where the convergence of altitude error to the predefined recovery threshold after collision serves as a reliable reference for identifying the recovery time.

The collision recovery process is illustrated in Figure 5.2. In this figure, point **C** marks the collision location where the quadrotor begins to deviate from its planned trajectory (depicted in blue). Following the impact, the drone temporarily follows a disturbed trajectory (in red). Recovery time is measured from the collision point **C** to the recovery point **R**, which represents the moment the drone's altitude error returns to within the acceptable error threshold.

Maximum Deviation

Maximum deviation is defined as the greatest distance between the drone's actual trajectory and the reference trajectory during the post-collision recovery phase. As shown in Figure 5.2, this is represented by the green line, which measures the furthest positional deviation from the reference path.

This metric provides a direct measure of the collision's immediate effect on trajectory accuracy and indicates how severely the drone's stability is disrupted before recovery begins. A system with superior impact resilience will generally

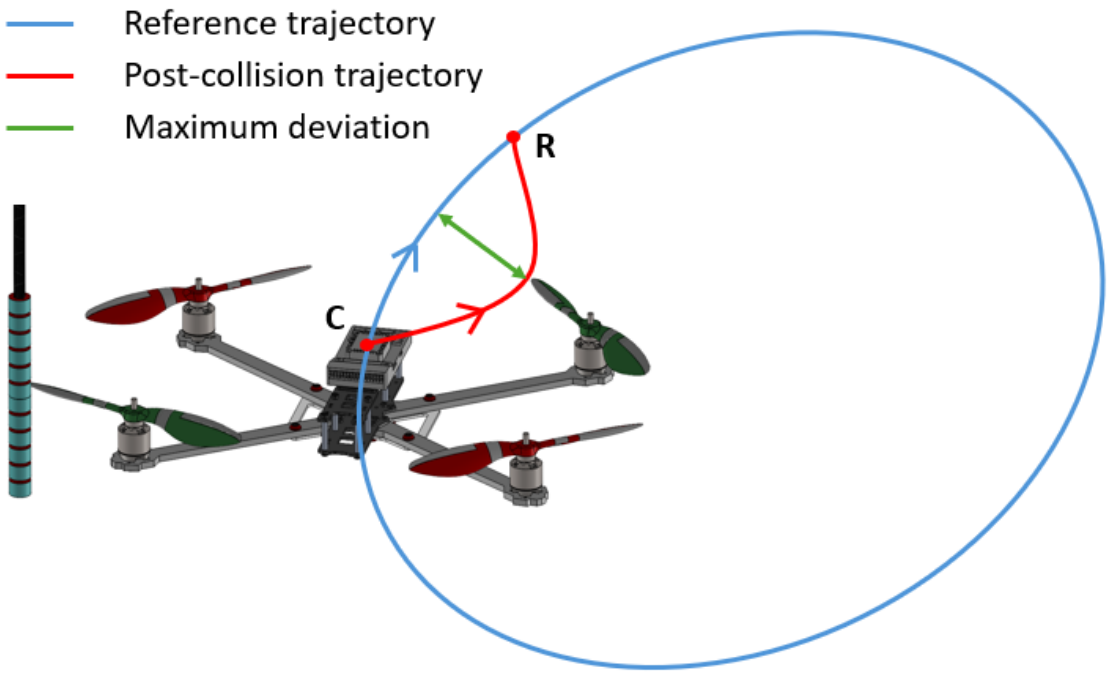


Figure 5.2: Illustration of Drone Trajectory Deviation During Collision. Point **C** indicates the collision location, while point **R** marks the recovery point. The green line represents the maximum deviation from the reference trajectory.

exhibit smaller maximum deviations, demonstrating better control authority and disturbance rejection.

Summary of Metric Selection

The combination of these two metrics—recovery time and maximum deviation—offers a comprehensive evaluation of the drone's post-collision performance. Recovery time captures the system's dynamic response and speed of stabilization, while maximum deviation reflects the system's ability to resist trajectory disturbances.

These metrics are not only suitable for evaluating the sensorless collision handling approach developed in Chapter 3 but can also serve as standardized criteria for future studies that aim to benchmark different hardware configurations, control strategies, or collision scenarios. By focusing on quantifiable and reproducible outcomes, this framework establishes a solid foundation for comparative analysis in collision recovery research.

5.4 Summary and Future Utility

This chapter presents a benchmarking framework that offers a systematic and repeatable methodology for evaluating post-collision recovery performance in quadrotors. Unlike conventional assessments that often focus solely on drone survivability, the proposed framework emphasizes detailed, quantitative recovery performance metrics, allowing for more meaningful comparisons across different hardware designs and control strategies.

Specifically, this framework introduces two essential and complementary performance metrics: *recovery time* and *maximum deviation*. These metrics are carefully selected to provide both dynamic and geometric perspectives on the drone's response to collisions, capturing the recovery process with clear physical meaning. The rationale for these metrics has been validated through the experiments and evaluations presented in Chapter 3, where the sensorless $\mathcal{L}_1$Quad control framework demonstrated its ability to achieve fast and stable recovery in various collision scenarios.

By focusing on these objective and quantifiable criteria, the framework serves as a valuable tool not only for validating the proposed sensorless collision handling system but also for future research efforts seeking to benchmark other drone configurations, hardware components, and control algorithms.

The benchmarking framework is designed to be widely applicable and adaptable to different UAV platforms and experimental conditions. It provides a foundation for future researchers to:

- Objectively evaluate and compare the collision recovery capabilities of various impact mitigation hardware, including deformable propellers, protective cages, or compliant frame designs.
- Benchmark different control strategies on the same drone structure to identify the most effective collision handling methods for specific hardware

setups.

- Design systematic and reproducible collision experiments that can generate comparable results across different studies.
- Develop more robust and safer UAV systems capable of maintaining operational stability in complex and unpredictable environments.

The introduction of this benchmarking framework is a significant academic contribution of this thesis. It establishes a universal reference for quantitatively evaluating drone resilience to collisions and provides a standardized methodology that future research can adopt to assess and compare UAV safety solutions. By bridging the gap between isolated experimental validations and broader, comparative assessments, this framework is expected to facilitate the advancement of collision-tolerant UAV technologies in both research and industrial applications.

Chapter 6

Conclusion

6.1 Discussion

This thesis presents two complementary approaches to collision handling for the Tombo quadrotor, aimed at enhancing safety and flight reliability in collision-prone environments. The first approach involves sensorless collision mitigation by integrating the $\mathcal{L}_1$Quad controller, compensating for uncertainties caused by Tombo propellers and quickly recovers trajectory tracking performance after collisions. The second approach introduces a sensor-based collision detection and response framework utilizing rotary encoders embedded in the Tombo propellers, enabling timely identification of collisions and facilitating adaptive control responses. Together, these solutions significantly improve the resilience and stability of the Tombo quadrotor during and following in-flight collisions.

6.1.1 Sensorless Collision Handling for Tombo Quadrotor

The integration of Tombo propellers with the $\mathcal{L}_1$Quad control framework offers a robust solution to address the dynamic uncertainties introduced by deformable rotor systems. While Tombo propellers improve collision resilience by absorbing impact forces and reducing structural damage, their compliant nature induces instabilities during aggressive maneuvers or minor impacts, such as transient thrust losses and variations in torque generation. $\mathcal{L}_1$Quad effectively mitigates these effects through adaptive compensation, particularly enhancing trajectory

tracking along the z-axis where deviations are most pronounced.

Experimental evaluations in diverse scenarios—including collision-free flight, in-flight contact with foam and carbon fiber obstacles, and hover collisions—consistently demonstrate that the combination of Tombo propellers and $\mathcal{L}_1$Quad ("Tombo $\mathcal{L}_1$ on") outperforms other configurations. It enables stable flight in the presence of disturbances, faster recovery after impacts, and smaller trajectory deviations compared to setups using rigid propellers or lacking adaptive control. However, without $\mathcal{L}_1$Quad, the flexibility of Tombo propellers may degrade flight performance due to unmitigated instabilities, underscoring the necessity of adaptive augmentation.

Despite these advantages, challenges remain when the quadrotor encounters strong or prolonged collisions. In such events, the sustained loss of propeller speed may prevent the controller from generating adequate compensatory thrust, leading to loss of control or instability. This limitation reveals the need for complementary strategies beyond adaptive control to ensure safe operation under severe impact conditions.

6.1.2 Sensor-Based Collision Handling for Tombo Quadrotor

The integration of rotary encoders on all four motors of the quadrotor equipped with Tombo propellers enables real-time and highly reliable detection of propeller collisions. By directly measuring propeller rotational velocity, the encoder-based approach exhibits strong resistance to external disturbances and noise, allowing for accurate and consistent detection. Crucially, it enables precise localization of the impacted propeller, which facilitates the design of targeted control responses depending on which rotor is affected. The proposed collision detection algorithm is grounded in both theoretical modeling of collision dynamics and extensive experimental validation, ensuring its robustness under various conditions. Building

upon this capability, two post-collision recovery strategies have been developed and experimentally validated, both demonstrating effective recovery performance when triggered by encoder-based collision events.

Despite its effectiveness, the proposed system still presents several limitations. First, the shared processing of four encoder signals limits the overall sampling rate, introducing latency in collision detection that may prolong obstacle contact and affect flight stability. Second, the current recovery strategies have only been tested in scenarios involving a single collision along a predefined trajectory. Consequently, the system is not yet equipped to handle multiple or successive impacts, as it lacks memory or trajectory adaptation mechanisms to optimize recovery in such cases. Third, since encoders are only installed on the propellers, the detection framework remains limited to rotor-level impacts and does not provide a comprehensive awareness of collisions involving other parts of the quadrotor structure.

6.2 Conclusions

This thesis has successfully demonstrated two complementary and effective solutions that significantly enhance the safety, resilience, and operational reliability of quadrotors equipped with Tombo propellers in collision-prone environments.

First, the implementation of the $\mathcal{L}_1$Quad control framework—a combination of geometric control and $\mathcal{L}_1$ adaptive augmentation—proves to be a powerful, sensorless strategy for handling light or brief collisions as well as compensating for aerodynamic uncertainties caused by the deformable nature of Tombo propellers. This adaptive controller enables rapid stabilization and robust trajectory tracking without relying on explicit collision detection, directly addressing disturbances and uncertainties in real time. Experimental results confirm that $\mathcal{L}_1$Quad effectively mitigates flight instabilities, maintains precise path following, and provides superior recovery performance compared to traditional rigid propeller setups, fully fulfilling the objective of fast and reliable sensorless collision recovery.

Second, the integration of rotary encoders with the Tombo propellers establishes a precise sensor-based collision detection and response mechanism capable of identifying strong or sustained collisions. By monitoring real-time variations in propeller rotational speeds, the system accurately detects which propeller has been impacted, enabling timely activation of reactive flight modes such as hovering stabilization and trajectory replanning. This direct sensing approach overcomes the limitations of indirect fault detection methods, reducing false positives and improving localization accuracy. Despite some constraints in sampling frequency and positional resolution, the encoder-based system demonstrates practical effectiveness in real-world dynamic environments, meeting the goal of precise and reliable collision identification for enhanced safety and mission continuity.

Together, these two strategies form a unified and robust framework that addresses a wide range of collision scenarios—from subtle disturbances to significant impacts—while leveraging the unique mechanical advantages of Tombo propellers. The synergy between the adaptive control capabilities of $\mathcal{L}_1$Quad and the encoder-enabled collision awareness yields improved flight stability, faster recovery times, and minimized trajectory deviations, outperforming rigid propeller configurations under diverse operating conditions.

By achieving the research objectives outlined in the Introduction, this work validates the practical applicability of these solutions for real-world UAV operations. The combined sensing and control approach not only enhances quadrotor resilience in cluttered and unpredictable environments but also sets a foundation for future advances in autonomous aerial systems requiring safe, precise, and adaptive flight performance. Potential applications include urban air mobility, search-and-rescue missions, and industrial inspection tasks, where maintaining structural integrity and operational continuity is critical.

Overall, this thesis establishes a comprehensive methodology that harnesses both sensorless adaptive control and sensor-based detection to realize collision-resilient quadrotors, offering a meaningful step forward in the design and deploy-

ment of next-generation UAV technologies.

In addition to these contributions, this thesis proposes a benchmarking framework for evaluating collision recovery performance, which provides a systematic and flexible tool for objectively assessing drone resilience after collisions. By defining standardized experimental setups and quantitative performance metrics, the framework enables future researchers to consistently compare the effectiveness of different hardware designs, control strategies, or their combinations in mitigating collision effects. This contribution offers a valuable reference for advancing collision-tolerant UAV research and supports the development of safer, more robust aerial platforms across diverse applications.

6.3 Future Works

While this thesis has made significant strides in enhancing UAV safety and resilience, several avenues for future research remain. First, improving the localization accuracy of collision events by integrating absolute encoders to obtain the angular position of the propeller at the moment of impact is essential. Combining this data with IMU measurements from the flight controller could enable precise estimation of the impact force direction, facilitating more accurate obstacle localization. Second, increasing the speed update rate of the encoders beyond the current limit of 50 Hz would reduce detection delays, minimizing the risk of prolonged obstacle contact and thrust loss. Third, developing mapping strategies for multipoint collision scenarios, such as building a collision map and designing a smart avoidance algorithm, would enable UAV to determine optimal flight paths in complex real-world conditions involving multiple collisions.

Furthermore, future work should focus on integrating collision avoidance control modes with the $\mathcal{L}_1$Quad framework (see Figure 6.1). Such an approach would allow UAV to simultaneously detect collisions using encoder data, leverage $\mathcal{L}_1$Quad for rapid stabilization and recovery, and navigate to the intended destination after

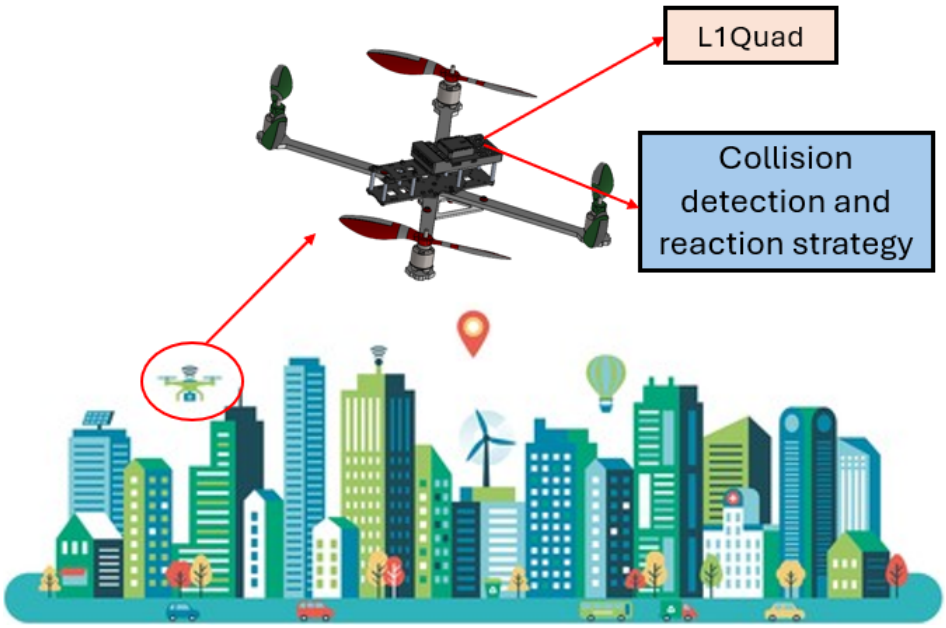


Figure 6.1: Flight testing of the Tombo quadrotor using the $\mathcal{L}_1$Quad controller and the quadrotor's collision detection and response strategy while navigating to a target destination in an outdoor environment.

collision. This integration could combine reactive flight modes with adaptive control to ensure collision avoidance and trajectory tracking under a unified control architecture. In addition, implementing both encoder-based collision detection and the $\mathcal{L}_1$Quad controller on the same drone platform enables the system to operate effectively in outdoor environments without relying on indoor motion capture systems. For outdoor deployment, global positioning system (GPS) can be used to provide position data, while onboard IMU sensors estimate the drone's orientation in real-time, replacing the role of vision-based tracking systems typically used indoors. This sensor configuration supports both global navigation and high-frequency attitude estimation, essential for collision detection and adaptive control in outdoor missions. Additionally, incorporating the collision simulation of Tombo propellers into a comprehensive drone simulation program could provide deeper insights into the UAV's behavior during collisions, enabling the design of more robust and generalized response strategies for a wider range of operational scenarios.

In addition, this study's proposed collision detection method is specifically

designed to detect impacts on the propellers. As a result, it cannot detect collisions on the drone's main body. However, numerous previous studies have explored body collision detection using IMU-based approaches or other sensing modalities such as vision-based systems. In future work, I plan to integrate these complementary techniques to enhance overall safety, enabling robust and comprehensive collision detection for both the propellers and the drone body.

Finally, these proposed methods should be implemented and tested in real-world outdoor flight environments to evaluate their performance and refine their effectiveness in handling actual collision scenarios.

References

[1] G. M. I. Inc., *Commercial Drone Market Report*. Global Market Insights Inc., July 2024.

[2] P. Kornatowski, S. Mintchev, and D. Floreano, "An origami-inspired cargo drone," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vancouver, BC, Canada, 2017, pp. 6855–6862.

[3] S. Mintchev, S. de Rivaz, and D. Floreano, "Insect-inspired mechanical resilience for multicopters," *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1248–1255, 2017.

[4] F. Ruiz, B. Arrue, and A. Ollero, "Bio-inspired deformable propeller concept for smooth human-uav interaction and efficient thrust generation," *IEEE Robotics and Automation Letters*, vol. 8, no. 6, pp. 3430–3437, 2023.

[5] J. Jang, K. Cho, and G.-H. Yang, "Design and experimental study of dragonfly-inspired flexible blade to improve safety of drones," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4200–4207, 2019.

[6] D. Q. Nguyen, G. Loianno, and V. A. Ho, "Towards design of a deformable propeller for drone safety," in *2020 3rd IEEE International Conference on Soft Robotics (RoboSoft)*, 2020, pp. 464–469.

[7] S. T. Bui, Q. K. Luu, D. Q. Nguyen, N. D. M. Le, G. Loianno, and V. A. Ho, "Tombo propeller: Bioinspired deformable structure toward collision-accommodated control for drones," *IEEE Transactions on Robotics*, vol. 39, no. 1, pp. 521–538, 2023.

[8] Z. Wu, S. Cheng, P. Zhao, A. Gahlawat, K. A. Ackerman, A. Lakshmanan, C. Yang, J. Yu, and N. Hovakimyan, "L1quad: L1 adaptive augmentation of geometric control for agile quadrotors with performance guarantees," *IEEE Transactions on Control Systems Technology*, vol. 33, no. 2, pp. 597–612, 2025.

[9] V. Scognamiglio, J. Cacace, F. Ruggiero, and V. Lippiello, "Fault detection and isolation for a standard quadrotor using a deep neural network trained on a momentum-based estimator," in *2024 IEEE 20th International Conference on Automation Science and Engineering (CASE)*, 2024, pp. 730–735.

[10] S. Sun, G. Cioffi, C. de Visser, and D. Scaramuzza, "Autonomous quadrotor flight despite rotor failure with onboard vision sensors: Frames vs. events," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 580–587, 2021.

[11] X. Zhang, Z. Zhao, Z. Wang, and X. Wang, "Fault detection and identification method for quadcopter based on airframe vibration signals," *Sensors*, vol. 21, no. 2, 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/2/581

[12] G. Chen, W. Dong, X. Sheng, X. Zhu, and H. Ding, "An active sense and avoid system for flying robots in dynamic environments," *IEEE/ASME Transactions on Mechatronics*, vol. 26, no. 2, pp. 668–678, 2021.

[13] K. N. McGuire, C. D. Wagter, K. Tuyls, H. J. Kappen, and G. C. H. E. de Croon, "Minimal navigation solution for a swarm of tiny flying robots to explore an unknown environment," *Science Robotics*, vol. 4, no. 35, p. eaaw9710, 2019.

[14] M. Lu, H. Chen, and P. Lu, "Perception and avoidance of multiple small fast moving objects for quadrotors with only low-cost rgbd camera," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 11 657–11 664, 2022.

[15] L. Lu, G. Fasano, A. Carrio, M. Lei, H. Bavle, and P. Campoy, "A comprehensive survey on non-cooperative collision avoidance for micro aerial vehicles: Sensing and obstacle detection," *Journal of Field Robotics*, vol. 40, no. 6, pp. 1697–1720, 2023. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.22189

[16] B. Herissé, T. Hamel, R. Mahony, and F.-X. Russotto, "Landing a vtol unmanned aerial vehicle on a moving platform using optical flow," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 77–89, 2012.

[17] A. J. Barry, P. R. Florence, and R. Tedrake, "High-speed autonomous obstacle avoidance with pushbroom stereo," *Journal of Field Robotics*, vol. 35, no. 1, pp. 52–68, 2018. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21741

[18] M. C. P. Santos, L. V. Santana, A. S. Brandão, and M. Sarcinelli-Filho, "Uav obstacle avoidance using rgb-d system," in *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2015, pp. 312–319.

[19] H. Oleynikova, D. Honegger, and M. Pollefeys, "Reactive avoidance using embedded stereo vision for mav flight," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 50–56.

[20] J. N. Yasin, S. A. S. Mohamed, M.-H. Haghbayan, J. Heikkonen, H. Tenhunen, and J. Plosila, "Unmanned aerial vehicles (uavs): Collision avoidance systems and approaches," *IEEE Access*, vol. 8, pp. 105 139–105 155, 2020.

[21] D. Falanga, K. Kleber, and D. Scaramuzza, "Dynamic obstacle avoidance for quadrotors with event cameras," *Science Robotics*, vol. 5, no. 40, 2020.

[22] S. U. Sharma and D. J. Shah, "A practical animal detection and collision avoidance system using computer vision technique," *IEEE Access*, vol. 5, pp. 347–358, 2017.

[23] N. Gageik, P. Benz, and S. Montenegro, "Obstacle detection and collision avoidance for a uav with complementary low-cost sensors," *IEEE Access*, vol. 3, pp. 599–609, 2015.

[24] Y. Luo, J. Remillard, and D. Hoetzer, "Pedestrian detection in near-infrared night vision system," in *2010 IEEE Intelligent Vehicles Symposium*, 2010, pp. 51–58.

[25] M.-g. Cho, "A study on the obstacle recognition for autonomous driving rc car using lidar and thermal infrared camera," in *2019 Eleventh International Conference on Ubiquitous and Future Networks (ICUFN)*, 2019, pp. 544–546.

[26] D. Laurijssen, R. Kerstens, G. Schouten, W. Daems, and J. Steckel, "A flexible low-cost biologically inspired sonar sensor platform for robotic applications," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 9617–9623.

[27] H. S. and, "An aeromagnetic survey carried out using a rotary-wing uav equipped with a low-cost magneto-inductive sensor," *International Journal of Remote Sensing*, vol. 42, no. 23, pp. 8805–8818, 2021. [Online]. Available: https://doi.org/10.1080/01431161.2021.1930269

[28] R. Reynolds, B. Barrowes, T. Shubitidze, C. Hartshorn, B. Quinn, and F. Shubitidze, "Electromagnetic induction sensing of unexploded ordinance and soil properties from unmanned aerial systems," in *Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XXVI*, S. S. Bishop and J. C. Isaacs, Eds., vol. 11750, International Society for Optics and Photonics. SPIE, 2021, p. 1175002. [Online]. Available: https://doi.org/10.1117/12.2585537

[29] M. C. P. Santos, C. D. Rosales, M. Sarcinelli-Filho, and R. Carelli, "A novel null-space-based uav trajectory tracking controller with collision avoidance," *IEEE/ASME Transactions on Mechatronics*, vol. 22, pp. 2543–2553, 12 2017.

[30] A. S. Huang, A. Bachrach, M. Krainin, D. Maturana, and N. Roy, *Robotics Reseach*, ser. Springer Tracts in Advanced Robotics. Springer, Cham, 2017, vol. 100.

[31] P. Castillo, R. Lozano, and A. Dzul, "Stabilization of a mini-rotorcraft having four rotors," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, vol. 3, 2004, pp. 2693–2698 vol.3.

[32] S. Bouabdallah, P. Murrieri, and R. Siegwart, "Towards autonomous indoor micro vtol," *Autonomous Robots*, vol. 18, no. 2, pp. 171–183, 2005.

[33] T. Madani and A. Benallegue, "Backstepping control for a quadrotor helicopter," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 3255–3260.

[34] M. Labbadi and M. Cherkaoui, "Robust adaptive backstepping fast terminal sliding mode controller for uncertain quadrotor uav," *Aerospace Science and Technology*, vol. 93, p. 105306, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1270963819300318

[35] D. Lee, H. J. Kim, and S. Sastry, "Feedback linearization vs. adaptive sliding mode control for a quadrotor helicopter," *International Journal of Control, Automation and Systems*, vol. 7, no. 3, pp. 419–428, 2009. [Online]. Available: https://doi.org/10.1007/s12555-009-0311-8

[36] H. Voos, "Nonlinear control of a quadrotor micro-uav using feedback-linearization," in *2009 IEEE International Conference on Mechatronics*, 2009, pp. 1–6.

[37] J. HILKERT, "Inertially stabilized platform technology concepts and principles," *IEEE Control Systems Magazine*, vol. 28, no. 1, pp. 26–46, 2008.

[38] V. Jurdjevic, *Geometric Control Theory*, ser. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 1996.

[39] D. Mellinger and V. Kumar, "Minimum snap trajectory generation and control for quadrotors," in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 2520–2525.

[40] T. Lee, M. Leok, and N. H. McClamroch, "Geometric tracking control of a quadrotor uav on se(3)," in *49th IEEE Conference on Decision and Control (CDC)*, 2010, pp. 5420–5425.

[41] T. Lee, M. Leok, and N. Mcclamroch, "Control of complex maneuvers for a quadrotor uav using geometric methods on se(3)," 03 2010.

[42] G. Antonelli, E. Cataldi, F. Arrichiello, P. Robuffo Giordano, S. Chiaverini, and A. Franchi, "Adaptive trajectory tracking for quadrotor mavs in presence of parameter uncertainties and external disturbances," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 1, pp. 248–254, 2018.

[43] F. A. Goodarzi, D. Lee, and T. Lee, "Geometric adaptive tracking control of a quadrotor unmanned aerial vehicle on se(3) for agile maneuvers," *Journal of Dynamic Systems, Measurement, and Control*, vol. 137, no. 9, p. 091007, 06 2015. [Online]. Available: https://doi.org/10.1115/1.4030419

[44] Z. T. Dydek, A. M. Annaswamy, and E. Lavretsky, "Adaptive control of quadrotor uavs: A design trade study with flight evaluations," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 4, pp. 1400–1406, 2013.

[45] L. Besnard, Y. B. Shtessel, and B. Landrum, "Quadrotor vehicle control via sliding mode controller driven by sliding mode disturbance observer," *Journal of the Franklin Institute*, vol. 349, no. 2, pp. 658–684, 2012, advances in Guidance and Control of Aerospace Vehicles using

Sliding Mode Control and Observation Techniques. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0016003211001839

[46] A. Castillo, R. Sanz, P. Garcia, W. Qiu, H. Wang, and C. Xu, "Disturbance observer-based quadrotor attitude tracking control for aggressive maneuvers," *Control Engineering Practice*, vol. 82, pp. 14–23, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S096706611830563X

[47] K. Guo, W. Zhang, Y. Zhu, J. Jia, X. Yu, and Y. Zhang, "Safety control for quadrotor uav against ground effect and blade damage," *IEEE Transactions on Industrial Electronics*, vol. 69, no. 12, pp. 13 373–13 383, 2022.

[48] W.-H. Chen, J. Yang, L. Guo, and S. Li, "Disturbance-observer-based control and related methods—an overview," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 2, pp. 1083–1095, 2016.

[49] E. Tal and S. Karaman, "Accurate tracking of aggressive quadrotor trajectories using incremental nonlinear dynamic inversion and differential flatness," *IEEE Transactions on Control Systems Technology*, vol. 29, no. 3, pp. 1203–1218, 2021.

[50] A. Saviolo, J. Frey, A. Rathod, M. Diehl, and G. Loianno, "Active learning of discrete-time dynamics for uncertainty-aware model predictive control," *IEEE Transactions on Robotics*, vol. 40, pp. 1273–1291, 2024.

[51] M. O'Connell, G. Shi, X. Shi, K. Azizzadenesheli, A. Anandkumar, Y. Yue, and S.-J. Chung, "Neural-fly enables rapid learning for agile flight in strong winds," *Science Robotics*, vol. 7, no. 66, p. eabm6597, 2022. [Online]. Available: https://www.science.org/doi/abs/10.1126/scirobotics.abm6597

[52] K. Y. Chee, T. Z. Jiahao, and M. A. Hsieh, "Knode-mpc: A knowledge-based data-driven predictive control framework for aerial robots," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2819–2826, 2022.

[53] G. Torrente, E. Kaufmann, P. Föhn, and D. Scaramuzza, "Data-driven mpc for quadrotors," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3769–3776, 2022. [Online]. Available: https://doi.org/10.5167/uzh-216588

[54] N. Hovakimyan and C. Cao, *1 Adaptive Control Theory*. Society for Industrial and Applied Mathematics, 2010. [Online]. Available: https://epubs.siam.org/doi/abs/10.1137/1.9780898719376

[55] Z. Zuo and P. Ru, "Augmented l1 adaptive tracking control of quad-rotor unmanned aircrafts," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, no. 4, pp. 3090–3101, 2014.

[56] M. Q. Huynh, W. Zhao, and L. Xie, "L1 adaptive control for quadcopter: Design and implementation," in *2014 13th International Conference on Control Automation Robotics Vision (ICARCV)*, 2014, pp. 1496–1501.

[57] I. Gregory, C. Cao, E. Xargay, N. Hovakimyan, and X. Zou, *L1 Adaptive Control Design for NASA AirSTAR Flight Test Vehicle*. [Online]. Available: https://arc.aiaa.org/doi/abs/10.2514/6.2009-5738

[58] K. Ackerman, E. Xargay, R. Choe, N. Hovakimyan, M. C. Cotting, R. B. Jeffrey, M. P. Blackstun, T. P. Fulkerson, T. R. Lau, and S. S. Stephens, *L1 Stability Augmentation System for Calspan's Variable-Stability Learjet*. [Online]. Available: https://arc.aiaa.org/doi/abs/10.2514/6.2016-0631

[59] K. A. Ackerman, E. Xargay, R. Choe, N. Hovakimyan, M. C. Cotting, R. B. Jeffrey, M. P. Blackstun, T. P. Fulkerson, T. R. Lau, and S. S. Stephens, "Evaluation of an l1 adaptive flight control law on calspan's variable-stability learjet," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 4, pp. 1051–1060, 2017. [Online]. Available: https://doi.org/10.2514/1.G001730

[60] I. Kaminer, A. Pascoal, E. Xargay, N. Hovakimyan, C. Cao, and V. Dobrokhodov, "Path following for small unmanned aerial vehicles using

l1 adaptive augmentation of commercial autopilots," *Journal of Guidance, Control, and Dynamics*, vol. 33, no. 2, pp. 550–564, 2010. [Online]. Available: https://doi.org/10.2514/1.42056

[61] H. Jafarnejadsani, D. Sun, H. Lee, and N. Hovakimyan, "Optimized l1 adaptive controller for trajectory tracking of an indoor quadrotor," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 6, pp. 1415–1427, 2017. [Online]. Available: https://doi.org/10.2514/1.G000566

[62] R. R. Lima and G. A. S. Pereira, "Drone collision detection and classification using proprioceptive data," in *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2022, pp. 562–569.

[63] S. Wang, N. Anselmo, M. Garrett, R. Remias, M. Trivett, A. Christoffersen, and N. Bezzo, "Fly-crash-recover: A sensor-based reactive framework for online collision recovery of uavs," in *2020 Systems and Information Engineering Design Symposium (SIEDS)*, 2020, pp. 1–6.

[64] R. Wang, Z. Guo, Y. Chen, X. Wang, and B. M. Chen, "Air bumper: A collision detection and reaction framework for autonomous mav navigation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 15 735–15 741.

[65] G. Dicker, F. Chui, and I. Sharf, "Quadrotor collision characterization and recovery control," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 5830–5836.

[66] X. Huang, G. Liu, and Y. Liu, "Collision detection and recovery control of drones using onboard inertial measurement unit," *Drones*, vol. 9, no. 5, 2025. [Online]. Available: https://www.mdpi.com/2504-446X/9/5/380

[67] K. Patnaik, S. Mishra, and Z. Chase, "Collision recovery control of a foldable quadrotor," 05 2021.

[68] A. Bakir, D. Özbek, A. Abazari, and O. Özcan, "Scorer: Sensorized collision resilient aerial robot," in *2023 IEEE International Conference on Soft Robotics (RoboSoft)*, 2023, pp. 1–7.

[69] Z. Liu and K. Karydis, "Toward impact-resilient quadrotor design, collision characterization and recovery control to sustain flight after collisions," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 183–189.

[70] Y. Mulgaonkar, W. Liu, D. Thakur, K. Daniilidis, C. J. Taylor, and V. Kumar, "The tiercel: A novel autonomous micro aerial vehicle that can map the environment by flying into obstacles," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 7448–7454.

[71] H. T. Pham, D. Q. Nguyen, S. T. Bui, G. Loianno, and V. A. Ho, "Collision dynamics of motorized deformable propellers for drones," in *2024 21st International Conference on Ubiquitous Robots (UR)*, 2024, pp. 176–183.

[72] L. Bauersfeld and D. Scaramuzza, "Range, endurance, and optimal speed estimates for multicopters," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2953–2960, 2022.

[73] Z. Wu, S. Cheng, K. A. Ackerman, A. Gahlawat, A. Lakshmanan, P. Zhao, and N. Hovakimyan, "L1adaptive augmentation for geometric tracking control of quadrotors," in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 1329–1336.

[74] S. Cheng, M. Kim, L. Song, C. Yang, Y. Jin, S. Wang, and N. Hovakimyan, "Difftune: Autotuning through autodifferentiation," *IEEE Transactions on Robotics*, vol. 40, pp. 4085–4101, 2024.

[75] A. Öchsner, *Timoshenko Beam Theory*, 06 2021, pp. 67–104.

[76] M. Tummers, V. Lebastard, F. Boyer, J. Troccaz, B. Rosa, and M. T. Chikhaoui, "Cosserat rod modeling of continuum robots from newtonian and lagrangian perspectives," *IEEE Transactions on Robotics*, vol. 39, no. 3, pp. 2360–2378, 2023.

[77] F. Janabi-Sharifi, A. Jalali, and I. D. Walker, "Cosserat rod-based dynamic modeling of tendon-driven continuum robots: A tutorial," *IEEE Access*, vol. 9, pp. 68 703–68 719, 2021.

[78] P. E. Nikravesh and M. Poursina, "Determination of effective mass for continuous contact models in multibody dynamics," *Multibody System Dynamics*, vol. 58, no. 3, pp. 253–273, 2023.

[79] W. J. Stronge, *Planar or Two-Dimensional Rigid Body Impact*. Cambridge University Press, 2018, p. 35–64.