

Modeling and control of a flying wing Tailsitter unmanned aerial vehicle

Romain Chiappinelli

Master of Engineering

Department of Mechanical Engineering

McGill University

Montreal, Quebec

April, 2018

A thesis submitted to McGill University in partial fulfillment of the requirements
of the degree of Master of Engineering

©Romain Chiappinelli, 2018

ACKNOWLEDGEMENTS

First of all, I would like to acknowledge my supervisor, Professor Meyer Nahon for his availability, responsivity and guidance through my thesis. He redirected me towards the correct path when I was trying to run in all directions.

I also would like to thank my colleagues at the Aerospace Mechatronics Laboratory for their help and comments in the lab and their hands during the flight tests: Eitan Bulka, Bassam Haq, Josh Levin, Adrian Battison, Mikkel Jorgensen, Sam El Toufaili, Fares El Tin, Juan Carlos Hernandez, Sahand Rezaei Shoshtari and Michael Verrecchia.

In addition, I would like to thank Waqas Khan for his work on flat plate aerodynamics whose model is used in this research.

I would like to thank my parents and family for their moral support throughout this work.

Finally, I would like to thank my partner Émilie Julien for her patience, energy and source of motivation. She helped me climb the slope everytime my motivation was going down.

This research was funded by "le Fonds de recherche du Québec - Nature et technologies" (FRQNT) and "the Natural Sciences and Engineering Research Council of Canada" (NSERC).

ABSTRACT

Tailsitters are a special class of fixed-wing unmanned aerial vehicle intended to bridge the gap between rotorcraft and conventional fixed-wing aircraft. These systems are able to perform aerobatic and stationary maneuvers, including vertical takeoff and landing, as well as efficient level flight. However, this flying ability brings a control challenge due to the two distinct flight regimes. During vertical maneuvers, the wings are stalled and only the thrust forces support the aircraft's weight. The rear control surfaces, called elevons, are kept effective due to the slipstream generated by the thrusters. During level flight, the aircraft flies at a substantial forward velocity which generates lift from the wings as well as control authority from the elevons. In this research, a real time simulator is developed for the full flight envelope range, based on a component breakdown method. The simulator includes a flat plate aerodynamics model which includes the effect of control surfaces deflection, a ground contact model, as well as a semi-empirical thruster model. A single quaternion-based controller is developed and implemented in this simulated environment and also tested on the real platform. The autonomous maneuvers needed for a real flight mission are demonstrated through experiments, including vertical takeoff and climb, transition to level flight, back transition, stationary flight, vertical descent and landing. The results from both simulations and flight experiments are compared and used to qualitatively evaluate the performances of the simulator.

ABRÉGÉ

Les "Tailsitters" sont une catégorie spéciale de véhicules aériens sans pilote à voilure fixe destinés à combler l'écart entre les giravions et les avions conventionnels à voilure fixe. Ces systèmes sont capables d'effectuer des manœuvres acrobatiques et stationnaires, y compris décollages et atterrissages verticaux, ainsi qu'un vol en palier efficace. Cependant, cette capacité de vol amène un défi du point de vue du contrôleur en raison des deux régimes de vol distincts. Pendant les manœuvres verticales, les ailes sont en situation de décrochage et seuls les propulseurs soutiennent l'avion dans les airs. Les surfaces de contrôle arrière, appelées élevons, sont maintenues efficaces en raison du flux d'air généré par le sillage des propulseurs. Alors qu'en vol en palier, l'avion vole à une vitesse importante qui génère une portance des ailes ainsi qu'un contrôle par les élevons. Dans cette recherche, un simulateur temps réel est développé pour l'entièreté de l'enveloppe de vol, basé sur une méthode de décomposition des segments de l'aile. Le simulateur comprend une aérodynamique complète basé sur un profil d'aile plate, qui modélise les effets de la défexion des surfaces de contrôle, un modèle pour les contacts avec le sol, ainsi qu'un modèle semi-empirique pour les propulseurs. Un unique contrôleur, basé sur les quaternions est ensuite développé et implémenté dans cet environnement simulé et enfin testé sur la plate-forme réelle. Les manœuvres autonomes nécessaires à une véritable mission de vol sont démontrées au moyen d'expériences, notamment le décollage et la montée verticale, la transition vers le vol en palier, la transition inverse vers le vol stationnaire, ainsi que la descente et l'atterrissage verticale. Les résultats des simulations et des expériences en vol sont finalement comparés et utilisés pour évaluer qualitativement les performances du simulateur.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	ii
ABSTRACT	iii
ABRÉGÉ	iv
LIST OF TABLES	viii
LIST OF FIGURES	ix
1 Introduction	2
1.1 Background and Motivation	2
1.2 Platform Selection	3
1.3 Literature Review	4
1.3.1 Tailsitter UAVs	4
1.3.2 Real Time Simulator	6
1.4 Research Objectives	7
1.5 Thesis Organization	8
2 Aircraft Simulation	9
2.1 Frames and Conventions	9
2.2 Aircraft Equations of Motion	12
2.3 Dynamics Modeling	13
2.3.1 Velocity and Angle of Attack	15
2.3.2 Gravity	17
2.3.3 Wing Aerodynamics	17
2.3.4 Structural Segments Aerodynamics	21
2.3.5 Thrusters and Gyroscopic Moments	24
2.3.6 Thruster Force and Torque	26
2.3.7 Ground Contact Dynamics	28
2.4 Simplified Analytical Aerodynamic Model	29
2.5 Measurements	32
2.5.1 Thruster Coefficients	32
2.5.2 Control Surface Deflection Map	36
2.5.3 Control Surface Effectiveness	37

3	Control Strategy	47
3.1	Maneuver Generator	48
3.1.1	Vertical Maneuvers	49
3.1.2	Level Flight	50
3.1.3	Transitions	52
3.1.4	Banked Turn	53
3.2	Controller	56
3.2.1	Position	56
3.2.2	Attitude	57
3.2.3	Thrust	58
3.2.4	Actuator Allocation	59
4	Flight Tests Results	62
4.1	Simulated Flights	62
4.1.1	Minimal Flight Mission	62
4.1.2	Banked Turn	65
4.2	Hardware-In-the-Loop Flights	67
4.2.1	Performance Evaluation	68
4.2.2	Results	69
4.3	Tethered Indoor Flights	71
4.3.1	Manually Assisted Hover Controller	72
4.3.2	Tethered Experiment	73
4.4	Experimental Flights	76
4.4.1	Vertical Flight	76
4.4.2	Minimal Flight Mission	80
4.4.3	Banked Turn	83
4.4.4	Aggressive Turnaround	87
4.5	Effect of the Center of Mass Location	89
5	Conclusion	92
5.1	Summary of Research	92
5.2	Main Conclusions	93
5.3	Suggestions for Future Work	95
	REFERENCES	97
	APPENDICES	100
A	Inertia and Center of Mass	100
B	Wind Model	102
C	Full Flight Envelope Simulator Measurements	105
D	Pitching Moment at Zero Deflection	107
E	Inverse Thruster Model	108
F	Hardware Components	110

G	Quaternion Definitions and Relations	115
H	Aircraft Parameters	118

LIST OF TABLES

<u>Table</u>		<u>page</u>
2–1 Horizontal Wing Segments		19
2–2 Vertical Wing Segments		20
2–3 Structural Rods		23
2–4 Reference Points for Contact Force		29
2–5 Rolling Moment Measures		39
2–6 Pitching Moment Measures		40
A.1 Tailsitter Components List		101
F.1 Hardware List		112
H.1 Aircraft Properties		118
H.2 Controller Gains		119

LIST OF FIGURES

<u>Figure</u>		<u>page</u>
1–1 Convair Pogo		3
1–2 Boeing V22 Osprey		3
1–3 X-Vert VTOL Platform		3
1–4 FireFLY6 Platform		3
2–1 Screenshot of the Simulator		9
2–2 Tailsitter CAD Model		10
2–3 Simulator Block Diagram		13
2–4 Component Breakdown		18
2–5 Aerodynamic Forces Coefficients of the Entire Aircraft		21
2–6 Pitching Moment Coefficient of the Entire Aircraft		21
2–7 Rod Segment in a Wind Field		22
2–8 Right Thruster		24
2–9 Two Blocks Diagram of the Thruster Model		27
2–10 Ground Contact Forces		28
2–11 Diagram of the Simplified Aerodynamics Model		29
2–12 Propeller Angular Speed Function of Throttle and Battery Voltage . .		33
2–13 Propeller Force and Torque Testbed		34
2–14 Static Propeller Coefficients Function of Angular Speed		35
2–15 Thrust and Power Coefficients Models, Function of Advance Ratio .		36
2–16 Elevon Deflection Map		37
2–17 Control Surface Effectiveness Testbed		38
2–18 Rolling Moment Deflection Coefficient from Simulator		41
2–19 Pitching Moment Deflection Coefficient from Simulator		42

2–20 Rolling Moment Deflection Coefficient from Simulator	43
2–21 Rolling Moment Deflection Coefficient from Simulator	44
2–22 Rolling Moment Coefficient Scaling	45
2–23 Pitching Moment Coefficient Scaling	46
3–1 Block Diagram of the Cascaded Quaternion Controller.	47
3–2 Minimal Flight Mission Reference	48
3–3 Flying Wing Dimensions	50
3–4 Level Flight Free Body Diagram	50
3–5 Equilibrium Pitch Angle in Level Flight	52
3–6 Banked Turn Free Body Diagram	54
3–7 Banked Turn Reference Trajectory	54
4–1 Schematic of the Conventional Simulation	62
4–2 Minimal Flight Mission Trajectory in Simulation	63
4–3 Minimal Flight Mission States in Simulation	63
4–4 Minimal Flight Mission Control States in Simulation	64
4–5 Banked Turn Trajectory in Simulation	65
4–6 Banked Turn States in Simulation	66
4–7 Banked Turn Control States in Simulation	67
4–8 HIL Simulation Diagram	68
4–9 HIL Update Rate	69
4–10 HIL Update Rate, Probability Density	69
4–11 Unstable Hover Trajectory in HIL	71
4–12 Manually Assisted Hover Controller Diagram	71
4–13 Radio Control Transmitter	72
4–14 Tethered Setup for Indoor Flight Experiments	74
4–15 Tethered Hover Experiment Trajectory	74
4–16 Tethered Hover Experiment States	75

4–17 Vertical Flight Trajectory	78
4–18 Vertical Flight Mosaic.	78
4–19 Vertical Flight States	79
4–20 Vertical Flight Control Signals	79
4–21 Minimal Flight Mission Trajectory	81
4–22 Minimal Flight Mission Top View	81
4–23 Minimal Flight Mission States	82
4–24 Minimal Flight Mission Control Signals	83
4–25 Banked Turn Trajectory	84
4–26 Banked Turn States	85
4–27 Banked Turn Control Signals	86
4–28 Aggressive Turnaround Trajectory	88
4–29 Aggressive Turnaround Control Commands	89
4–30 Pitching Moment Coefficient for different CM	90
4–31 Back Transitions for different CM	91
A.1 CAD Model of the Tailsitter	100
B.1 Simulated Wind Speed Time History and Power Spectrum	103
C.1 Aerodynamics and Rod Drag Blocks of the Simulator	105
D.1 Model of the Pitching Moment Coefficient at zero Deflection	107
E.1 Block Diagram of the Inverse Thruster Model	108
F.1 Aircraft Hardware	110
F.2 Hardware Wiring Diagram	111
F.3 EKF Position Error	113
F.4 Distance Sensor Operational Diagram	114

CHAPTER 1

Introduction

1.1 Background and Motivation

Unmanned Aerial Vehicles (UAVs) are proposed in several applications such as search and rescue, monitoring, agriculture management, aerial photography, and 3D mapping [1]. Today's UAV market is mainly divided into rotorcraft and fixed wing aircraft. Rotorcraft possess vertical takeoff and landing (VTOL), hovering flight capabilities, and high maneuverability. By contrast, fixed wing aircraft are capable of efficient and fast level flight, suited for long range and endurance applications; but need a runway or other apparatus, such as a launcher and a net, for takeoff and landing. Furthermore, conventional fixed wing aircraft are not capable of hovering flight. The idea of combining the advantages of these two categories of aircraft has led to a new class called VTOL aircraft, or hybrid aircraft. Early designs of VTOL aircraft appeared in the 1950's with the Convair Pogo which was the first reported manned tailsitter [2], shown in Figure 1–1. The project was cancelled after few successful flights due to the difficulty for the pilot to perform the transitions from vertical to horizontal flight. More successful designs such as the Harrier Jump Jet or the tilt rotor Boeing V22 Osprey, shown in Figure 1–2, are still in use today. However, those designs have never been used in civilian applications.

While they are not well suited for transportation of people or goods, these concepts may have more promise as UAVs. Progress in modeling, control theory, and hardware miniaturization are all enabling technologies for autonomous VTOL aircraft developments, leading to an increased interest in VTOL aircraft in academic research on UAVs.

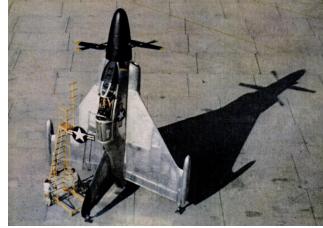


Figure 1–1: Convair Pogo [3].



Figure 1–2: Boeing V22 Osprey [4].



Figure 1–3: X-Vert VTOL Platform from Horizon Hobby [5].



Figure 1–4: FireFLY6 Platform from BirdsEyeView Aerobotics [6].

1.2 Platform Selection

Several hybrid designs are proposed today, many of which are home made. In this research, the objective is not to design an aircraft, but rather to study the dynamics and control. This is why an available airframe from the market is selected. Based on the maturity and reliability of the available platforms, only two VTOL airframes are considered:

- X-Vert VTOL from Horizon Hobby, shown in Figure 1–3.
- FireFLY6 from BirdsEyeView Aerobotics, shown in Figure 1–4.

The X-Vert tailsitter is an affordable platform with a wing span of about 0.5 m . This is a flying wing aircraft with two thrusters which takes off and lands nose up. In order to perform the transition from hovering flight, the entire aircraft needs to pitch down to enter level flight. This platform is available with a custom hardware and attitude controller able to perform the transition from vertical to horizontal flight and back transition from horizontal to vertical flight. However, if this platform is chosen, the entire hardware and controller will need to be replaced, as the embedded system is not open source. The high thrust to weight ratio allows this aircraft to

hover and fly vertically, where the thrusters provide the force to support the aircraft's weight. In this regime, the airstream blown by the thrusters, known as slipstream, ensures authority of the rear control surfaces to provide attitude control. During level flight, the aircraft is oriented almost horizontally and its weight is supported mainly by the lift force produced by the wings. In this second regime, the control surfaces are effective due to the airflow created by the substantial aircraft forward velocity.

The FireFLY6 is a tilt rotor design using an open source Pixhawk hardware controller. It is a flying wing of about 1.5 m wing span, with three counter-rotating thruster pairs. The two front thruster pairs can tilt to perform the transitions. During vertical maneuvers, this aircraft acts as a Y shaped hexarotor.

The first airframe is chosen due to its the mechanical simplicity and minimized number of actuators which make this tailsitter an elegant academic tool. One attraction is that tailsitters come with a bigger dynamics and control challenge due to the two flight regime which are aerodynamically very different.

1.3 Literature Review

1.3.1 Tailsitter UAVs

Several successful control strategies for tailsitter UAVs are reported by the research community. Tailsitters are able to fly at different flight regime, it results very nonlinear aerodynamics which are challenging to control with a single controller. To overcome this difficulty, attitude and height controllers that vary with the flight mode were derived, such as a switching logic choosing between hover and level flight linear controller [7] tested indoor on a agile fixed wing aircraft, a gain scheduling with adaptation for attitude control [8] tested outdoor on an 18 kg tailsitter, or a hybrid controller [9], which uses different dynamics models for hover, transition, and level flight, with a separate controller for each, this last controller was tested in simulation only. Other types of adaptive based controllers were developed, such as in [10],

which uses an on board recursive least squares to estimate a linear function relating deflections to angular acceleration. A backstepping controller is then applied on this estimated dynamics model and was able to control the aircraft during outdoor environment flight tests. More recently, an adaptive method was used to estimate the aerodynamics from basis functions, then a cascaded architecture controller was able to track a path in an indoor environment [11]. Precomputed optimal body rates as a function of the attitude error were used for the attitude controller.

A more empirical method was used in [12], where the forces and moments model was implemented as a quadratic curve fit of the velocity, angle of attack, motor rpm and elevon deflection data, based on wind tunnel experiments of half the aircraft. The control structure was then separated into an attitude controller operating in $SO(3)$, that generates the desired moments and an actuator allocation that mixes actuators to achieve those moments and the desired longitudinal force. This method was able to track a path in an outdoor environment.

Another physics based method, using quaternions for attitude control is reported in [13]. First, the desired acceleration is computed from the position error, velocity error and aerodynamics model. Then, the desired attitude is chosen to point the thrusters towards the desired acceleration. Finally, the desired moments are computed as a PD law on the quaternion error and body rates. Outdoor experiments are reported, but nothing is reported about the model used for the aerodynamic coefficients nor the model used to relate the control surfaces deflections to the moments.

In this work, the controller implemented is derived from a single quaternion-based controller, developed for aerobatic flights, which does not assume any particular operating point. Because it accounts for the full range of rotation kinematics, it is able to perform well through large attitude changes [14]. Its cascaded architecture allows it to be applied to different platforms with minimal modifications. This

control strategy is chosen as it has been proven, through experimental flight tests, to be able to control a fixed wing aircraft during aerobatic maneuvers, including level flight, hover, transitions, aggressive turnaround, knife edge and rolling harrier. The transition from level flight to hover performed in prior work [14] is very similar to the transition needed for a tailsitter.

1.3.2 Real Time Simulator

Aerodynamic forces and moments are usually modeled scale with the inflow velocity squared. This allows the use of aerodynamic coefficients function of the angle of attack (AOA) only. The lift coefficient is often modeled linearly for low AOA and the drag coefficient is modeled as a quadratic function of the lift coefficient [15], this is the common model for conventional fixed wing aircraft that captures the lift-induced drag [16]. The lift coefficient is known to be a function of the aspect ratio of the aircraft. Commonly used empirical formulas are given for different aspect ratio ranges [17]. For flying wing aircraft, empirical lift coefficient formula also exists as a function of the sweep angle [18].

To cover the full flight envelope, aerodynamic coefficients are sometimes modeled as second order sine and cosine functions of the angle of attack, such as in [19]. However, the stalling effect, between low and high AOA is not represented in models. In [11], third order trigonometric functions are used to capture the stalling effects. These third order trigonometric functions behaves similarly to flat plate models [16]. The effects of stall can also be modeled as a sigmoid function to blend the linear range with the flat plate model for high AOA [20], as it is known that once stalled, the profile of an airfoil has negligible effect on its aerodynamic performance [21]. The effects of control surface deflections are often modeled linearly, such as in [19] or more generally using stability derivatives [15].

Another approach, consisting of a fully empirical model is proposed in [12]. The flying wing tailsitter is separated into left and right wings and aerodynamic forces

and moments are curve fitted with a quadratic polynomial as a function of the aircraft velocity, propeller angular speed, and control surfaces deflections. Although, this method can be very accurate for the given platform, it is purely based on experiments and therefore requires extensive wind tunnel tests.

One of the most relevant simulator is the one from Selig [22]. This model is designed for small scale aerobatic airplanes. It uses a component breakdown approach, where the wings are split into several segments having their own surface and producing a lift force, drag force, and pitching moment. The aerodynamic coefficients for those three quantities are precomputed for the full angle of attack and control surface deflection ranges, and are retrieved during simulation by linear interpolation.

In this research, the component breakdown approach is selected to model the wings aerodynamics. Flat-plate theory, based on a previous work from the Aerospace Mechatronics Laboratory [23] is used to compute the aerodynamic forces and moments in real-time for each segment of the wing based on the inflow velocity and the eventual control surface deflection.

1.4 Research Objectives

The objectives of this thesis are:

1. To develop a full flight envelope, real-time simulator for the chosen platform.
A simulator will be built around the thin flat plate aerodynamics model [23] and relevant components, including modeling of ground contact and structure aerodynamics, will be incorporated into it.
2. To develop a single controller for all flight modes. The quaternion-based controller developed for aerobatic fixed wing aircraft [14] will be modified for implementation on the tailsitter platform.
3. To demonstrate the autonomous ability of the controller. The controller will be tested in the simulated environment. It will then be implemented and

tested on the real platform. The objective is to show through experiments whether the single controller is suitable for the tailsitter platform, including the transition maneuvers.

4. To perform qualitative evaluation of the simulator, from experimental flight data. The data from the flight tests in simulation and from the real experiments will be compared and discrepancies will be investigated.

1.5 Thesis Organization

This thesis is divided into three parts. First, the real time simulator is documented in Chapter 2, where the equations of motions are presented, followed by the detailed dynamics model and measurements necessary for semi empirical models. Then, the control strategy is presented in Chapter 3, where the flight missions are divided into basic maneuvers and the control architecture is detailed. In Chapter 4, the flight tests in simulation and on the real platform are presented and a qualitative evaluation of the simulator performance is undertaken. Finally, Conclusions are presented in Chapter 5 along with recommendations for future work.

CHAPTER 2

Aircraft Simulation

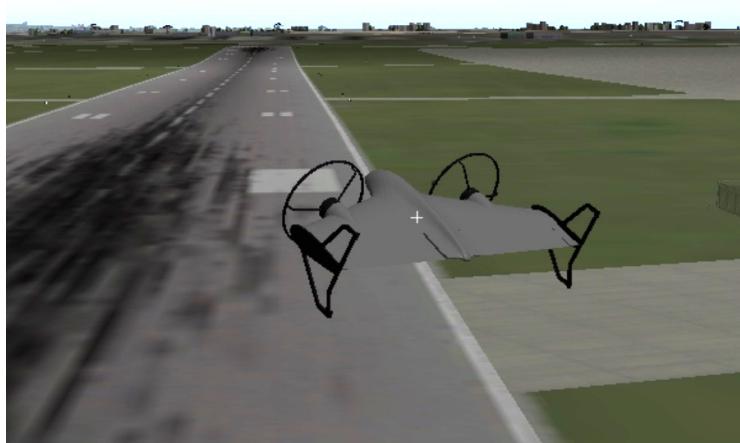


Figure 2–1: Screenshot of the Simulator, which uses X-Plane for visualization.

This chapter describes the modeling of the tailsitter aircraft and the associated real-time simulator. It is divided into five sections. The first section presents the frames and conventions used in the model development. The second section describes the equations of motion of a rigid aircraft. The third section develops the simulator dynamics, namely the model for the forces and moments acting on the aircraft during the simulation. The fourth section presents a closed form aerodynamics model used to perform measurements and used by the controller's actuator mixer described in Section 3.2.4. The last section reports the experimental measurements made to provide parameters for the model.

The simulator is implemented in Matlab/Simulink and X-Plane 9 is used for the visualization of the trajectory in real time, a screenshot is shown in Figure 2–1.

2.1 Frames and Conventions

The platform used is a bi-thruster flying wing tailsitter shown in Figure 2–2. For this system, three reference frames are defined:

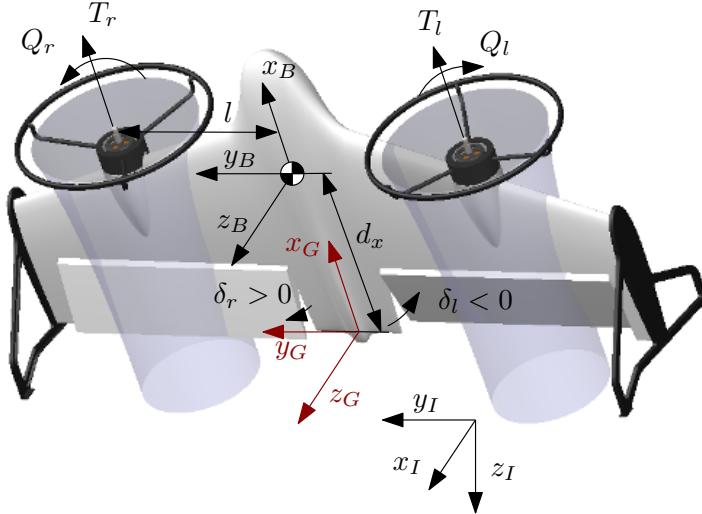


Figure 2–2: Tailsitter CAD Model showing the body frame B , geometric frame G , inertial frame I , the four actuators, and the slipstream acting on a portion of the control surfaces. The deflections are defined positive along y_B using the right hand rule.

- The Inertial frame I : $[x_I, y_I, z_I]^T$, which is fixed on the ground and is chosen as North-East-Down (NED). The ground is assumed to be locally flat in the aircraft’s workspace.
- The Geometric frame G : $[x_G, y_G, z_G]^T$, which is fixed on the trailing edge of the flying wing, laterally centered and inside the chord line. The geometric frame G is used for geometric references independent of the center of mass location.
- The body frame B : $[x_B, y_B, z_B]$, which is fixed on the aircraft and located at its center of mass CM . The center of mass is assumed to be laterally centered, inside the chord line and located at a distance d_x from the trailing edge.

This aircraft possesses four actuators, namely:

- The left thruster, which generates a force T_l , and a torque Q_l due to the drag of the propeller blades. In this context, a thruster is composed of a propeller, a Brushless Direct Current motor (BLDC) and an Electronic Speed Controller (ESC). A brushless motor is a electric motor which uses the magnetism to

produce a rotation, powered by three sine waves shifted 120° apart. The BLDC is a synchronous motor, which means the motor rotational speed is proportional to the sine waves frequency [24]. Generating the sine waves is the purpose of the ESC which controls their frequency according to the throttle signal τ . The hardware components are listed in Appendix F. The left propeller rotates clockwise looking at the vehicle from the rear. The torque opposes the rotation meaning Q_l is negative along x_B . The rotating propeller blows air reward, which is called the slipstream effect.

- The right thruster, which generates a force T_r and torque Q_r . Since it rotates counterclockwise, its torque is opposite that of the left thruster and is positive along x_B .
- The left elevon (elevon stands for elevator and aileron combined), which is a control surface located at the rear of the vehicle, behind the left thruster. Its location is intentionally in the slipstream to ensure it maintains effectiveness at low speeds.
- The right elevon, which is located behind the right thruster. By convention, a control surface deflection increases the camber, so both elevon deflections are defined positive along the y_B axis.

The actuators work in combination to produce longitudinal force and roll, pitch, and yaw moments. Longitudinal force is achieved by simultaneous thrusts while a yawing moment is produced by differential thrusts. The same applies for the elevons deflections; Common deflections create a pitching moment while differential deflections produce a rolling moment.

In hover (nose up), the tailsitter is able to maintain a fixed position, but it is also able to perform low speed displacements by tilting about the z_B axis, achieved by differential thrust, and pitching about the y_B , achieved by common deflections. Those gentle tilt and pitch angle will have the effect of orienting the longitudinal

thrust force towards the desired position. In this situation, the heading direction can be controlled by differential elevon deflections.

To avoid confusion in the terminology, this thesis will use the following:

- Euler angles. The set used in this thesis is yaw-pitch-roll $\mathbf{R}_1(\psi)\mathbf{R}_2(\theta)\mathbf{R}_3(\phi)$, as defined in [15], which correspond to a rotation by ψ about the z_B axis, followed by a rotation by θ about the new y_B axis and finally a rotation by ϕ about the new x_B axis.
- The rotations of the aircraft during hovering maneuvers will be described by the terms heading, pitching, and lateral tilting. The heading will be a rotation on the spot of the aircraft about a vertical axis, in that case x_B . The term pitching describes a rotation of the aircraft about its y_B axis and lateral tilting will be a rotation of the aircraft about the z_B axis.

2.2 Aircraft Equations of Motion

The equations of motion are a standard set of equations that describes the dynamics of the aircraft as a rigid body, and can be found in aircraft textbooks such as [20].

Due to the ability of this aircraft to perform wide attitude change, unit quaternions q are used for attitude representation. This convention is chosen over Euler angles because the latter have a singularity when the pitch angle approaches $\theta = \pm\pi/2$, known as gimbal lock. In that condition, the yaw and roll angle are not unique. Unit quaternions are free of kinematic singularities. The definitions and relations are given in Appendix E.

The equations of motion of a rigid aircraft with the use of quaternions [25] are given by:

$$\begin{aligned}\dot{\mathbf{p}} &= \mathbf{R}(\mathbf{q}) \mathbf{v}_B \\ \dot{\mathbf{v}}_B &= \frac{1}{m} \mathbf{F}_B - \boldsymbol{\omega}_B \times \mathbf{v}_B \\ \dot{\mathbf{q}} &= \frac{1}{2} \mathbf{q} \otimes \begin{bmatrix} 0 \\ \boldsymbol{\omega}_B \end{bmatrix} \\ \dot{\boldsymbol{\omega}}_B &= \mathbf{I}^{-1} (\mathbf{M}_B - \boldsymbol{\omega}_B \times \mathbf{I} \boldsymbol{\omega}_B)\end{aligned}\tag{2.1}$$

where $\mathbf{R}(\mathbf{q})$ is the rotation matrix from body frame B to inertial frame I , which is retrieved from the unit quaternion, \mathbf{p} is the inertial position, $\mathbf{v}_B = [u, v, w]^T$ is the velocity expressed in the body frame, m is the mass of the vehicle, $\boldsymbol{\omega}_B = [p, q, r]^T$ are the body angular rates in the body frame, \otimes is the quaternion product, \mathbf{I} is the inertia matrix of the vehicle about the body frame axes centered at the center of mass (CM), and \times is the 3D cross product. \mathbf{F}_B and \mathbf{M}_B are the net body forces and moments exerted at the CM, which is the origin of the body frame B . The expression for $\mathbf{R}(\mathbf{q})$ is given in Appendix G by Equation G.14.

2.3 Dynamics Modeling

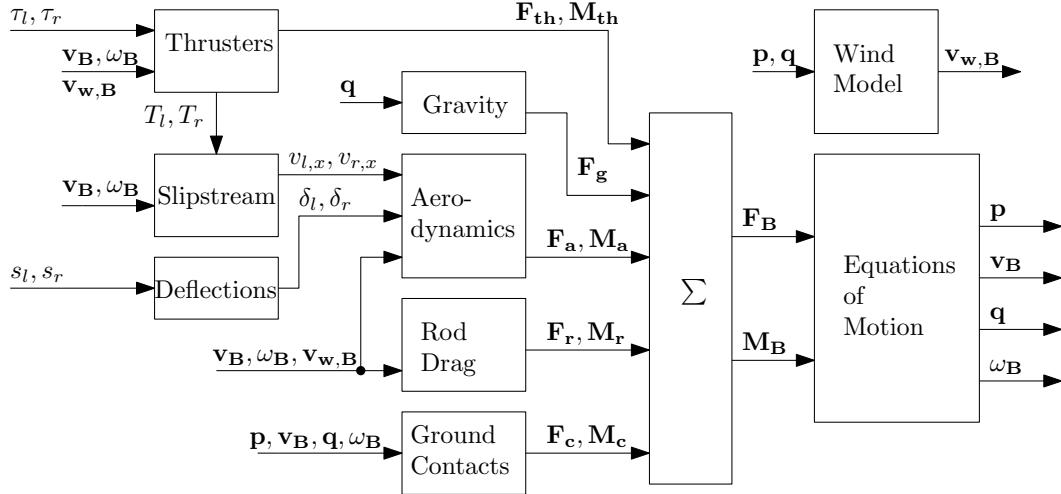


Figure 2–3: Simulator Block Diagram.

The real time simulator is shown graphically in Figure 2–3. A number of dynamics blocks calculate the forces and moments at the CM, and these forces and moments are then added and fed to the equations of motion (Equation 2.1), which are then integrated to generate the states \mathbf{p} , \mathbf{v}_B , \mathbf{q} , ω_B .

The body forces \mathbf{F}_B acting at the CM are the sum of the gravity force \mathbf{F}_g , thruster forces \mathbf{F}_{th} , aerodynamic forces \mathbf{F}_a , rod drag forces \mathbf{F}_r and ground contact forces \mathbf{F}_c :

$$\mathbf{F}_B = \mathbf{F}_g + \mathbf{F}_{th} + \mathbf{F}_a + \mathbf{F}_r + \mathbf{F}_c \quad (2.2)$$

Similarly, the body moments acting at the CM, $\mathbf{M}_B = [L, M, N]^T$ are the sum of the thruster and gyroscopic moments \mathbf{M}_{th} , aerodynamic moments \mathbf{M}_a , rod drag moments \mathbf{M}_r and ground contacts moments \mathbf{M}_c :

$$\mathbf{M}_B = \mathbf{M}_{th} + \mathbf{M}_a + \mathbf{M}_r + \mathbf{M}_c \quad (2.3)$$

The force due to the gravity is described in Section 2.3.2. It acts directly at the CM, and so no moment is created.

To calculate the aerodynamic forces generated by the wings, a component breakdown approach is implemented, which is reported in Section 2.3.3. The wings are partitioned into trapezoidal segments. Each surface moves at its own velocity, described in Section 2.3.1, and creates a lift force, drag force and pitching moment about its aerodynamic center. The forces and moments are then summed up and translated to the CM. This method allows the modeling of partial flow conditions over the wing surfaces, due to the slipstream or due to a high rotation rate of the aircraft. The aerodynamic forces and moments are obtained by considering the wing as flat plate [23]. This model predicts the full flight envelope aerodynamics, stalling effects, the effects of low aspect ratio and sweep, and large control surfaces with large deflections.

A similar component breakdown method is used to model the drag force due to the landing gears and other appendages, which are considered as rod segments. This is described in Section 2.3.4.

The thruster forces and torques are evaluated by considering the throttle signals and inflow velocity. The gyroscopic moments are also considered, these are moments generated from conservation of angular momentum of the propeller when the aircraft rotates. The thruster model is described in Section 2.3.5.

Finally, to simulate the takeoff and landing phases, a ground contact model is developed. This model is presented in Section 2.3.7, where specific exterior points on the aircraft are considered. When those points make contact with the ground, a reaction force proportional to the penetrated distance is created as well as a friction force.

2.3.1 Velocity and Angle of Attack

The velocity at different points on the aircraft needs to be evaluated for the aerodynamics model and thrusters model. Working in the body frame B , the velocity relative to the air, at any point i , located at a position \mathbf{r}_i on the aircraft can be found from:

$$\mathbf{v}_i = \mathbf{v}_B + \boldsymbol{\omega}_B \times \mathbf{r}_i - \mathbf{v}_{w,B} \quad (2.4)$$

where $\mathbf{v}_{w,B}$ is the wind velocity in the body frame generated from a Dryden wind model, as discussed in Appendix B. This equation holds for any point outside of the slipstream.

The velocity can be decomposed into components as $\mathbf{v}_i = [u_i, v_i, w_i]^T$. The angle of attack of the horizontal segments α_i , which is the incidence angle formed between the inflow velocity and the chord line of the airfoil, can be found from:

$$\alpha_i = \text{atan2}(w_i, u_i) \quad (2.5)$$

The angle of attack of the vertical segments (i.e. the winglets), also called sideslip angle β_i can be computed as:

$$\beta_i = \text{atan}2(v_i, u_i) \quad (2.6)$$

When a point is inside the slipstream, the longitudinal component u_i is modified since the thrusters blow along the longitudinal axis. The velocity u_i is modified to $v_{l,x}$ and $v_{r,x}$, defined below, for the segments behind the left and right thrusters, respectively. Momentum theory can be used to predict the velocity far downstream the propeller disc [17]. As reported from experiments on a small scale aircraft propeller [26], momentum theory gives reasonable predictions for distances between 1 and 3 propeller radii downstream of the propeller disc. This distance range corresponds to the location of the control surface behind the propeller disc. Based on this, the left and right propeller will create a longitudinal airflow, whose velocity given by:

$$v_{l,x} = \sqrt{v_{in,l}^2 + \frac{2T_l}{\rho\pi r_p^2}}, \quad v_{r,x} = \sqrt{v_{in,r}^2 + \frac{2T_r}{\rho\pi r_p^2}} \quad (2.7)$$

where r_p is the propeller radius and $v_{in,l}$, $v_{in,r}$ are the left and right thruster's inflow velocity. They can be computed as the first component of the velocity taken at the propeller location using Equation 2.4. According to the same theory, when the airplane is static (i.e. no inflow velocity to the thrusters), the diameter of the slipstream is given by $d_s = \sqrt{2}r_p$. This quantity is chosen for the width of the wing segments exposed to the slipstream velocity.

The velocity at the propeller disc v_d according to the same theory is given by:

$$v_{d,l,x} = (v_{l,x} + v_{in,l})/2, \quad v_{d,r,x} = (v_{r,x} + v_{in,r})/2 \quad (2.8)$$

these quantities are used to model the velocity on the propeller guards.

2.3.2 Gravity

Gravity creates a force \mathbf{F}_g acting at the center of mass (CM):

$$\mathbf{F}_g = \mathbf{R}^T(\mathbf{q}) \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \quad (2.9)$$

where g is the usual gravitational acceleration in $[m/s^2]$. No moment is created, since this force acts directly at the CM.

The center of mass location is located at a distance $d_x = 0.13 [m]$ ahead of the trailing edge and is assumed to be in the same horizontal plane as the thrusters' axis of revolution and the chord line of the wings.

The mass m was measured on a scale, while the inertia matrix \mathbf{I} and center of mass location were evaluated using a high fidelity CAD model, described in Appendix A.

2.3.3 Wing Aerodynamics

The aerodynamic forces and moments of the wings are modeled through component breakdown. Each segment has its own velocity, evaluated using equation 2.4, on its surface S_i and creates its own lift force, drag force, and pitching moment about its aerodynamic center. The forces and moments of each segment are then summed to form the aerodynamic force \mathbf{F}_a and aerodynamic moment \mathbf{M}_a . Although the airfoils of the tailsitter are not flat, a model developed for flat plate profile [23] is chosen as data is available to evaluate the aerodynamic forces and moments for the full flight envelope as well as for large control surfaces with large deflections. For this tailsitter, the flying wing is broken into nine horizontal segments, which constitute the minimal number of segments to capture the slipstreams and the portions with control surfaces. One vertical segment is added at each wingtip, refer to Figure 2–4.

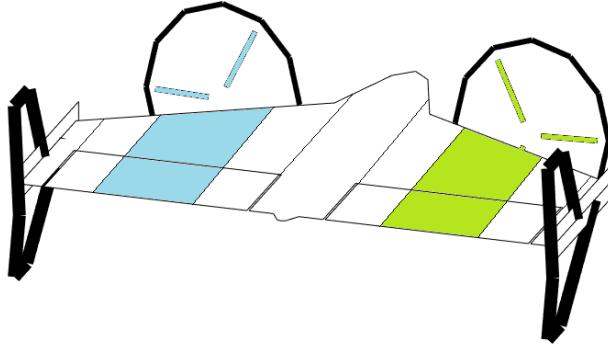


Figure 2–4: Component Breakdown. The blue segments are exposed to the left slipstream and the green segments are exposed to the right slipstream. The thick black components represent the structural components, modeled as rod segments creating drag forces.

Each horizontal segment i , having a mean aerodynamic chord c_i and span b_i , produces a force \mathbf{F}_i at its aerodynamic center \mathbf{r}_i as:

$$\mathbf{F}_i = \frac{1}{2} \rho b_i c_i (u_i^2 + w_i^2) \begin{bmatrix} C_L \sin \alpha_i - C_D \cos \alpha_i \\ 0 \\ -C_L \cos \alpha_i - C_D \sin \alpha_i \end{bmatrix} \quad (2.10)$$

where C_L is the lift coefficient and C_D the drag coefficient, which are described later. Each horizontal segment, knowing their pitching moment at the aerodynamic center $C_{M,ac}$, also produces a moment $\mathbf{M}_i = [0, M_i, 0]^T$:

$$M_i = \frac{1}{2} \rho b_i c_i^2 (u_i^2 + w_i^2) C_{M,ac} \quad (2.11)$$

Similarly, each vertical segment i , produces a force \mathbf{F}_i at its aerodynamic center \mathbf{r}_i as:

$$\mathbf{F}_i = \frac{1}{2} \rho b_i c_i (u_i^2 + v_i^2) \begin{bmatrix} C_L \sin \beta_i - C_D \cos \beta_i \\ -C_L \cos \beta_i - C_D \sin \beta_i \\ 0 \end{bmatrix} \quad (2.12)$$

area S_i [mm ²]	span b_i [mm]	MAC c_i [mm]	flap chord $c_{f,i}$ [mm]	position $r_{i,G}$ [mm]
2 408	22.5	110.0	0	[82.5, -239.3, 0]
4 785	38.3	125.0	63	[93.75, -208, 0]
13 051	88.4	148.2	63	[111.15, -143.4, 0]
11 100	63.3	175.9	63	[131.9, -68.4, 0]
17 042	75.0	231.4	0	[173, 0, 0]
11 100	63.3	175.9	63	[131.9, 68.4, 0]
13 051	88.4	148.2	63	[111.15, 143.4, 0]
4 785	38.3	125.0	63	[93.75, 208, 0]
2 408	22.5	110.0	0	[82.5, 239.3, 0]

Table 2–1: Horizontal Wing Segments, from left to right.

where β_i is the sideslip angle of the segment i defined in Section 2.3.1. The vertical segments moments are defined as $\mathbf{M}_i = [0, 0, N_i]^T$, where:

$$N_i = -\frac{1}{2}\rho b_i c_i^2 (u_i^2 + v_i^2) C_{M,ac} \quad (2.13)$$

The segment's aerodynamic forces and moments are then translated to the center of mass of the aircraft as:

$$\mathbf{F}_a = \sum \mathbf{F}_i, \quad \mathbf{M}_a = \sum (\mathbf{M}_i + \mathbf{r}_i \times \mathbf{F}_i) \quad (2.14)$$

Table 2–1 reports the geometric characteristics of the horizontal segments of the flying wing. The parameters shown are the segments area S_i , span b_i , MAC c_i , flap chord $c_{f,i}$ and aerodynamic center $\mathbf{r}_{i,G}$ given in the geometric frame G . From this table, the aerodynamic center of the wing can be computed as:

$$AC_G = \frac{\sum S_i \mathbf{r}_{i,G}}{\sum S_i} = \begin{bmatrix} 126.4 \\ 0 \\ 0 \end{bmatrix} [mm] \quad (2.15)$$

Similarly Table 2–2 reports the characteristics of the two vertical segments, i.e. the winglets.

area	span	MAC	flap chord	position
$S_i \text{ [mm}^2]$	$b_i \text{ [mm]}$	$c_i \text{ [mm]}$	$c_{f,i} \text{ [mm]}$	$\mathbf{r}_{i,G} \text{ [mm]}$
3 120	26	120	0	[85, -252, 0]
3 120	26	120	0	[85, 252, 0]

Table 2–2: Vertical Wing Segments, from left to right.

In the linear range, the following lift curve slope is used for the flying wing segments, as described in [18]:

$$C_{L\alpha} = \frac{2\pi \cos \Lambda}{\frac{2\cos \Lambda}{A_R} + \sqrt{1 + \left(\frac{2\cos \Lambda}{A_R}\right)^2}} \quad (2.16)$$

where Λ is the sweep angle taken as 19.8° , and the aspect ratio A_R is taken as 3.13 for the horizontal segments. Also in the linear range, the following equations are used to describe the lift and drag coefficients:

$$C_L = C_{L\alpha} \alpha, \quad C_D = C_{D,0} + \frac{C_L^2}{\pi k_0 A_R} \quad (2.17)$$

where $C_{D,0}$ is the drag coefficient due to skin friction, and k_0 is Oswald's efficiency factor.

As the angle of attack increases, the airflow separates from the upper surface in a process known as "stall", which occurs progressively. To account for this, the model from [23] includes a term for progressive separation of the flow from the upper surface of the airfoil, starting from the trailing-edge, as well as turbulent leading edge vortices, which both cause a loss of lift. The angle of attack at complete stall α_{high} , which corresponds to a complete flow separation, is retrieved from an empirical database listing it for different flat plate aspect ratios.

The control surface deflection has the effect of increasing the camber of the wing segment and reducing the effective chord length. The consequences are a shift in the lift, drag, and moment curves and a modification of the stalling angle of attack α_{high} . For a complete set of equations, including stalling model, control surface

deflection effects, vertical segments, and moment coefficients, the reader can refer to [21].

The full flight envelope lift and drag forces coefficients of the entire aircraft are shown in Fig. 2–5 and the pitching moment coefficient of the entire aircraft is shown in Fig. 2–6.

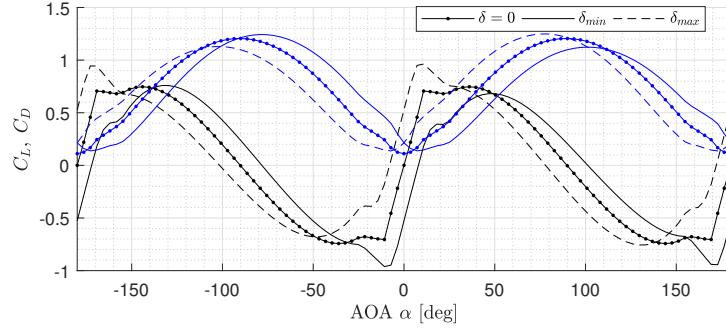


Figure 2–5: Aerodynamic Forces Coefficients of the Entire Aircraft. The lift coefficient C_L is shown in black and the drag coefficient C_D is shown in blue. Both forces are represented with both elevons at minimal, zero and maximal deflection angle denoted respectively by plain, plain with dots, and dashed lines.

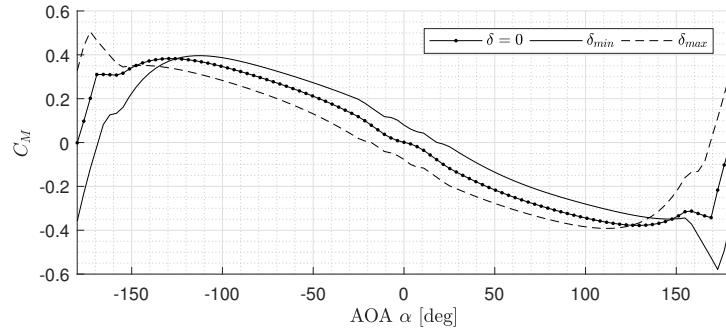


Figure 2–6: Pitching Moment Coefficient of the Entire Aircraft. It is represented with both elevons at minimal, zero and maximal deflection angle denoted respectively by plain, plain with dots, and dashed lines.

2.3.4 Structural Segments Aerodynamics

The structural parts such as the propeller protectors and the landing gear are modeled as rod segments creating a drag force. Consider a rod segment as shown in Figure 2–7. Each rod segment j , of length l_j and diameter d_j , creates a drag force

N_j normal to its length. The drag model proposed [27], is given as:

$$N_j = \frac{1}{2} \rho v_j^2 l_j d_j C_{D,rods} \sin^2 \theta_j \quad (2.18)$$

where the drag coefficient of the rods $C_{D,rods}$ is taken as 1.1. The rod segment velocity v_j can be computed using Eq. 2.4 by considering the center of the rod segment r_j as the location of the computed velocity.

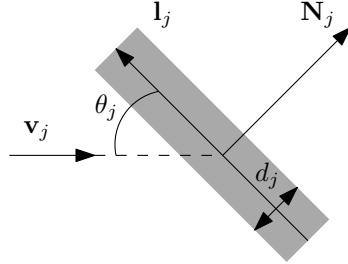


Figure 2–7: Rod segment j in a wind field \mathbf{v}_j , with incidence angle θ_j , creating a normal force \mathbf{N}_j .

The force can then be evaluated as the product of the magnitude in equation 2.18 and a unit normal vector \mathbf{u}_j , to ensure it in the same direction as the wind:

$$\mathbf{u}_j = \begin{cases} \frac{(\mathbf{v}_j \times \mathbf{l}_j) \times \mathbf{l}_j}{|\mathbf{v}_j| |\mathbf{l}_j|^2}, & \text{if } \mathbf{v}_j \cdot ((\mathbf{v}_j \times \mathbf{l}_j) \times \mathbf{l}_j) \geq 0 \\ -\frac{(\mathbf{v}_j \times \mathbf{l}_j) \times \mathbf{l}_j}{|\mathbf{v}_j| |\mathbf{l}_j|^2}, & \text{otherwise} \end{cases} \quad (2.19)$$

The sin of the incidence angle can be expressed as:

$$\sin \theta_j = \frac{|\mathbf{v}_j \times \mathbf{l}_j|}{|\mathbf{v}_j| |\mathbf{l}_j|} \quad (2.20)$$

A special case is accounted for during implementation to avoid a division by zero when the inflow velocity is zero. In this case, the drag force is set to zero.

The rod segment forces are then translated to the center of mass using:

$$\mathbf{F}_r = \sum N_j \mathbf{u}_j, \quad \mathbf{M}_r = \sum \mathbf{r}_j \times N_j \mathbf{u}_j \quad (2.21)$$

start [mm]			end [mm]			\emptyset [mm]
$r_{1,G,x}$	$r_{1,G,y}^*$	$r_{1,G,z}$	$r_{2,G,x}$	$r_{2,G,y}$	$r_{2,G,z}$	d_j
46.5	252.0	0.0	19.0	252.0	-66.0	7
19.0	252.0	-66.0	-10.0	252.0	-66.0	7
-10.0	252.0	-66.0	0.0	252.0	25.0	7
0.0	252.0	25.0	-10.0	252.0	69.0	7
-10.0	252.0	69.0	14.5	252.0	69.0	7
14.5	252.0	69.0	81.0	252.0	0.0	7
177.0	207.8	36.2	177.0	182.3	62.8	3
177.0	181.3	62.8	177.0	146.0	72.5	3
177.0	145.0	72.5	177.0	109.8	62.8	3
177.0	108.8	62.8	177.0	83.2	36.2	3
177.0	82.2	36.2	177.0	73.5	0.0	3
177.0	72.5	0.0	177.0	83.2	-36.2	3
177.0	82.2	-36.2	177.0	109.7	-62.8	3
177.0	108.7	-62.8	177.0	146.0	-72.5	3
177.0	145.0	-72.5	177.0	182.3	-62.8	3
177.0	181.3	-62.8	177.0	208.8	-36.3	3
177.0	207.8	-36.3	177.0	218.5	-0.0	3
177.0	217.5	-0.0	177.0	208.8	36.3	3
172.0	140.5	9.5	175.0	114.5	54.6	3
172.0	140.5	-9.5	175.0	114.5	-54.6	3
172.0	157.0	-0.0	175.0	209.0	-0.0	3

Table 2–3: Structural Rods, right side. The left side can be obtained by reversing the sign on $r_{1,G,y}$ and $r_{2,G,y}$.

Each propeller protector is modeled as 12 rod segments composing the circle, and 3 radial rods, all of them having a diameter of 3 [mm]. Each landing gear is composed of 6 rod segments, of diameter 7 [mm], refer to Figure 2–4. According to this model, the rod drag force represents 40% of the total aerodynamic drag force when the aircraft is in level flight at 8 [m/s].

Table 2–3 provides the characteristics of the structural rod segments in the geometric frame G . The segments are located as a start point $r_{1,G}$, an end point $r_{2,G}$, and a diameter \emptyset . The length vector can be found as the difference between the end point and the start point.

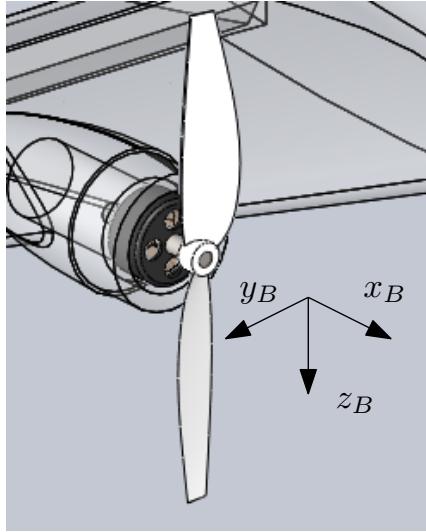


Figure 2–8: Right Thruster, the rotating parts (rotor and propeller) shown in opaque, spin about the x_B axis.

2.3.5 Thrusters and Gyroscopic Moments

The left and right thrusters create a thrust force T_l and T_r pointing forward along the x_B axis. Therefore the thruster force \mathbf{F}_{th} is purely longitudinal:

$$\mathbf{F}_{th} = \begin{bmatrix} T_l + T_r \\ 0 \\ 0 \end{bmatrix} \quad (2.22)$$

The right and left thrusters are located at a lateral distance l and $-l$, from the CM, respectively. The thrusters are located in the same horizontal plane as the CM and their location along the longitudinal axis does not influence the generated moments because the thrusters produce a force along the x_B axis only. The thrust forces generate a yawing moment about the z_B axis. Furthermore, each propeller generates a torque Q . On this platform, the left propeller rotates clockwise and the right propeller counterclockwise, which results in torques in opposite directions.

The moments generated by the thrusters are given by \mathbf{M}_T :

$$\mathbf{M}_T = \begin{bmatrix} Q_r - Q_l \\ 0 \\ l(T_l - T_r) \end{bmatrix} \quad (2.23)$$

The gyroscopic moments \mathbf{M}_{gyro} are then summed to \mathbf{M}_T to produce \mathbf{M}_{th} . The reactive gyroscopic moments are due to the rate of change of angular momentum of the rotating thrusters by the body rates and are given by:

$$\mathbf{M}_{gyro} = -\boldsymbol{\omega}_B \times \mathbf{I}_{th} \boldsymbol{\omega}_p \quad (2.24)$$

Where $\boldsymbol{\omega}_B = [p, q, r]^T$ are the body rates of the airplane, \mathbf{I}_{th} is the inertia matrix of the spinning parts of the thruster, taken at a point on the revolution axis and $\boldsymbol{\omega}_p$ is the angular speed of the thruster. Because the thruster is spinning at high rotational speed ω_p about the x_B axis, it is chosen to consider the propeller as an equivalent disc with fixed inertia. The symmetries of a disc result in a diagonal inertia matrix which simplifies equation 2.24 to:

$$\mathbf{M}_{gyro} = -\boldsymbol{\omega}_B \times I_{th} \omega_p \mathbf{x}_B \quad (2.25)$$

where I_{th} is the scalar inertia about x_B axis, of the spinning parts of the thrusters.

Expanding the cross product yields:

$$\mathbf{M}_{gyro} = I_{th} \omega_p \begin{bmatrix} 0 \\ -r \\ q \end{bmatrix} \quad (2.26)$$

This equation is linear with respect to the thruster angular speed. Assuming same inertia for right and left thrusters, ω_p can be replaced by $\omega_l - \omega_r$:

$$\mathbf{M}_{gyro} = I_{th} (\omega_l - \omega_r) \begin{bmatrix} 0 \\ -r \\ q \end{bmatrix} \quad (2.27)$$

where ω_l and ω_r are the left and right thruster angular speed and are both positive.

I_{th} was estimated as $1.626 \cdot 10^{-6} [kg\ m^2]$ from the CAD model.

Finally, the total thruster moments \mathbf{M}_{th} are obtained as:

$$\mathbf{M}_{th} = \begin{bmatrix} Q_r - Q_l \\ 0 \\ I(T_l - T_r) \end{bmatrix} + I_{th} (\omega_l - \omega_r) \begin{bmatrix} 0 \\ -r \\ q \end{bmatrix} \quad (2.28)$$

2.3.6 Thruster Force and Torque

The thruster force and torque need to be generated from throttle signals and the effect of the inflow velocity on the thrust force also need to be considered. Blade element momentum theory [28] can be used to model the propeller aerodynamics. In this context, the propeller needs to be decomposed into segments, which are then considered as airfoils that generate lift and drag as functions of their own velocity. The thrust and torque are found by integrating along the propeller radius. However, this would require a high fidelity model of the propeller shape as well as lift and drag coefficients function of the blade segment. For these reasons, we chose instead to rely on the model described below, based on experimental data.

The thruster dynamic model is proposed as a two block representation shown in Fig. 2–9. The first block is the Electronic Speed Controller (ESC) and brushless motor block, which generates the rotation ω_p of the propeller from the throttle signal τ and the battery voltage V_{batt} . The second block represents the propeller, which, given its rotational speed ω_p and inflow velocity v_{in} generates a thrust T and a torque

Q . Each thruster is composed of an ESC BeeRotor BS20A, an electric motor BL280 2600KV and a propeller with 125mm diameter and 75mm pitch value. Refer to Appendix Table F.1. The same model describes each thruster, with its own inflow velocity, angular speed and throttle signal, though the battery voltage is common to both thrusters, since they are connected in parallel to the same battery.

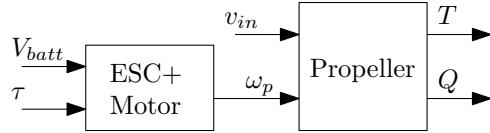


Figure 2–9: Two Blocks Diagram of the Thruster Model.

The propeller angular speed ω_p is modeled as an empirical function of V_{batt} and τ , and will be discussed in Section 2.5.1:

$$\omega_p = \omega_p(V_{batt}, \tau) \quad (2.29)$$

Knowing the propeller angular speed ω_p and the inflow velocity v_{in} , allows a calculation of the propeller thrust and torque as follows. First, the advance ratio J is computed as the ratio of the inflow velocity over the propeller tip velocity. The definition from UIUC [29] is used:

$$J = \frac{\pi v_{in}}{\omega_p r_p} [1/rev] \quad (2.30)$$

where r_p is the radius of the propeller.

Then, the thrust coefficient C_T and power coefficient C_P are modeled as functions of the advance ratio J . The model for these two coefficients will be described in Section 2.5.1.

$$C_T = C_T(J), \quad C_P = C_P(J) \quad (2.31)$$

Finally, the thrust force T and torque Q are computed as:

$$T = \frac{4}{\pi^2} \rho \omega_p^2 r_p^4 C_T, \quad Q = \frac{4}{\pi^3} \rho \omega_p^2 r_p^5 C_P \quad (2.32)$$

The model is the same for the left and right thruster. The torque Q is computed as a magnitude and equation 2.28 accounts for the direction.

2.3.7 Ground Contact Dynamics

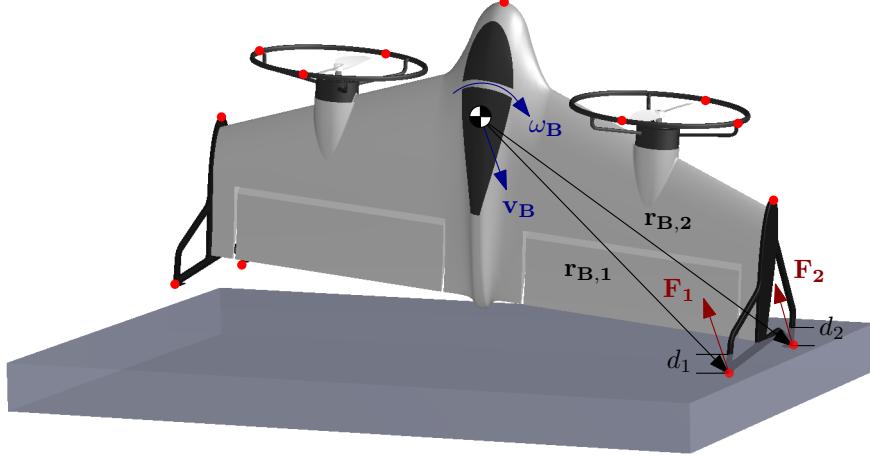


Figure 2–10: Ground Contact Forces. The selected points are shown in red.

Since this study focuses on the full flight mission, including vertical takeoff and landing, there is a need to model ground contact dynamics. The contact force is modeled using a spring-damper analogy. First, to detect the contact with the ground, thirteen specific exterior points are chosen as shown in Figure 2–10.

Then, working in the inertial frame, each point k penetrating the ground at a depth d_k , and having a velocity $v_{I,k}$, produces a force $F_{I,k}$ at its respective location as:

$$\mathbf{F}_{I,k} = \begin{bmatrix} 0 \\ 0 \\ -m k_p d_k \end{bmatrix} - m k_v v_{I,k} \quad (2.33)$$

The inertial velocity $v_{I,k}$ is computed from the point's location in the body frame \mathbf{r}_k as:

$$v_{I,k} = \mathbf{R}(q)(v_B + \omega_B \times \mathbf{r}_k) \quad (2.34)$$

The locations of the points in the geometric frame G are given in Table 2–4.

	corners	[mm]
177.0	145.0	72.5
177.0	217.5	0.0
177.0	145.0	-72.5
177.0	-145.0	72.5
177.0	-217.5	0.0
177.0	-145.0	-72.5
249.4	0.0	0.0
118.0	251.0	0.0
118.0	-251.0	0.0
-15.0	251.0	71.0
-15.0	251.0	-71.0
-15.0	-251.0	71.0
-15.0	-251.0	-71.0

Table 2–4: Reference Points for Contact Force Calculations, given in the geometric reference frame G .

Then, to avoid unrealistic attraction to the ground, an upper bound of zero is imposed on the z_I component of the inertial force $\mathbf{F}_{I,k}$. These inertial forces are finally translated into the body frame to generate the contact force \mathbf{F}_c and the contact moment \mathbf{M}_c :

$$\mathbf{F}_c = \sum \mathbf{R}(\mathbf{q})^T \mathbf{F}_{I,k}, \quad \mathbf{M}_c = \sum \mathbf{r}_k \times \mathbf{R}(\mathbf{q})^T \mathbf{F}_{I,k} \quad (2.35)$$

Satisfactory results are obtained for $k_p = 100$ [1/ s^2] and $k_v = 5$ [1/ s], i.e., the airplane landing phase was found to appear realistic during real time visualization and the airplane reaches a complete stop quickly, in about half a second.

2.4 Simplified Analytical Aerodynamic Model

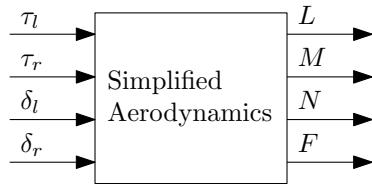


Figure 2–11: Diagram of the Simplified Aerodynamics Model.

Distinct from the high fidelity simulator, a simplified analytic model describing the aircraft aerodynamics is needed for the controller developed in this research. As

shown in Figure 2–11, this function needs to model the relation for the left and right throttle signals τ_l, τ_r and left and right elevon deflections δ_l, δ_r to the body moments $\mathbf{M}_B = [L, M, N]^T$ and the forward force F .

To distinguish variables evaluated in this model from those in the more exact model, we use an overhat notation to denote estimates. For example, \hat{M} denotes the pitching moment in the simplified model.

The forward force is estimated as the sum of the left and right thrust forces:

$$\hat{F} = T_l + T_r \quad (2.36)$$

where T_l and T_r are evaluated from τ_l and τ_r using the thruster model described in Figure 2–9.

The yawing moment is estimated by:

$$\hat{N} = l(T_l - T_r) \quad (2.37)$$

where l is the lateral position of the right thruster in the body frame.

The rolling moment \hat{L} is assumed to be the sum of the rolling moment generated by the elevons in the slipstream, the rolling moment generated by the elevons out of the slipstream, and the propeller torques:

$$\hat{L} = \underbrace{\frac{1}{2}\rho c_x(\delta_l v_l^2 - \delta_r v_r^2)}_{\text{in slipstream}} + \underbrace{\frac{1}{2}\rho(u^2 + w^2)b_x(\delta_l - \delta_r)}_{\text{outside slipstream}} + \underbrace{(Q_r - Q_l)}_{\text{prop. torques}} \quad (2.38)$$

where c_x is the rolling moment deflection coefficient accounting for the elevons in the slipstream, b_x is the rolling moment deflection coefficient accounting for the elevons outside the slipstream, Q_l and Q_r are the left and right propeller torques evaluated from the thruster model described in Figure 2–9, ρ is the air density, u and w are the aircraft velocity along x_B and z_B axes and v_l, v_r are the left and right

slipstream velocities modeled using momentum theory:

$$v_l^2 = u^2 + w^2 + \frac{2T_l}{\rho\pi r_p^2}, \quad v_r^2 = u^2 + w^2 + \frac{2T_r}{\rho\pi r_p^2} \quad (2.39)$$

where r_p is the radius of the propeller.

Similarly, the pitching moment is assumed to be the sum of the pitching moment generated by the elevons in the slipstream, the pitching moment generated by the elevons outside the slipstream and the pitching moment at zero deflection.

$$\hat{M} = \underbrace{-\frac{1}{2}\rho c_y(\delta_l v_l^2 + \delta_r v_r^2)}_{in \ slipstream} - \underbrace{\frac{1}{2}\rho(u^2 + w^2)b_y(\delta_l + \delta_r)}_{outside \ slipstream} + \underbrace{\hat{M}_0}_{no \ def.} \quad (2.40)$$

where c_y is the pitching moment deflection coefficient of the elevons in the slipstream, b_y is the pitching moment deflection coefficient of the elevons outside the slipstream. \hat{M}_0 is the estimated pitching moment of the entire aircraft at zero deflection, which is a function of the angle of attack and the aircraft velocity. \hat{M}_0 is described in Appendix D.

The moment deflection coefficients c_x, c_y, b_x and b_y will be evaluated in Section 2.5.3.

For compaction purpose, the XZ dynamic pressure $P_d = \frac{1}{2}\rho(u^2 + w^2)$ as well as the left and right slipstream velocities defined in equation 2.39 can be plugged into equations 2.38 and 2.40. We finally obtain the following set of equations describing the simplified analytical aerodynamic model:

$$\begin{aligned} \hat{L} &= \frac{c_x}{\pi r_p^2}(\delta_l T_l - \delta_r T_r) + P_d b_x(\delta_l - \delta_r) + (Q_r - Q_l) \\ \hat{M} &= -\frac{c_y}{\pi r_p^2}(\delta_l T_l + \delta_r T_r) - P_d(c_y + b_y)(\delta_l + \delta_r) + \hat{M}_0 \\ \hat{N} &= l(T_l - T_r) \\ \hat{F} &= T_l + T_r \end{aligned} \quad (2.41)$$

This simplified model will be used in Section 3.2.4 where the controller must determine the control deflections to use to obtain desired control moments and force.

2.5 Measurements

This section describes the measurements carried out to complement the simulator model. The first experiment measured the thruster coefficients from the thrusters mounted on a force torque sensor. The second experiment measured the elevon deflection angles as a function of the PWM command signals. The third experiment aimed to measure the effectiveness of the control surfaces in the slipstream, by mounting the entire aircraft on a force torque sensor.

2.5.1 Thruster Coefficients

This section reports the experiments carried out to define the thruster model, given in Section 2.3.6. First, a model is proposed for the ESC and motor to estimate the angular speed of the propeller from the battery voltage and the throttle signal.

To measure the ESC and motor properties, the whole thruster with propeller was fixed on a static surface. First, the ESC was calibrated to use Pulse Width Modulation signals (PWM) [30], with a high time ranging from 1000 [μs] to 2000 [μs]. In this context, the throttle τ ranging from 0 to 1 is related to the PWM high time PWM_h in [μs] as:

$$\tau = \frac{PWM_h - 1000}{1000} \iff PWM_h = 1000 + 1000\tau \quad (2.42)$$

A DC power source HY3005F-3 was used to provide voltage to the ESC. The throttle signals were converted to PWM signals sent to the ESC. The angular speed of the propeller was retrieved from an oscilloscope Hantek DSO5102B measuring the differential frequency between two phases of the brushless motor. This measure was validated by an optical RPM sensor Hangar 9 HAN156, which is less accurate but provides a cross-check of the oscilloscope measure.

The propeller angular speed was measured for different voltages and throttle signals, providing the data points shown in Figure 2–12. To better understand the

effect of the battery voltage, the data points at each voltage were scaled as functions of that voltage. It was found that, when they were scaled by $V_{batt}^{0.8}$ the data at different voltages collapsed approximately onto a single line. That line was then curve-fitted using a second-order polynomial, leading to the following model:

$$\omega_p = V_{batt}^{0.8} \left(\underbrace{-84.75}_{\omega_{p,2}} \tau^2 + \underbrace{356.34}_{\omega_{p,1}} \tau - \underbrace{4.27}_{\omega_{p,0}} \right) \quad (2.43)$$

This model with the recorded data are shown in Figure 2–12. Equation 2.43 was used to fill the role of equation 2.29 described earlier.

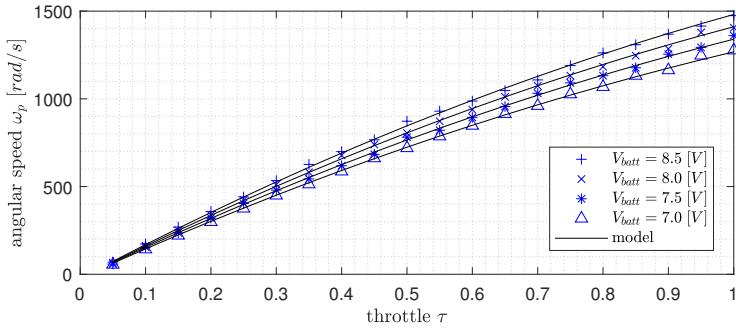


Figure 2–12: Angular speed ω_p as a function of throttle τ at different battery voltages V_{batt} .

To evaluate the thrust coefficient C_T and power coefficient C_P , a wind tunnel experiment is needed to estimate them as function of the inflow velocity v_{in} , or equivalently as a function of the advance ratio J . Rather than conduct our own wind tunnel experiments, we started by performing experiments at zero inflow velocity ($J = 0$) and relied on external wind tunnel measurements to complete the model for nonzero J .

The motor was mounted upward on a force torque sensor ATI mini40 calibrated SI-20-1, able to measure forces up to 20 [N] and torques up to 1 [Nm], refer to Figure 2–13. In this setup, the drag of the mount was assumed negligible since the motor diameter was greater than the mount rod diameter. The thrust T and torque

Q can be measured directly from the sensor as:

$$\begin{aligned} T &= F_z \\ Q &= M_z \end{aligned} \tag{2.44}$$

The table on which the apparatus was mounted might have an effect on the measured thrust. Cheeseman and Bennett [31] proposed an equation to estimate this effect, based on conservation of power. They predicted the increase of the thrust due to ground effect as the ratio:

$$\left[\frac{T_{\text{ground}}}{T} \right]_{P=\text{const}} = \frac{1}{1 - \left(\frac{r_p}{4z} \right)^2} \tag{2.45}$$

In this experiment, $r_p = 62.5 \text{ [mm]}$ is the propeller radius and $z = 258.5 \text{ [mm]}$ is the propeller's location above the ground. Equation 2.45 predicts an increase of thrust force of 0.37 %, and so this effect was assumed to be negligible.

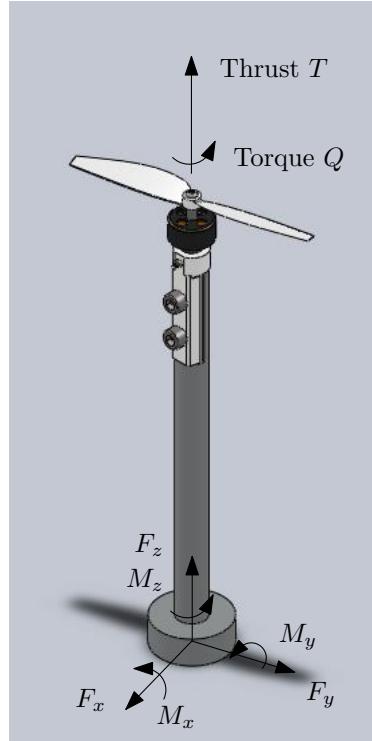


Figure 2–13: Propeller Force and Torque Testbed.

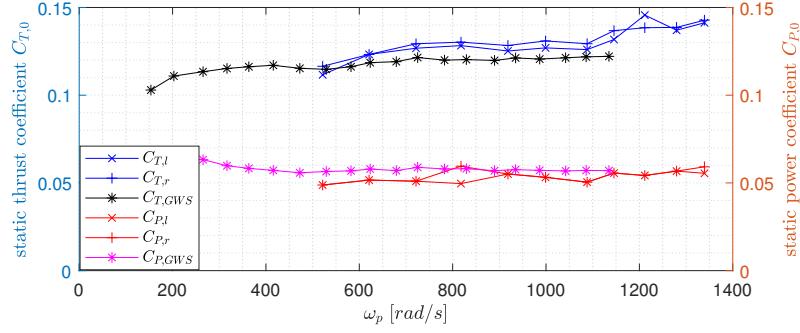


Figure 2–14: Static Propeller Coefficients Function of Angular Speed. $C_{T,l}$, $C_{T,r}$ and $C_{T,GWS}$ are the static thrust coefficients of the left and the right propeller in our experiment, and the GWS propeller from the database. Similarly, $C_{P,l}$, $C_{P,r}$ and $C_{P,GWS}$ are the static power coefficients.

The propeller thrust T and torque Q were recorded at different propeller angular speeds ω_p . Then, the static thrust and power coefficients were obtained using:

$$C_{T,0} = \frac{T \pi^2}{4 \rho \omega_p^2 r_p^4}, \quad C_{P,0} = \frac{Q \pi^3}{4 \rho \omega_p^2 r_p^5} \quad (2.46)$$

where ρ is the ambient air density and r_p the radius of the propeller. The variation of these coefficients with ω_p is shown in Figure 2–14.

The measured thrust and power coefficients function of the angular speed were compared with propellers from UIUC database [29]. A propeller with similar coefficient mean value and trend was chosen: GWS Direct Drive 4.5x3 inches. Its static coefficients are also shown in Figure 2–14. The GWS propeller thrust and power coefficients at nonzero values of J were curve fitted and scaled to match the static coefficients of the propeller in our experiment. The resulting propeller coefficients as functions of the advance ratio are shown in Figure 2–15 and are given as:

$$\begin{aligned} C_T(J) &= \underbrace{-0.1281 J^2}_{C_{T,2}} \underbrace{-0.1196 J}_{C_{T,1}} \underbrace{+ 0.1342}_{C_{T,0}} \\ C_P(J) &= \underbrace{-0.0602 J^2}_{C_{P,2}} \underbrace{-0.0146 J}_{C_{P,1}} \underbrace{+ 0.0522}_{C_{P,0}} \end{aligned} \quad (2.47)$$

The model used assumes that, at negative advance ratios, which occur during vertical descent, the static coefficients are valid. For small negative advance ratios ($-0.3 < J < 0$), this assumption is validated by experimental data [32].

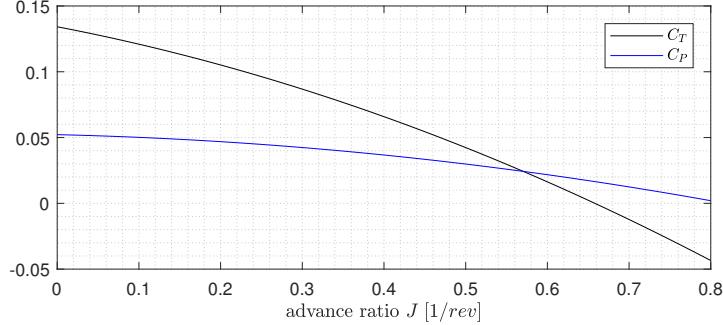


Figure 2–15: Thrust and Power Coefficients Models, Function of Advance Ratio.

2.5.2 Control Surface Deflection Map

In order to model the control surfaces, measurements were made to relate the normalized servo signal s to the deflection angle δ . Deflection angles were measured with a digital angle gauge (Capri Tools CP20005) as the servo signal was varied.

The input to the servomotor is a Pulse Width Modulation (PWM) signal [30], which is generated by the Pixhawk onboard computer. For this system, the PWM high time ranges from 850 to 2150 [μs]. The mapping from normalized servo signal to PWM high time is linear and is given for the right elevon as:

$$PWM_h = 1500 + 650 s_r \iff s_r = \frac{PWM_h - 1500}{650} \quad (2.48)$$

and for the left elevon as:

$$PWM_h = 1500 - 650 s_l \iff s_l = \frac{1500 - PWM_h}{650} \quad (2.49)$$

where s_l, s_r are the normalized servo signal ranging from -1 to 1 and PWM_h is the PWM high time ranging from 850 to 2150 [μs].

The corresponding measured elevon deflections are shown in Figure 2–16. The relationship is observed to be linear and symmetric and the maximum deflection

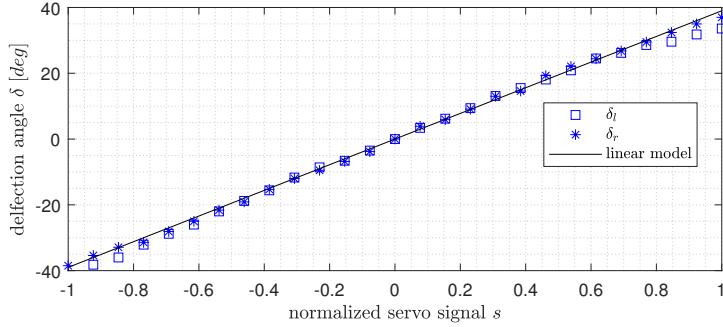


Figure 2–16: Elevons Deflection Map.

angle is taken as:

$$\delta_{max} = 39^\circ \quad (2.50)$$

As a result, the linear relationship relating the deflection δ and servo signal s is simply given by:

$$\delta_l = s_l \delta_{max}, \quad \delta_r = s_r \delta_{max} \quad (2.51)$$

2.5.3 Control Surface Effectiveness

This section discusses the measurements performed to investigate the effect of the control surface deflections δ_l and δ_r . The entire airplane was first mounted on the force torque sensor to measure the elevon deflection coefficients c_x and c_y to be used in the simplified aerodynamic model in equation 2.41. Similar measurements were performed in the simulator to measure c_x , c_y , b_x and b_y .

Bench Test on the Force/Torque Sensor

The experimental setup is shown in Figure 2–17. The aircraft center of mass was located at a distance $z = 0.236$ [m] above the sensor's origin. A constant thrust command τ and battery voltage V_{batt} was applied to each thruster. Based on the thruster model presented earlier, these corresponded to a thrust of 0.66 [N] from each thruster (about a third of the maximum thrust). At this thruster operating point and zero elevon deflections, the force torque sensor was zeroed. It results a

difference of forces and moments read by the sensor as:

$$\begin{aligned} F_x &= -\Delta D \\ M_x &= \Delta L \\ M_y &= -z\Delta D - \Delta M = zF_x - \Delta M \end{aligned} \quad (2.52)$$

where D is the drag of the aircraft due to the propellers downwash, and L and M are the rolling and pitching moments of the aircraft, assumed due to the control surface deflections only. The Δ symbol before these quantities symbolizes the difference from the zeroed operating point. Because both thrusters were running at same rotational speed, their torques cancel.

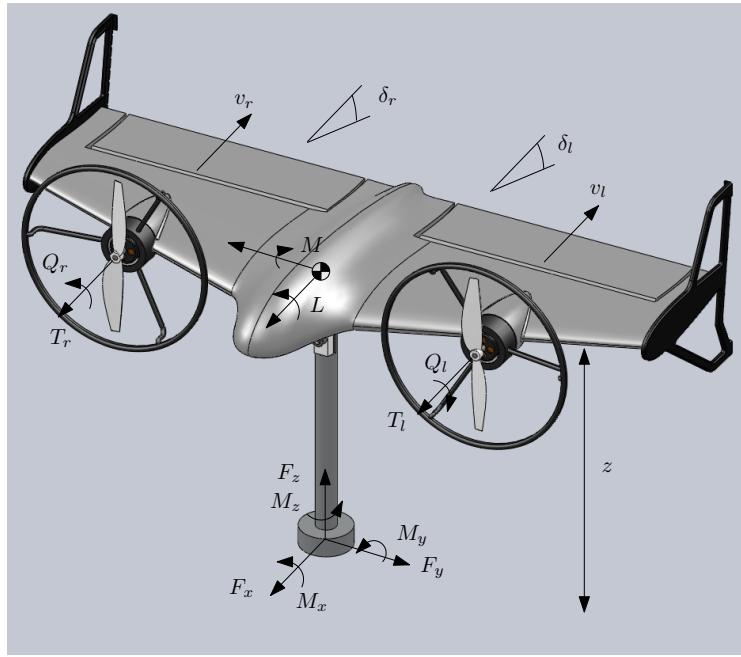


Figure 2–17: Control Surface Effectiveness Testbed. The aircraft center of mass is located at a distance z above the sensor’s origin.

Referring to the simplified model, for the rolling moment \hat{L} and pitching moment \hat{M} given by equation 2.41, and considering that the airplane is static (i.e. dynamic pressure $P_d = 0$) and that both thrust forces are equal and constant $T_l = T_r = T$, the

δ_l [deg]	δ_r [deg]	ΔL [Nm]	c_x [m^3/rad]
-30	30	-0.051	$9.82 \cdot 10^{-4}$
-20	20	0.032	$9.24 \cdot 10^{-4}$
20	-20	0.036	$10.4 \cdot 10^{-4}$
30	-30	0.053	$10.2 \cdot 10^{-4}$

Table 2–5: Rolling Moment Measures

following relations can be used for this case:

$$\begin{aligned}\hat{L} &= \Delta L = \frac{c_x}{\pi r_p^2} T(\delta_l - \delta_r) \\ \hat{M} &= \Delta M = -\frac{c_y}{\pi r_p^2} T(\delta_l + \delta_r)\end{aligned}\tag{2.53}$$

The rolling moment coefficient c_x can be retrieved from the rolling moment L measured by the torque sensor as M_x , by applying symmetrically opposite deflections to the left and right elevons $\delta_l = -\delta_r = \delta$:

$$c_x = \frac{M_x \pi r_p^2}{2T \delta}\tag{2.54}$$

Similarly, the pitching moment coefficient c_y can be retrieved from the pitching moment M given by the measurements as $M = zF_x - M_y$, by applying symmetric and equal deflections to the elevons $\delta_l = \delta_r = \delta$:

$$c_y = -\frac{(zF_x - M_y) \pi r_p^2}{2T \delta}\tag{2.55}$$

The coefficients were taken as the mean of several measurements, shown in Tables 2–5 and 2–6. The measurements were intentionally taken at high deflections to measure large moments and avoid reaching the sensor accuracy limit. The resulting values are:

$$c_x = 9.91 \cdot 10^{-4} [m^3/rad], \quad c_y = 4.74 \cdot 10^{-4} [m^3/rad]\tag{2.56}$$

δ_l [deg]	δ_r [deg]	ΔM [Nm]	c_y [m^3/rad]
-30	-30	0.026	$4.96 \cdot 10^{-4}$
-20	-20	0.017	$4.86 \cdot 10^{-4}$
20	20	-0.016	$4.59 \cdot 10^{-4}$
30	30	-0.024	$4.54 \cdot 10^{-4}$

Table 2–6: Pitching Moment Measures

Bench Test in the Simulator

The same procedure was conducted in the simulator to find c_X and c_y coefficients in order to compare them with the values obtained from the force/torque sensor. Considering \hat{L} and \hat{M} in the simplified model from equations 2.38 and 2.40 and assuming the airplane is static (i.e. $u = 0, w = 0$), and both thrusters are spinning at same rotational speed (i.e. $Q_r = Q_l$ and the slipstream $v_l = v_r = v_s$), we obtain:

$$\begin{aligned}\hat{L} &= \frac{1}{2}\rho c_x v_s^2 (\delta_l - \delta_r) \\ \hat{M} &= -\frac{1}{2}\rho c_y v_s^2 (\delta_l + \delta_r)\end{aligned}\tag{2.57}$$

For the following measurements in the simulator, the slipstream velocities are set to $v_s = 1$ [m/s], (the thrusters are not running and v_s is hard-coded).

The rolling coefficient c_x can be obtained by applying equal and opposite deflections, $\delta_l = -\delta_r = \delta$:

$$\begin{aligned}\hat{L} &= \rho c_x \delta v_s^2 \\ \hat{M} &= -\frac{1}{2}\rho c_y v_s^2 (\delta - \delta) = 0\end{aligned}\tag{2.58}$$

The rolling moment deflection coefficient is measured while the deflection δ is set at 10 different positions δ_i linearly spaced from δ_{min} to δ_{max} . Solving for c_x , we obtain:

$$c_{x,i} = \frac{L_i}{\rho \delta_i v_s^2}\tag{2.59}$$

The resulting data points $c_{x,i}$ are shown in Figure 2–18 and the value for c_x is taken at the mean value of the 10 measurements:

$$c_{x,sim} = 2.35 \cdot 10^{-3} [m^3/rad] \quad (2.60)$$

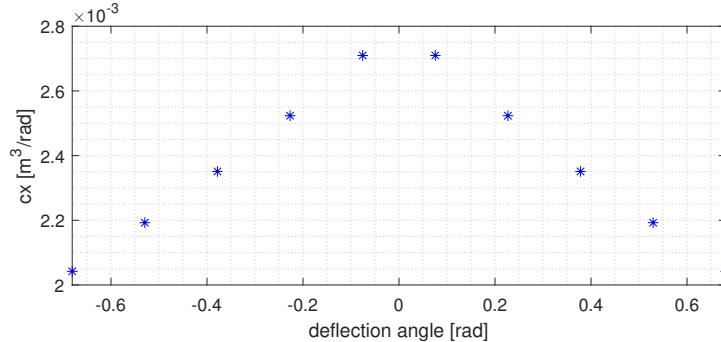


Figure 2–18: Rolling Moment Deflection Coefficient from Simulator.

Similarly, the pitching moment deflection coefficient c_y can be obtained by applying equal deflections, $\delta_l = \delta_r = \delta$:

$$\begin{aligned} \hat{L} &= \frac{1}{2}\rho c_x v_s^2 (\delta - \delta) = 0 \\ \hat{M} &= -\rho c_y \delta v_s^2 \end{aligned} \quad (2.61)$$

The pitching moment is measured while the deflection δ is set at 10 different positions δ_i linearly spaced from δ_{min} to δ_{max} . Solving for c_y , we obtain:

$$c_{y,i} = \frac{M_i}{-\rho \delta_i v_s^2} \quad (2.62)$$

The data point $c_{y,i}$ are shown in Figure 2–19 and the value for c_y is taken at the mean value of the 10 measurements:

$$c_{y,sim} = 3.5 \cdot 10^{-4} [m^3/rad] \quad (2.63)$$

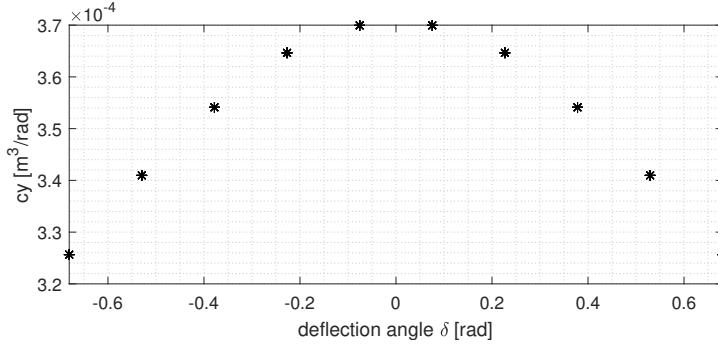


Figure 2–19: Pitching Moment Deflection Coefficient from Simulator.

To measure the deflection moment coefficients outside the slipstream b_x and b_y , it is proposed to consider the airplane with an inflow wind velocity u with zero angle of attack and the two thrusters not running, then b_x, b_y can be extracted knowing c_x, c_y . Considering \hat{L} and \hat{M} in the simplified model from equations 2.38 and 2.40 and assuming $w = 0, v_l = u, v_r = u, Q_r = 0$ and $Q_l = 0$, we obtain:

$$\begin{aligned}\hat{L} &= \frac{1}{2}\rho u^2(c_x + b_x)(\delta_l - \delta_r) \\ \hat{M} &= -\frac{1}{2}\rho u^2(c_y + b_y)(\delta_l + \delta_r)\end{aligned}\quad (2.64)$$

The rolling coefficient b_x can be obtained by applying equal and opposite deflections, $\delta_l = -\delta_r = \delta$:

$$\begin{aligned}\hat{L} &= \rho u^2(c_x + b_x)\delta \\ \hat{M} &= -\frac{1}{2}\rho u^2(c_y + b_y)(\delta - \delta) = 0\end{aligned}\quad (2.65)$$

The rolling moment deflection coefficient is measured while the deflection δ is set at 10 different positions δ_i linearly spaced from δ_{min} to δ_{max} . Solving for b_x , using $c_x = c_{x,sim}$, we obtain:

$$b_{x,i} = \frac{L_i}{\rho u^2}\delta_i - c_{x,sim} \quad (2.66)$$

The resulting data points $b_{x,i}$ are shown in Figure 2–20 and the value for b_x is taken at the mean value of the 10 measurements:

$$b_{x,sim} = 2.22 \cdot 10^{-3} \text{ [m}^3/\text{rad]} \quad (2.67)$$

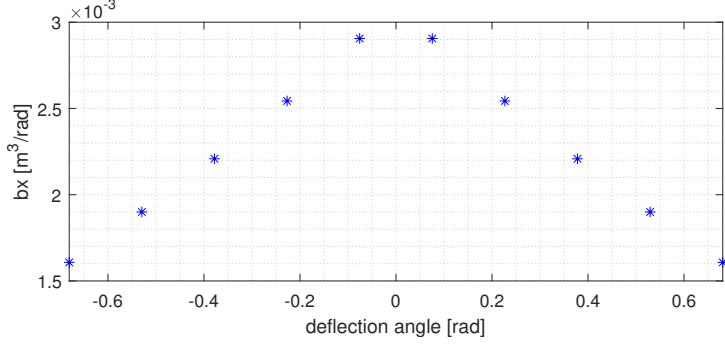


Figure 2–20: Rolling Moment Deflection Coefficient from Simulator.

The pitching coefficient b_y can be obtained by applying equal deflections, $\delta_l = \delta_r = \delta$:

$$\begin{aligned}\hat{L} &= \frac{1}{2}\rho u^2(c_x + b_x)(\delta - \delta) = 0 \\ \hat{M} &= -\rho u^2(c_y + b_y)\delta\end{aligned}\tag{2.68}$$

The pitching moment deflection coefficient is measured while the deflection δ is set at 10 different positions δ_i linearly spaced from δ_{min} to δ_{max} . Solving for b_y , using $c_y = c_{y,sim}$, we obtain:

$$b_{y,i} = \frac{M_i}{-\rho u^2 \delta_i} - c_{y,sim}\tag{2.69}$$

The data point $b_{y,i}$ are shown in Figure 2–21 and the value for b_y is taken at the mean value of the 10 measurements:

$$b_{y,sim} = 2.57 \cdot 10^{-4} \text{ [m}^3/\text{rad]}\tag{2.70}$$

The value of c_x from the simulator is about 2.5 times the measured value, while the value of c_y is about 25% lower in the simulator, indicating that the control surfaces are more effective than predicted for the pitching moment and much less effective than predicted for the rolling moment. The same trend was observed when flying the real platform manually and flying the aircraft manually in the

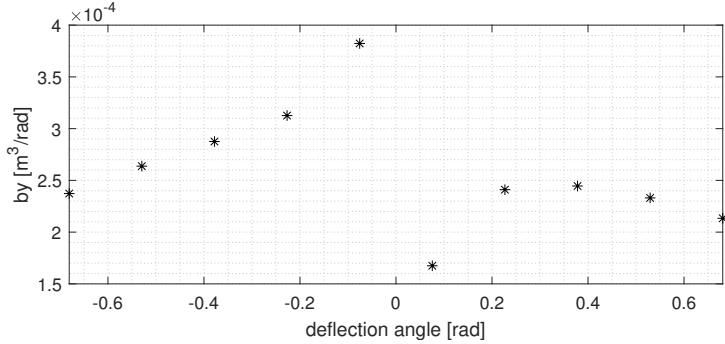


Figure 2–21: Rolling Moment Deflection Coefficient from Simulator.

simulated environment. In real flight, loop maneuvers were observed to be quicker and roll maneuvers were observed to be much slower than when accomplishing the same maneuvers in the simulator. This large disparity between the measured and simulated values is likely due to assumptions in the model, such as the fact that momentum theory was used to predict the slipstream over the elevons and the fact that flat plate was assumed for the wing profile. The measured values are considered more representative of the real aerodynamics than the simulated coefficient values.

Knowing the discrepancies between the simulated values for $c_{x,sim}$, $c_{y,sim}$ and the experimental values for c_x , c_y , it is chosen to use the following values for b_x and b_y :

$$\begin{aligned} b_x &= b_{x,sim} \cdot c_x / c_{x,sim} = 9.37 \cdot 10^{-4} \text{ [m}^3/\text{rad]} \\ b_y &= b_{y,sim} \cdot c_y / c_{y,sim} = 3.48 \cdot 10^{-4} \text{ [m}^3/\text{rad]} \end{aligned} \quad (2.71)$$

where the multiplication by the ratio of the experimental value over the simulated value is expected to capture the discrepancies. In other words, the same discrepancy ratio is assumed for the elevon effectiveness outside the slipstream and in the slipstream. The values for the control derivative coefficients c_x , c_y , b_x and b_y used by the controller are summarized in Appendix H, Table H.1.

Simulator Modification

In order to make use of the experimental measurements to improve the simulator, we chose to scale the rolling moment due to deflection by $c_x / c_{x,sim}$ and the pitching

moment due to deflection by $c_y/c_{y,sim}$. For this purpose, the rolling and pitching moments at the given elevon deflection and the rolling and pitching moments at zero deflection are computed and the chosen moments are set to:

$$L = L(\delta = 0) + \frac{c_x}{c_{x,sim}}(L(\delta_l, \delta_r) - L(\delta = 0)) \quad (2.72)$$

$$M = M(\delta = 0) + \frac{c_y}{c_{y,sim}}(M(\delta_l, \delta_r) - M(\delta = 0))$$

The effect of scaling is shown in Figure 2–22 and 2–23. Reducing the rolling moment greatly improved the simulator fidelity relative to manual flight tests with the real platform. The effect of scaling the pitching moment does not seem to have on a big impact on the curve in Figure 2–23, but it has a substantial impact on the observed maneuvers. For instance, the back transition distance at saturated elevon deflection was reduced by about 40%.

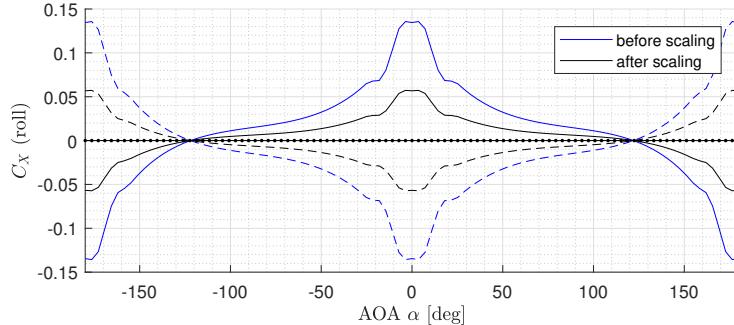


Figure 2–22: Rolling Moment Coefficient shown in blue without scaling and in black once the deflection effects are scaled by $c_x/c_{x,sim}$. The moments at $\delta_l = \delta_{max}$, $\delta_r = \delta_{min}$ are shown in plain line and the moments at $\delta_l = \delta_{min}$, $\delta_r = \delta_{max}$ are shown in dashed line.

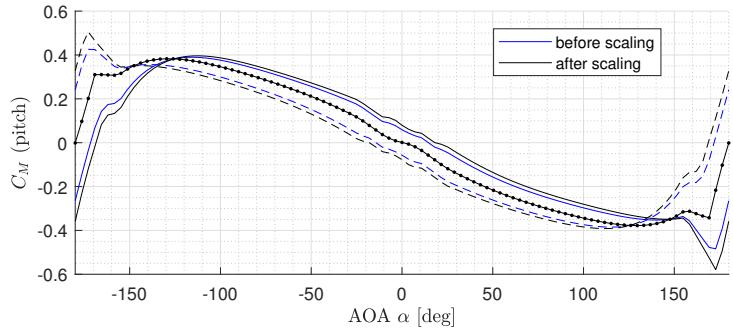


Figure 2–23: Pitching Moment Coefficient shown in blue without scaling and in black once the deflection effects are scaled by $c_y/c_{y,sim}$. The moments at minimum deflections are shown in plain line and the moments at maximum deflections are shown in dashed line.

CHAPTER 3

Control Strategy

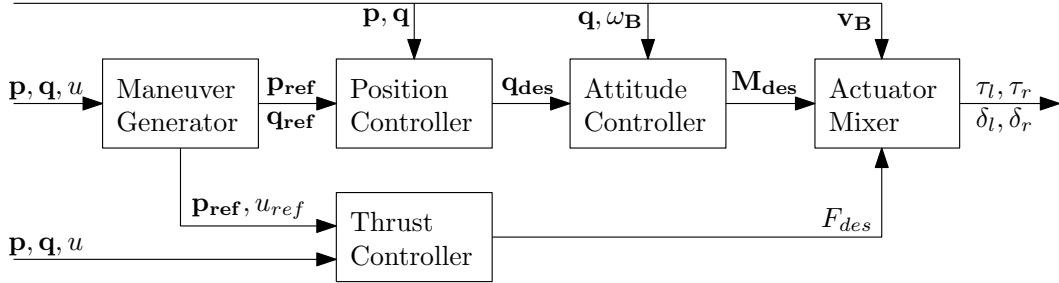


Figure 3–1: Block Diagram of the Cascaded Quaternion Controller.

The block diagram of the controller is shown in Figure 3–1. Its objective is to generate actuators signals $\tau_l, \tau_r, \delta_l, \delta_r$ to achieve a reference behavior. The behavior is characterized by the states of the system, namely the position p , the velocity v_B , the attitude q and the body rates ω_B . The states are estimated from the onboard sensor measurements by an Extended Kalman Filter (EKF) that is packaged with the PX4 flight stack [33]. More detail about the EKF and its accuracy can be found in Appendix F.

The reference states are generated by the maneuver generator, which produces the reference position p_{ref} , attitude q_{ref} and forward velocity u_{ref} . The reason not to include the other states as reference is the under-actuated nature of this system. For example, since the thrusters can produce only a forward force, only the forward component of the velocity u_{ref} is used as reference. The maneuver generator is described in Section 3.1.

The controller is then responsible for tracking the reference position, attitude and forward velocity. Due to the under-actuated nature of the system, a position controller first reorients the desired attitude of the aircraft to make the thrusters point

towards the reference position. Then, the attitude controller computes the moments needed to achieve this reorientation. In parallel, the thrust controller regulates the longitudinal force to ensure velocity and altitude tracking. Finally, the actuator mixer generates the actuators signals from the desired force and moments. The controller is described in Section 3.2.

The control algorithm is based on a cascaded quaternion controller developed for aerobatic maneuvering of an agile fixed wing UAV [14]. The controller is globally the same, except few modifications, the major one is the addition of the actuator mixer. This controller does not assume any particular operating point, which means the same controller is used for all the maneuvers regardless of the flight mode.

3.1 Maneuver Generator

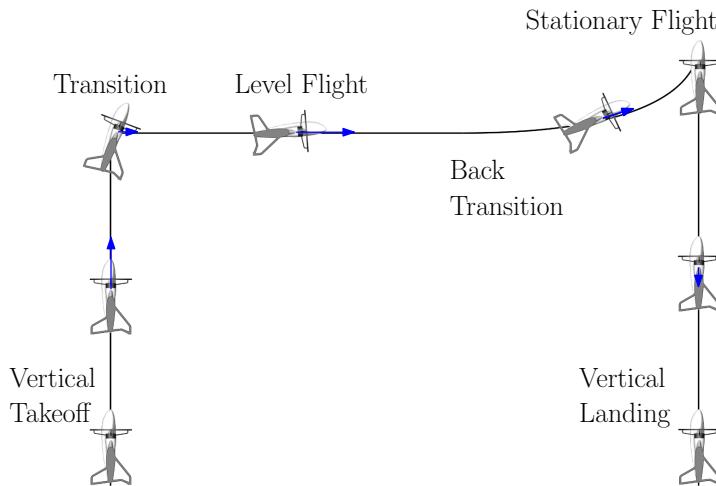


Figure 3–2: Minimal Flight Mission Reference.

The maneuver generator is implemented as a finite state machine to ensure a consistent logic. It generates the reference position p_{ref} , attitude q_{ref} and longitudinal velocity u_{ref} at each instant.

To evaluate the feasibility of the developed controller, the minimal flight mission shown in Figure 3–2 is used as a first test. In this mission, the aircraft starts on the ground, pointing nose up. It takes off and climbs vertically to the reference altitude, and then transitions to level flight. During level flight, the aircraft tracks

an horizontal line, altitude and velocity. After a specified distance being traveled, the aircraft performs the back transition by pitching up, and then descends slowly in backward flight until a few centimeters above the ground where the actuators are stopped in order to land.

The second tested maneuver is the banked turn which is used to evaluate the ability of the aircraft to also perform lateral maneuvers.

3.1.1 Vertical Maneuvers

The vertical maneuvers such as hover, climb and descent are described in this section. During these maneuvers, the reference attitude is set to vertical attitude, which is retrieved from Euler angles, as:

$$\mathbf{q}_{vert} = \mathbf{q}(\psi = \psi_{ref}, \theta = \pi/2, \phi = 0) \quad (3.1)$$

where ψ_{ref} is a user-prescribed reference heading.

During takeoff, the reference position is set at least one meter above the initial position \mathbf{p}_0 to ensure a quick maneuver and to minimize the time the aircraft is in contact with the ground with the thrusters running. During that time, the reference velocity is set to $u_{ref} = 0$.

Hover reference can be generated by specifying a constant reference position and setting a zero reference velocity u_{ref} .

A vertical climb can be obtained by increasing the reference altitude $h_{ref} = -p_{ref,z}$ linearly and keeping the reference velocity constant $u_{ref} = dh_{ref}/dt$.

The same procedure can be used to generate a vertical descent, with a negative reference velocity.

The landing is implemented as a vertical descent down to an altitude of few centimeters, where the throttles are cut and the control surfaces set to zero deflection. This allows the aircraft to be dropped on the ground from this small height completing the landing phase.

3.1.2 Level Flight

During level flight, the aircraft tracks an horizontal line at the reference altitude h_{lvl} . The reference position is set as \mathbf{p}_{\parallel} : the current position, projected along a line created at the level altitude with the initial heading. It can be constructed from a unit vector pointing horizontally towards the initial heading direction \mathbf{u}_0 :

$$\mathbf{p}_{\parallel} = \mathbf{p}_1 + (\mathbf{p} - \mathbf{p}_1)^T \mathbf{u}_0 \mathbf{u}_0, \quad \mathbf{u}_0 = \begin{bmatrix} \cos \psi_0 \\ \sin \psi_0 \\ 0 \end{bmatrix} \quad (3.2)$$

During level flight, the reference attitude is set as the level flight attitude :

$$\mathbf{q}_{lvl} = \mathbf{q}(\psi = \psi_0, \theta_{lvl}(V), \phi = 0) \quad (3.3)$$

where θ_{lvl} is the pre-computed equilibrium pitch angle accounting for the wing lift, as a function of the level flight velocity V . As proposed for the agile fixed wing aircraft [14], this reference pitch angle is computed from the free body diagram during level flight as depicted in Figure 3–4. The angle of attack can be retrieved from the body frame velocities:

$$\alpha = \text{atan}2(w, u) \quad (3.4)$$

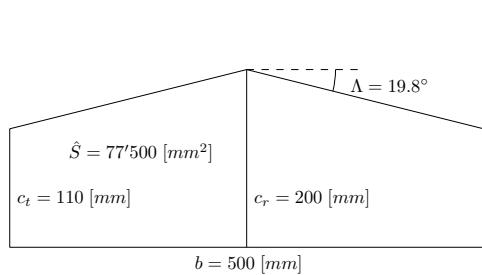


Figure 3–3: Flying Wing Dimensions.

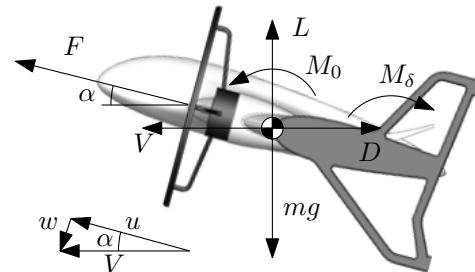


Figure 3–4: Level Flight Free Body Diagram.

In this particular case, the pitch angle equals the angle of attack, $\theta_{lvl} = \alpha$. From Figure 3–4, the sum of the forces along the vertical and horizontal axes can be computed as:

$$\begin{aligned} F \sin \theta + L &= mg \\ F \cos \theta &= D \end{aligned} \quad (3.5)$$

We can eliminate F by combining these equations:

$$F = \frac{mg - L}{\sin \theta} = \frac{D}{\cos \theta} \implies mg - L = \tan \theta D \quad (3.6)$$

The lift and drag forces can now be decomposed into their standard aerodynamic equations to show the dependence on the cruise velocity V :

$$mg - \frac{1}{2}\rho V^2 S C_L = \tan \theta \frac{1}{2}\rho V^2 S C_D \quad (3.7)$$

where ρ is the air density, S is the wing surface area, estimated by \hat{S} in the flying wing model in Figure 3–3, and C_L and C_D are the lift and drag coefficients. The velocity can then be isolated as:

$$V = \sqrt{\frac{mg}{\frac{1}{2}\rho S(C_L + C_D \tan \theta)}} \quad (3.8)$$

The lift and drag coefficients can be approximated by linear models as the aircraft is expected to fly level at low angles of attack. In this linear range, the following equations are used to describe the lift and drag coefficients:

$$C_L = C_{L\alpha} \alpha, \quad C_D = C_{D,0} + \frac{C_L^2}{\pi k_0 A_R} \quad (3.9)$$

where $C_{D,0} = 0.02$ is the drag coefficient due to skin friction, and $k_0 = 0.87$ is Oswald's efficiency factor [21]. For a flying wing, the lift curve slope can be approximated [18] as:

$$C_{L\alpha} = \frac{2\pi \cos \Lambda}{\frac{2\cos \Lambda}{A_R} + \sqrt{1 + \left(\frac{2\cos \Lambda}{A_R}\right)^2}} \quad (3.10)$$

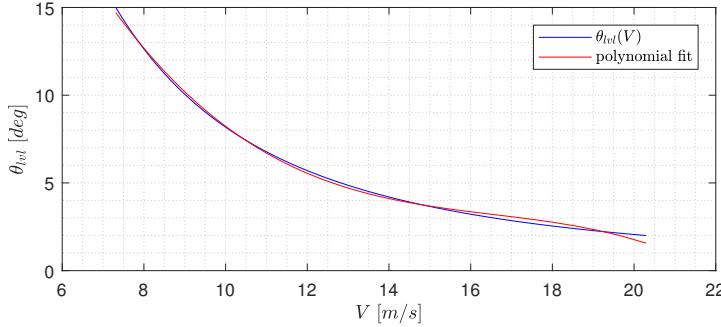


Figure 3–5: Equilibrium Pitch Angle in Level Flight.

where $\Lambda = 19.8^\circ$ is the sweep angle and $A_R = b^2/\hat{S} = 3.2$ is the aspect ratio, both obtained from Figure 3–3.

The level flight velocity at any pitch angle can now be found by combining Equations 3.8, 3.10 and 3.9. From this, an inverse relation for $\theta_{lvl}(V)$ can be Found, as shown in Figure 3–5. Finally, a curve fit to this data yields:

$$\theta_{lvl}(V) = -2.4 \cdot 10^{-4} V^3 + 0.011 V^2 - 0.178 V + 1.03 \text{ [rad]} \quad (3.11)$$

The last needed reference for the level flight is the forward velocity u_{ref} , which is found by trigonometry from the reference cruise velocity V :

$$u_{ref} = V \cos \theta_{ref} \quad (3.12)$$

3.1.3 Transitions

In the simulator, the reference attitude \mathbf{q}_{ref} during transition and back transition is discontinuous, but for the real flight experiment a smooth transition was necessary to improve stability. Discontinuities can be smoothed out by using half a cosine function on the reference pitch angle as follows:

$$\theta_{ref}(t) = \begin{cases} \theta_v, & t < t_{t,0} \\ \theta_{lvl} + \frac{1}{2} \left(1 + \cos \left(\pi \frac{t - t_{t,0}}{T_{tr}} \right) \right) (\theta_v - \theta_{lvl}), & t_{t,0} \leq t \leq t_{t,0} + T_{tr} \\ \theta_{lvl}, & t > t_{t,0} + T_{tr} \end{cases} \quad (3.13)$$

where $\theta_v = \pi/2$ is the reference vertical pitch angle, θ_{lvl} is the reference level flight pitch angle described in section 3.1.2, $t_{t,0}$ is the time when the transition starts, recorded during the flight, and T_{tr} is the duration of the transition.

Equivalently, the back transition can be set as follows:

$$\theta_{ref}(t) = \begin{cases} \theta_{lvl}, & t < t_{b,0} \\ \theta_v + \frac{1}{2} \left(1 + \cos \left(\pi \frac{t - t_{b,0}}{T_b} \right) \right) (\theta_{lvl} - \theta_v), & t_{b,0} \leq t \leq t_{b,0} + T_b \\ \theta_v, & t > t_{b,0} + T_b \end{cases} \quad (3.14)$$

where $t_{b,0}$ is the time when the back transition starts, recorded during the flight and T_b is the duration of the back transition.

The position and velocity reference values during the transition are set as $\mathbf{p}_{ref} = \mathbf{p}_{\parallel}$ and $u_{ref} = V \cos \theta_{ref}$.

The back transition is more elaborate. When initiating a back transition, the current projected position \mathbf{p}_{\parallel} is saved and used as the reference position for the whole back transition:

$$\mathbf{p}_{ref} = \mathbf{p}_{\parallel}(t_{b,0}) \quad (3.15)$$

Starting from level flight, the aircraft will initially continue moving forward and \mathbf{p}_{ref} will be behind it to ensure a short back transition. The velocity is set as $u_{ref} = V \cos \theta_{ref}$. The back transition ends when the aircraft starts to point backward, i.e. the heading changes direction:

$$\min(|\psi - \psi_0|, |\psi - \psi_0 + 2\pi|, |\psi - \psi_0 - 2\pi|) > 90^\circ \quad (3.16)$$

where the minimum of the three terms ensures continuity of the heading difference.

3.1.4 Banked Turn

For the system to be functional in maneuvers out of the vertical plane, the maneuver generator should be able to generate a banked turn. In this scenario, the aircraft needs to describe an arc of radius R at a horizontal velocity V . From Figure

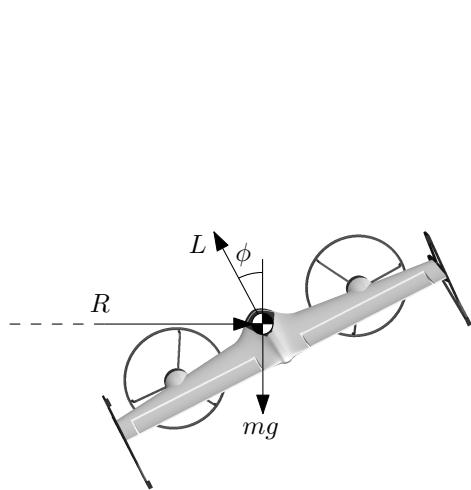


Figure 3–6: Banked Turn Free Body Diagram, rear view.

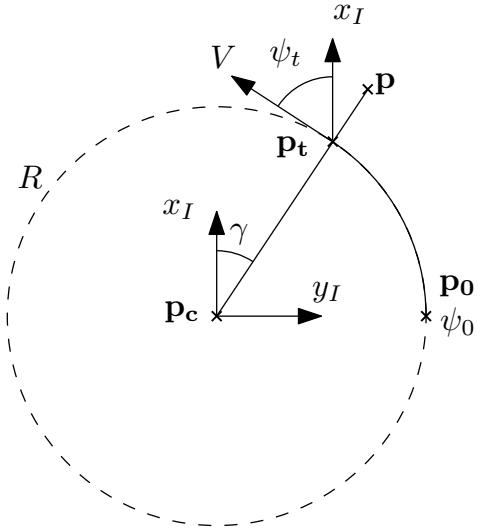


Figure 3–7: Banked Turn Reference Trajectory, top view.

3–6, the aircraft equation of motion in the horizontal direction is:

$$L \sin |\phi| = \frac{m V^2}{R} \quad (3.17)$$

where V^2/R is the centripetal acceleration inward toward the center of the turn and ϕ is the roll angle.

Expanding the lift force yields:

$$\frac{1}{2} \rho V^2 S C_L \sin |\phi| = \frac{m V^2}{R} \quad (3.18)$$

Again, using the linear model for the lift force given by Eq. 3.9, the positive reference bank angle can be retrieved as:

$$\sin \phi_{ref} = \frac{2 m}{R \rho S C_{L\alpha} \theta} \quad (3.19)$$

The vertical force balance in order to maintain constant altitude is given by:

$$L \cos \phi + F \sin \theta \cos \phi = m g \quad (3.20)$$

To avoid losing altitude, the lift force should ideally be increased during the banked turn relative to level flight. This effect will be neglected as the thrust and position

controller are assumed to take care of the required lift force modification. This leads to the following attitude reference in level flight:

$$\mathbf{q}_{ref} = \mathbf{q}(\psi = \psi_t, \theta_{lvl}(V), \phi = \sigma\phi_{ref}) \quad (3.21)$$

where $\theta_{lvl}(V)$ is defined in equation 3.11, ϕ_{ref} is given in equation 3.19, and ψ_t is defined below. The symbol σ represents the sign of the turn, defined as $\sigma = 1$ for a right turn and $\sigma = -1$ for a left turn.

From Figure 3–7, the center of the circle knowing the initial position \mathbf{p}_0 and initial heading ψ_0 can be computed as:

$$\mathbf{p}_c = \mathbf{p}_0 + R \begin{bmatrix} \cos(\psi_0 + \sigma\pi/2) \\ \sin(\psi_0 + \sigma\pi/2) \\ 0 \end{bmatrix} \quad (3.22)$$

Knowing the center of the circle \mathbf{p}_c , the turn position \mathbf{p}_t can be obtained by projecting the current position \mathbf{p} over the circle of radius R describing the trajectory, as shown in Figure 3–7.

$$\gamma = atan2(p_y - p_{c,y}, p_x - p_{c,x}) \quad (3.23)$$

$$\mathbf{p}_{ref} = \mathbf{p}_t = \mathbf{p}_c + R \begin{bmatrix} \cos \gamma \\ \sin \gamma \\ 0 \end{bmatrix} \quad (3.24)$$

The reference heading during the turn ψ_t is given as:

$$\psi_t = \gamma + \sigma\pi/2 \quad (3.25)$$

where ψ_t is adjusted to stay in its defined range $[-\pi; \pi]$.

During the banked turn, the reference velocity is the same as in level flight:

$$u_{ref} = V \cos \theta_{ref} \quad (3.26)$$

3.2 Controller

From the current states, namely the position \mathbf{p} , the orientation \mathbf{q} and the forward velocity u , and body rates ω_B , the controller generates the actuators signals, which are the left and right throttles τ_l, τ_r and left and right elevon deflections δ_l, δ_r . The objective is to track the reference signals given by the maneuver generator, namely the position \mathbf{p}_{ref} , orientation \mathbf{q}_{ref} and forward velocity u_{ref} .

From the feed forward reference attitude \mathbf{q}_{ref} , the position controller provides a modified orientation \mathbf{q}_{des} to point the aircraft thrusters towards the reference position \mathbf{p}_{ref} . Then the attitude controller applies a PD law on the attitude error between \mathbf{q}_{des} and \mathbf{q} to compute the desired moments \mathbf{M}_{des} . In parallel, the desired forward force F_{des} is regulated by the thrust controller to track the altitude and forward velocity. Finally the actuator mixer inverts the aircraft dynamics to generate the actuators signals $\tau_l, \tau_r, \delta_l, \delta_r$ to achieve the desired force and moments.

3.2.1 Position

An aircraft is underactuated and cannot apply forces in arbitrary directions. The idea behind the position controller is to orient the longitudinal force F towards the reference position \mathbf{p}_{ref} . For this purpose, it modifies the reference attitude \mathbf{q}_{ref} into the desired one \mathbf{q}_{des} based on position error.

Its first step is to generate the correction angles Θ_y and Θ_z , from the position error, by computing a PD control law, using the positive gains k_{pp} and k_{pd} respectively:

$$\begin{bmatrix} - \\ \Theta_z \\ \Theta_y \end{bmatrix} = \mathbf{R}_r \left(k_{pp} (\mathbf{p}_{ref} - \mathbf{p}) + k_{pd} \frac{d}{dt} (\mathbf{p}_{ref} - \mathbf{p}) \right) \quad (3.27)$$

where $\mathbf{R}_r = \mathbf{R}(\mathbf{q}_{ref})$ is the rotation matrix from the inertial to the reference attitude and can be retrieved from the reference quaternion \mathbf{q}_{ref} . Additionally, the correction

angles Θ_y and Θ_z are bounded to $\pm\Theta_{max} = \pm 15^\circ$ to avoid extreme corrections during hover.

At this point, Θ_y will correct the pitch (up-down) and Θ_z will correct the yaw (right-left). Another way to reduce the lateral position (right-left) error is by making use of the lift force by performing a banked turn. This is enabled only during level flight, not during hover. The roll correction Θ_x can be based on Θ_z , corrected by the cosines of the current pitch and the roll angles, to account smoothly for changes in the expected lift force. This result in:

$$\Theta_x = \Theta_z \cos \theta \cos \phi \quad (3.28)$$

The correction angles are converted into unit quaternions by a proper quaternion construction, refer to Appendix G:

$$\begin{aligned} \mathbf{q}_x &= [\cos \frac{\Theta_x}{2}, \sin \frac{\Theta_x}{2}, 0, 0] \\ \mathbf{q}_y &= [\cos \frac{\Theta_y}{2}, 0, -\sin \frac{\Theta_y}{2}, 0] \\ \mathbf{q}_z &= [\cos \frac{\Theta_z}{2}, 0, 0, \sin \frac{\Theta_z}{2}] \end{aligned} \quad (3.29)$$

Finally the desired attitude \mathbf{q}_{des} is computed by rotating the reference attitude obtained from the maneuver generator \mathbf{q}_{ref} around the z axis (yaw), then around the y axis (pitch) and finally around the x axis (roll):

$$\mathbf{q}_{des} = \mathbf{q}_{ref} \otimes \mathbf{q}_z \otimes \mathbf{q}_y \otimes \mathbf{q}_x \quad (3.30)$$

3.2.2 Attitude

The attitude controller generates the desired moments $L_{des}, M_{des}, N_{des}$ from the attitude error $\Delta\mathbf{q}$. The attitude error can be retrieved by multiplying the conjugate attitude \mathbf{q}^* by the desired attitude \mathbf{q}_{des} or its negation (which describes the same

attitude):

$$\Delta \mathbf{q} = \begin{cases} \mathbf{q}^* \otimes \mathbf{q}_{des}, & \text{if } \|\mathbf{q} - \mathbf{q}_{des}\| \leq \|\mathbf{q} + \mathbf{q}_{des}\| \\ \mathbf{q}^* \otimes (-\mathbf{q}_{des}), & \text{if } \|\mathbf{q} - \mathbf{q}_{des}\| > \|\mathbf{q} + \mathbf{q}_{des}\| \end{cases} \quad (3.31)$$

where the case ensures the angular errors remain less than 180° .

It is known that a linear PD control law on the vector part of the quaternion can stabilize the attitude [34]. This is why the desired moments $\mathbf{M}_{des} = [L_{des}, M_{des}, N_{des}]^T$ are computed by multiplying the inertia with a PD control law on the vector part of the quaternion $\Delta \mathbf{q}_v$ and the body rates $\boldsymbol{\omega}_B = [p, q, r]^T$:

$$\begin{aligned} L_{des} &= I_x (k_{ap_L} \Delta q_{v,1} - k_{ad_L} p) \\ M_{des} &= I_y (k_{ap_M} \Delta q_{v,2} - k_{ad_M} q) \\ N_{des} &= I_z (k_{ap_N} \Delta q_{v,3} - k_{ad_N} r) \end{aligned} \quad (3.32)$$

where k_{ap} and k_{ad} are the P and D gains.

In the simulator, the same gains are used for all three axes:

$$\begin{aligned} k_{ap} &= k_{ap_L} = k_{ap_M} = k_{ap_N} \\ k_{ad} &= k_{ad_L} = k_{ad_M} = k_{ad_N} \end{aligned} \quad (3.33)$$

3.2.3 Thrust

In parallel to the position and attitude controller, the thrust controller generates the desired longitudinal force F_{des} to track the reference altitude $h_{ref} = -p_{ref,z}$ and reference longitudinal velocity u_{ref} . The control law is defined as follows:

$$F_{des} = \max \left(0, \underbrace{mg \sin \theta}_{F_w} + \underbrace{m k_{up} (u_{ref} - u)}_{F_u} + \underbrace{m k_{hp} (h_{ref} - h) \sin \theta}_{F_h} \right) \quad (3.34)$$

where the max function ensures a positive longitudinal force. F_w is the term that cancels the weight of the aircraft along the longitudinal axis, F_u regulates the velocity and F_h tracks the altitude. The aerodynamic lift and drag forces are not accounted

for in the thrust controller as they are highly nonlinear and have little effect on the altitude and longitudinal velocity tracking.

3.2.4 Actuator Allocation

The last module of the controller is the actuator mixer. Its objective is to translate the desired longitudinal force F_{des} and desired moments $\mathbf{M}_{des} = [L_{des}, M_{des}, N_{des}]^T$ into actuator signals, namely the left and right throttle signals τ_l, τ_r and the left and right elevons deflections δ_l, δ_r . For this purpose, the simplified dynamic model from Eq. 2.41 is inverted.

First, a upper bound of 95% of the maximum longitudinal force at the current flying conditions is set on F_{des} . This is used to keep room for yawing authority achieved by differential thrust. The maximum thrust force of one thruster T_{max} can be retrieved using the forward thruster model in Section 2.3.6 and feeding it the current battery voltage, forward velocity and maximum throttle signal. This upper bound F_{max} is then set as:

$$F_{max} = 2 \cdot 0.95 \cdot T_{max} \quad (3.35)$$

If F_{des} is greater than F_{max} , it is limited to F_{max} .

The actuator mixer then solves the left and right thrust forces from F_{des} and N_{des} :

$$\begin{aligned} T_l &= \frac{1}{2} F_{des} + \frac{1}{2l} N_{des} \\ T_r &= \frac{1}{2} F_{des} - \frac{1}{2l} N_{des} \end{aligned} \quad (3.36)$$

where l is the lateral distance from the center of mass to the right thruster.

Then, a lower bound T_{min} is set on the thrust forces to keep it positive and to keep a minimum slipstream velocity $v_{s,min}$ over the elevons which ensures a minimal authority of the control surfaces:

$$T_{min} = \max \left(0, \frac{1}{2} \rho \pi r_p^2 (v_{s,min}^2 - u^2) \right) \quad (3.37)$$

At this point, the left and right thruster torques Q_l, Q_r and the left and right throttles τ_l, τ_r , can be computed from the left and right thrust forces T_l, T_r , and the longitudinal velocity u . The propeller model from Section 2.3.5 is inverted analytically since all the sub models are second order polynomials. This inverse thruster model is described in Appendix E.

Knowing $T_l, T_r, Q_l, Q_r, L_{des}$ and M_{des} , the left and right elevon deflections can be solved as a linear system:

$$\mathbf{A} \begin{bmatrix} \delta_l \\ \delta_r \end{bmatrix} = \underbrace{\begin{bmatrix} L_{des} - (Q_r - Q_l) \\ M_{des} - M_0 \end{bmatrix}}_b \Rightarrow \begin{bmatrix} \delta_l \\ \delta_r \end{bmatrix} = \mathbf{A}^{-1} \mathbf{b} \quad (3.38)$$

$$\mathbf{A} = \begin{bmatrix} \frac{c_x}{\pi r_p^2} T_l + P_d b_x & -\frac{c_x}{\pi r_p^2} T_r - P_d b_x \\ -\frac{c_y}{\pi r_p^2} T_l - P_d(c_y + b_y) & -\frac{c_y}{\pi r_p^2} T_r - P_d(c_y + b_y) \end{bmatrix} \quad (3.39)$$

where b_x, b_y, c_x, c_y are the aerodynamic deflection coefficients defined in the simplified analytic model in section 2.4 and $P_d = \frac{1}{2}\rho(u^2 + w^2)$ is the dynamic pressure.

The matrix \mathbf{A} is invertible as long as there is a nonzero velocity on the control surfaces, so \mathbf{A} is always invertible due to $v_{s,min}$.

In reality, in the onboard flight controller, the elevon deflections are commanded by a normalized servomotor signal ranging from -1 to 1 . Those left and right signals can be easily obtained from the left and right deflection, knowing the maximum allowable deflection δ_{max}

$$\begin{aligned} s_l &= \delta_l / \delta_{max} \\ s_r &= \delta_r / \delta_{max} \end{aligned} \quad (3.40)$$

A special treatment is performed on the desired pitching moment M_{des} , as it is the principal moment acting during the transitions maneuvers. If the elevon deflections are saturated, it is chosen to boost this moment by increasing the slipstream velocities. The output moment M_{out} , which is the pitching moment after saturation of

the elevon deflections into their range $[-\delta_{max}, \delta_{max}]$, computed from the simplified model in Section 2.4, is used to check the saturation condition. If $|M_{out}| < |M_{des}|$, then a saturation of the pitching moment is detected. This method allows detection of saturation of the pitching moment without necessarily a saturation of both elevon deflections. Then, the mean elevon deflection is computed as:

$$\bar{\delta} = \frac{1}{2}(\delta_l + \delta_r) \quad (3.41)$$

If this mean deflection $\bar{\delta}$ is null, increasing the slipstream velocities will increase the rolling moment but not the pitching moment, so the boost is not applied. Whereas if this mean deflection is non zero, the slipstream velocities are boosted by modifying the desired longitudinal force as:

$$F_{des} = \frac{M_{des} - \hat{M}_0 + 2P_d(c_y + b_y)\bar{\delta}}{-\frac{c_y}{\pi r_p^2}\bar{\delta}} \quad (3.42)$$

and the mixer is recomputed again from Eq. 3.35 to Eq. 3.40.

CHAPTER 4

Flight Tests Results

This chapter reports the flight experiments carried out to demonstrate the performance of the control algorithm to accomplish stable and autonomous flight. The results are divided into four sections, reporting the flight tests from the simulator, the hardware in the loop simulation, the tethered indoor flights, and experimental flights.

4.1 Simulated Flights

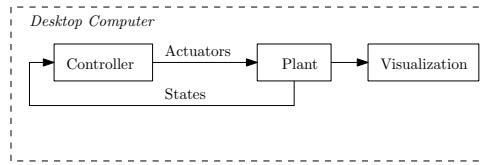


Figure 4–1: Schematic of the Conventional Simulation. The controller and plant are simulated in Simulink, the visualization is realized in X-Plane.

Two flight missions, originally described in Chapter 3.1, were tested in the real-time simulator.

The controller presented in Chapter 3 is implemented in Simulink and closes the loop around the simulated plant, as shown in Figure 4–1. The states are simulated without noise but a North East wind with mean velocity of 1 [m/s] is present for realism. A greater wind speed would topple the tailsitter before takeoff. The aircraft parameters are shown in Appendix Table H.1 and the controller gains are shown in Appendix Table H.2.

4.1.1 Minimal Flight Mission

In order to evaluate the aircraft controller performance, we devised a minimal flight mission that would include some of the key flight phases of a tailsitter aircraft. The flight mission is set as follows, the airplane starts vertically facing the North

direction. It takes off vertically and climbs to an altitude of 6 [m], transitions to level flight pointing North, at an altitude of 6 [m] and a ground velocity of 7 [m/s]. At this velocity, the reference pitch angle is given as $\theta_{lvl} = 15.7^\circ$. After travelling 40 [m] horizontally, a back transition is commanded. The vehicle then descends in backward flight at a velocity of 0.5 [m/s]. Finally, when the airplane reaches an altitude of 0.2 [m], the throttles are stopped, resulting in a vertical landing.

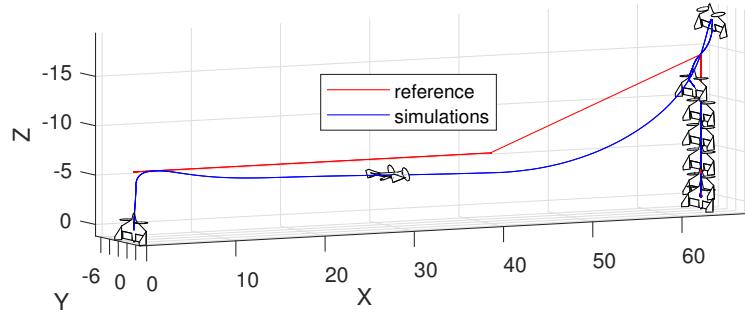


Figure 4–2: Minimal Flight Mission Trajectory in Simulation. The airplane is not represented to scale.

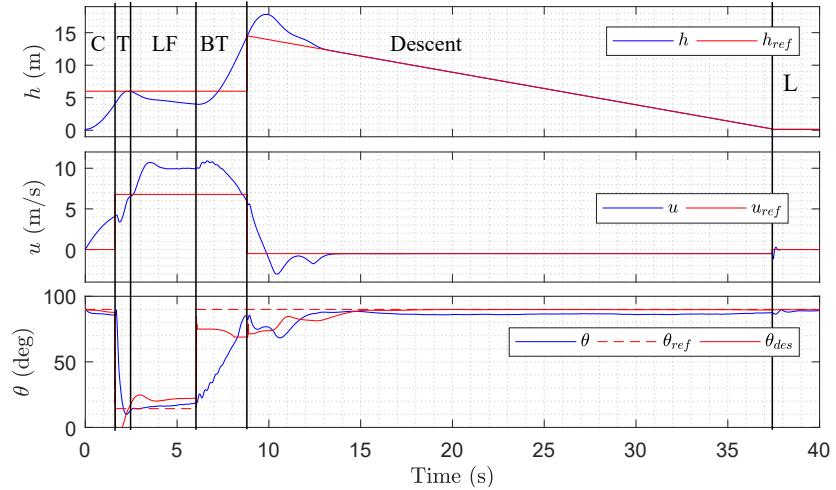


Figure 4–3: Minimal Flight Mission States in Simulation. From top to bottom, altitude h and reference h_{ref} , forward velocity u and reference u_{ref} , pitch angle θ , reference θ_{ref} and desired θ_{des} . The flight phases are: the climb C, the transition T, the level flight LF, the back transition BT, the descent, and the landing L.

The simulation results are shown in Fig.4–2 to 4–4. A first observation is the discontinuity of the reference signals sent to the controller, such as the altitude h_{ref}

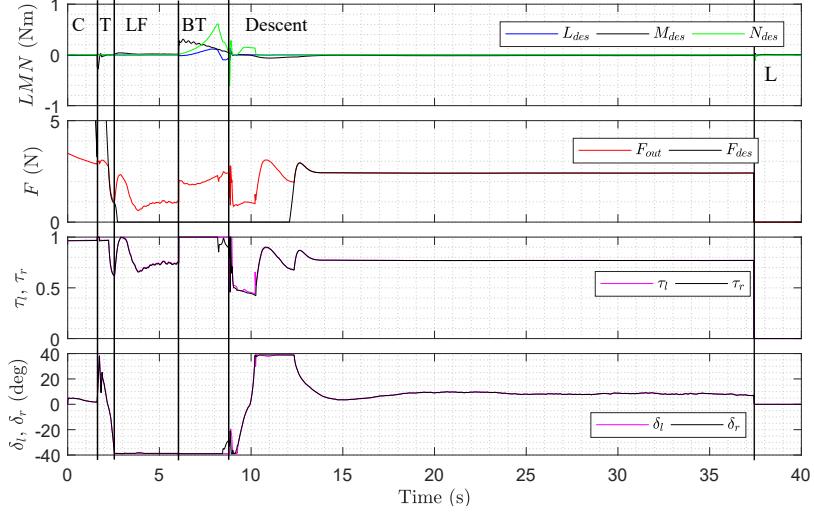


Figure 4–4: Minimal Flight Mission Control States in Simulation. From top to bottom, desired moments L_{des} , M_{des} , N_{des} , desired forward force F_{des} , left and right throttles τ_l , τ_r , left and right elevon deflections δ_l , δ_r . The flight phases are: the climb C, the transition T, the level flight LF, the back transition BT, the descent, and the landing L.

and the pitch angle θ_{ref} . In spite of these discontinuities, the controller is able to maintain stability during all the flight stages.

The climb phase is observed to last about 2 [s] to climb 6 [m] during which the thrusters are almost saturated. The remaining 5% of the throttle signal are reserved for yaw authority as described in the actuator mixer in section 3.2.4. Then the transition and level flight last about 5 [s] and the aircraft covers a distance of 40 [m], with an altitude error up to 2 [m] and a velocity error up to 4 [m/s]. The thrust controller tries to increase the longitudinal force F_{des} in order to climb, but also tries to reduce it to slow down to track the velocity u_{ref} . As a result, $F_{des} = 0$ [N] during level flight because the velocity error is very large. However, the actuator mixer overrides this to set a non zero longitudinal force to ensure elevon effectiveness with a minimum slipstream velocity. The elevons are observed to be saturated upward during level flight, which shows that the aircraft is flying at its maximum angle of attack ability, given its cruise speed.

During the transition from hover to level, the elevon deflections are observed not to be fully saturated, but during the back transition, the elevons are saturated. Because the aircraft is stable in pitch, this tends to help forward transition, but makes back transition more difficult to accomplish. The back transition maneuver is observed to occupy about 27 [m] horizontally and 14 [m] vertically.

During the vertical descent at 0.5 [m/s], very good stability and tracking are observed. Finally the aircraft successfully performs the vertical landing by cutting the throttles when the aircraft's CM is 20 [cm] above the ground, corresponding to a drop from 5.5 [cm].

4.1.2 Banked Turn

The goal of this mission is to demonstrate the controller ability to perform a banked turn. For this mission, the aircraft starts in level flight, then tracks two horizontal left loops with a radius of 15 [m], which leads to a reference bank angle $\phi_{ref} = -24.1^\circ$. The aircraft finally exists the loop and finishes the mission in level flight.

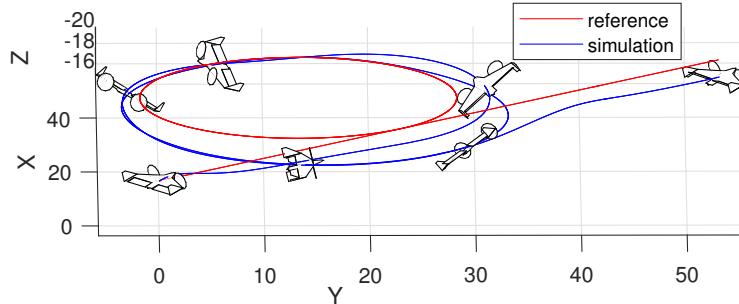


Figure 4–5: Banked Turn Trajectory in Simulation. The airplane is not represented to scale.

The banked turn results in simulation are shown in Figure 4–5 to 4–7. The aircraft tracks the loop about 1.5 meters below the reference and up to 5 meters out of the circle. The roll angle ϕ is well tracked. The forward velocity is poorly tracked, as before, due to the conflicting objective of tracking the altitude at the same time. The actuators are observed to be almost saturated during the turn. The

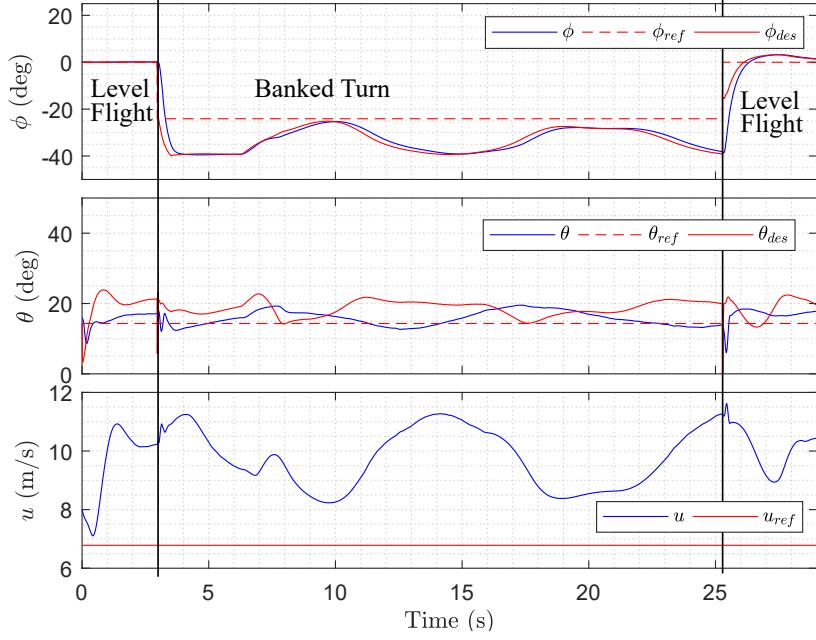


Figure 4–6: Banked Turn States in Simulation. From top to bottom, bank angle ϕ , reference ϕ_{ref} , and desired ϕ_{des} , pitch angle θ , reference θ_{ref} and desired θ_{des} , forward velocity u and reference u_{ref} .

aircraft initiates the turn at time $t = 3$ [s] and ends it at time $t = 26$ [s]. At these times, the control surfaces are observed to be suddenly changing, in order to modify the roll angle quickly.

The system is able to perform the banked turn maneuver. This shows that the controller is suited for this type of maneuvers, but the poor tracking performance and actuator saturation show that the platform is overweight. This was confirmed by other simulation with reduced aircraft mass, not shown here.

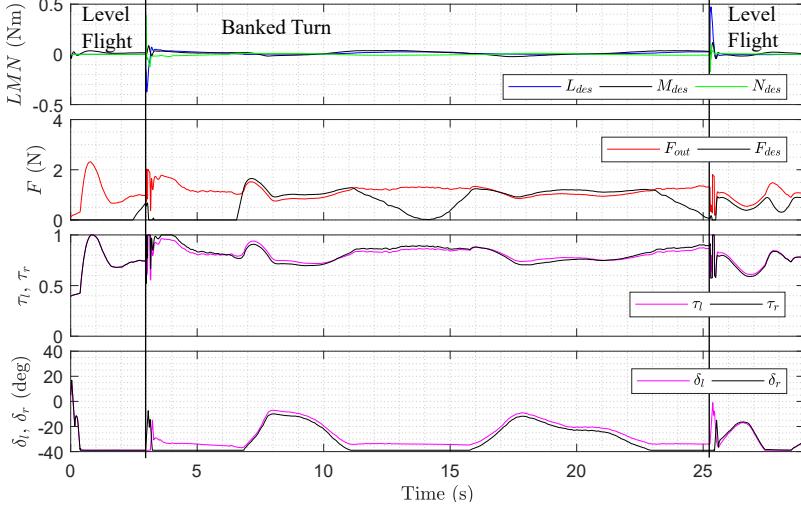


Figure 4–7: Banked Turn Control States in Simulation. From top to bottom, desired moments L_{des} , M_{des} , N_{des} , desired forward force F_{des} , left and right throttles τ_l , τ_r , left and right elevon deflections δ_l , δ_r .

4.2 Hardware-In-the-Loop Flights

Once the aircraft and controller performance were satisfactory in the pure simulation environment, they were then tested in a Hardware-In-the-Loop (HIL) simulator. Unlike in a conventional simulator, in the HIL environment, the controller runs on the onboard hardware, which in this case is a Pixhawk autopilot. The hardware sends actuators signals to the desktop computer via a USB cable. The simulator running on the desktop computer evaluates the dynamics given the actuator signals and the previous states of the system. It then sends back raw sensor measurements corrupted with realistic noise. The state estimator on the Pixhawk generates the states from those simulated measurements and the controller closes the loop by computing and sending the actuator signals for the next time step, as shown in Figure 4–8. This setup enables the simulation of the embedded control software and allows testing of the hardware related issues, such as memory management, state estimation, controller discretization, and timing delays.

A HIL environment for an agile fixed wing platform was previously developed in the laboratory [35]. That setup uses the built-in HIL firmware from the PX4 flight

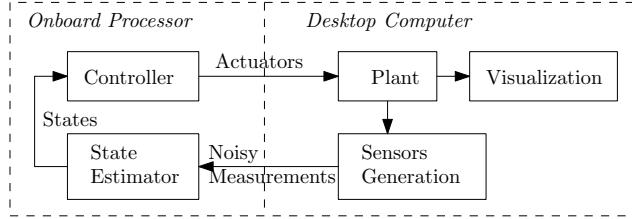


Figure 4–8: HIL Simulation.

stack which communicates to QGroundControl via MavLink messages¹ through the USB cable. The communication from QGroundControl to Simulink is done via UDP messages [36]. The whole setup being already functional, the communication aspect did not need to be modified. The real-time simulator of the tailsitter described in Section 2.3 was incorporated into the HIL environment and the embedded code inside the PX4 flight stack was modified accordingly to handle the four actuators signals used by this tailsitter vehicle. The sensor noise characteristics were previously determined for prior research, from high pass filtered flight data [35]. The same noise was used since the same sensors were used in this research.

4.2.1 Performance Evaluation

From the firsts HIL tests, poor performance of the system was observed. The system was unstable and not able to maintain a hover. The reasons of this loss of performance compared to the conventional simulation, were therefore investigated.

The simulator update rate was evaluated. The desired update rate was set to $t_{s,d} = 5 \text{ [ms]}$ in the "Real-Time Sync" Simulink block². This corresponds to the Pixhawk update rate in the real aircraft, as well as the update rate used in the pure simulation. We measured the update rate recorded by the Pixhawk hardware during the execution in order to compare it with this desired value. The update rate

¹<http://qgroundcontrol.org/mavlink/start>

²<https://www.mathworks.com/help/sldrt/ref/realtimesync.html>

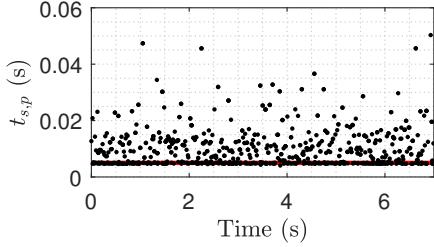


Figure 4–9: HIL Update Rate $t_{s,p}$, recorded by the Pixhawk.

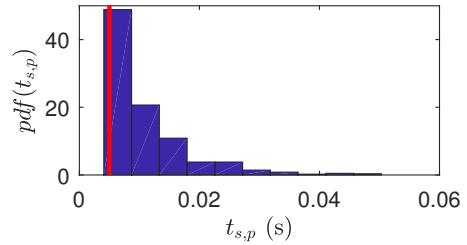


Figure 4–10: Corresponding Probability Density Function $pdf(t_{s,p})$, the reference $t_{s,d} = 0.005$ [s] is shown in red.

recorded by the Pixhawk hardware $t_{s,p}$ is shown in Figure 4–9 and its corresponding probability density function is shown in Figure 4–10. The mean and Root Mean Square (RMS) of the recorded update rate are given by:

$$\bar{t}_{s,p} = 0.0103 \text{ [s]}, \quad RMS(t_{s,p}) = 0.127 \text{ [s]} \quad (4.1)$$

We observe the mean update rate to be twice the desired update rate. This is likely due to timing inaccuracies in the execution of the simulation. This inaccurate and erratic update rate will lead to inaccurate results and may be the cause of instability.

The other source of problem, which is intimately related to the first one is the rate at which the raw data are sent to the Extended Kalman Filter (EKF). Some of the real system's sensors operate at much higher rate; for instance, the IMU operates at 1 [kHz], which is ten time faster than the mean recorded rate. The resulting downsampling effects lead to less accurate and slower state estimation which will affect the control loop performance.

4.2.2 Results

As described above, the flight experiments in the HIL environment were not successful. However, these experiments helped to better understand the challenges associated with the control of this system. Three modifications on the controller were performed:

- The control gain values were decreased as described in [35]. This was an expected result due to the introduction of noise in the feedback loop.
- The discrete derivative of the quaternion vector was replaced by the body rates. In the attitude controller, the PD law was originally set as:

$$\mathbf{M}_{des} = \mathbf{I} \left(k_{ap} \Delta \mathbf{q}_v - k_{ad} \Delta \dot{\mathbf{q}}_v \right) \quad (4.2)$$

where the differentiation of the quaternion vector part for evaluation of $\Delta \dot{\mathbf{q}}_v$ was implemented as a discrete derivative, which created noise. A low-pass filter was implemented, as well as removal of outlier data. However, this filtering led to a trade-off between noise and delay, where neither effect was desirable. Instead, it was proposed to use the body rates ω_B estimated by the EKF which are smoother:

$$\mathbf{M}_{des} = \mathbf{I} \left(k_{ap} \Delta \mathbf{q}_v - k_{ad} \omega_B \right) \quad (4.3)$$

This is the PD control law described in Equation 3.32, which has an analytically proven convergence [34].

- It has been observed that the attitude controller performed better about some axes. In hover for instance, the lateral tilt angle was observed to diverge slowly while the pitch angle was observed to be oscillatory, implying that the attitude gains were too high for the y-axis but too low for the z-axis. The solution to this problem was to use different control gains for each axis:

$$\mathbf{M}_{des} = \begin{bmatrix} L_{des} \\ M_{des} \\ N_{des} \end{bmatrix} = \begin{bmatrix} I_x(k_{ap,L} \Delta q_1 - k_{ad,L} p) \\ I_y(k_{ap,M} \Delta q_2 - k_{ad,M} q) \\ I_z(k_{ap,N} \Delta q_3 - k_{ad,N} r) \end{bmatrix} \quad (4.4)$$

After these modifications, the system was still unstable, but instead of being chaotic, only a slow divergence in the pitch angle was observed. Figure 4–11 shows the trajectory of an unsuccessful hover maneuver in the HIL. Due to the

absence of feasible short-term feasible solution for improving the timing of the HIL environment, it was chosen the proceed to the flight experiments.

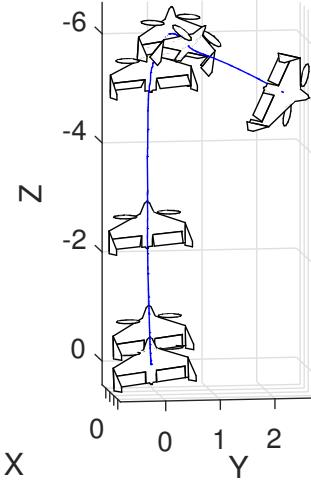


Figure 4–11: Unstable Hover Trajectory in HIL. The aircraft is supposed to takeoff, climb and perform a hover at 5 [m], but the aircraft becomes unstable during the hover. Aircraft not represented to scale.

4.3 Tethered Indoor Flights

Indoor flight experiments were carried out to test the correct behavior of the controller in hover and to tune the control gains. The tests were performed in the laboratory, in a constrained environment, where GPS signal is not available. In this context, the onboard position controller cannot be used to maintain a position. Instead, the controller was modified to accept user inputs from the radio control transmitter to control the position. Thus, these indoor experiments only test the attitude and thrust controllers.

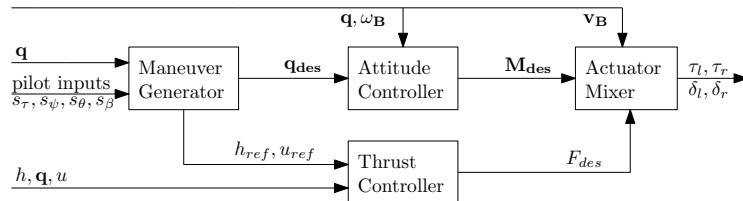


Figure 4–12: Manually Assisted Hover Controller Block Diagram.

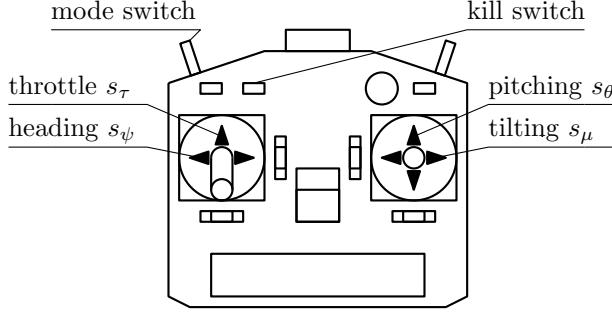


Figure 4–13: Radio Control Transmitter.

4.3.1 Manually Assisted Hover Controller

To incorporate the user inputs from the radio control transmitter, the position controller is removed and the maneuver generator directly generates the desired attitude \mathbf{q}_{des} , as shown in Figure 4–12.

The radio control transmitter is equipped with two sticks, each with two degrees of freedom, as shown in Figure 4–13. The throttle stick generates a normalized signal ranging from 0 to 1, and the heading, pitching and tilting sticks generate normalized signals ranging from -1 (left or down) to $+1$ (right or up). The throttle stick s_τ is used to generate the reference altitude h_{ref} which is fed through the thrust controller:

$$h_{ref} = s_\tau h_{max} \quad (4.5)$$

where h_{max} is the maximum altitude of the maneuver and has been set to half a meter for the indoor experiments.

The heading stick s_ψ is used to control the rotation about a vertical axis by modifying the reference yawing angle as:

$$\psi_{ref} = \psi_0 + \int_0^t \dot{\psi}_{max} s_\psi dt \quad (4.6)$$

where $\dot{\psi}_{max}$ is the maximum heading rate and is set to $\pi/3 [rad/s]$. ψ_{ref} is rolled over the range $[-\pi, \pi]$, to ensure it remains within its feasible range. Then the

nominal vertical attitude \mathbf{q}_v in hover is formed from the Euler angles:

$$\mathbf{q}_v = \mathbf{q}(\phi = 0, \theta = \pi/2, \psi = \psi_{ref}) \quad (4.7)$$

The pitching and tilting sticks, s_θ and s_μ are used to generate the correction angles Θ_y and Θ_z as:

$$\Theta_y = s_\theta \Theta_{max}, \quad \Theta_z = s_\mu \Theta_{max} \quad (4.8)$$

where Θ_{max} is the same maximum correction error as defined in the position controller, i.e. 15° .

To form the desired attitude \mathbf{q}_{des} , a procedure similar to the position controller is used. The correction angles are translated into unit quaternions as follows:

$$\mathbf{q}_y = [\cos \frac{\Theta_y}{2}, 0, -\sin \frac{\Theta_y}{2}, 0], \quad \mathbf{q}_z = [\cos \frac{\Theta_z}{2}, 0, 0, \sin \frac{\Theta_z}{2}] \quad (4.9)$$

and finally the desired attitude \mathbf{q}_{des} is computed by rotating the vertical attitude \mathbf{q}_v around the z axis and then around the y axis:

$$\mathbf{q}_{des} = \mathbf{q}_v \otimes \mathbf{q}_z \otimes \mathbf{q}_y \quad (4.10)$$

The maneuver generator also generates the reference forward velocity, which is set to $u_{ref} = 0 [m/s]$ because the maneuver considered is a hover.

4.3.2 Tethered Experiment

To perform the flight experiments in a safe environment, the aircraft was first tethered by three fishing lines able to handle a maximum force of $176 [N]$ ($18 [kg]$) each. One fishing line was connected on each landing gear and the third one was connected at the nose of the vehicle. This setup allowed the aircraft to evolve freely in a sphere about half a meter in diameter, without ever making contact with the floor or ceiling, as shown in Figure 4–14. Interestingly enough, the same kind of tethered experiments were carried out on the Convair XFY Pogo in order for the pilot to get accustomed to the control of this tailsitter aircraft [2].

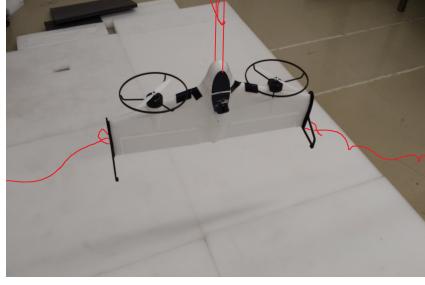


Figure 4–14: Three Points Tethered Setup for Hover Indoor Flight Experiments.

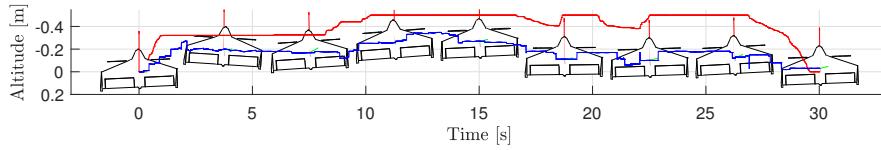


Figure 4–15: Tethered Hover Experiment Trajectory. Because only the altitude is available but not the position, time is plotted on the horizontal axis. The reference altitude is shown in red, the actual altitude in blue and the reference longitudinal orientation is indicated by the red arrow.

The measured results are shown in Figures 4–15 and 4–16, showing that the controller works well for this hover maneuver. The local altitude was measured with a distance sensor mounted on the aircraft and pointing to the rear, shown in Appendix Figure F.4, while the forward velocity u was estimated from the EKF relying on the barometer and IMU sensors. The estimated u , which is fed back to the thrust controller, is observed to drift with time, which leads to poor altitude tracking: up to 0.4 meter. However, this altitude error is satisfactory for a typical application where the aircraft will fly at an altitude of few meters, or more.

The stability of the aircraft can be appreciated from the pitch angle data shown in Figure 4–16. The pitch angle never goes below 78° , and the pitch angle error stays within 10° . We observe that both throttle signals τ_l, τ_r remain within 0.7 and 0.8 during the stationary flight, which leaves 20 – 30% of the throttle for excess thrust. This shows the airplane is a bit overweight, but as a result of the controller, the throttle signals are very steady. The elevon deflections are more oscillatory with

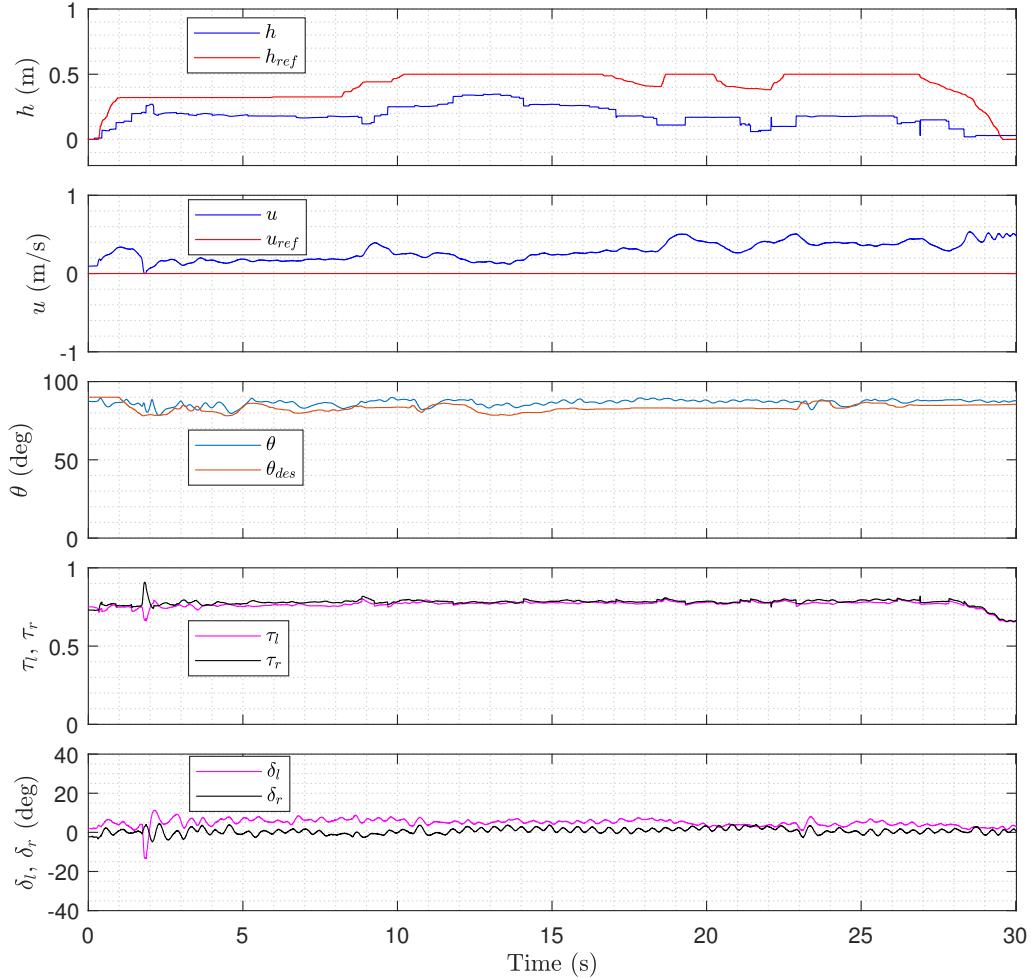


Figure 4–16: Tethered Hover Experiment States. From top to bottom, altitude h and reference h_{ref} , forward velocity u and reference u_{ref} , pitch angle θ and reference θ_{des} , left and right throttles τ_l, τ_r , left and right elevon deflections δ_l, δ_r .

an amplitude about 4° , which is very reasonable. The fact that the left deflection seems to have an offset of 4° above the right deflection could be explained by a length difference of the control rods linking the servomotors to the elevons.

Testing the vertical takeoff and landing was also performed in the laboratory. For this purpose, the aircraft was tethered with one line connected at each landing gear. This allowed the aircraft to make contact with a foam mattress covering the ground, but limits its movements above the mattress. This experiment was successful but as

it is very similar to the previous experiment, it is not reported here. Other takeoff and landing experiments are reported in the next section.

4.4 Experimental Flights

This section reports the experimental flight tests, which were carried out inside a large air-supported dome that normally serves as an indoor football field. The ground is covered by synthetic grass and GPS signals can be received inside this structure. The maneuvers were performed autonomously using the control strategy discussed in Chapter 3 and the control gains given in Appendix H Table H.2. Furthermore, the manually assisted hover controller, described in Section 4.3.1, was used as an emergency backup in case of problems.

4.4.1 Vertical Flight

For this experiment, the airplane takes off vertically, climbs to a given altitude, then descends at a given speed and finally lands on its tail when it is close enough to the ground. In order to leave the ground quickly and avoid unmodeled ground contact forces interfering with the controller, the reference altitude is set to 1 meter to perform a takeoff at full throttle. Then the reference commands the aircraft to climb vertically at a speed of 0.33 [m/s]. This controlled climb is used to avoid throttle saturation, which would interfere with lateral tilt angle regulation, that is controlled by differential thrust. For this maneuver, when the reference reaches the desired altitude of 3 meters, the maneuver generator initiate the descent at a constant speed of 0.25 [m/s]. The climb and descent speeds are specified by the user.

The maneuver trajectory is shown in Figure 4–17 along with a mosaic created from a statically recorded video in Figure 4–18. We observe that the video shows a much larger position error than indicated by the measurements. The error distance at landing is about 5 meters but the logged error distance is 1.2 meters. This is an illustration of the inaccuracy of the GPS used to estimate the position, as discussed in Appendix F.

The vertical flight results are shown in Figure 4–19. It can be seen from the throttle curves that the aircraft takes off at full throttle. It can also be seen that the desired longitudinal force is saturated in the first few instants of the maneuver. The saturation for the desired control signals appears in red on Figure 4–20. The throttle saturation has the effect of preventing control of the yawing moment N_{out} , which can be seen to saturate at zero. After this flight test, the mixer was modified to leave room for differential thrust by introducing equation 3.35. This lack of yawing moment authority means the aircraft cannot regulate its lateral tilt angle until it reaches the reference altitude. This loss of lateral tilt control can be perceived in the pitch angle tracking, at the first instants of the climb (because Euler angles suffer from the singularity in vertical flight, the tilt angle is captured by the pitch angle). The thrust controller performs well during the climb and the descent, with an altitude tracking error of about half a meter. The performance of the thrust controller is also reflected in the forward velocity tracking. Satisfactory results are observed with less than 0.3 [m/s] of velocity tracking error. Small oscillations are observed on the pitch angle, corresponding to the oscillations of the control surfaces. The effect of the position controller can be observed on the pitch angle data. Indeed, θ_{des} is reduced as time goes on, because the position error is growing and the position controller tries to reduce it. An interesting phenomenon is observed on the throttle and desired forward force. This is a consequence of using the unfiltered distance sensor data to estimate the altitude. The distance sensor Lightware SF20 operates at a slower sampling rate (50 Hz), with a resolution of 1 [cm], which results in the presence of discrete steps in the measured distance. The thrust controller includes a derivative term, which acts as a high pass filter over those steps. A similar phenomenon occurs when the maneuver switches from climb to descent, the reference altitude is continuous but the reference velocity is discontinuous, resulting in a sawtooth on the throttle signal and desired forward force. As described in Equation 3.34, the

desired forward force is made up of three components:

$$F_{des} = F_w + F_u + F_h \quad (4.11)$$

where these three forces represents the three goals of the thrust controller: to cancel the weight of the vehicle projected onto the longitudinal axis, to track the forward velocity and to track the altitude. According to Figure 4–20, the largest component of the thrust controller serves to cancel the weight of the vehicle. The velocity tracking is observed to have almost no contribution once the takeoff is completed. The altitude tracking term is observed to reduce the total forward force because the vehicle is consistently above the altitude reference. The weight of the vehicle was measured on a high-accuracy scale (Acculab VI-1200, with a resolution of 0.1g) and is therefore unlikely to be wrong. A more likely explanation for this performance is that the thrust produced by the thruster is a little higher than that predicted by the thruster model derived in Section 2.3.5. The last variable of interest is the battery voltage which is observed to drop as the thrusters are running. Although, this effect has not been modeled, it is expected and is due to the internal resistance of the battery. The current flowing out of the battery to power the motors create this voltage drop while flowing through this resistance. The controller adjusts the throttle signals when the battery drops, to ensure the desired thrust force.

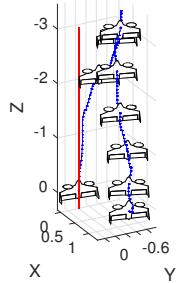


Figure 4–17: Vertical Flight Trajectory. The reference attitude is represented by the red-green-black frame.



Figure 4–18: Vertical Flight Mosaic. Each frame is separated by 2.5 seconds.

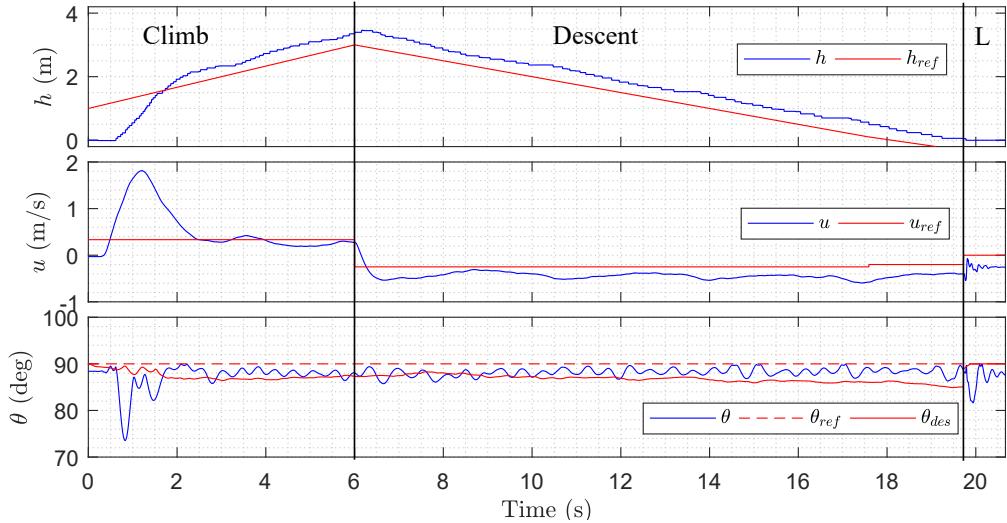


Figure 4–19: Vertical Flight States. From top to bottom, altitude h and reference h_{ref} , forward velocity u and reference u_{ref} , pitch angle θ , reference θ_{ref} and desired θ_{des} . The flight phases are: the climb, the descent, and the landing L.

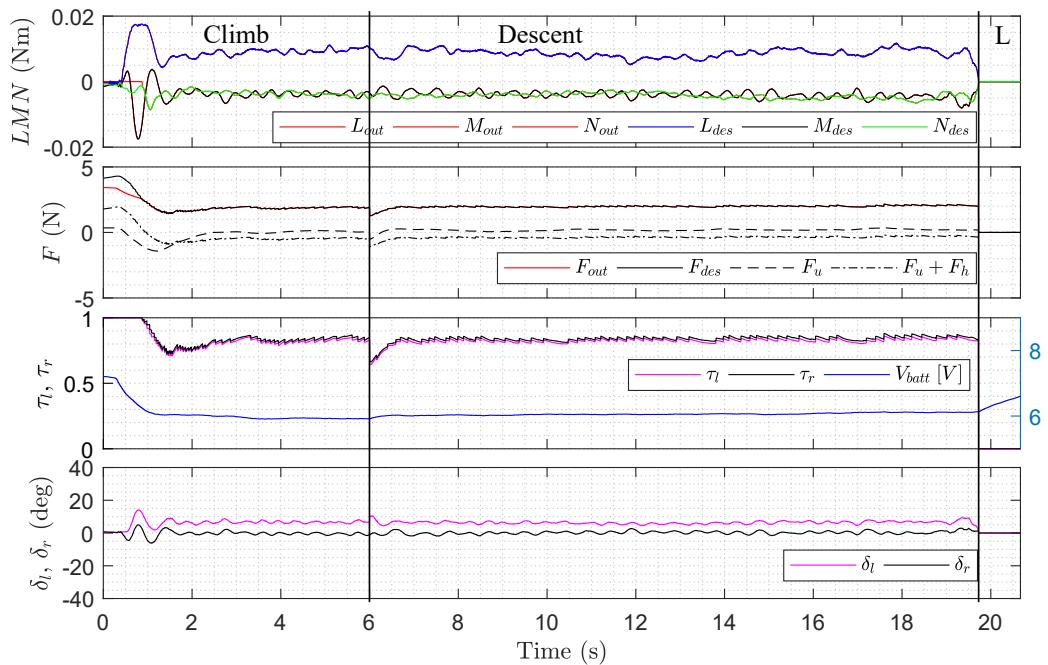


Figure 4–20: Vertical Flight Control Signals. From top to bottom, desired moments L_{des} , M_{des} , N_{des} , desired forward force F_{des} and battery voltage, left and right throttles τ_l , τ_r , left and right elevon deflections δ_l , δ_r . The red curves show the output signals. The flight phases are: the climb, the descent, and the landing L.

4.4.2 Minimal Flight Mission

This section reports the minimal flight mission, that includes vertical takeoff and climb, transition to level flight, back transition to hover, vertical descent and landing. We will compare this real flight mission with the same mission performed in the simulator, as reported in Section 4.1.1. Because the test was performed in an indoor environment, the maneuver was limited in space. The level altitude was set to 3 meters and the horizontal distance before back transition was set to 25 meters.

The flight mission trajectory is shown on Figure 4–21. The top view trajectory in Figure 4–22 shows that the aircraft’s heading is not pointing in the correct direction before the forward transition. Thus, once the aircraft is in level flight, it tries to follow the reference by performing a slight banked turn, leading to a position tracking error during level flight of about 2 meters.

The data collected during this flight mission are shown in Figure 4–23. The altitude tracking errors are less than 0.4 meter during the climb and descent, and are less than a meter during the level flight. The forward velocity tracking error is very satisfactory with less than 0.2 [m/s] during steady climb and descent, and less than 0.5 [m/s] during level flight. These tracking errors are naturally larger during takeoff and forward and back transitions. Regarding the attitude, the smooth transition using the cosine function, described in section 3.1.3, can be observed on θ_{ref} . The pitch angle θ is well tracked during the takeoff climb and transition. During the back transition, the pitch angle is less stable, going as low as 66°, but the aircraft remains stable. The discontinuity in the altitude estimate at time $t = 14.8$ [s] is due to the distance sensor whose values are suddenly considered when the pitch angle becomes greater than $\theta = \pi/4$ as described in Appendix F.

The level flight pitch angle is observed to oscillate between 35° and 40°. This oscillation can be explained by two factors: First, this high pitch angle in level flight likely results in an angle of attack beyond stall. This implies that the wing portions

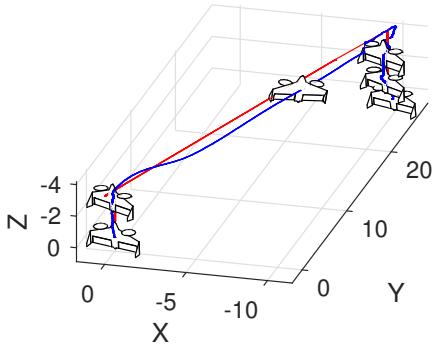


Figure 4–21: Minimal Flight Mission Trajectory. The estimated path is shown in blue and the reference path is in red. The reference attitude is represented by the red-green-black frame. Aircraft not to scale.

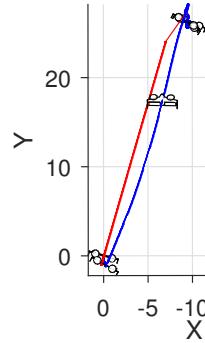


Figure 4–22: Minimal Flight Mission Top View, (same legend).

out of the slipstream do not contribute to generate the lift predicted by the linear model used to compute equilibrium pitch angle in cruise, as described in Section 3.1.2. The second effect is due to the predicted pitching moment at zero deflection M_0 described in Appendix D. According to the simulator, beyond an angle of attack of 20° , the pitching moment of the wing is greater than the moment generated by the control surfaces at maximum upward deflection. Thus, it is not possible to fly level above 20° pitch angle in the simulator, but the real experiment clearly shows level flight at a higher angle of attack. Thus, either the control surfaces are able to generate a greater pitching moment in reality, but it has been tested in Section 2.5.3, or the pitching moment at zero deflection is not as large as predicted by the simulator. The second option seems more plausible. The actuator mixer described in Section 3.2.4 uses this pitching moment model, which may be in error and may cause the observed oscillations.

The oscillations are also observed on the desired pitching moment M_{des} in Figure 4–24, and on the control surfaces deflections δ_l and δ_r . Because the reference velocity u_{ref} is a function of $\cos \theta$, oscillations are also present there. After that flight, it was chosen to set the reference velocity u_{ref} function of $\cos \theta_{des}$ instead, to

avoid having an oscillatory reference, which could lead to instabilities. The oscillations in u_{des} are also propagated to the desired forward force F_{des} and ultimately to the throttle signals. During the level flight, δ_l is observed to be smaller than δ_r , as a result of the banking control. Finally, during the back transition, the forward force T_{out} is observed to be greater than the desired F_{des} (at around $t = 15$ [s]). This is a result of the boost to increase the slipstream over the control surfaces and therefore increase the control surface pitching moment, as described in Section 3.2.4 for the actuator mixer.

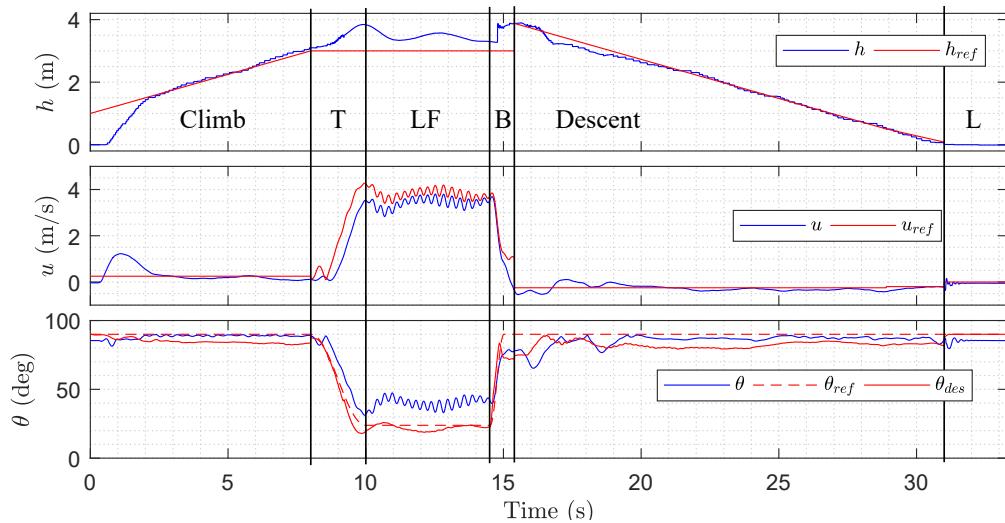


Figure 4–23: Minimal Flight Mission States. From top to bottom, altitude h and reference h_{ref} , forward velocity u and reference u_{ref} , pitch angle θ , reference θ_{ref} and desired θ_{des} . The flight phases are: the climb, the transition T, the level flight LF, the back transition B, the descent, and the landing L.

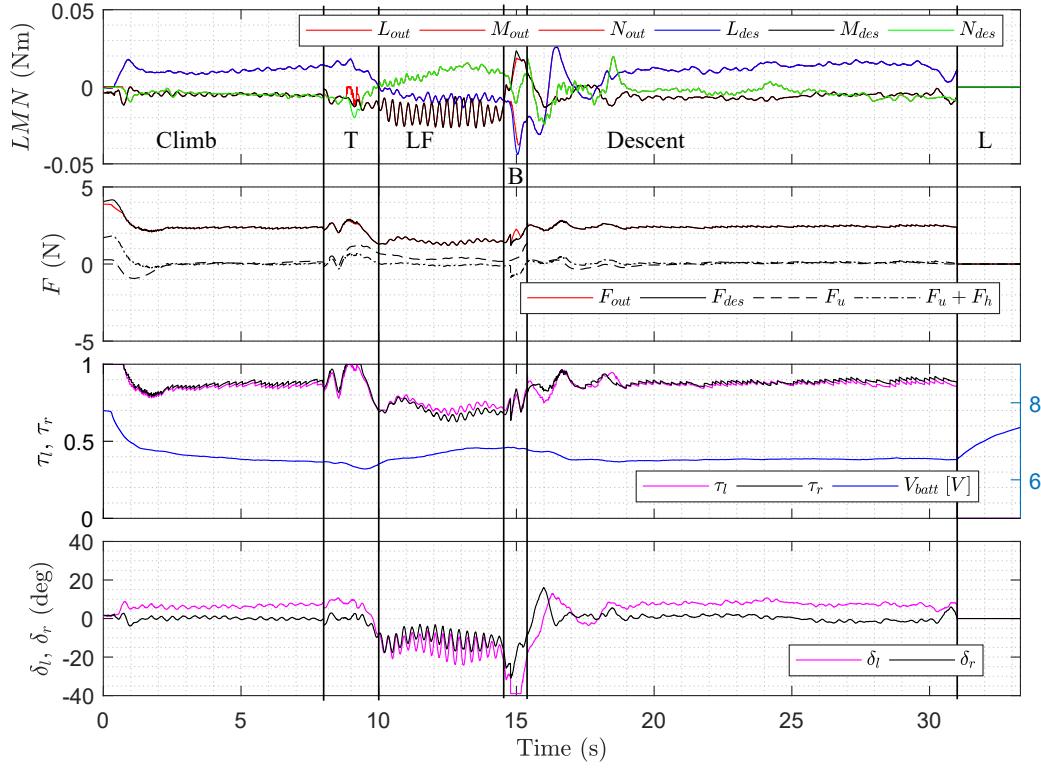


Figure 4–24: Minimal Flight Mission Control Signals. From top to bottom, desired moments L_{des} , M_{des} , N_{des} , desired forward force F_{des} and battery voltage, left and right throttles τ_l , τ_r , left and right elevon deflections δ_l , δ_r . The red curves show the output signals. The flight phases are: the climb, the transition T, the level flight LF, the back transition B, the descent, and the landing L.

4.4.3 Banked Turn

In order for the aircraft to be capable of performing a complete flight mission, it must be able to turn to modify its heading. The smoothest and most energy efficient way to perform a turn is by banking the aircraft, refer to Section 3.1.4. The maneuver presented here is a left turn, with a radius of 10 meters, accomplished with a reference bank angle $\phi_{ref} = -33.9^\circ$, allowing the aircraft to perform two complete loops. The recorded trajectory is presented in Figure 4–25. The radial tracking error is less than a meter and the altitude error is about two meters. Surprisingly, the aircraft is able to perform a sharper turn than its reference, implying that the lift force is actually greater than the estimate, which is used in the maneuver generator to

calculate the desired roll angle, as discussed in Section 3.1.4. This can be explained by the fact that the aircraft wing section is cambered, while a thin flat airfoil was assumed in the simulator for lift computation. As shown in Figure 4–26, the roll angle ϕ is well tracked with less than 5° error during the banked turn. The pitch angle θ is above its desired value and the forward velocity is below its reference, as observed in the other experiments. The forward velocity tracking is related to the altitude tracking by the thrust controller, which explains their poor performance. Regarding the controls in Figure 4–27, all the control signals are observed to be below saturation. The same oscillations as the level flight in the minimal flight mission are present on the pitch angle. However, the desired forward velocity is not oscillatory because it was changed to be a function of $\cos \theta_{des}$ for this flight.

This maneuver is useful to compare the energy requirement of the aircraft. During a hover, the recorded longitudinal force needed to sustain the aircraft is about 2 [N], refer to Figure 4–20. By contrast, the recorded thrust needed to maintain a banked turn is about 1.1 [N]. This reflects the advantage of a fixed-wing platform over a rotorcraft.

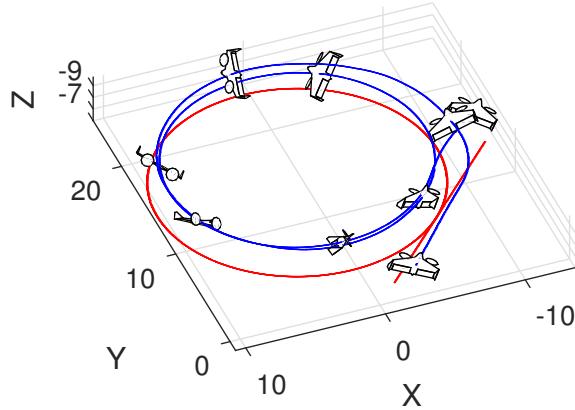


Figure 4–25: Banked Turn Trajectory. The reference is shown in red and the estimated trajectory is blue, the aircraft is not to scale.

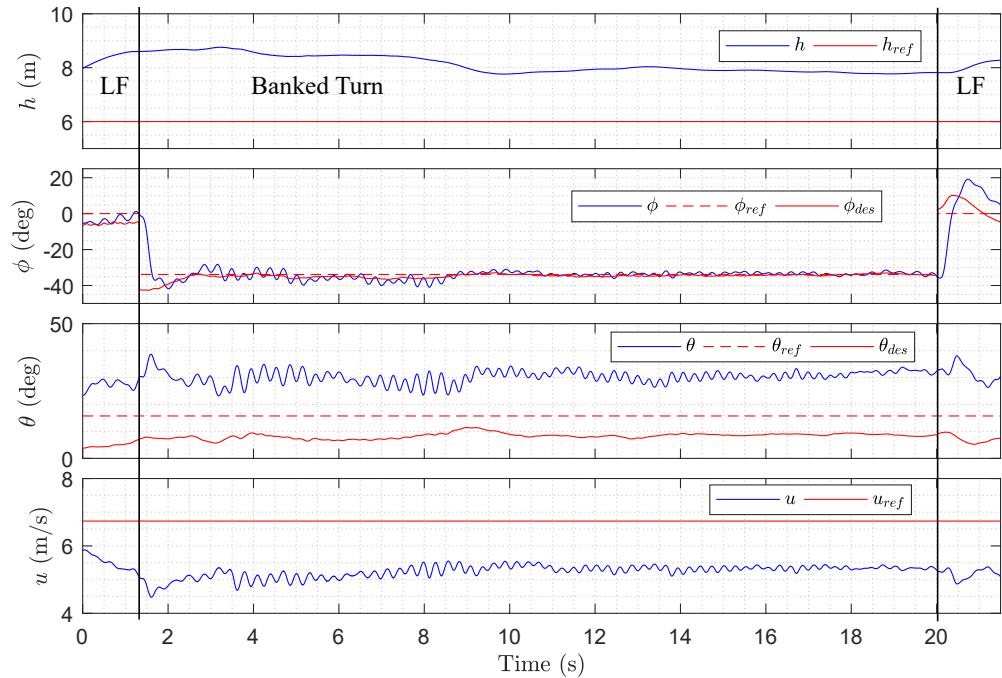


Figure 4–26: Banked Turn States. From top to bottom: roll angle ϕ reference ϕ_{ref} and desired ϕ_{des} , pitch angle θ reference θ_{ref} and desired θ_{des} , forward velocity u and reference u_{ref} . The flight phases are: the level flights LF and the banked turn.

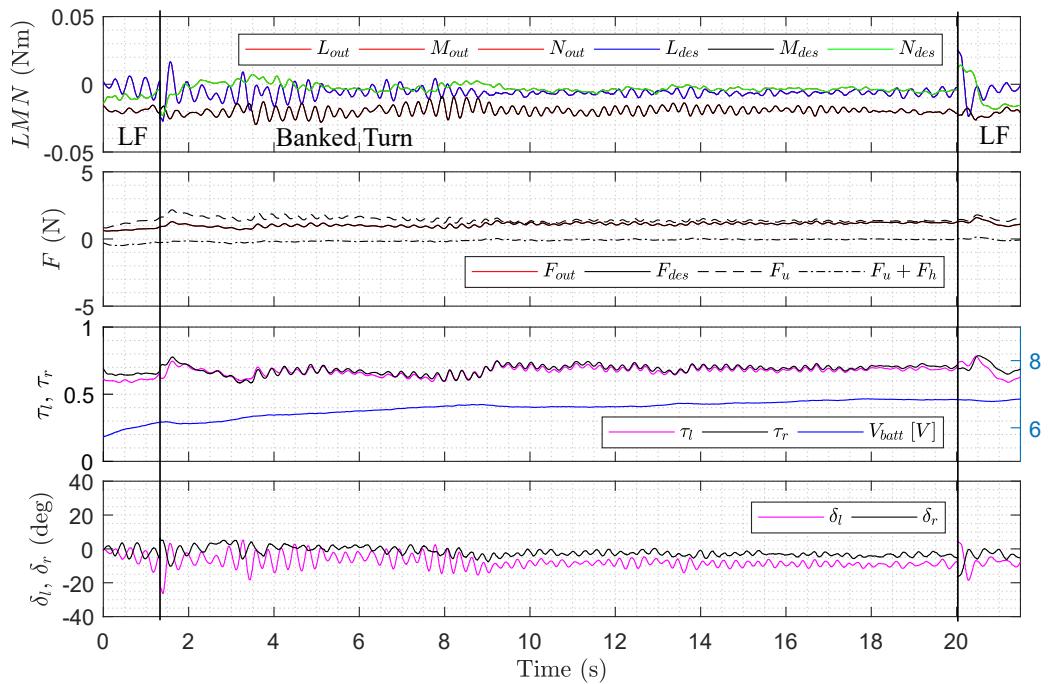


Figure 4–27: Banked Turn Controls. From top to bottom: desired moments L_{des} , M_{des} , N_{des} , desired forward force F_{des} , left and right throttles τ_l , τ_r and battery voltage V_{batt} , left and right elevon deflections δ_l , δ_r . The flight phases are: the level flights LF and the banked turn.

4.4.4 Aggressive Turnaround

To demonstrate the aerobatic ability of this platform and to test its limits, an aggressive turnaround maneuver was performed. The objective of this maneuver is to realize a U-turn as quickly as possible. To realize this maneuver, the reference commands the airplane to pitch as much as possible until the aircraft is upside down, aligned with the desired heading, then the aircraft is commanded to roll half a turn to return to its original level attitude. This maneuver is known as an Immelman turn and is the same maneuver as described in [14].

This maneuver starts from a level flight and is executed in three stages:

1. The maneuver starts similarly as a back transition, the vertical attitude is set as reference:

$$\mathbf{q}_{ref} = \mathbf{q}(\psi = \psi_0, \theta = \pi/2, \phi = 0) \quad (4.12)$$

where, ψ_0 is the initial heading from the level flight. The reference position is set as the last reference position from level flight:

$$\mathbf{p}_{ref} = \mathbf{p}_{\parallel}(t_{b,0}) \quad (4.13)$$

where $t_{b,0}$ is the time when the aggressive turnaround starts and \mathbf{p}_{\parallel} is described in Section 3.1.2.

2. When the aircraft's pitch angle is above 45° , the reference attitude is set as backward flight:

$$\mathbf{q}_{ref} = \mathbf{q}(\psi = \psi_0 + 180^\circ, \theta = 0, \phi = 180^\circ) \quad (4.14)$$

and the reference position is set as the projection on the level flight line:

$$\mathbf{p}_{ref} = \mathbf{p}_{\parallel} \quad (4.15)$$

3. Finally, when the pitch angle becomes lower than the reference pitch in flight velocity: $\theta < \theta_{lvl}$, the reference is set as level flight in opposite direction:

$$\mathbf{q}_{ref} = \mathbf{q}(\psi = \psi_0 + 180^\circ, \theta = \theta_{lvl}, \phi = 0) \quad (4.16)$$

and the reference position is kept as the projection:

$$\mathbf{p}_{ref} = \mathbf{p}_{\parallel} \quad (4.17)$$

During the three stages, as the pitch angle varies quickly, it is chosen to base the reference velocity on it:

$$u_{ref} = V \cos \theta \quad (4.18)$$

where V is the reference cruise velocity in level flight.

The recorded trajectory from a flight experiment is shown in Figure 4–28. The complete turnaround maneuver lasts 1.5 s, the distance needed is less than 3 meters and results in a drop of 1 meter. The saturation of the control signals on Figure 4–29 illustrate the large control action needed to realize this maneuver.

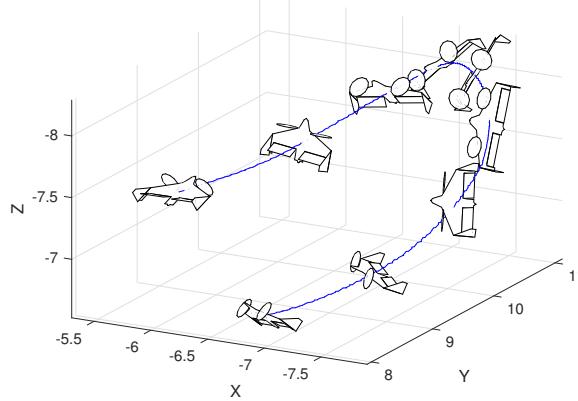


Figure 4–28: Aggressive Turnaround Trajectory. The aircraft is represented to scale. The complete maneuver lasts 1.5 s.

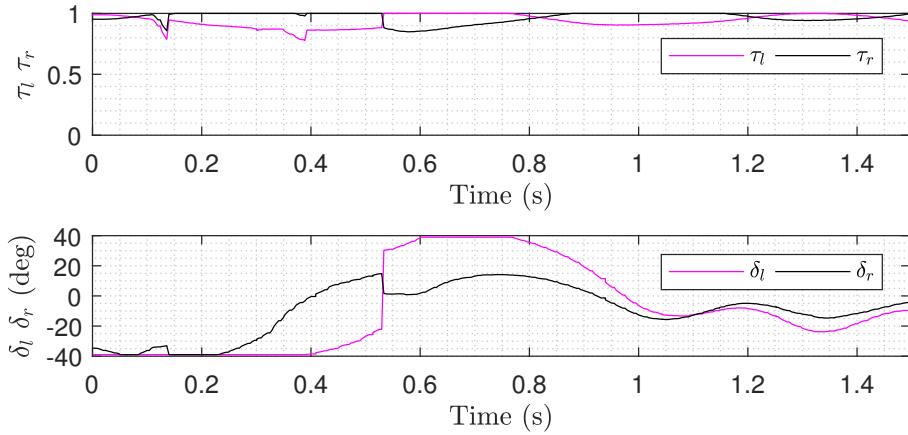


Figure 4–29: Aggressive Turnaround Control Commands.

4.5 Effect of the Center of Mass Location

The experimental results are mostly observed to match the simulated results except during the back transition in the minimal flight mission, which needs much more length and altitude in simulation than in real flight. This maneuver is strongly influenced by the pitching moment, which is itself strongly influenced by the position of the center of mass. This section reports the effect of moving the center of mass location in the simulated environment.

On a conventional aircraft, the tail is located far away from the center of mass. This remote location acts as a lever arm resulting in effective pitching moment control authority. By contrast, on a flying wing, this lever arm is very short and can even change sign with a small modification of the CM location. This has a strong effect on the pitching moment, which will have a substantial influence on the back transition.

The simulator was initially set with the CM set at a distance $d_x = 130$ [mm] from the trailing edge of the wings. The computed aerodynamic center is at $AC_x = 126.4$ [mm]. It is known that if the center of mass is in front of the aerodynamic center, the aircraft should be naturally stable [15]. To evaluate the effect of a more

rearward CM location, we choose to evaluate the case where, the CM is placed at $d_x' = 118$ [mm], i.e. 8 [mm] behind the aerodynamic center. Although the aircraft should be naturally unstable, it is expected that the controller should stabilize it. The effect of the center of mass location on the pitching moment coefficient is shown in Figure 4–30. It is observed that the aircraft can be maintained at equilibrium ($C_M = 0$) up to an AOA of $\alpha = 18^\circ$ with $\delta = \delta_{max}$, when the CM is located at $d_x = 130$ [mm] (blue solid line in Figure 4–30). When the CM is located at $d_x = 118$ [mm], the aircraft can be maintained at equilibrium up to 28° (red solid line in Figure 4–30).

On the same graph, the instability of the aircraft when the CM is located behind the aerodynamic center can be observed where the slope of the red curve becomes positive between $\alpha = -11^\circ$ and $\alpha = 11^\circ$.

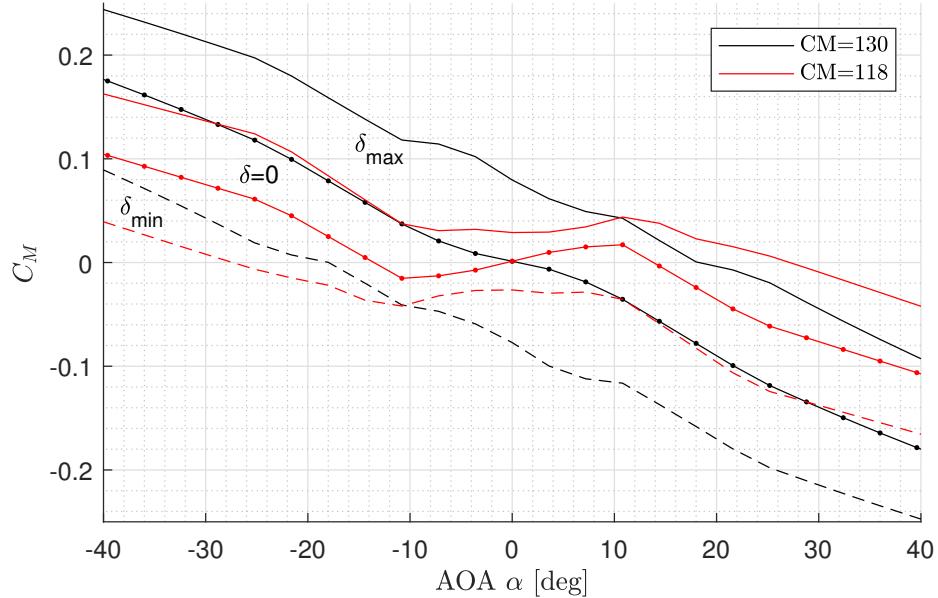


Figure 4–30: Pitching Moment Coefficient for different CM. The center of mass location is shown at a distance $d_x = 130$ [mm] in black and $d_x' = 118$ [mm] in red.

The back transitions for these two CM locations are shown in Figure 4–31. The maneuvers were realized in the simulator with the same algorithm and gains as in

Section 4.1. The aircraft enters the back transition from a level flight at 7 [m/s]. As expected the controller is able to stabilize the aircraft despite the unstable behavior of the natural pitching moment, in the case where the CM is located behind the aerodynamic center. The back transition requires 27 [m] horizontally and about 14 [m] in height when the CM is located at $d_x = 130$ [mm]. This is reduced to 13 [m] horizontally and 7 [m] in height when the CM is located at $d_x' = 118$ [mm]. This modification of 12 [mm] on the center of mass location reduces the back transition length to half of its original value. Although this modification of the CM location brings the simulation results closer to the real flight results, the real platform achieves even smaller stopping distance. Furthermore, it is unlikely that the CM location on the CAD model would be in error by more than a centimeter. While an inaccurate CM location may be contributing to the discrepancy between simulation and real flight, there are likely also other effects coming into play, such as the effect of the airfoil camber which has not been accounted for in the simulation.

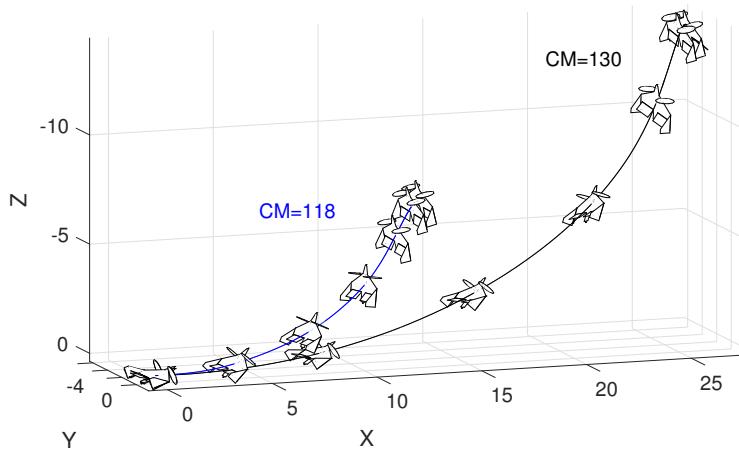


Figure 4–31: Back Transitions for different CM locations, in simulation. The center of mass location is shown at a distance $d_x = 130$ [mm] in black and $d_x' = 118$ [mm] in blue. Aircraft not to scale.

CHAPTER 5

Conclusion

5.1 Summary of Research

The research undertaken here on a tailsitter aircraft was separated into three main tasks. First the modeling of the chosen platform was done to develop a complete dynamic model, which was then implemented in a real time simulator. Secondly, a single controller was developed to be used on this platform, and finally, this controller was implemented in hardware to perform flight experiments.

The research started with the selection of a reliable platform. An off-the-shelf tailsitter aircraft was chosen based on its availability and on the modeling and control challenges associated with this system. Modeling began with the development of a detailed CAD model of the aircraft to accurately estimate the inertia and wing geometry. The simulator was then developed based on previous work in our laboratory [21]. The aerodynamic forces and moments were evaluated based on a component breakdown approach. The thrusters were modeled semi-empirically from static bench test measurements and extrapolations of data from a propeller database. Finally a simple yet effective ground contact model was developed. This dynamics model was incorporated into a previously developed hardware-in-the-loop environment [35].

The control strategy used a quaternion-based, single controller previously developed in our laboratory [14]. This controller was chosen for its proven ability to perform aerobatic maneuvers as well as its cascaded architecture that promotes reusability. A simplified aerodynamics model, to be used by the controller, was developed. This simplified model needed some coefficients for the control surface effectiveness, and these were measured on static bench tests as well as retrieved from

the simulator. Several iterations of the controller were realized and tested in the simulator, then on the hardware-in-the-loop environment, leading to performance improvements.

The hardware and sensor components necessary for fully autonomous flights were then selected and implemented on the tailsitter platform. The system was first tested inside the lab in a safe environment constrained by fishing lines. This setup allowed the controller to be tested in vertical maneuvers and the control gains to be tuned. Finally, flight experiments took place, in a GPS transparent environment, inside an air supported dome. These flights demonstrated the ability of the system to perform autonomously all the maneuvers necessary for a typical application, as well as aerobatic maneuvers.

5.2 Main Conclusions

The observations and comments of this section refer directly to the four objectives presented in the introduction:

1. Develop a full flight envelope, real-time simulator for the tailsitter.
2. Develop a single controller valid for all the flight modes.
3. Demonstrate the autonomous ability of the controller.
4. Perform qualitative evaluation of the simulator from flight data.

The first objective is partially fulfilled. The simulator developed for the tailsitter platform is indeed real-time and considers the full flight envelope aerodynamics of the aircraft, but the aerodynamic forces and moments do not accurately represent the real platform. This is observed through the three following points:

- Discrepancies are observed in particular for the back transition maneuver where the simulation predicted much greater distances needed to complete it. Two explanations are likely. First, the pitching moment generated by the control surfaces may be predicted too low, but this effect was partially measured by considering the effect of the slipstream over the control surfaces

while the aircraft was fixed on the force torque sensor. The second explanation is that the pitching moment of the aircraft at zero control surface deflection is predicted too large, which seems more likely. Firstly, the pitching moment can be overestimated due to the flat plate model used to predict the characteristics of the aircraft's wing. The tailsitter is equipped with a reflexed camber airfoil, which is commonly used in flying wing design to achieve close to zero pitching moment in the low angle of attack regime. Secondly, the location of the center of mass has a significant influence on the pitching moment. Unlike on a conventional aircraft that is equipped with a tail, the pitching moment on a flying wing is a function of the distance between the center of mass and the aerodynamic center, which is very small. Therefore a small modification of the center of mass location can produce large changes of the pitching moment. This was demonstrated in simulation in Section 4.5.

- Discrepancies are also observed during the banked turn, where the aircraft is observed to fly inside and above the reference trajectory while the simulator predicts the aircraft to fly outside and below it. A reasonable explanation is that the simulator underestimates the generated lift force, which is evaluated from the flat plate model.
- Another important point is that the aircraft cannot be stabilized during hover in the simulator with the same gains values used to stabilize the real platform. The gains need to be about twice as big to stabilize the system in simulation. Similarly, the aircraft could not be stabilized in hover in the hardware-in-the-loop simulation. This may also be due to an inaccurate aerodynamics model; or it may simply be due to computer hardware issues.

The second and third objectives are fulfilled, the developed controller was able to control the aircraft in simulation and was also able to control the real platform

demonstrated through flight experiments. The different maneuvers used for a complete flight mission such as vertical takeoff and climb, transition, level flight, banked turn, back transition, vertical descent and autonomous landing were demonstrated successfully both in simulation and in real flight. Furthermore, the real aircraft was able to perform aerobatic maneuvers beyond expectations, such as the aggressive turnaround and short back transition.

The performance evaluation of the simulator, although qualitative only, satisfies the last objective. While the performance can only be evaluated through recorded trajectories and attitude, the detailed flight variables such as lift force or pitching moment cannot easily be estimated from the flight tests.

5.3 Suggestions for Future Work

Two directions could be taken for future work. The first one would be to improve the simulator to be more representative of the real platform. For this purpose, wind tunnel tests will be necessary for validation of the current aerodynamic model. These tests could be used to retrieve the lift force, drag force, and pitching moment, as functions of the angle of attack and control surface deflection. Similar wind tunnel measurements were proposed in the literature [12], where the collected data were then curve fitted to create a quadratic model. The propeller characteristics could also be improved by developing an empirical model as a function of the inflow velocity instead of relying on a propeller database. This would also require access to a wind tunnel.

The other direction would be to focus on the real system, which has already proven its autonomous abilities. The maneuver generator could be extended to be more systematic, for example by generating a reference trajectory, given as a sequence of positions. The reference attitude could also be derived from the sequence of positions instead of being user specified. Solutions to this problem were proposed [11], such as keeping the aircraft in coordinated flight for instance.

The distance sensor used to measure the altitude during vertical flight could be fused into the extended Kalman filter to provide smoother estimated altitude and a more accurate rate of climb.

The developed platform could also be used to investigate particular research problems regarding tailsitters, such as study the ground effect or takeoff and landing phases. The ground effect is known to increase the thrust force, but on this platform, the propellers are located far enough from the ground, that this is negligible. However, just before landing, the elevons are located very close to the ground. Because the airflow separates when reaching the ground, this could have an impact on the quality of the airflow around the control surfaces and therefore influence their effectiveness near the ground. Another phenomenon occurs when the aircraft is in contact with the ground, a stuck condition can occur where the actuators cannot recover the aircraft from falling, due to the presence of contact forces. Studying these effects could allow a better understanding of the takeoff and landing phases and lead to safer procedures to accomplish them.

REFERENCES

- [1] N. Muchiri and S. Kimathi, “A review of applications and potential applications of uav,” in *Proceedings of Sustainable Research and Innovation Conference*, pp. 280–283, 2016.
- [2] W. F. Chana *et al.*, “World’s first vtol airplane convair/navy xfy-1 pogo,” tech. rep., SAE Technical Paper, 1996.
- [3] Life Inc, “New u.s. aim in the air,” *LIFE*, vol. 36, p. 182, March 22 1954.
- [4] J. Hamilton, *V-22 Osprey. A & D xtreme*, Abdo Group, 2013.
- [5] Horizon Hobby LLC, “X-VERT VTOL RTF.” <https://www.horizonhobby.com/x-vert-vtol-rtf-eft1800>. Accessed: 2018-03-28.
- [6] BirdsEyeView Aerobotics, “FireFLY6 PRO.” <https://www.birdseyeview.aero/products/firefly6>. Accessed: 2018-03-28.
- [7] V. Myrand-Lapierre, A. Desbiens, E. Gagnon, F. Wong, and É. Poulin, “Transitions between level flight and hovering for a fixed-wing mini aerial vehicle,” in *American Control Conference (ACC), 2010*, pp. 530–535, IEEE, 2010.
- [8] Y. Jung and D. H. Shim, “Development and application of controller for transition flight of tail-sitter uav,” *Journal of Intelligent & Robotic Systems*, vol. 65, no. 1, pp. 137–152, 2012.
- [9] P. Casau, D. Cabecinhas, and C. Silvestre, “Hybrid control strategy for the autonomous transition flight of a fixed-wing aircraft,” *IEEE Transactions on control systems technology*, vol. 21, no. 6, pp. 2194–2211, 2013.
- [10] N. B. Knoebel and T. W. McLain, “Adaptive quaternion control of a miniature tailsitter uav,” in *American Control Conference, 2008*, pp. 2340–2345, IEEE, 2008.
- [11] R. Ritz and R. D’Andrea, “A global controller for flying wing tailsitter vehicles,” in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pp. 2731–2738, IEEE, 2017.
- [12] S. Verling, B. Weibel, M. Boosfeld, K. Alexis, M. Burri, and R. Siegwart, “Full attitude control of a vtol tailsitter uav,” in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, pp. 3006–3012, IEEE, 2016.

- [13] R. Bapst, R. Ritz, L. Meier, and M. Pollefey, “Design and implementation of an unmanned tail-sitter,” in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pp. 1885–1890, IEEE, 2015.
- [14] E. Bulka and M. Nahon, “Automatic control for aerobatic maneuvering of agile fixed-wing uavs,” *Journal of Intelligent & Robotic Systems*, pp. 1–16, 2018.
- [15] J. Russell, *Performance and stability of aircraft*. Butterworth-Heinemann, 1996.
- [16] R. F. Stengel, *Flight dynamics*. Princeton University Press, 2015.
- [17] B. W. McCormick, *Aerodynamics, aeronautics, and flight mechanics*, vol. 2. Wiley New York, 1995.
- [18] J. D. Anderson Jr, *Fundamentals of aerodynamics*. Tata McGraw-Hill Education, 2010.
- [19] L. R. Lustosa, F. Defay, and J.-M. Moschetta, “Longitudinal study of a tilt-body vehicle: modeling, control and stability analysis,” in *Unmanned Aircraft Systems (ICUAS), 2015 International Conference on*, pp. 816–824, IEEE, 2015.
- [20] R. W. Beard and T. W. McLain, *Small unmanned aircraft: Theory and practice*. Princeton university press, 2012.
- [21] W. Khan, *Dynamics modeling of agile fixed-wing unmanned aerial vehicles*. Phd thesis, McGill University, 2016.
- [22] M. S. Selig, “Real-time flight simulation of highly maneuverable unmanned aerial vehicles,” *Journal of Aircraft*, vol. 51, no. 6, pp. 1705–1725, 2014.
- [23] W. Khan and M. Nahon, “Modeling dynamics of agile fixed-wing uavs for real-time applications,” in *Unmanned Aircraft Systems (ICUAS), 2016 International Conference on*, pp. 1303–1312, IEEE, 2016.
- [24] P. Yedamale, “Brushless dc (bldc) motor fundamentals,” *Microchip Application Notes*, vol. AN885, 2003.
- [25] A. L’Afflitto, *A Mathematical Perspective on Flight Dynamics and Control*. Springer, 2017.
- [26] W. Khan and M. Nahon, “Development and validation of a propeller slipstream model for unmanned aerial vehicles,” *Journal of Aircraft*, vol. 52, no. 6, pp. 1985–1994, 2015.
- [27] S. F. Hoerner, *Fluid-dynamic drag: practical information on aerodynamic drag and hydrodynamic resistance*. Hoerner Fluid Dynamics, 1965.

- [28] J. M. Seddon and S. Newman, *Basic helicopter aerodynamics*. John Wiley & Sons, 2011.
- [29] J.B. Brandt, R.W. Deters, G.K. Ananda, and M.S. Selig, “UIUC Propeller Database.” <http://m-selig.ae.illinois.edu/props/propDB.html>, 2005-2015.
- [30] C. P. R. Francisco Ramirez Fuentes, Marco Trujillo and Mendoza, “Controlling dc motors and servo motors,” *Freescale Semiconductor Application Notes*, vol. AN4251, 2010.
- [31] I. C. Cheeseman and W. E. Bennett., “The effect of the ground on a helicopter rotor in forward flight.,” Tech. Rep. 3021, 1957.
- [32] O. Shetty and M. Selig, “Small-scale propellers operating in the vortex ring state,” in *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, p. 1254, 2011.
- [33] L. Meier, D. Honegger, and M. Pollefeys, “Px4: A node-based multithreaded open source robotics framework for deeply embedded platforms,” in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pp. 6235–6240, IEEE, 2015.
- [34] B. Wie, H. Weiss, and A. Arapostathis, “Quaternions feedback regulator for spacecraft eigenaxis rotations,” *Journal of Guidance, Control, and Dynamics*, vol. 12, no. 3, pp. 375–380, 1989.
- [35] E. Bulk and M. Nahon, “Autonomous fixed-wing aerobatics: From theory to flight,” in *Accepted to 2018 IEEE International Conference of Robotics and Automation (ICRA)*, May 2018.
- [36] J. Postel, “User datagram protocol,” tech. rep., 1980.
- [37] K. Wilhelm, “Military standard-flying qualities of piloted aircraft,” tech. rep., MIL-STD-1797, US Air Force Department, 1990.
- [38] F. M. Hoblit, *Gust loads on aircraft: concepts and applications*. American institute of Aeronautics and Astronautics, 1988.
- [39] J. Sola, “Quaternion kinematics for the error-state Kalman filter,” *arXiv preprint arXiv:1711.02508*, 2017.
- [40] J. Diebel, “Representing attitude: Euler angles, unit quaternions, and rotation vectors,” *Matrix*, vol. 58, no. 15-16, pp. 1–35, 2006.

APPENDICES

A Inertia and Center of Mass

A complete Computer Aided Design model (CAD) of the tailsitter was realized on Solidworks to estimate the inertia matrix \mathbf{I} and the location of the center of mass CM, refer to Figure A.1. Each component was measured on an ACCULAB VI-1200 scale with a resolution of 0.1 g. A 3D model was then created for each component and the density was adjusted to fit the mass measured on the scale. Table A.1 lists the components of the tailsitter with their respective masses. The total mass from the CAD model is 92% of the total measured aircraft mass, so the CAD-derived values for I and CM location are expected to be representative of the real values. The difference between the measured mass the estimated mass from the CAD model is attributable to unmodeled components, such as the electric wires and the control rods. It is not surprising that the electric wires represents about 20 [g] on the total mass, as they are made of copper which is dense, and the thick cables on the power board are not modeled, nor their soldered connections.

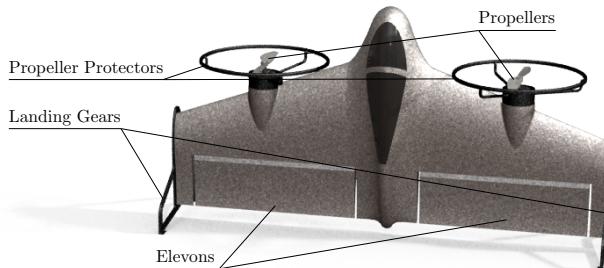


Figure A.1: CAD Model of the Tailsitter.

Component	Mass [g]	Quantity
Fuselage	52.2	1
Battery LiPo 450 mAh	31.2	1
Motor	19.8	2
GPS Module (without case)	14.0	1
Distance Sensor	9.4	1
Pixhawk mini (without case)	8.3	1
Landing gear	8.1	2
Power Module	7.2	1
Receiver	7.2	1
Propeller Protector	6.7	2
Servo	4.2	2
ESC	4.0	2
Splitter	3.0	1
Propeller	1.3	2
Total (CAD)	220.7	
Total (Scale), m	240.0	

Table A.1: Tailsitter Components List

The estimated center of mass in the geometric frame CM_G given by the CAD is:

$$CG_G = \begin{pmatrix} 128.34 \\ 1.10 \\ -3.44 \end{pmatrix} [mm] \quad (\text{A.1})$$

The estimated inertia matrix at the center of mass is given by:

$$\mathbf{I} = \begin{pmatrix} 2.986344 & -0.022259 & -0.012405 \\ -0.022259 & 0.637364 & 0.00121 \\ -0.012405 & 0.00121 & 3.487448 \end{pmatrix} \cdot 10^{-3} [kg\ m^2] \quad (\text{A.2})$$

B Wind Model

A wind model is integrated into the simulator to add realism and test the controller robustness. It generates 3D wind velocity components in the body frame $\mathbf{v}_{w,B}$. It is composed of a Dryden wind turbulent model summed to a mean horizontal wind component.

In a Dryden model, the wind is interpreted as linear transfer functions $H(s)$ filtering white noise to generate turbulent inertial velocity components $\mathbf{v}_{t,I} = [u_{t,I}, v_{t,I}, w_{t,I}]^T$ [37]:

$$H_u(s) = \sigma_u \sqrt{\frac{2L_u}{\pi V}} \frac{1}{1 + \frac{L_u}{V}s} \quad (\text{B.1})$$

$$H_v(s) = \sigma_v \sqrt{\frac{2L_v}{\pi V}} \frac{1 + 2\sqrt{3}L_v s}{\left(1 + \frac{2L_v}{V}s\right)^2} \quad (\text{B.2})$$

$$H_w(s) = \sigma_w \sqrt{\frac{2L_w}{\pi V}} \frac{1 + 2\sqrt{3}L_w s}{\left(1 + \frac{2L_w}{V}s\right)^2} \quad (\text{B.3})$$

Where $V = w_6$ is the velocity at an altitude of 6 [m], $\sigma_u, \sigma_v, \sigma_w$ are the turbulence intensities in [m/s] and L_u, L_v, L_w are the turbulence scale lengths in [m]. On the Simulink Dryden Block, those quantities are computed from the altitude h and the velocity at 6 [m] w_6 :

$$L_u = \frac{h}{(0.177 + 0.0027h)^{1.2}}, \quad L_v = L_u/2, \quad L_w = h/2 \quad (\text{B.4})$$

$$\sigma_w = \frac{w_6}{10}, \quad \sigma_u = \sigma_v = \frac{\sigma_w}{(0.177 + 0.0027h)^{0.4}} \quad (\text{B.5})$$

An evaluation of the wind model results was performed to compare the simulated wind spectrum to the expected Dryden spectrum, shown in Figure B.1. The recorded wind velocity time history was converted to a power spectrum using a Discrete Fourier Transform (DFT). The theoretical spectrum is plotted with the simulated spectrum showing good agreement.

The theoretical spectrum is obtained by the following formula [37]:

$$\Phi_u(\Omega) = \sigma_u^2 \left(\frac{2L_u}{\pi} \right) \frac{1}{1 + (L_u \Omega)^2} \left[\frac{(m/s)^2}{rad/m} \right] \quad (\text{B.6})$$

where Ω is the spatial frequency in $[rad/m]$.

Because the units of the power spectrum are dependent of the unit of the variable, changing the variable will modify the power spectrum as follows [38]:

$$\Phi_u(f) = \frac{2\pi}{V} \Phi_u(\Omega) = \sigma_u^2 \left(\frac{4L_u}{V} \right) \frac{1}{1 + \left(\frac{L_u 2\pi f}{V} \right)^2} \left[\frac{(m/s)^2}{Hz} \right] \quad (\text{B.7})$$

where f is the frequency in $[Hz]$.

The power spectrum using angular frequency ω in $[rad/s]$ would be:

$$\Phi_u(\omega) = \frac{1}{V} \Phi_u(\Omega) = \sigma_u^2 \left(\frac{2L_u}{\pi V} \right) \frac{1}{1 + \left(\frac{L_u \omega}{V} \right)^2} \left[\frac{(m/s)^2}{rad/s} \right] \quad (\text{B.8})$$

The variables are related as follows:

$$\Omega V = 2\pi f = \omega \quad (\text{B.9})$$

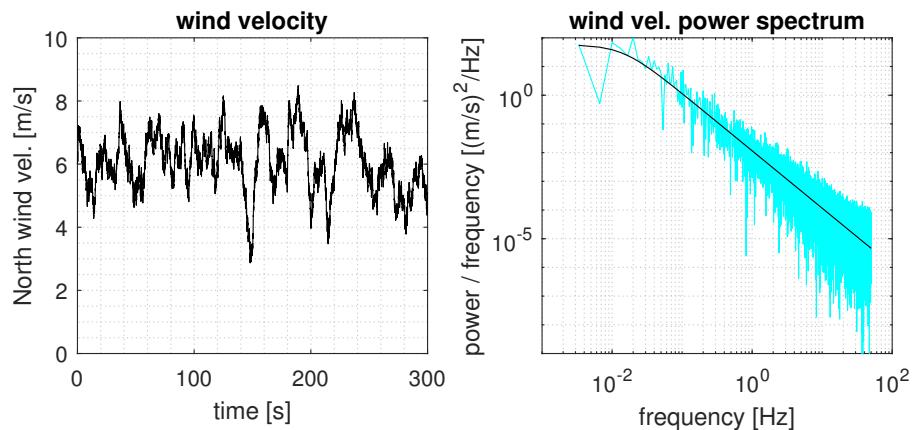


Figure B.1: Wind speed time history at $h = 10 [m]$ and $V = 6 [m/s]$. Its corresponding power spectrum is shown on the right, the plain line is the theoretical spectrum.

The mean wind velocity $\mathbf{v}_{m,I}$ is computed as an horizontal wind making an angle ψ_w from the North axis:

$$\mathbf{v}_{m,I} = V \begin{bmatrix} \cos \psi_w \\ \sin \psi_w \\ 0 \end{bmatrix} \quad (\text{B.10})$$

The wind velocity is finally the sum of the mean and turbulent components and can be translated into the body frame as:

$$\mathbf{v}_{w,B} = \mathbf{R}(q)^T(\mathbf{v}_{m,I} + \mathbf{v}_{t,I}) \quad (\text{B.11})$$

C Full Flight Envelope Simulator Measurements

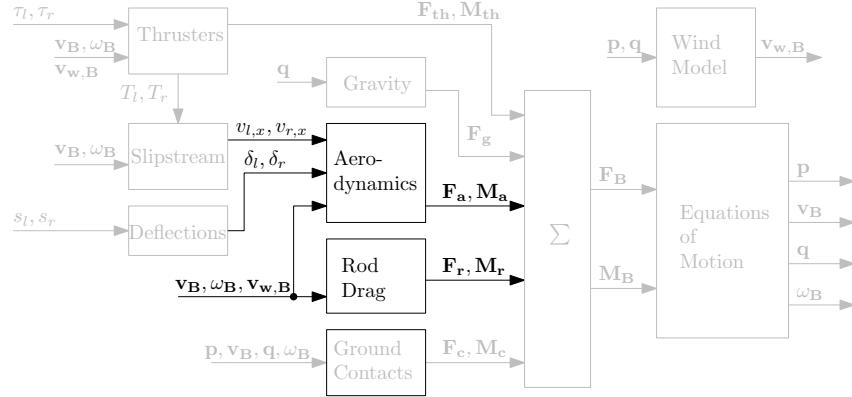


Figure C.1: Aerodynamics and Rod Drag Blocks of the Simulator.

This section reports how the full flight envelope data points were obtained from the simulator to generate the curves in Figures 2–5, 2–6, 2–22 and 2–23. For this purpose, the Aerodynamics and Rod Drag Blocks were considered as shown in Figure C.1. The inputs were set to $\omega_B = \mathbf{0}$, $v_{w,B} = \mathbf{0}$, $v_{l,x} = 0$, $v_{r,x} = 0$, and $v_B = [\cos \alpha, 0, \sin \alpha]^T$ which correspond to a unit forward speed of the aircraft. In that case, the inputs that can be modified are δ_l , δ_r and α . The forces and moments are summed and the outputs are then \mathbf{F}_o and \mathbf{M}_o :

$$\begin{aligned}\mathbf{F}_o &= \mathbf{F}_a + \mathbf{F}_r \\ \mathbf{M}_o &= \mathbf{M}_a + \mathbf{M}_r\end{aligned}\tag{C.1}$$

The measurements were realized by varying the angle of attack α over its range $[-\pi; \pi]$ at 101 linearly spaced data points α_i . The lift and drag forces were obtained as:

$$\begin{aligned}L_i &= F_{o,x,i} \sin(\alpha_i) - F_{o,z,i} \cos(\alpha_i) \\ D_i &= -F_{o,x,i} \cos(\alpha_i) - F_{o,z,i} \sin(\alpha_i);\end{aligned}\tag{C.2}$$

The lift and drag coefficients data points were the obtained as:

$$\begin{aligned}C_{L,i} &= \frac{L_i}{\frac{1}{2}\rho(1)^2 S} \\ C_{D,i} &= \frac{D_i}{\frac{1}{2}\rho(1)^2 S}\end{aligned}\tag{C.3}$$

where ρ is the air density and S is the wing area.

The rolling and pitching moments were simply obtained as the first and second component of \mathbf{M}_o . The rolling and pitching moments coefficients data points were obtained as:

$$\begin{aligned} C_{X,i} &= \frac{M_{o,x}}{\frac{1}{2}\rho(1)^2 S b} \\ C_{M,i} &= \frac{M_{o,y}}{\frac{1}{2}\rho(1)^2 S \bar{c}} \end{aligned} \quad (\text{C.4})$$

where b is the span and \bar{c} is the mean aerodynamic chord of the wing.

Regarding the elevon deflections, different cases were applied:

- To obtain the curves at zero deflection, the elevons applied were set to $\delta_l = \delta_r = 0$.
- To obtain the curves at minimum deflection, the elevons applied were set to $\delta_l = \delta_r = \delta_{min}$.
- To obtain the curves at maximum deflection, the elevons applied were set to $\delta_l = \delta_r = \delta_{max}$.

D Pitching Moment at Zero Deflection

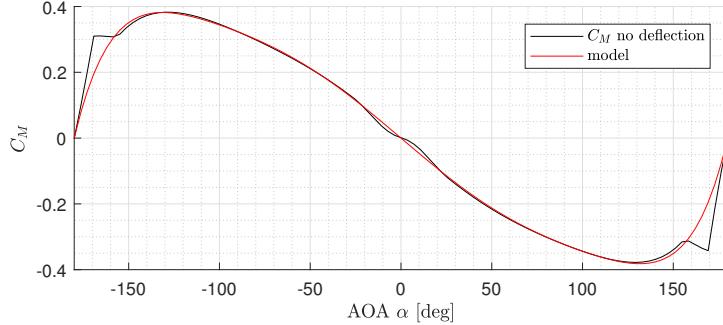


Figure D.1: Model of the Pitching Moment Coefficient at zero Deflection.

The estimated pitching moment at zero deflection \hat{M}_0 is used in the simplified aerodynamic model in equation 2.41, itself used by the actuator mixer. It is chosen to model \hat{M}_0 as a light function to ensure fast computation in the control loop. \hat{M}_0 is defined as a function of the estimated pitching moment coefficient \hat{C}_M :

$$\hat{M}_0 = k_m \frac{1}{2} \rho (u^2 + w^2) S \bar{c} \hat{C}_M(\alpha) \quad (\text{D.1})$$

where ρ is the air density, u and w are the aircraft velocities along the x_B and the z_B axes, $S = 0.08 \text{ [m}^2]$ is the wing area, $\bar{c} = 0.17 \text{ [m]}$ is the mean aerodynamic chord of the flying wing, α is the angle of attack, and k_m is a gain used in the controller only to scale the pitching moment for possible discrepancies between the real system and the simulator. As the other gains on the controller, it was tuned in flight, and a value of $k_m = 0.8$ gave satisfactory results.

The pitching moment coefficient of the entire aircraft at zero deflection C_M is shown in Figure 2–6, and its evaluation is described in Appendix C. It is chosen to estimate it by an 7th order polynomial, odd function, having roots in $\alpha = \pm \pi$ and $\alpha = 0$, as shown in Figure D.1:

$$\hat{C}_M(\alpha) = (5.18 \cdot 10^{-4} \alpha^5 - 1.03 \cdot 10^{-3} \alpha^3 + 2.72 \cdot 10^{-2} \alpha)(\alpha - \pi)(\alpha + \pi) \quad (\text{D.2})$$

where α is in $[rad]$ and \hat{C}_M is unitless.

E Inverse Thruster Model

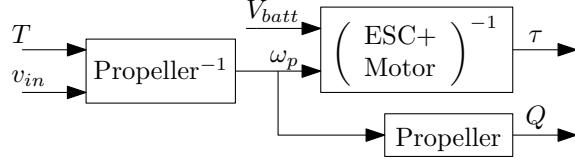


Figure E.1: Block Diagram of the Inverse Thruster Model.

During the computation of the actuator mixer, the inverse relation of the thruster is needed to predict the throttle signal τ in order to achieve the desired thrust T . As shown in Figure E.1, the inputs of the inverse thruster model are the thrust force T and the inflow velocity v_{in} . The output of the first block is the angular speed of the propeller ω_p . The first function to find is $\omega_p(T, v_{in})$. For this purpose, the thrust equation in 2.32 is considered, along with the curve fitted thrust coefficient in equation 2.47, and the advance ratio J given by equation 2.30:

$$\begin{aligned} T &= \frac{4}{\pi} \rho \omega_p^2 r_p^4 C_T(J) \\ C_T(J) &= C_{T,2} J^2 + C_{T,1} J + C_{T,0} \\ J &= \frac{\pi v_{in}}{\omega_p r_p} \end{aligned} \tag{E.1}$$

These can be combined into a second order equation for the angular speed ω_p , if we assume a fixed v_{in} :

$$\underbrace{C_{T,0} r_p^2}_{a} \omega_p^2 + \underbrace{C_{T,1} \pi v_{in} r_p}_{b} \omega_p + \underbrace{C_{T,2} \pi^2 v_{in}^2 - \frac{T \pi^2}{4 \rho r_p^2}}_{c} = 0 \tag{E.2}$$

This can be solved for ω_p as $\omega_p = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$, giving:

$$\omega_p = \frac{\pi}{2 C_{T,0} r_p} \left[\sqrt{C_{T,1}^2 v_{in}^2 + 4 C_{T,0} \left(\frac{T}{4 \rho r_p^2} - C_{T,2} v_{in}^2 \right)} - C_{T,1} v_{in} \right] \tag{E.3}$$

Because $C_{T,0} > 0$, $C_{T,1} < 0$ and $C_{T,2} < 0$, the term under the square root is always positive and ω_p is always positive too. In the derivation of the thruster forward model, in Section 2.3.6, a lower bound of zero was set on v_{in} to ensure a positive

advance ratio J . The thrust force T is also defined positive.

Then, the desired propeller rotational speed ω_p is bounded to stay in the feasible range. Once the propeller speed ω_p is known, the torque Q can be computed using equation 2.32.

The second part of the inverse thruster model is to predict the throttle signal τ knowing the propeller rotational speed ω_p and the battery voltage V_{batt} , which can be measured from the power module during the flight. This consists of inverting Equation 2.43:

$$\underbrace{\omega_{p,2}}_a \tau^2 + \underbrace{\omega_{p,1}}_b \tau + \underbrace{\omega_{p,0} - \frac{\omega_p}{V_{batt}^{0.8}}}_c = 0 \quad (\text{E.4})$$

and can be done using the root of a quadratic as before:

$$\tau = \frac{-\omega_{p,1} + \sqrt{\omega_{p,1}^2 - 4\omega_{p,2}\left(\omega_{p,0} - \frac{\omega_p}{V_{batt}^{0.8}}\right)}}{2\omega_{p,2}} \quad (\text{E.5})$$

Because the desired propeller angular speed ω_p was previously bounded to its feasible range, this equation always yields a real root.

F Hardware Components

The chosen platform is a commercial tailsitter used by hobbyists. It can be bought ready to fly and comes with an attitude and altitude controller, based on an onboard Inertial Measurement Unit (IMU) and barometer. This controller is able to stabilize the system during hover and level flight and the whole system weighs less than 200 [g]. Because the source code of this flight controller is not available, it is chosen to replace the flight controller by a Pixhawk Mini and other necessary hardware components. The block diagram of the hardware components is shown in Figure F.2 and a picture of the aircraft carrying the hardware is shown in Figure F.1.

The Lithium Polymer (LiPo) battery is a high efficiency battery that produces a voltage of around 7.4 [V] which can vary depending on the current delivered and the battery level. The power module is an electrical board that transmits the battery voltage to the ESCs and generates a regulated 5 [V] voltage for the low voltage components. It also provides battery current and voltage measurement signals for the Pixhawk Mini. The radio receiver is paired with the radio transmitter and provide a remote communication link from the user to the platform. The Global Positioning System (GPS) sensor provides information about the position of the vehicle. It works by reading radio signals emitted by a constellation of satellites. The compass

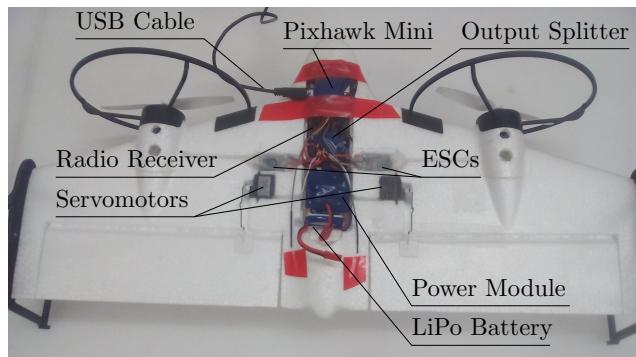


Figure F.1: Aircraft Hardware. The GPS is below the Pixhawk Mini and the radio receiver.

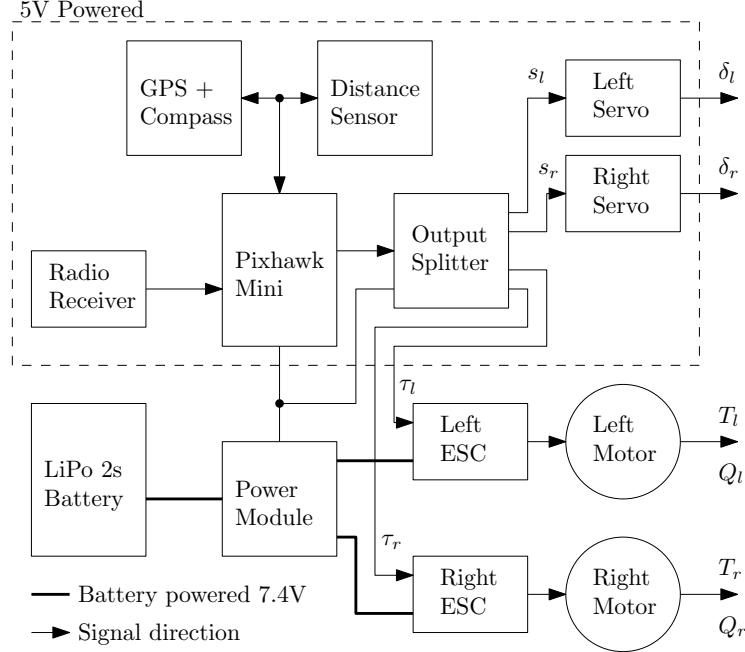


Figure F.2: Hardware Wiring Diagram.

is a sensor that provides heading information based on the Earth's magnetic field. The distance sensor is laser-based rangefinder and used to measure the aircraft's altitude, and is described further. The Pixhawk Mini is the microcontroller board. Three additional sensors are embedded inside it, the Inertial Measurement Unit (IMU), composed of a 3-axis accelerometer to measure acceleration and a 3-axis gyroscope to measure angular rates, a barometer to estimate the altitude, and another compass. On it runs an Extended Kalman Filter (EKF) as well as the controller. The EKF estimates the states of the system, namely \mathbf{p} , \mathbf{v}_B , \mathbf{q} and $\boldsymbol{\omega}_B$, from the raw sensor measurements. The controller generates actuator signals to achieve the desired behavior, based on the estimated states. These actuator signals are sent to the output splitter which separates them. The first two signals, s_l and s_r are the normalized servomotor signals ranging from -1 to 1 . A servomotor (servo) is an electrical actuator that is able to hold an angular position. The servomotor is linked to the control surface by a control rod, which creates the elevon deflections δ_l and δ_r . The other two signals are the left and right throttles τ_l and τ_r , ranging from

0 to 1, which are fed through the ESCs to generate the thrust forces T_l and T_r as explained in Section 2.1. The hardware components are listed in Table F.1 along with their designation.

Component	Designation
Battery	LiPo 450mAh 2S 7.4V 30C
Power Module	Quad Power Distribution Board
Radio Receiver	Futaba R6303SB
Radio Transmitter	Futaba T7C 2.4GHz
GPS	ublox Neo-M8N
Distance Sensor	Lightware SF20
Microcontroller Board	3DR Pixhawk Mini
Output Splitter	8 Channel PWM Breakout Board
Servomotor	SPMSA220
ESC	BeeRotor BS20A
Electric motor	BL280 2600KV
Propeller	125x75mm

Table F.1: Hardware List

Sensor Limitations

The states of the system are estimated from onboard sensors measurements by the EKF running on the onboard computer. The sensors fused are:

- The IMU composed of a 3-axis accelerometer to measure acceleration and a 3-axis gyroscope to measure angular rates. The accelerometer also provides information about the aircraft attitude by measuring the direction of gravity acceleration g .
- The compass, is a 3-axis magnetometer to estimate the heading based on Earth magnetic field.
- GPS to estimate the position.
- Barometer to estimate the altitude.

The EKF of the PX4 flight stack is used as a black box and the estimated states are assumed to be accurate enough to be treated as the real ones.

An experiment was carried out to estimate the inaccuracy of the position estimation, refer to Figure F.3. For this experiment, the aircraft is placed on the ground and

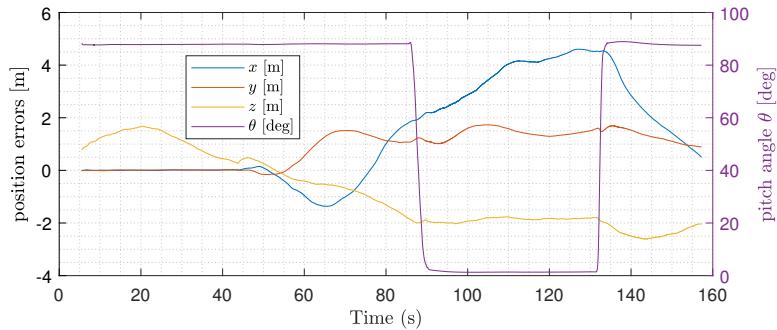


Figure F.3: EKF Estimated Position during vertical and horizontal attitude.

left static. The EFK is run and the position is recorded for about a minute. The position recorded in this static experiment is in fact the error of the EKF estimations. The x and y errors are due to the GPS inaccuracies and the z errors are due to the barometer inaccuracies. Furthermore, the aircraft attitude was changed from vertical to horizontal at $t = 90$ [s] and back to vertical at $t = 130$ [s], to check if the attitude has an impact on the GPS measurements. This hypothesis come from the fact that the GPS sensor reads radio waves from the satellite constellation through a receiving antenna. It is assumed that this antenna is directional in the sense that signal reception power is orientation-dependent. The GPS sensor is mounted flat inside the aircraft body, parallel to the wings. We observe position inaccuracy up to 6 meters and the effect of the GPS sensor orientation on the position estimation seems minimal. We also observe altitude inaccuracy up to 4 meters. The position inaccuracy is not much of a problem because the drift is much slower than the aircraft dynamics, and should have no effect on the controller stability. The altitude inaccuracy is a bigger problem, as the aircraft needs to perform autonomous landing. We therefore chose to mount a distance sensor on the aircraft along its longitudinal axis and pointing to the tail, refer to Figure F.4. During vertical maneuvers, the distance measured will provide information about the altitude, assuming a flat ground.

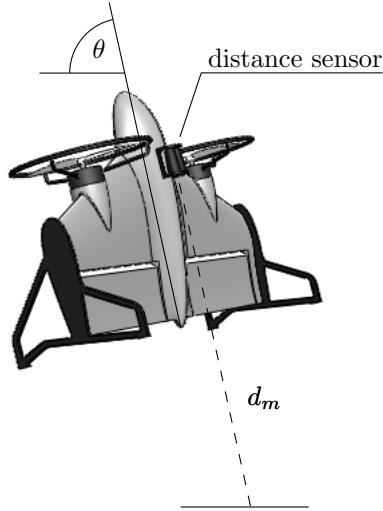


Figure F.4: Distance Sensor Operational Diagram to Estimate the Altitude.

In this system, the relative altitude $h = -z$ can be retrieved by the distance d_m measured by the sensor by:

$$h = d_m \sin \theta \quad (\text{F.1})$$

where θ is the pitch angle estimated by the EKF. This method allows retrieval of the altitude only during vertical maneuvers. It is chosen to use this sensor to correct the altitude when the pitch angle is greater than 60° and to rely on the estimation from the barometer for smaller pitch angle.

G Quaternion Definitions and Relations

All the following quaternion definitions and relations are taken from [39] and [40]. A unit quaternion is a mathematical object having a scalar part q_0 and a vector part \mathbf{q}_v apposed over a unit imaginary orthonormal basis.

$$\mathbf{q} = q_0 + q_1 i + q_2 j + q_3 k = q_0 + \mathbf{q}_v^T \begin{bmatrix} i \\ j \\ k \end{bmatrix} \quad (\text{G.1})$$

where i, j, k are unit imaginary numbers defined as:

$$ij = -ji = k, \quad jk = -kj = i, \quad ki = -ik = j \quad (\text{G.2})$$

$$i^2 = j^2 = k^2 = ijk = -1 \quad (\text{G.3})$$

Furthermore, the Euclidean norm of the unit quaternion must be 1 at all time:

$$\|\mathbf{q}\| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} = 1 \quad (\text{G.4})$$

A quaternion \mathbf{q} can be represented as a 4-dimensional vector:

$$\mathbf{q} = \begin{bmatrix} q_0 \\ \mathbf{q}_v \end{bmatrix} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \quad (\text{G.5})$$

where \mathbf{q}_v is the vector part.

The rotation of an angle Θ about an axis given by a unit vector \mathbf{u} ($\|\mathbf{u}\| = 1$) can be translated into a quaternion as:

$$\mathbf{q} = \begin{bmatrix} \cos(\Theta/2) \\ \mathbf{u} \sin(\Theta/2) \end{bmatrix} \quad (\text{G.6})$$

It follows that a quaternion \mathbf{q} and its negation $-\mathbf{q}$ describe the same rotation, i.e. a rotation Θ about \mathbf{u} equals a rotation $-\Theta$ about $-\mathbf{u}$.

Inversely, a quaternion \mathbf{q} can be translated into a rotation described by an angle Θ about a unit vector \mathbf{u} as:

$$\Theta = 2 \cos^{-1}(q_0), \quad \mathbf{u} = \frac{\mathbf{q}_v}{\|\mathbf{q}_v\|} \quad (\text{G.7})$$

The inverse rotation of a unit quaternion \mathbf{q} is given by its conjugate:

$$\mathbf{q}^{-1} = \mathbf{q}^* = \begin{bmatrix} q_0 \\ -\mathbf{q}_v \end{bmatrix} \iff \mathbf{q}^* \otimes \mathbf{q} = 1 \quad (\text{G.8})$$

i.e. this is the rotation of an angle $-\Theta$ about a unit vector \mathbf{u} .

The identity rotation is given by the quaternion:

$$\mathbf{q} = \begin{bmatrix} 1 \\ \mathbf{0} \end{bmatrix} \quad (\text{G.9})$$

The quaternion product \otimes of two quaternions \mathbf{a} and \mathbf{q} represents the rotation by \mathbf{a} followed by a rotation by \mathbf{q} and is defined as:

$$\mathbf{a} \otimes \mathbf{q} = \begin{bmatrix} a_0 q_0 - \mathbf{a}_v^T \mathbf{q}_v \\ a_0 \mathbf{q}_v + q_0 \mathbf{a}_v + \mathbf{a}_v \times \mathbf{q}_v \end{bmatrix} \quad (\text{G.10})$$

where \times is the 3D cross product which can be defined as a matrix product using a skew symmetric matrix $[\bullet]_\times$:

$$\mathbf{a} \times \mathbf{b} = [\mathbf{a}]_\times \mathbf{b} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \quad (\text{G.11})$$

notice that the quaternion product is non-commutative due to the cross product term:

$$\mathbf{a} \otimes \mathbf{q} \neq \mathbf{q} \otimes \mathbf{a} \quad (\text{G.12})$$

Let's consider a 3D vector \mathbf{v} in the inertial frame, its representation in the body frame \mathbf{v}' is given by:

$$\begin{bmatrix} 0 \\ \mathbf{v}' \end{bmatrix} = \mathbf{q} \otimes \begin{bmatrix} 0 \\ \mathbf{v} \end{bmatrix} \otimes \mathbf{q}^* \quad (\text{G.13})$$

The 3 by 3 rotation matrix can be retrieved from the unit quaternion as:

$$\mathbf{R}(\mathbf{q}) = (q_0^2 - \mathbf{q}_v^T \mathbf{q}_v) \mathbf{I}_3 + 2\mathbf{q}_v \mathbf{q}_v^T + 2q_0 [\mathbf{q}_v]_\times \quad (\text{G.14})$$

where \mathbf{I}_3 is the 3D identity matrix.

The quaternion can be retrieved from standard yaw-pitch-roll (ψ, θ, ϕ) Euler angles as:

$$\mathbf{q}(\psi, \theta, \phi) = \begin{bmatrix} \cos \psi/2 \\ 0 \\ 0 \\ \sin \psi/2 \end{bmatrix} \otimes \begin{bmatrix} \cos \theta/2 \\ 0 \\ \sin \theta/2 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} \cos \phi/2 \\ \sin \phi/2 \\ 0 \\ 0 \end{bmatrix} \quad (\text{G.15})$$

The Euler angles in sequence yaw-pitch-roll can be retrieved from unit quaternion as:

$$\begin{aligned} \phi &= \text{atan2}(2q_2q_3 + 2q_0q_1, q_3^2 - q_2^2 - q_1^2 + q_0^2) \\ \theta &= \text{asin}(2q_0q_2 - 2q_1q_3) \\ \psi &= \text{atan2}(2q_1q_2 + 2q_0q_3, q_0^2 + q_1^2 - q_2^2 - q_3^2) \end{aligned} \quad (\text{G.16})$$

H Aircraft Parameters

Parameter	Symbol	Value	Unit
Mass	m	0.21	kg
Moments of Inertia	I_x	$3.002 \cdot 10^{-3}$	$kg\ m^2$
	I_y	$6.245 \cdot 10^{-4}$	$kg\ m^2$
	I_z	$3.538 \cdot 10^{-3}$	$kg\ m^2$
Non-zero Products of Inertia	I_{xz}	$-14.03 \cdot 10^{-6}$	$kg\ m^2$
Wing Area	S	0.0798	m^2
Wing Span	b	0.5	m
Wing Aspect Ratio	A_R	3.13	
Winglet Aspect Ratio	$A_{R,t}$	0.217	
Control Derivative Coefficients	c_x	$9.91 \cdot 10^{-4}$	m^3/rad
	c_y	$4.74 \cdot 10^{-4}$	m^3/rad
	b_x	$9.37 \cdot 10^{-4}$	m^3/rad
	b_y	$3.48 \cdot 10^{-4}$	m^3/rad
Maximum Elevons Deflection	δ_{max}	39	deg
Propeller Radius	r_p	62.5	mm
Thruster Rotational Inertia	I_{th}	$1.626 \cdot 10^{-6}$	$kg\ m^2$
Thrusters Lateral Position	l	145	mm
CM Position from Trailing Edge	d_x	130	mm
Battery Voltage	V_{batt}	7.4	V
Delta Wing Sweep Angle	Λ	19.8	deg
Skin Friction Coefficient	$C_{D,0}$	0.02	
Oswald's Efficiency Factor	k_0	0.87	

Table H.1: Aircraft Properties

Parameter	Symbol	Sim.	Exp.	Unit
Position Proportional Gain	k_{pp}	0.05	0.06	rad/m
Position Derivative Gain	k_{pd}	0.1	0.1	$rad \cdot s/m$
Attitude Proportional Gain along x	k_{ap_L}	500	30	$1/s^2$
Attitude Proportional Gain along y	k_{ap_M}	500	155	$1/s^2$
Attitude Proportional Gain along z	k_{ap_N}	500	65	$1/s^2$
Attitude Derivative Gain along x	k_{ad_L}	60	4	$1/s$
Attitude Derivative Gain along y	k_{ad_M}	60	5	$1/s$
Attitude Derivative Gain along z	k_{ad_N}	60	7	$1/s$
Speed Proportional Gain	k_{up}	8	4	$1/s$
Height Proportional Gain	k_{hp}	18	6	$1/s^2$
Minimum slipstream velocity	$v_{s,min}$	8	7	m/s
Pitching Moment Scaling Gain	k_m	1	0.8	

Table H.2: Controller Gains