Adaptive Control of the Transition from Vertical to Horizontal Flight Regime of a Quad-Tailsitter UAV

Grant I. Carter

Thesis submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

Masters of Science

in

Mechanical Engineering

Andrea L'Afflitto, Chair

Alexander Leonessa

Pinhas Ben-Tzvi

May 12th, 2021

Blacksburg, Virginia

Keywords: VTOL UAV, Model Reference Adaptive Control (MRAC), tailsitter, drone

Adaptive Control of the Transition from Vertical to Horizontal Flight Regime of a Quad-Tailsitter UAV

Grant I. Carter

(ABSTRACT)

Tailsitter UAVs (Unmanned Aerial Vehicles) are a type of VTOL (Vertical Take off and Landing) aircraft that combines the agility of a quadrotor drone with the endurance and speed of a fixed-wing aircraft. For this reason, they have become popular in a wide range of applications from tactical surveillance to parcel delivery. This thesis details a clean sheet design process for a tailsitter UAV that includes the dynamic modeling, control design, simulation, vehicle design, vehicle prototype fabrication, and testing of a tailsitter UAV. The goal of this process was to design a robust controller that is able to handle uncertainties in the system's parameters and external disturbances and subsequently can control the vehicle through the transition between vertical and horizontal flight regimes. It is evident in the literature that most researchers choose to model and control tailsitter UAVs using separate methods for the vertical and horizontal flight regimes and combine them into one control architecture. The novelty of this thesis is the use of a single dynamical model for all flight regimes and the robust control technique used. The control algorithm used for this vehicle is a MRAC (Model Reference Adaptive Control) law, which relies on reference models and gains that adapt according to the vehicle's response in all flight regimes. To validate this controller, numerical simulations in Matlab and flight tests were conducted. The combination of these validation methods confirms our adaptive controller's ability to control the transition between the vertical and horizontal flight regimes when faced with both parametric uncertainties and external disturbances.

Adaptive Control of the Transition from Vertical to Horizontal Flight Regime of a Quad-Tailsitter UAV

Grant I. Carter

(GENERAL AUDIENCE ABSTRACT)

Unmanned aircrafts have been a topic of constant research and development recently due to their wide range of applications and their ability to fly without directly involving pilots. More specifically, VTOL UAVs have the advantage of being able to take off without a runway while retaining the efficiency of a classical aircraft. A tailsitter UAV behaves as a traditional quadrotor drone when in its vertical configuration and can rotate to a horizontal configuration, where it takes advantage of its wings to fly as a conventional aircraft. Modeling the dynamics of the tailsitter UAV and designing an autopilot controller is the main focus of this thesis. An adaptive controller was chosen for the tailsitter UAV due to its ability to modify the gains of the system based on the behavior of the vehicle to adapt to the unknown vehicle properties. This controller was validated using both computer simulations and actual flight tests. It was found that the adaptive controller was able to successfully control the transition between the vertical and horizontal flight regimes despite the uncertainties in the parameters of the vehicle.

Acknowledgments

First and foremost, I would like to thank my advisor, Dr. L'Afflitto, for the guidance throughout this project. I owe a debt of gratitude to Dr. L'Afflitto and all the rest of the professors at Virginia Tech who have taught me over the past five years. I would also like to thank my fiancé Danielle and all of my family for their continued support through this process. Additionally, this work has been funded by the Center for Unmanned Aircraft Systems (C-UAS), a National Science Foundation Industry/University Cooperative Research Center (I/UCRC) under NSF award No. CNS-1650465 along with significant contributions from C-UAS industry members and the US Department of the Navy, Grant no. N004212110004

Contents

List of Figures

List of Acronyms

ACSL Advanced Control Systems Lab

ARL Army Research Lab

CAD Computer Aided Modeling

COM Center of Mass

ESC Electronic Speed Controller

FAA Federal Aviation Administration

FPV First-Person View

GPS Global Positioning System

LQR Linear–Quadratic Regulator

MIMO Multi-input, Multi-output

MRAC Model Reference Adaptive Control

NACA National Advisory Committee for Aeronautics

NED North-East-Down

PC Polycarbonate

PETG Polyethylene Terephthalate with Glycol

PID Proportional-Integral-Derivative

QRBP Quadrotor Biplane

RTK Real-Time Kinematic

UAV Unmanned Aerial Vehicle

VLM Vortex Lattice Method

VTOL Vertical Take off and Landing

Chapter 1

Introduction

1.1 Motivation & Scope

In this thesis, we design and test an adaptive control system for the longitudinal dynamics of a VTOL (Vertical Take off and Landing) tailsitter UAV. A tailsitter UAV is able to take off vertically, pitch 90 degrees to a horizontal orientation, and then fly like a traditional fixed-wing aircraft. The applications of a vehicle with this ability are nearly endless. Most existing tailsitter UAVs employ a generic control system that combines quadrotor and fixed-wing aircraft controllers and fail to properly model the dynamics of the vehicle. The goal of this thesis is to develop a proper dynamic model of the vehicle and design a robust adaptive control architecture that allow the transition between vertical and horizontal flight regimes despite modeling uncertainties and external disturbances.

In general, previous researchers have designed the control architectures for VTOL UAVs separately for the vertical and the horizontal flight regimes. While this strategy seems valid, it fails to account for the increasingly important aerodynamic forces during the transition between the vertical and horizontal flight regimes. Numerous researchers had also applied control techniques that were unable to guarantee satisfactory performance despite uncertainties in the systems parameters and external disturbances, especially in the nonlinear regime characterizing the transition between vertical and horizontal flight. In this thesis, we develop a control strategy that is fluent between flight regimes and robust against vehicle

parametric uncertainties.

The wide range of possible applications for this vehicle is also a motivating factor in this research. The desire for VTOL UAVs in both military and private industry has been apparent in recent years. The ability for these vehicles to take off and land in tight spaces, without an airstrip, while maintaining the endurance of a fixed-wing aircraft is the driving factor for the recent research and application of this class of aircraft. The demand for VTOL vehicles has been present in the Untied States since the 1950s, shortly after the benefits of the first helicopters were observed [13]. For the first VTOL vehicles, researchers had experimented with attaching turbojet engines to helicopters and called them "convertiplanes". While the execution seems elementary, this concept was the origin of modern VTOL vehicles.

A prime modern example of a VTOL aircraft that has been used by the military is the Boeing V-22 Osprey [1]. The V-22 Osprey is a manned transport vehicle that combines the vertical abilities of a helicopter and the efficiency of a fixed-wing aircraft and is shown in Figure 1.1. The V-22 Osprey utilizes tilting motors, known as vector thrusting, to enable forward flight after a vertical take off [14]. Boeing has supplied the United States military with V-22 Osprey vehicles since 2007. Another example that emphasizes the wide range of applications for VTOL aircraft is the VTOL UAV developed by Wing for commercial payload delivery [2]. Their VTOL UAV combines the vertical take off ability of a helicopter with the endurance of a fixed-wing aircraft by simply adding vertical lift propellers to a fixed-wing aircraft; shown in Figure 1.2. This vehicle operates autonomously and has been conducting payload deliveries from stores such as Walgreens in Christiansburg, Virginia since 2019. These two examples provide a brief look at the wide range of possible applications for VTOL aircraft.

Some of the prominent technical challenges that were faced when conducting this research included the aerodynamic performance, lack of control surfaces, and the mechanical under-

Figure 1.1: Boeing V-22 Osprey [1]

Figure 1.2: Project Wing [2]

actuation. It is inherent that any tailsitter UAV faces aerodynamic challenges during a rapid transition between vertical and horizontal flight regimes. While the vehicle pitches to its horizontal flight mode, it must undergo a period of wing stall, in which there is a decrease in lift and an increase in drag. In the design of the vehicle, we choose to not incorporate any control surfaces, and regulate the vehicle only by the means of the propellers' differential thrust. This make the vehicle lighter and less mechanically complex, but makes controlling the vehicle more difficult. Although this work is primarily focused on the design of a control law for the longitudinal motion of a tailsitter UAV, we designed an outer loop to guarantee satisfactory lateral-direction behavior in actual flight tests.

1.2 Thesis Outline

In this thesis, we review the current literature on the configurations of tailsitter UAVs. We describe in detail the different configurations used in research for tailsitter UAVs dating back to the first vehicle's constructed in the 1950s. Following that, we provide a brief description of the configuration of our tailsitter UAV.

In Chapter 2, firstly we review the dynamic modeling strategies from the literature on tailsitter UAVs. Specifically, we focus on the design decisions and assumptions that were made to simplify currently available dynamic models. Next, we discuss the notation used in this thesis to represent the tailsitter UAV and capture its longitudinal dynamics. Following that, we present the classical kinematic and dynamic equations of an aircraft and the equations of motion used in this thesis to capture the longitudinal motion of a tailsitter and hence, design adaptive control laws. Finally, we analyze these equations of motion to determine the controllability of the tailsitter UAV.

In Chapter 3, we review the literature on tailsitter UAVs to survey the control methods employed by previous researchers. Both the control strategies and the assumptions used by previous researchers when developing their control algorithms for tailsitter UAVs are described in detail. Following that, we present the control architecture used for our tailsitter UAV, namely Model Reference Adaptive Control (MRAC). To apply MRAC to our vehicle's translation and rotation, we begin by feedback linearizing the equations of motion derived in Chapter 2. The MRAC control laws are then presented for the translation and rotation of the vehicle. Subsequently, we present an outer loop for the vehicle to regulate the vehicle's lateral-directional dynamics during actual flight tests. Finally, we compute the thrust force that each motor needs to produce to realize the desired control inputs.

In Chapter 4, we validate the control algorithms designed in the previous chapter using numerical simulations in the Matlab environment. We begin by describing the user-defined trajectory that we employ to simulate a flight profile that encompasses take off, horizontal flight, and landing. The results of Matlab simulations of the transition between vertical and horizontal flight regimes are presented and analyzed. From these simulations, we also determine the motor thrusts required to complete this transition, which is useful in the design process of the tailsitter UAV. Finally, we evaluate the performance of the proposed

outer loop controller. These simulations give us an idea of how the vehicle behaves in real flight tests.

In Chapter 5, we describe in detail the design and fabrication of the vehicle along with the flight tests completed to validate our modeling and control designs. For the design of the vehicle, we describe the entire design process from the design decisions related to the vehicle's aerodynamics to the materials used in the 3D printed mounts. Then, we describe the fabrication and assembly process for the tailsitter UAV prototype used for our real flight tests. For the flight tests, we begin by conducting autonomous hover tests, both indoors and outdoors. We also perform tests to validate the outer loop control of the vehicle. Finally, we present the results of an outdoor flight test that comprises vertical take off, horizontal flight, and vertical landing.

In Chapter 6, we provide a summary of the work completed for this thesis and the results obtained. Then we discuss the possibilities for future work including vehicle upgrades, modeling changes, control improvements, and applications' studies. Lastly, we discuss the conclusions drawn from the work completed for this thesis.

1.3 Overview of Tailsitter Configurations

Tailsitter UAVs have a unique configuration. The basic configuration for a tailsitter UAV has one or multiple propellers providing vertical thrust during take off and landing and horizontal thrust during forward flight. To perform forward flight, the vehicle pitches 90 degrees and exploits its wings to provide the necessary lift. For its ability to take off and land vertically and hover, tailsitter UAVs are classified as VTOL (Vertical Take off and Landing) aircraft. Therefore, tailsitter UAV's have the advantages of a traditional quadcopter in vertical flight mode and the endurance of a plane in horizontal flight mode. The hybrid nature

of this vehicle allows for a wide range of applications from tactical surveillance operations [15] to product delivery [16, 17].

An early example of a tailsitter configuration would be that of the first tailsitter developed in 1954, the Lockheed XFV [3, 4]; shown in Figure 1.3. This vehicle is characterized by a single propeller on the nose of the plane, and ailerons on the wings, which provide the pitching moment to transition into horizontal flight. This tailsitter was developed by Lockheed Martin for the United States Navy, who tasked Lockheed Martin and Convair with developing a VTOL plane. Only Lockheed's prototype plane was ever able to fly any successful test flights. It is worthwhile to mention that the project was ultimately canceled due to the complexity of controlling the vehicle and the lack of power from the motors. Other vehicles of similar configuration include a miniature tailsitter UAV developed at Brigham Young University (BYU) in 2008 [5]; shown in Figure 1.4. Another tailsitter called the "Vertical Bat" was developed in 2013 at BYU and it had a single propeller at the tail of the vehicle instead of the nose [18]. The AVALON vehicle was also developed at Universidade de São Paulo in 2017 [19]. These prototype vehicles show a simple configuration of a tailsitter UAV.

Figure 1.3: Lockheed XFV [3, 4]

Figure 1.4: BYU tailsitter [5]

One way to generate a larger pitching moment to assist with the transition from vertical to horizontal flight is to add propellers away from the axis of rotation of the tailsitter and

apply a differential thrust to these propellers. This idea was explored at NASA AMES almost two decades ago [20]. Their tailsitter had four propellers, a single wing. A tailsitter with a configuration similar to the NASA tailsitter was developed at Tohoku University in Japan [6]. Their tailsitter had a wing span of 1.19m, a chord of 0.28m, a NACA-0010(National Advisory Committee for Aeronautics-0010) airfoil shape, and a mass of 2.0kg; see Figure 1.5.

Researchers at the University of Sydney developed a tailsitter in 2004 that they called the "T-wing" [21]. The "T-wing" had two motors, one wing, and both ailerons and elevators, which acted as control surfaces in both vertical and horizontal flight regimes. In the vertical flight regime, the control surfaces relied on propeller wash and in the horizontal flight regime, the control surfaces relied mostly on the wind. This vehicle had a 2.18m wing span, weighed 65 pounds, and was powered by two 6.5 hp two-stroke engines. In 2007, at the Technology University of Compiegne in France [22], a tailsitter was studied that had two propellers, one wing, and both ailerons and elevators. This unique configuration relied on both the differential in thrust of the propellers and the deflection of the ailerons to transition from vertical to horizontal flight. The Technology University of Compiegne in France developed an interesting tailsitter a year later called the "T-Plane" [7]. The "T-Plane" tailsitter had a single tilting propeller on the nose, two bisecting wings, and ailerons on the tail; see Figure 1.6. The tilting propeller helped to improve roll control and robustness when external disturbances were present. The goal of this project was to stabilize hover flight of the tailsitter. Another tailsitter with two propellers, one wing, and ailerons was developed in 2018 at McGill University in Canada [23].

A commonly studied configuration of the tailsitter UAV is the quadrotor biplane (QRBP) tailsitter, which distinguishes itself for the agility and its ability to generate lift with smaller wing platform areas. For these appealing specifications, this is the configuration of the UAV for this thesis. An example of a QRBP tailsitter UAV is the CRC-20 (Common Research

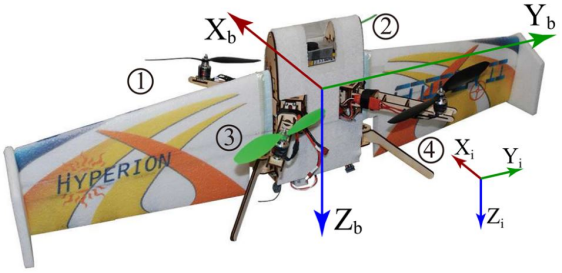

Figure 1.5: Tohoku tailsitter UAV [6]

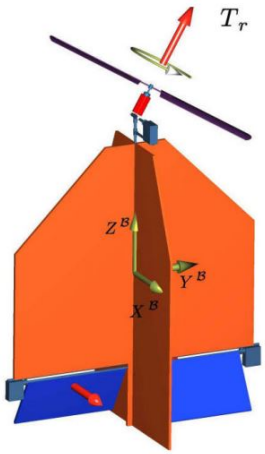

Figure 1.6: T-Plane tailsitter UAV [7]

Configuration-20) which was developed at the University of Maryland (UMD) in cooperation with the Army Research Lab (ARL) [8, 24]. The CRC-20 was developed in 2015 and was a scaled up version of their previous prototype and weighed 21 pounds with a wingspan of 4.9 feet and a rotor diameter of 2 feet. The previous prototype, the AirEZ, weighed 8 pounds with a wingspan of 3.3 feet and a rotor diameter of 1.2 feet [17]. The scientific literature about the CRC-20 UAV substantially concerns its aerodynamics, whereas its dynamics and control have been examined in lesser extent. An interesting variation on the QRBP developed at the UMD in 2019 is the morphing winglet tailsitter [9]. This vehicle was similar to the basic CRC-20 but had wing extensions that can actuate to form a box wing or open winglet; see Figure 1.8. The morphing winglet allowed for the lift to be increased at low speeds and decreased at high speeds. An additional example of a QRBP tailsitter UAV is the APT (Autonomous Pod Transport) currently being developed at Bell Helicopters [8, 25, 26, 27]. The larger version of the Bell APT can reach speeds of 110 knots and weighs 300 pounds. This vehicle was just recently able to carry a 100 pound payload for 8 miles and made its first payload delivery in Fort Worth, Texas [16, 27]. The Bell APT is a quadrotor biplane with stabilizing fins in the back and a pod in the middle, which houses the electronics and

a storage area for a payload. This vehicle is a good example of the scalability potential of a tailsitter UAV. Another example of a QRBP was developed at the Indian Institute of Technology Kanpur in 2018 [28]. The vehicle developed was a quadrotor tailsitter UAV with no control surfaces on the wings and a single stabilizing wing in the middle.

Figure 1.7: UMD CRC-20 [8]

Figure 1.8: UMD Morphing winglet tailsitter [9]

The tailsitter developed for this thesis is a QRBP tailsitter with stabilizing wings shown in Figure 1.9. This configuration is similar to the UAVs developed at UMD and Bell Helicopters. Stabilizing wings perpendicular to the main wings have been introduced to guarantee lateral-direction stability during horizontal flight. This design aspect is similar to that of the morphing winglet tailsitter developed at UMD. Control surfaces, such as ailerons and elevators, have not been included to mimic the CRC-20, and enable future collaborations with ARL and UMD. The center of the vehicle houses both the electronics and the battery on a 3D printed mount. The onboard electronics include a Pixhawk autopilot, used for sensors, and an ODROID, used as the companion computer to run the control law. Four carbon fiber rods provide the UAV's frame and support the four motors and the aerodynamic surfaces. The motor mounts and the wings mounts, which act as the landing gear for the vehicle, were also 3D printed. We designed this vehicle with two planes of symmetry, namely a horizontal and a vertical plane, to guarantee lateral direction stability, and focus the controller's effort to perform a successful transition between vertical and horizontal regimes despite uncer-

tainties in the UAV's dynamics and aerodynamics. We also designed this vehicle assuming that the center of mass coincides with the center of the vehicle. A prototype of our vehicle was fabricated to test the control method discussed in Chapter 3. This prototype tailsitter weighs 2.6kg, has a 0.75m wingspan, 0.5m of spacing between the wings, a chord of 0.29m, a planform area of 0.41m^2, and a symmetric airfoil similar to the NACA 0012 airfoil. More details of the design process and fabrication of this prototype are outlined in Chapter 5.

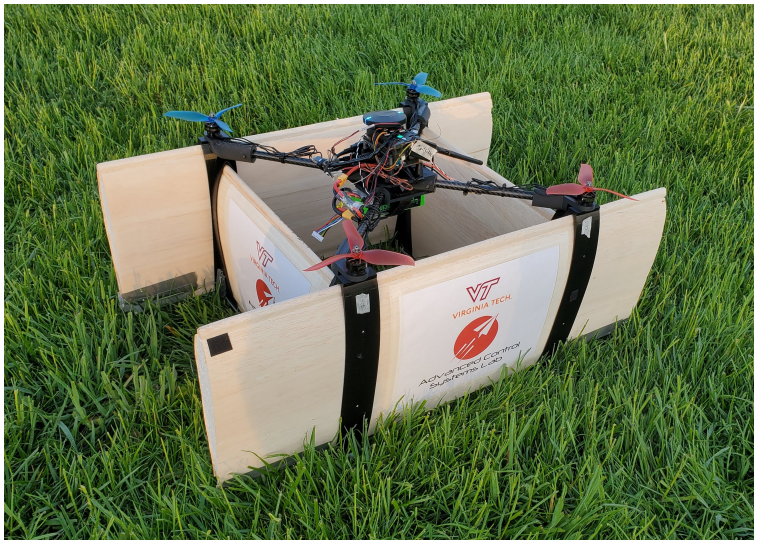

Figure 1.9: Advanced Control Systems Lab tailsitter UAV

Chapter 2

Dynamical Modeling

2.1 Introduction

The goal of this thesis is to design, implement, and test a control system for a tailsitter UAV (Unmanned Aerial Vehicle) able to control their transition between the vertical and the horizontal flight regimes, and vice-versa. This transition is particularly critical since in the vertical flight mode, this vehicle behaves like a classical quadcopter UAV and in the horizontal flight mode, this vehicle behaves like a classical fixed-wing biplane. Although numerous autopilots for these two vehicles have been developed, synthesizing a control system for tailsitter UAVs that simply juxtaposes the control systems for quadcopters and fixed-wing UAVs have been proven to be ineffective. Indeed, in the transition between vertical and horizontal flight modes, the aerodynamic forces start playing an increasingly important role, and control systems for quadcopter UAVs are usually designed neglecting the aerodynamic forces [29]. Furthermore, autopilots for fixed-wing UAVs are usually designed assuming small displacements from the aircraft's trim condition, but this assumption has been proven to be inaccurate in the transition between vertical and horizontal flight modes. For these reasons, in this thesis we focus on the design of a control system that allows for a successful transition between the horizontal and the longitudinal flight regimes.

This chapter is organized as follows. Firstly, we overview the current literature on modeling a tailsitter UAV. Then, the frames of reference necessary and the notation needed to

model the tailsitter UAV are discussed, and the translational and rotational equations of motion for a tailsitter UAV are derived. Successively, assuming that the UAV has a vertical plane of symmetry and that external forces act in the UAV's plane of symmetry, we deduce the longitudinal equation of motion for tailsitter UAVs. Finally, the controllability of our tailsitter UAV is examined.

2.2 Literature Review on Dynamic Modeling of Tailsitter UAVs

In this section, we overview the current literature on dynamic modeling of tailsitter UAVs to help understand the challenges that are faced when controlling these vehicles.

One of the first attempts at developing a quadrotor tailsitter UAV was at NASA AMES almost two decades ago [20]. They were interested in demonstrating successful hover, transition, and some forward flight. However, they were unable to maintain steady forward flight with their vehicle. These test were directly controlled by a pilot using a R/C controller and not an autopilot. While this was an impressive feat, it did not address the issue of modeling or designing automatic control algorithms for tailsitter UAVs. The Lockheed XFV was also controlled by a pilot and therefore did not have a control system or dynamic model [3]. These earlier results are worthwhile being recalled since they show both the complexity and novelty of this problem.

An early attempt at modeling and controlling a tailsitter UAV was completed at the University of Sydney [21]. A large portion of their research revolved around the propeller wash model. To start, Goldstein's vortex theory was used to analyze the blade element solution for the propellers. Next, they used a full azimuthal solution to determine the aerodynamic

forces produced by the wing as a result of the slipstream velocity. The slipstream velocity induced by the propeller was combined with the freestream velocity to accurately model the total aerodynamic forces. The aerodynamic coefficients were then determined using a panel-method solution with two-dimensional viscous corrections to account for a large angle of attack during the transition from vertical to horizontal flight regimes. Since this was an early attempt at modeling a tailsitter UAV, they performed a multi-disciplinary design optimization which helped to analyze the interactions between different components of the vehicle. The areas studied included the aerodynamics, propulsion, vehicle structures, stability and control, and weight and balance. This study helped to determine the optimum vehicle parameters and flight conditions.

Shortly after, the Technology University of Compiegne in France also attempted to model and control their tailsitter [22]. Their model only captured the vehicle's longitudinal dynamics, and made some important modeling assumptions. For instance, they assumed that the vehicle aerodynamic surface was fully submerged in the propeller slipstream and the slipstream velocity was much greater than the body normal velocity. These assumptions lead them to model the aerodynamic coefficients as functions of the slip stream velocity and hence, as a function of the axial velocity and the current thrust of the vehicle. A backstepping technique was employed to control the system. The modeling strategy for the T-Plane tailsitter, developed at the Technology University of Compiegne in France, also relied on the simplification of the longitudinal equations of motion based on assumptions about the aerodynamics [7].

Another modeling and control technique for a tailsitter UAV was developed at the Brigham Young University (BYU) [5]. The configuration of this vehicle was different than the one discussed in this thesis, but the modeling and control strategies were similar. Specifically, the tailsitter designed at BYU had a single propeller on the nose and ailerons on the wings.

The similarities in the modeling come about in their assumption that the vehicle maintains symmetric flight and therefore the longitudinal and lateral equations can be separated. They also created a reference model to help smooth the desired attitude trajectories and maintain a realistic reference trajectory. Their reference model was designed with the physical limitations of the actual aircraft in mind. They used an adaptive backstepping controller to stabilize the system. Finally, they used a stabilized recursive least squares method to estimate the system parameters for the controller.

The Vertical Bat tailsitter was developed at BYU in collaboration with the MLB Company [18]. Their aerodynamic lift was modeled as a mixture of a laminar airfoil model and a flat plate model and they used a blending function to calculate the total lift coefficient while their aerodynamic drag was modeled as a parasitic drag. Since their vehicle had a duct around their airfoil, they modeled the radial airflow velocity, dynamic pressure, and angle of attack around the duct as a function of the angular position around the duct to determine the lift and drag induced by the duct. To help control the Vertical Bat tailsitter, they incorporated eight control veins and two ailerons and modeled independent local reference frames for each of these control surfaces.

The CRC-20 tailsitter developed at UMD underwent extensive aerodynamic design and modeling studies [24]. The design parameters investigated included the wing airfoil shape, motor propeller, wing aspect ratio, wing spacing, and the alignment of the propeller-chord. All of these parameters were tested to maximize the vehicle's aerodynamic efficiency and once these parameters were chosen, tests in wind tunnels were performed. These tests helped researchers at UMD develop an accurate model of the vehicle and determine minimum operating speeds for transition and forward flight. Quaternions were employed to capture the vehicle's attitude, and a proportional-derivative (PD) control law was employed. They were able to achieve transition to forward flight, but they had poor control over the vehicle once

in forward flight. The next evolution of this project, the morphing winglet tailsitter, also went through an extensive dynamic modeling investigation [9]. They focused on the effects of the winglets on the lift force generated in different flight regimes. They first used a Vortex Lattice Method (VLM) 3D flow solver to determine the lift generated at different dihedral angles of the winglets, which allowed them to more accurately model the lift coefficient. Following that, they verified the simulation results with experimental tests, which showed the improved horizontal flight efficiency over their original vehicle designs.

A tailsitter UAV was produced and tested at Tohoku University in Japan [6]. They separated the longitudinal and lateral dynamic equations when modeling the tailsitter UAV and modeled both the lift coefficient and the drag coefficient as functions of the angle of attack deduced by means of tests in a wind tunnel. One of the main differences with this thesis lies in the control laws employed, where they implemented a proportional-integral-derivative (PID) controller based on the altitude and attitude error dynamics. Their goal was to study the effect of different transition strategies from vertical to horizontal flight, while minimizing the transition time and maintaining a constant altitude during transition.

The AVALON tailsitter UAV was developed at the Universidade de São Paulo [19]. The vehicle was designed to mimic a typical fixed-wing symmetric aircraft with a single propeller, one wing, two ailerons, two elevators, and one rudder. To model the vehicle, the assumptions were made that the center of the body-fixed reference frame coincided with the center of gravity of the vehicle and that the inertial frame is fixed with the Earth and tangent to gravity. Both the kinematics and dynamics of the vehicle were used to model the vehicle. They assumed that the propeller is aligned with the center of gravity of the vehicle, which justifies modeling the propeller thrust only in one axis of the body fixed reference frame. Their model was tested using a simulation made in Matlab.

A tailsitter UAV with a very similar configuration to the one addressed in this thesis but a

different modeling technique was studied at the Indian Institute of Technology Kanpur [28]. Multiple dynamical models and control algorithms were used for the different flight modes. In their vertical flight regime, they modeled the vehicle only using the thrust provided by the propellers neglecting the aerodynamic forces, they captured the UAV dynamics as a second-order system, and designed their controllers by setting the desired damping ration and natural frequency for the closed-loop system. In the horizontal flight regime, they modeled the aircraft by employing the classical set of translational and rotational equations of motion in the UAV's plane of symmetry and then their fixed wing controller was used to determine the necessary throttle input to achieve their desired velocity. During the transition between vertical and horizontal flight regimes, they choose a reference pitch angle to switch from using their vertical flight controller to using their horizontal flight controller. While this strategy is practical, it fails to account for the aerodynamic forces in the quadrotor mode and the transition mode.

The tailsitter UAV developed at McGill University had an extensive model and was verified with a simulation [23]. To ensure an accurate simulation, they modeled the typical aerodynamic forces and thrust forces but went on to model the drag induced by the propeller protectors, landing gear, propellers and the gyroscopic moments induced by the rotation of the propellers. They also modeled the ground contact forces during take off and landing and then used a Dryden wind turbulence model to represent the inflow velocity from the propellers in the aerodynamic model. To determine the coefficients for the propellers and the control surfaces, static bench tests were conducted. They used quaternions to represent body rotations without any singularities. This comprehensive model was used for the simulation, but for the controller, a simplified model was used that predicts the effect of the propeller thrust and elevator deflection on the body moments and forward force.

2.3 Notation

In this section, we present the notation used in this thesis to capture the dynamics of a tailsitter UAV. To model the UAV, three orthonormal references frames are introduced. Let $\mathbb{I} \triangleq \{O; X, Y, Z\}$ denote the orthonormal *inertial reference frame* centered at the point O and fixed with the Earth. This is a North-East-Down (NED) reference frame, that is, X points North, Y points East, and Z is along the direction of gravity so that the weight of the UAV is denoted by $F_g^{\mathbb{I}} = mgZ$, where $m > 0$ denotes the *UAV's mass* and $g > 0$ denotes the *gravitational acceleration*; see Figure 2.1. Vectors expressed in the reference frame $\mathbb{I}$ are denoted by the superscript $\mathbb{I}$. A reference frame fixed with the Earth is not inertial due to the revolution of the Earth about its axis and the rotation of the Earth about the Sun. However, the assumption can be made that a reference frame fixed with the Earth is inertial because the UAV is relatively slow and operates at a low altitude [30, Ch. 2]. The orthonormal *body reference frame* is denoted by $\mathbb{J}(\cdot) \triangleq \{A(\cdot); x_{\text{body}}(\cdot), y_{\text{body}}(\cdot), z_{\text{body}}(\cdot)\}$ and is centered at the UAV's center of mass, $A(\cdot)$. A vector expressed in $\mathbb{J}(\cdot)$ will not have a superscript. Lastly, $\mathbb{W}(\cdot) \triangleq \{A(\cdot); x_{\text{wind}}(\cdot), y_{\text{wind}}(\cdot), z_{\text{wind}}(\cdot)\}$ denotes the orthonormal *wind frame*. Vectors expressed in the wind frame $\mathbb{W}(\cdot)$ are denoted by the superscript $\mathbb{W}$. The $x_{\text{wind}}(\cdot)$ axis is in the direction of the UAV's velocity with respect to the freestream wind velocity, the $z_{\text{wind}}(\cdot)$ axis is orthogonal and in the UAV's plane of symmetry, and the $y_{\text{wind}}(\cdot)$ axis is chosen so that the reference frame adheres to the right hand rule. The position of $A(\cdot)$ relative to the origin of the inertial reference frame $\mathbb{I}$ is denoted by $r_A : [t_0, \infty) \to \mathbb{R}^3$. The orientation of the body reference frame with respect to the inertial reference frame is represented by a 3-2-1 sequence of explicit Bryan-Tait angles, whereby $\psi : [t_0, \infty) \to [0, 2\pi)$ captures the *yaw angle*, $\theta : [t_0, \infty) \to (-\frac{\pi}{2}, \frac{\pi}{2})$ captures the *pitch angle*, and $\phi : [t_0, \infty) \to [0, 2\pi)$ captures the *roll angle*.

The orientation of the wind reference frame $\mathbb{W}(\cdot)$ with respect to the body reference frame

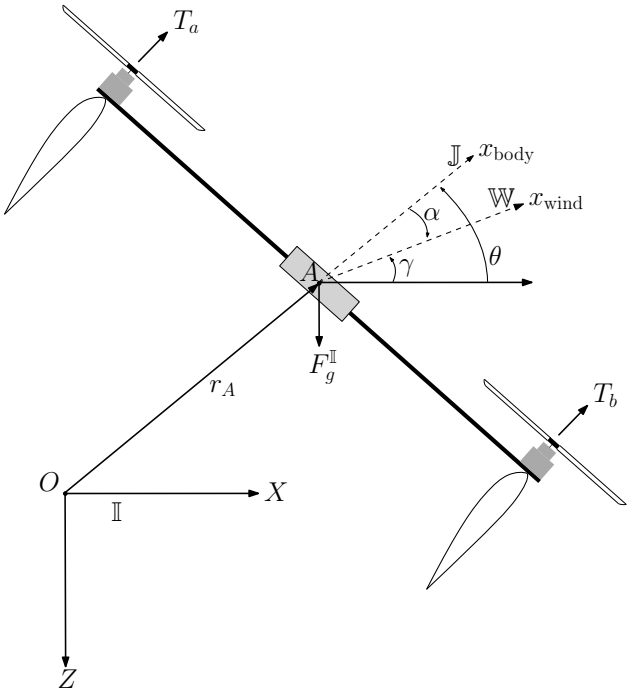

Figure 2.1: Tailsitter UAV with reference frames

$\mathbb{J}(\cdot)$ is captured by the the polar coordinates $\beta : [t_0, \infty) \to [-\pi, \pi]$ and $\alpha : [t_0, \infty) \to [0, 2\pi)$, which denote the *sideslip angle* and the *angle of attack*, respectively. In this thesis, we assume that the freestream wind velocity does not have any component orthogonal to the UAV's plane of symmetry and hence, we assume that $\beta(t) = 0$, $t \geq t_0$. Finally, the angle between the wind reference frame $\mathbb{W}(\cdot)$ and the inertial reference frame $\mathbb{I}$ is captured by the *flight path angle*, $\gamma(t) \triangleq \theta(t) - \alpha(t), t \geq t_0$; see Figure 2.1.

A tailsitter UAV has four control inputs that form the *control vector*, $u(t) \triangleq [u_1(t), u_2(t), u_3(t), u_4(t)]^{\mathrm{T}}$, $t \geq t_0$, where $u_1(t) \geq 0$ captures the propellers' total thrust force, and $u_2(t)$, $u_3(t)$, and $u_4(t) \in \mathbb{R}$ capture the moment induced by the propellers' thrust force about the $x_{\text{body}}(\cdot)$, $y_{\text{body}}(\cdot)$, and $z_{\text{body}}(\cdot)$ axes, respectively. Since we are only studying the longitudinal motion, we assume that $u_2(t) = u_4(t) = 0$, $t \geq t_0$, and introduce T_a and $T_b : [t_0, \infty) \to [0, \infty)$ so that $u_1(t) = T_a(t) + T_b(t)$ and $u_3(t) = \frac{l}{2}(T_b(t) - T_a(t))$, where $l > 0$ denotes the distance between propellers along the $z_{\text{body}}(\cdot)$ axis and T_a and T_b denote the force produced by the

top two and bottom two propellers, respectively. A fixed wing aircraft typically has control surfaces on the wings in the form of ailerons and on the vertical stabilizer in the form of a rudder. As discussed in Section 1.3, our vehicle is a hybrid of a quadcopter and a classical fixed-wing UAV, and we chose not to introduce ailerons to reduce the UAV's total weight.

2.4 Modeling Assumptions

In this section, we present the hypotheses underlying the proposed dynamical models for a tailsitter UAV, which can be summarized as follows. The mass of the aircraft is assumed to be constant, which is a suitable assumption for applications such as reconnaissance and surveillance.

An additional assumption is that the aircraft velocity vector with respect to the free stream air is contained in its plane of symmetry. Although restrictive, this assumption is justified by the fact that in this study, we are interested in controlling the transition between vertical and horizontal flight regimes while keeping the roll angle as small as possible. Additionally, to minimize the control effort, it is recommended that in this stage, the take off maneuver is performed by pointing the UAV in the same direction as the wind.

In this thesis, we model the UAV as a rigid body. This assumption is verified whenever the aircraft does not carry any dangling payload and whenever the gyroscopic effect produced by the propellers is sufficiently small compared to the Coriolis forces produced by the UAV's rotational motion.

Our final modeling assumption is that the lift and drag coefficients of the wings are estimated in the linear regimes, whereas the UAV's aerodynamic properties are completely unknown beyond the linear regime. A key point of this thesis is to produce a control law for

a tailsitter UAV that allows for a successful transition between vertical and horizontal flight regimes despite this limiting assumption.

2.5 Longitudinal Equations of Motion

In this section, the longitudinal equations of motion for the tailsitter UAV will be deduced from the general set of equations of motion of an aircraft. The translational kinematic equations of an aircraft modeled as a rigid body are given by [30, Ch. 2]

$$\frac{d}{dt} r_A^{\mathbb{I}}(t) = v^{\mathbb{I}}(t), \qquad r_A^{\mathbb{I}}(t_0) = r_{A,0}^{\mathbb{I}}, \qquad t \geq t_0, \tag{2.1}$$

where $v(\cdot)$ denotes the *velocity of the center of mass* with respect to the wind, and the rotational kinematic equations are captured by

$$\frac{d}{dt}\begin{bmatrix}\phi(t)\\ \theta(t)\\ \psi(t)\end{bmatrix} = \begin{bmatrix}1 & \sin\phi(t)\tan\theta(t) & \cos\phi(t)\tan\theta(t)\\ 0 & \cos\phi(t) & -\sin\phi(t)\\ 0 & \sin\phi(t)\sec\theta(t) & \cos\phi(t)\sec\theta(t)\end{bmatrix}\omega(t), \qquad \begin{bmatrix}\phi(t_0)\\ \theta(t_0)\\ \psi(t_0)\end{bmatrix} = \begin{bmatrix}\phi_0\\ \theta_0\\ \psi_0\end{bmatrix}. \tag{2.2}$$

The translational dynamic equations are captured by

$$\frac{d}{dt}v^{\mathbb{I}}(t) = \frac{1}{m}R_{\mathbb{J}}^{\mathbb{I}}(\phi(t),\theta(t),\psi(t))\left(F_t(t) + R_{\mathbb{W}}^{\mathbb{J}}(\alpha(t),\beta(t))\begin{bmatrix}D(t,v)\\ C(t,v)\\ L(t,v)\end{bmatrix}^{\mathbb{W}}\right) + \begin{bmatrix}0\\ 0\\ g\end{bmatrix},$$
$$v(t_0) = v_0, \qquad t \geq t_0, \tag{2.3}$$

where the orientation of the body reference frame $\mathbb{J}(\cdot)$ relative to the inertial reference frame $\mathbb{I}$ is captured by

$$R_{\mathbb{J}}^{\mathbb{I}}(\phi(t),\theta(t),\psi(t)) \triangleq \begin{bmatrix} \cos\phi(t) & -\sin\phi(t) & 0 \\ \sin\phi(t) & \cos\phi(t) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta(t) & 0 & \sin\theta(t) \\ 0 & 1 & 0 \\ -\sin\theta(t) & 0 & \cos\theta(t) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\psi(t) & -\sin\psi(t) \\ 0 & \sin\psi(t) & \cos\psi(t) \end{bmatrix}, \tag{2.4}$$

$F_t(t) \triangleq u_1(t)\mathbf{e}_{1,3}$ denotes the *thrust force* produced by the propellers, the orientation of the wind reference frame $\mathbb{W}(\cdot)$ relative to the body reference frame $\mathbb{J}(\cdot)$ is captured by [31, p. 117]

$$R_{\mathbb{W}}^{\mathbb{J}}(\alpha,\beta) \triangleq \begin{bmatrix} \cos\alpha(t)\cos\beta(t) & -\cos\alpha(t)\sin\beta(t) & -\sin\alpha(t) \\ \sin\beta(t) & \cos\beta(t) & 0 \\ \sin\alpha(t)\cos\beta(t) & -\sin\alpha(t)\sin\beta(t) & \cos\alpha(t) \end{bmatrix},$$

$D(t,v) \triangleq -\frac{1}{2}\rho||v(t)||^2 S C_D(\alpha(t))$ denotes the *drag force* in the wind reference frame $\mathbb{W}(\cdot)$, $C(t,v) \triangleq -\frac{1}{2}\rho||v(t)||^2 S C_C$ denotes the *side force* in the wind reference frame $\mathbb{W}(\cdot)$, $L(t,v) \triangleq -\frac{1}{2}\rho||v(t)||^2 S C_L(\alpha(t))$ denotes the *lift force* in the wind reference frame $\mathbb{W}(\cdot)$, ρ denotes the *air density*, S denotes the UAV's *planform area*, $C_L : \mathbb{R} \to \mathbb{R}$ denotes the *coefficient of lift* at any given angle of attack, $C_D : \mathbb{R} \to \mathbb{R}$ denotes the *coefficient of drag* at any given angle of attack, and C_C denotes the *coefficient of side force*. To model the relationship between the lift and the angle of attack, the coefficient of lift is defined by

$$C_L(\alpha) \triangleq C_{L,0} + C_{L,\alpha}\alpha(t), \qquad \alpha \in [0, 2\pi), \qquad t \geq t_0, \tag{2.5}$$

where $C_{L,\alpha} > 0$ denotes the *coefficient of lift slope* and $C_{L,0} \in \mathbb{R}$ denotes the *coefficient of lift at zero angle of attack* [32, p. 70]. Similarly, the coefficient of drag is modeled as

$$C_D(\alpha) \triangleq C_{D,0} + C_{D,\alpha}\alpha(t), \qquad \alpha \in [0, 2\pi), \qquad t \geq t_0, \tag{2.6}$$

where $C_{D,\alpha} > 0$ denotes the *coefficient of drag slope* and $C_{D,0} \in \mathbb{R}$ denotes the *coefficient of drag at zero angle of attack*.

Finally, the rotational dynamic equation is captured by

$$\frac{d}{dt}\omega(t) = \begin{bmatrix} -I_z(I_{xz}^2 - I_xI_z)^{-1} & 0 & -I_{xz}(I_{xz}^2 - I_xI_z)^{-1} \\ 0 & I_y^{-1} & 0 \\ -I_{xz}(I_{xz}^2 - I_xI_z)^{-1} & 0 & -I_x(I_{xz}^2 - I_xI_z)^{-1} \end{bmatrix} \left(\begin{bmatrix} K(v(t), \omega(t)) \\ M(v(t), \omega(t), \dot{\omega}(t)) \\ N(v(t), \omega(t)) \end{bmatrix} \right.$$
$$\left. -\omega^{\times}(t) \begin{bmatrix} I_x & 0 & -I_{xz} \\ 0 & I_y & 0 \\ -I_{xz} & 0 & I_z \end{bmatrix} \omega(t) + \begin{bmatrix} u_2(t) \\ u_3(t) \\ u_4(t) \end{bmatrix} \right), \qquad \omega(t_0) = \omega_0, \qquad t \geq t_0. \tag{2.7}$$

where $\omega(\cdot)$ denotes the *aircraft angular velocity* expressed in the body frame $\mathbb{J}$, $K : \mathbb{R}^3 \times \mathbb{R}^3 \to \mathbb{R}$, $M : \mathbb{R}^3 \times \mathbb{R}^3 \to \mathbb{R}$, and $N : \mathbb{R}^3 \times \mathbb{R}^3 \to \mathbb{R}$ denote the components of of the moment of the external forces acting on the aircraft, along the roll, pitch, and yaw axes, respectively, I_x, I_y, and $I_z > 0$ denote the *moments of inertia*, and $I_{xz} \in \mathbb{R}$ denote the *products of inertia*. The products of inertia, I_{xy} and I_{yz}, are zero because we assume a symmetric vehicle and that the center of the body reference frame $\mathbb{J}(\cdot)$ is at the center of mass of the vehicle. [30, pp. 32-33]. Equations (2.1)–(2.7) are the *equations of motion of an aircraft modeled as a rigid body*.

The tailsitter UAV is a symmetric fixed-wing aircraft and we assume that the lateral component of the aerodynamic force and the moment of the aerodynamic forces are equal

to zero. Thus the side slip angle is zero at all times [31, pp. 159-160] [33, Ch. 2] [30, Ch. 2]. Under these assumptions, it follows from (2.3) that the translational dynamic equations of motion reduce to

$$\frac{d}{dt}v^{\mathbb{I}}(t) = \frac{1}{m}\begin{bmatrix} \cos\theta(t) & 0 & \sin\theta(t) \\ 0 & 1 & 0 \\ -\sin\theta(t) & 0 & \cos\theta(t) \end{bmatrix}\begin{bmatrix} u_1(t) + D(t,v)\cos\alpha(t) - L(t,v)\sin\alpha(t) \\ 0 \\ D(t,v)\sin\alpha(t) + L(t,v)\cos\alpha(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}, \qquad v(t_0) = \begin{bmatrix} s_0 \\ 0 \\ w_0 \end{bmatrix}, \quad t \geq t_0. \tag{2.8}$$

Using the relation, $\gamma(t) = \theta(t) - \alpha(t)$, $t \geq t_0$ and (2.1), it follows that (2.8) can be further reduced to

$$\ddot{x}_A(t) \triangleq \frac{1}{m}[F_x^{\mathbb{I}}(t) + L(t,v)\sin\gamma(t) + D(t,v)\cos\gamma(t)],$$
$$x_A(t_0) = x_{A,0}, \qquad \dot{x}_A(t_0) = \dot{x}_{A,0}, \quad t \geq t_0, \tag{2.9}$$
$$\ddot{z}_A(t) \triangleq \frac{1}{m}[F_z^{\mathbb{I}}(t) + mg + L(t,v)\cos\gamma(t) - D(t,v)\sin\gamma(t)],$$
$$z_A(t_0) = z_{A,0}, \qquad \dot{z}_A(t_0) = \dot{z}_{A,0}, \tag{2.10}$$

where $x_A(\cdot)$ and $z_A(\cdot)$ denote the first and third components of $r_A(\cdot)$, $F_x^{\mathbb{I}}(\cdot) \triangleq (T_a(t) + T_b(t))\cos\theta(t)$ denotes the desired control force in the X inertial direction, and $F_z^{\mathbb{I}}(\cdot) \triangleq -(T_a(t) + T_b(t))\sin\theta(t)$ denotes the desired control force in the Z inertial direction.

Furthermore, (2.2)–(2.7) do not explicitly depend on the yaw angle $\psi(\cdot)$. Therefore, it

follows from (2.2) that [30, p. 44]

$$\frac{d}{dt}\begin{bmatrix}\phi(t)\\ \theta(t)\end{bmatrix} = \begin{bmatrix}1 & \sin\phi(t)\tan\theta(t) & \cos\phi(t)\tan\theta(t)\\ 0 & \cos\phi(t) & -\sin\phi(t)\end{bmatrix}\omega(t) \qquad \begin{bmatrix}\phi(t_0)\\ \theta(t_0)\end{bmatrix} = \begin{bmatrix}\phi_0\\ \theta_0\end{bmatrix}, \qquad t \geq t_0. \tag{2.11}$$

Using the same assumptions as for the translational equations of motion, (2.7) simplifies to

$$\frac{d}{dt}\omega(t) = \begin{bmatrix}0\\ I_y^{-1}u_3(t)\\ 0\end{bmatrix}, \qquad \omega(t_0) = \begin{bmatrix}0\\ q_0\\ 0\end{bmatrix}, \qquad t \geq t_0, \tag{2.12}$$

and following from (2.11) and (2.12), the rotational equation of motion of the tailsitter UAV is expressed as

$$\ddot{\theta}(t) \triangleq I_y^{-1}u_3(t), \qquad \theta(t_0) = \theta_0, \qquad \dot{\theta}(t_0) = \dot{\theta}_0, \tag{2.13}$$

where I_y denotes the *moment of inertia* about the $y_{\text{body}}(t)$ axis.

In this thesis, (2.9), (2.10), and (2.13) are considered as the equations of motion of a tailsitter UAV. Analyzing these equations of motion, we recognize that there is no need to limit $\theta(\cdot)$ within the interval $(-\frac{\pi}{2}, \frac{\pi}{2})$ as previously stated in Section 2.3 because no singularity is present in (2.13). Therefore, we let $\theta(t) \in [0, 2\pi)$, $t \geq t_0$. Similar examples of longitudinal translational and rotational dynamic equations of motion are provided by [32, pp. 69-70], [22, p. 3387], [31, pp. 160-161], and [34, p. 1893].

2.6 Controllability of the Tailsitter UAV

In the following, we recall the notion of underactuated dynamical systems. For the statement of this results consider the dynamical system [29, pp. 106-107]

$$\ddot{q}(t) = f(t, q(t), \dot{q}(t)) + G(q(t))u(t), \quad q(t_0) = q_0, \quad \dot{q}(t_0) = q_{d,0}, \quad t \geq t_0, \tag{2.14}$$

where $q(t) \in \mathcal{D} \subseteq \mathbb{R}^n$ denotes the vector of independent generalized coordinates, $u(t) \in \mathbb{R}^m$, $f : [t_0, \infty) \times \mathcal{D} \times \mathbb{R}^n \to \mathbb{R}^n$, and $G : \mathbb{R}^n \to \mathbb{R}^{n \times m}$.

Definition 2.1 ([35])**.** Consider the nonlinear dynamical system (2.14). If $\operatorname{rank}(G(q)) = n$ for all $q \in \mathcal{D}$, then (2.14) is fully actuated. Alternatively, if $\operatorname{rank}(G(q)) < n$ for all $q \in \mathcal{D}$, then (2.14) is underactuated.

Next we verify that the dynamical model given by (2.9), (2.10) and (2.13) is underactuated. To verify this fact, we recognize that (2.9), (2.10) and (2.13) is in the same form as (2.14) with $n = 3$, $m = 2$, $q(t) = [x_A(t), z_A(t), \theta(t)]^{\mathrm{T}}$, $t \geq t_0$, $u(t) = [T_a(t), T_b(t)]^{\mathrm{T}}$,

$$f(t, q(t), \dot{q}(t)) = \begin{bmatrix} \frac{1}{m}(L(t,v)\sin\gamma(t) + D(t,v)\cos\gamma(t)) \\ \frac{1}{m}(mg + L(t,v)\cos\gamma(t) - D(t,v)\sin\gamma(t)) \\ I_y^{-1} \end{bmatrix}, \tag{2.15}$$

$$G(q) = \begin{bmatrix} \cos\theta(t) & \cos\theta(t) \\ -\sin\theta(t) & -\sin\theta(t) \\ -\frac{l}{2} & \frac{l}{2} \end{bmatrix}. \tag{2.16}$$

We note that $\operatorname{rank}(G(q)) = 2 < n$ for the given $q \in \mathcal{D}$ and therefore the system is underactuated. If we only consider $q(t) = [z_A(t), \theta(t)]^{\mathrm{T}}$, $t \geq t_0$ then, $n = 2$, $m = 2$,

$u(t) = [T_a(t), T_b(t)]^{\mathrm{T}}$,

$$f(t, q(t), \dot{q}(t)) = \begin{bmatrix} \frac{1}{m}(mg + L(t,v)\cos\gamma(t) - D(t,v)\sin\gamma(t)) \\ I_y^{-1} \end{bmatrix}, \tag{2.17}$$

$$G(q) = \begin{bmatrix} -\sin\theta(t) & -\sin\theta(t) \\ -\frac{l}{2} & \frac{l}{2} \end{bmatrix}. \tag{2.18}$$

For this case, the $\text{rank}(G(q)) = n$ and therefore the system is fully actuated for all $q \in \mathcal{D}$. For this reason, we choose to directly control only $z_A(\cdot)$ and $\theta(\cdot)$.

Simply put, underactuation of a mechanical system occurs when there are less control inputs than generalized coordinates. This is the case for the longitudinal motion of a tailsitter UAV because there are three independent coordinates, the position in X and Z directions and the pitch angle, and there are only two control inputs, $T_a(\cdot)$ and $T_b(\cdot)$. This means that we have to choose two of the generalized coordinates to directly control and leave the third uncontrolled.

2.7 Conclusion

In this chapter, we first outlined the current literature on tailsitter UAVs. We reviewed the modeling decisions and assumptions made for other tailsitter UAVs. Through the exploration of the current literature on the modeling of tailsitter UAVs, we found examples of how other researchers using similar assumptions as ours to derive the vehicle's equations of motion. We then presented the notation, as well as the reference frames necessary to capture the tailsitter UAVs attitude and altitude. Following that, we proceeded with the general kinematic and dynamic equations of motion of an aircraft and employed our assumptions to deduce the

translational and rotational dynamic equations that are used in this thesis to model our tailsitter UAV. Some important assumptions that we made when simplifying the equations of motion were that the center of mass of the vehicle is known, that the inertial reference frame $\mathbb{I}$ is fixed with the Earth, that we will always operate in symmetric flight, and that the wind reference frame $\mathbb{W}(\cdot)$ is centered at the same point as the body reference frame $\mathbb{J}(\cdot)$ which is the center of mass of the vehicle. Finally, we examined the controllability of the system and determined that our vehicle is underactuated and therefore we will only be able to control two of the generalized coordinates, $z_A(t)$ and $\theta(t)$. In the next chapter, we will start with the equations of motion derived in this chapter and proceed to design the method of control used to stabilize the tailsitter UAV despite uncertainties.

Chapter 3

Adaptive Control of a Tailsitter UAV

3.1 Introduction

Once a dynamical system has been modeled, the next step is designing a controller. The "hybrid" nature of the tailsitter, which allows it to hover as a quadcopter and to fly forward as a fixed-wing airplane, is what makes this vehicle so attractive, but also what makes it more difficult to control than classical quadcopters and conventional fixed-wing airplanes. Control methods for both fixed-wing aircraft and traditional quadrotor UAVs have been extensively researched in recent years. However, controlling a tailsitter UAV is a relatively new research area.

When dealing with an uncertain, nonlinear system attempting to undergo a complex maneuver, such as a transition from vertical to horizontal flight, it is important to implement a robust control technique. An adaptive control technique, in which the gains of the control system adapt based on the response of the system, can be considered as robust for its ability to guarantee satisfactory trajectory tracking error performance despite modeling uncertainties. Research on adaptive control began in the 1950s to develop a new control technique for high-performance aircraft [36, Ch. 9]. For this thesis, we use a version of adaptive control called Model Reference Adaptive Control (MRAC). First proposed in 1958 by Whitaker et al. [37], MRAC introduces a user-defined reference model, based on an ideal system, for the system to track in the presence of uncertainty. The first proof of MRAC for a closed-loop

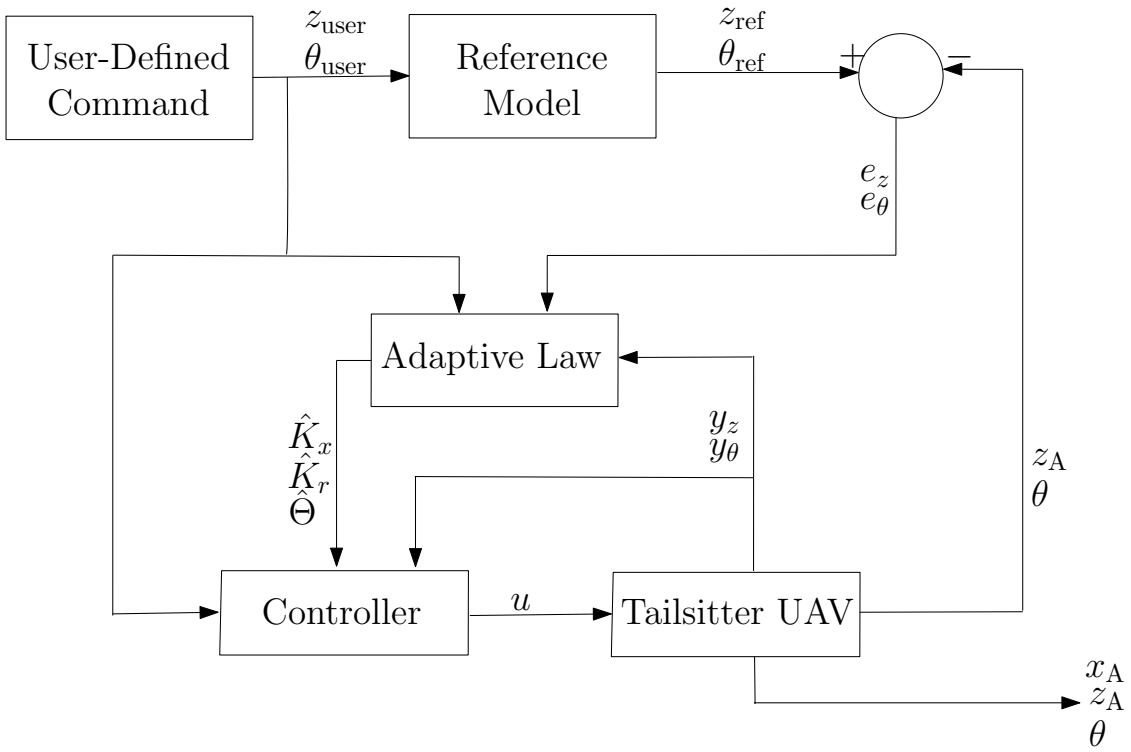

Figure 3.1: Schematic representation of the proposed control architecture for a tailsitter UAV

system based on Lyapunov stability was given in 1965 and was confirmed experimentally on an analog computer [38]. A simple block diagram of MRAC is shown in Figure 3.1.

We will begin this chapter by reviewing the types of control laws used by other tailsitter UAVs. Successively, we will overview the basics of the MRAC technique. Following from the equations of motion derived in Chapter 2, we will apply MRAC with a baseline PD controller to our tailsitter UAV to control its longitudinal motion. A control algorithm for the outer loop, which will be instrumental to enforce the symmetric flight assumptions model underlying the inner loop, will be presented. Finally, we will show how the tailsitter UAV will realize the desired forces and moments.

3.2 Literature Review on Control Methods for a Tailsitter UAV

Controlling the transition from vertical to horizontal flight for a tailsitter UAV is a challenging task. When a tailsitter UAV begins to transition to forward flight, the aerodynamic forces and moments of inertia play a significant role in the dynamics of the vehicle. Without a robust control technique, the transition flight is unlikely to be successful. Many control techniques have been tested on tailsitter UAVs and in this section we will summarize these efforts.

Some of the early tailsitter UAVs developed were not controlled autonomously due to their complexity and the state of technology at that time. These vehicles were operated by a pilot, whether that be in the vehicle or via R/C controller. The tailsitters that were not autonomously controlled were the Lockheed XFV [3, 4] and the tailsitter developed at NASA AIMES [20].

The T-Wing tailsitter UAV developed at the University of Sydney had a controller based on a simplified version of the dynamical models [21]. They began by linearizing the UAV's dynamical model about various trim conditions in different flight regimes and proceeded to design multiple low-level gain-scheduled controllers to stabilize the system in all flight conditions. The controllers included linear–quadratic regulator (LQR) controllers and other classical flight controllers. The purpose of gain-scheduled controllers is to stabilize a nonlinear system using a series of linear controllers in different flight conditions. Guidance controllers were implemented at a higher level to navigate the tailsitter between its desired waypoints. The researchers at the University of Sydney developed a simulator in Simulink to test their control law in the presence of unknown disturbances and uncertainties. As for real tests, their control law was tested for autonomous hovering but no transition flights

were tested.

The tailsitter developed at the University of Compiegne split its control strategies into a vertical flight controller, which neglected the aerodynamic terms, and a forward flight controller [22]. The vertical flight controller used feedback linearization in the thrust input and employed a backstepping technique to overcome the system's underactuation. The final vertical control law was designed such that the derivative of their Lyapunov function is negative definite, proving stability. Their forward flight control law incorporated a set-point Proportional-Derivative (PD) controller for the pitch of the vehicle in addition to the previously described controller for the altitude. To test their control laws, a simulation was run with perturbations in the attitude during vertical and horizontal flight. The vertical control law was proven to be robust to external perturbation, and experimental results confirmed their vertical flight control law. In [7], the T-Plane UAV developed at the University of Compiegne employed a nonlinear control law based on separated saturation functions to stabilize both the orientation and position of the vehicle. This controller was able to successfully stabilize the vehicle when an autonomous hover test was conducted with disturbances, but they did not move on to horizontal testing.

An adaptive control law was employed in an attempt to control the miniature tailsitter developed at BYU [5]. As already mentioned in Chapter 2, this tailsitter had control surfaces in the form of ailerons to assist with the transition from vertical to horizontal flight regimes. They explored the application of a computationally efficient least-squares-based model reference adaptive quaternion backstepping control method [5]. Their reference model was based on the physics of the actual system to ensure that their trajectories were obtainable. Their adaptive backstepping controller drove the error between the actual trajectory and the reference trajectory to zero and the stability of this controller was verified using Lyapunov stability theory. The system parameters needed for the controller were estimated using a

stabilized recursive least squares method. Both the simulation and hardware test showed the vehicle tracking the reference model and the estimated parameters quickly converging to their true values.

The Vertical Bat tailsitter developed at BYU implemented a hierarchical control structure based on Proportional-Integral-Derivative (PID) controllers, which contained a waypoint manager, flight mode controller, flight mode specific controllers and attitude controller [18]. The waypoint manager was the highest level of the control architecture and determined which waypoint should be tracked at any given time. One level down was the flight mode controller, which determined the maneuvers, such as transitioning from vertical to horizontal flight regimes, that the the tailsitter UAV must undergo to reach the desired waypoint. The next level of the control architecture comprised of the individual controllers that would change depending on the UAV's current flight mode. For horizontal forward flight, a closed loop PID controller was implemented, while in vertical hover flight, a feed-forward controller was used in addition to a PID controller. For transition modes, a combination of a proportional controller and a feed-forward controller was used. The last level of the control architecture was an attitude controller which consisted of a PID controller based on quaternions to represent orientations. Other examples of tailsitter UAVs implementing PID or PD controllers are given in [6, 9, 19, 23, 24].

A variation of the traditional PID control strategy was used to study the transition from vertical to horizontal flight of a tailsitter UAV at Tohoku University [6]. They verified that a PID controller did not successfully stabilize the system when a large attitude error was present. This led them to implement a "Resolved Tilt-Twist Angle Feedback Control" method for the attitude error in addition to the PID controller [39]. For this control technique, pitch and yaw errors were formed based on an analogy of the inverted pendulum. This control technique was found to stabilize the UAV in both simulations and experiments.

It is important to note that this vehicle implemented control surfaces in the form of ailerons and elevators, which assisted its transition between vertical and horizontal flight regimes.

The tailsitter UAV developed at the Indian Institute of Technology Kanpur used a non-linear dynamic inversion controller with an outer loop that was based on the translational dynamic equations, an inner loop that was based on the rotational dynamic equations, and a third loop that dynamically allocated control to the individual motors [28]. The contents of the outer and inner loops were different based on the flight regime. For the vertical flight regime, the aerodynamic forces were ignored in the derivation of the model for the control law and the error dynamics for both the inner and outer loops were designed based on a desired damping ratio and natural frequency. These researchers designed their tailsitter UAV to switch between the vertical and horizontal flight controllers at the pre-specified stall angle of attack. The horizontal flight controller was designed taking into account the aerodynamic forces and resembled a traditional fixed-wing airplane controller based on dynamic inversion. Numerical simulations showed the efficacy of this approach.

3.3 Fundamentals of Model Reference Adaptive Control

In this section, we will overview Model Reference Adaptive Control for multi-input multi-output (MIMO) systems. Let the dynamics of the plant be captured by

$$\dot{y}(t) \triangleq Ay(t) + B\Lambda[u(t) + \Theta^{\mathrm{T}}\Phi(t, y)], \qquad y(t_0) = y_0, \qquad t \geq t_0, \tag{3.1}$$

where $y(t) \in \mathbb{R}^n$ denotes the *plant state*, $u(t) \in \mathbb{R}^m$ denotes the *control input*, $B \in \mathbb{R}^{n\times m}$ denotes a known matrix, $A \in \mathbb{R}^{n\times n}$ and $\Lambda \in \mathbb{R}^{m\times m}$ denote unknown constant matrices, the

pair (A, B) is controllable, Λ is diagonal and positive-definite, $\Theta \in \mathbb{R}^{N\times m}$ denotes a constant matrix of unknown coefficients, and $\Phi(t, y) \in \mathbb{R}^N$ denotes the known *regressor vector*, which is continuous in t and Lipschitz continuous in y [36, Ch. 9]. The matrices A and Λ capture the parametric uncertainty of the plant, while $\Theta^{\mathrm{T}}\Phi(t, y)$ represents the matched uncertainty. MRAC laws steer the plant state towards the trajectory of the reference model, which is expressed as

$$\dot{y}_{\mathrm{ref}}(t) \triangleq A_{\mathrm{ref}} y_{\mathrm{ref}}(t) + B_{\mathrm{ref}} r(t), \qquad y_{\mathrm{ref}}(t_0) = y_{\mathrm{ref},0}, \qquad t \geq t_0, \tag{3.2}$$

where $A_{\mathrm{ref}} \in \mathbb{R}^{n\times n}$ is Hurwitz, $B_{\mathrm{ref}} \in \mathbb{R}^{n\times m}$, the pair $(A_{\mathrm{ref}}, B_{\mathrm{ref}})$ is controllable, and $r(t) \in \mathbb{R}^m$ is user-defined and bounded. Let $e(t) \triangleq y(t) - y_{\mathrm{ref}}(t)$, $t \geq t_0$, denote the *trajectory tracking error*. To guarantee that $\lim_{t\to\infty} e(t) = 0$, the adaptive control law is defined as

$$u(t) \triangleq \hat{K}_x^{\mathrm{T}}(t) y(t) + \hat{K}_r^{\mathrm{T}}(t) r(t) - \hat{\Theta}^{\mathrm{T}}(t)\Phi(t, y), \qquad t \geq t_0, \tag{3.3}$$

where $\hat{K}_x(t) \in \mathbb{R}^{n\times m}$, $\hat{K}_r(t) \in \mathbb{R}^{m\times m}$, $\hat{\Theta}(t) \in \mathbb{R}^{N\times n}$ denote estimates of the *adaptive gains*. These adaptive gains will be generated online by integrating the following adaptive laws

$$\dot{\hat{K}}_x(t) \triangleq -\Gamma_x y(t) e^{\mathrm{T}}(t) PB, \qquad \hat{K}_x(t_0) = \hat{K}_{x,0}, \qquad t \geq t_0, \tag{3.4}$$

$$\dot{\hat{K}}_r(t) \triangleq -\Gamma_r r(t) e^{\mathrm{T}}(t) PB, \qquad \hat{K}_r(t_0) = \hat{K}_{r,0}, \tag{3.5}$$

$$\dot{\hat{\Theta}}(t) \triangleq \Gamma_\Theta \Phi(t, y) e^{\mathrm{T}}(t) PB, \qquad \hat{\Theta}(t_0) = \hat{\Theta}_0, \tag{3.6}$$

where $\Gamma_x \in \mathbb{R}^{n\times n}$, $\Gamma_r \in \mathbb{R}^{m\times m}$, and $\Gamma_\Theta \in \mathbb{R}^{N\times N}$ are symmetric, positive-definite, and denote the user-defined *rates of adaptation*, $P = P^{\mathrm{T}} > 0$ is the solution of the algebraic Lyapunov

equation

$$A_{\text{ref}}^{\text{T}} P + P A_{\text{ref}} = -Q, \tag{3.7}$$

and $Q = Q^{\text{T}} > 0$ is user-defined. To prove the existence of a controller in the form of (3.3) such that the plant dynamics (3.1) behave like the ideal reference dynamics (3.2), there must exist gains $K_x \in \mathbb{R}^{n \times m}$ and $K_r \in \mathbb{R}^{m \times m}$ that satisfy the matching conditions

$$A + B\Lambda K_x^{\text{T}} = A_{\text{ref}}, \tag{3.8}$$

$$B\Lambda K_r^{\text{T}} = B_{\text{ref}}. \tag{3.9}$$

For a detailed proof of the effectiveness of the control law (3.3) to guarantee asymptotic convergence of the trajectory tracking error, see pp. 282–285 of [36].

3.4 Feedback Linearization of the Translational Equations

To design a controller for the translational motion of the tailsitter UAV, first we consider the translational dynamics for $z_A(\cdot)$ derived in Chapter 2. Due to the under-actuation of the UAV, only the vertical translation and pitch of the vehicle are controlled directly. The translational equations of motion of the UAV along the vertical direction are captured by (2.10). In this thesis, we assume that the UAV's wings are symmetric and their lift is poorly modeled. Thus, we prefer to consider the UAV's lift force as an external disturbance supporting the controller's effort rather than a term to design through the outer loop. Thus,

we set the control law

$$u_z(t,v) \triangleq -\hat{L}(t,v)\cos\gamma(t) + \hat{D}(t,v)\sin\gamma(t) - \hat{m}g + u_{r,z}(t), \qquad (t,v) \in [t_0,\infty)\times\mathbb{R}^3, \quad (3.10)$$

for $F_z^{\mathbb{I}}(\cdot)$, where $u_z(t,v)$ denotes the *control input*, $u_{r,z}(t)$ denotes the *adaptive control input*, $\hat{L}(t,v)) \triangleq -\frac{1}{2}\hat{\rho}||v(t)||^2\hat{S}\hat{C}_L(\alpha)$ denotes the *estimated lift force*, $\hat{D}(t,v)) \triangleq -\frac{1}{2}\hat{\rho}||v(t)||^2\hat{S}\hat{C}_D(\alpha)$ denotes the *estimated drag force*, $\hat{\rho} > 0$ denotes the *estimated air density*, $\hat{S} > 0$ denotes the *estimated planform area*, $\hat{C}_L : \mathbb{R} \to \mathbb{R}$ denotes the *estimated coefficient of lift*, $\hat{C}_D : \mathbb{R} \to \mathbb{R}$ denotes the *estimated coefficient of drag*, and $\hat{m} > 0$ denotes the *estimated mass*. We also model the relationship between $\hat{C}_L(\cdot)$ and the angle of attack α as

$$\hat{C}_L(\alpha) = \hat{C}_{L,\alpha}\alpha + \hat{C}_{L,0}, \qquad \alpha \in [0, 2\pi), \qquad (3.11)$$

where $\hat{C}_{L,\alpha} > 0$ denotes the *estimated coefficient of lift slope* and $\hat{C}_{L,0} \in \mathbb{R}$ denotes the *estimated coefficient of lift at zero angle of attack* [32, p. 70]. Similarly, the estimated coefficient of drag is modeled as

$$\hat{C}_D(\alpha) \triangleq \hat{C}_{D,\alpha}\alpha + \hat{C}_{D,0}, \qquad \alpha \in [0, 2\pi), \qquad (3.12)$$

where $\hat{C}_{D,\alpha} > 0$ denotes the *estimated coefficient of drag slope* and $\hat{C}_{D,0} \in \mathbb{R}$ denotes the *estimated coefficient of drag at zero angle of attack*.

It follows from (2.10), (3.10), (3.11), and (3.12) that the UAV's equations of motion in the inertial vertical direction can be expressed as

$$\dot{y}_z(t) = A_z y_z(t) + B_z\Lambda_z[u_{r,z}(t) + \Theta_z^{\mathrm{T}}\Phi_z(t,y_z)], \qquad y_z(t_0) = y_{z,0}, \qquad t \geq t_0, \qquad (3.13)$$

where $y_z(t) = \begin{bmatrix} z_A(t) \\ \dot{z}_A(t) \end{bmatrix}$, $A_z = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$, $B_z = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, $\Lambda_z \triangleq \frac{1}{m}$,

$$\Theta_z \triangleq \begin{bmatrix} \hat{\rho}\hat{S}\hat{C}_{L,\alpha} - \rho S C_{L,\alpha} \\ \hat{\rho}\hat{S}\hat{C}_{L,0} - \rho S C_{L,0} \\ \hat{\rho}\hat{S}\hat{C}_{D,\alpha} - \rho S C_{D,\alpha} \\ \hat{\rho}\hat{S}\hat{C}_{D,0} - \rho S C_{D,0} \\ \hat{m} - m \end{bmatrix}, \tag{3.14}$$

$$\Phi_z(t, y_z) \triangleq \begin{bmatrix} \frac{||v(t)||^2 \alpha \cos\gamma(t)}{2} \\ \frac{||v(t)||^2 \cos\gamma(t)}{2} \\ -\frac{||v(t)||^2 \alpha \sin\gamma(t)}{2} \\ -\frac{||v(t)||^2 \sin\gamma(t)}{2} \\ -g \end{bmatrix}. \tag{3.15}$$

The reference model for the $z_A(\cdot)$ dynamics is given by

$$\dot{y}_{\mathrm{z,ref}}(t) \triangleq A_{\mathrm{z,ref}} y_{\mathrm{z,ref}}(t) + B_{\mathrm{ref}} r_z(t), \qquad y_{z,ref}(t_0) = y_{z,ref,0}, \qquad t \geq t_0, \tag{3.16}$$

where $y_{\mathrm{z,ref}}(t) = [z_{A,\mathrm{ref}}(t), \dot{z}_{A,\mathrm{ref}}(t)]^{\mathrm{T}}$,

$$A_{\mathrm{z,ref}} \triangleq \begin{bmatrix} 0 & 1 \\ -K_{p,z} & -K_{d,z} \end{bmatrix}, \tag{3.17}$$

$$B_{\mathrm{ref}} \triangleq \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \tag{3.18}$$

$$r_z(t) \triangleq \ddot{z}_{user}(t) + K_{p,z} z_{user}(t) + K_{d,z} \dot{z}_{user}(t), \tag{3.19}$$

$K_{p,z} > 0$ and $K_{d,z} > 0$ denote the user-defined proportional and derivative gains, respectively, and $z_{\text{user}}(\cdot)$ denotes the user-defined reference trajectory for the UAV's altitude. Next, let

$$u_{r,z}(t) \triangleq \hat{K}_{x,z}^{\mathrm{T}}(t)y_z(t) + \hat{K}_{r,z}^{\mathrm{T}}(t)r_z(t) - \hat{\Theta}_z^{\mathrm{T}}(t)\Phi_z(t, y_z) + K_0 e_z(t) + K_1 \dot{e}_z(t), \qquad t \geq t_0, \tag{3.20}$$

denote the adaptive control input, where $K_0 > 0$ and $K_1 > 0$ denote the baseline controller gains and $e_z(t) \triangleq z_A(t) - z_{A,\text{ref}}(t)$. Applying (3.4)–(3.6), the adaptive gain rates are captured by

$$\dot{\hat{K}}_{x,z}(t) \triangleq -\Gamma_{x,z} y_z(t) e_z^{\mathrm{T}}(t) P_z B, \qquad \hat{K}_{x,z}(t_0) = \hat{K}_{x,z,0}, \qquad t \geq t_0, \tag{3.21}$$

$$\dot{\hat{K}}_{r,z}(t) \triangleq -\Gamma_{r,z} r_z(t) e_z^{\mathrm{T}}(t) P_z B, \qquad \hat{K}_r(t_0) = \hat{K}_{r,0}, \tag{3.22}$$

$$\dot{\hat{\Theta}}_z(t) \triangleq \Gamma_{\Theta,z} \Phi_z(t, y_z) e_z^{\mathrm{T}}(t) P_z B, \qquad \hat{\Theta}_z(t_0) = \hat{\Theta}_{z,0}, \tag{3.23}$$

where $\Gamma_{x,z} \in \mathbb{R}^{2\times 2}$, $\Gamma_{r,z} \in \mathbb{R}$, and $\Gamma_{\Theta,z} \in \mathbb{R}^{5\times 5}$ are symmetric, positive-definite, and denote the user-defined *rates of adaptation*, $P_z = P_z^{\mathrm{T}} > 0$ is the solution of the algebraic Lyapunov equation

$$A_{\text{ref,z}}^{\mathrm{T}} P_z + P_z A_{\text{ref,z}} = -Q_z, \tag{3.24}$$

and $Q_z = Q_z^{\mathrm{T}} > 0$.

3.5 Feedback Linearization of the Rotational Equations

It follows from from (2.13) that the rotational equations of motion for the pitch of the tailsitter UAV can be expressed as

$$\dot{y}_\theta(t) \triangleq A_{\text{rot}} y_\theta(t) + B_{\text{rot}} \Lambda_{\text{rot}} u_{\text{rot}}(t), \qquad y_\theta(t_0) = y_{\theta,0}, \qquad t \geq t_0, \tag{3.25}$$

where $y_\theta(t) \triangleq \begin{bmatrix} \theta(t) \\ \dot{\theta}(t) \end{bmatrix}$, $B_{\text{rot}} \triangleq \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, $A_{\text{rot}} \triangleq \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$, and $\Lambda_{\text{rot}} \triangleq \frac{1}{I_y}$. To control the pitch dynamics of the UAV a reference system is established in the form

$$\dot{y}_{\theta,\text{ref}}(t) \triangleq A_{\text{rot,ref}} y_{\theta,\text{ref}}(t) + B_{\text{rot}} r_{\text{rot}}(t), \qquad y_{\theta,\text{ref}}(t_0) = y_{\theta,\text{ref},0}, \qquad t \geq t_0, \tag{3.26}$$

where $y_{\theta,\text{ref}}(t) \triangleq [\theta_{\text{ref}}(t), \dot{\theta}_{\text{ref}}(t)]^{\text{T}}$,

$$A_{\text{rot,ref}} \triangleq \begin{bmatrix} 0 & 1 \\ -K_{\text{p,rot}} & -K_{\text{d,rot}} \end{bmatrix}, \tag{3.27}$$

$$r_{\text{rot}}(t) \triangleq \ddot{\theta}_{\text{user}}(t) + K_{\text{p,rot}} \theta_{\text{user}}(t) + K_{\text{d,rot}} \dot{\theta}_{\text{user}}(t), \tag{3.28}$$

$K_{\text{p,rot}}$ and $K_{\text{d,rot}}$ denote the user-defined proportional and derivative gains, respectively. The rotational control input is given by

$$u_{rot}(t) \triangleq \hat{K}_x^{\text{T}}(t) y_\theta(t) + K_2 e_\theta(t) + K_3 \dot{e}_\theta(t), \qquad t \geq t_0, \tag{3.29}$$

where $K_2 > 0$ and $K_3 > 0$ denote the baseline controller gains, $e_\theta(t) \triangleq \theta(t) - \theta_{\text{ref}}(t)$, and

$$\dot{\hat{K}}_x(t) = -\Gamma_{\text{rot}} y_\theta(t) e_\theta^{\text{T}}(t) P_{\text{rot}} B_{\text{rot}}, \qquad \hat{K}_x(t_0) = \hat{K}_{x,0}, \tag{3.30}$$

denotes the estimated adaptive gain rate, $\Gamma_{\rm rot} \in \mathbb{R}^{2\times 2}$ is symmetric, positive-definite, and denotes the user-defined *rate of adaptation*, $P_{\rm rot} = P_{\rm rot}^{\rm T} > 0$ is the solution of the algebraic Lyapunov equation

$$A_{\rm rot,ref}^{\rm T} P_{\rm rot} + P_{\rm rot} A_{\rm rot,ref} = -Q_{\rm rot}, \tag{3.31}$$

and $Q_{\rm rot} = Q_{\rm rot}^{\rm T} > 0$ is user-defined.

3.6 Outer Loop Design

In this section, we outline the outer loop system of our tailsitter UAV. The purpose of the outer loop is to ensure that the aircraft travels along the X axis of the inertial reference frame $\mathbb{I}$ at all times. We will design this system by controlling the roll angle $\phi(t)$, $t \geq t_0$. The strategy used in this section is similar to the outer loop strategies used in [29, 40], and we will assume that $\phi(t)$, $t \geq t_0$ and $\psi(t)$ are sufficiently small. Linearizing (2.1) and (2.3), it follows that

$$\begin{bmatrix} \ddot{x}_A(t) \\ \ddot{y}_A(t) \end{bmatrix} \triangleq \frac{1}{m} \begin{bmatrix} 0 & 0 \\ r_{2,1}(t, v(t)) & r_{2,2}(t, v(t)) \end{bmatrix} \begin{bmatrix} \phi(t) \\ \psi(t) \end{bmatrix},$$
$$\begin{bmatrix} x_A(t_0) \\ y_A(t_0) \end{bmatrix} = \begin{bmatrix} x_{A,0} \\ y_{A,0} \end{bmatrix}, \qquad \begin{bmatrix} \dot{x}_A(t_0) \\ \dot{y}_A(t_0) \end{bmatrix} = \begin{bmatrix} \dot{x}_{A,0} \\ \dot{y}_{A,0} \end{bmatrix}, \qquad t \geq t_0, \tag{3.32}$$

where

$$r_{2,1}(t, v(t)) \triangleq (u_1(t) - L(t, v)\sin\alpha(t) + D(t, v)\cos\alpha(t))\cos\theta(t) + (L(t, v)\cos\alpha(t) + D(t, v)\sin\alpha(t))\sin\theta(t),$$
$$r_{2,2}(t, v(t)) \triangleq -D(t, v)\sin\alpha(t) - L(t, v)\cos\alpha(t).$$

From (3.32), it follows that in first approximation, the $x_A(\cdot)$ dynamics do not depend on the roll or yaw angles. Therefore we will choose to only control the $y_A(\cdot)$ dynamics and leave the $x_A(\cdot)$ dynamics uncontrolled. This is a reasonable decision since this work is concerned with the transition between horizontal and vertical flight modes. Considering $\phi(\cdot)$ as a virtual control input for the second term of (3.32), we design the reference angle

$$\phi_{\text{ref}}(t) = \frac{1}{\hat{r}_{2,1}(t, v(t))} \left(\hat{\phi}_{\text{ref}}(t) - \hat{r}_{2,2}(t, v(t))\psi(t) \right), \qquad t \geq t_0, \tag{3.33}$$

where $\hat{r}_{2,1}(\cdot,\cdot)$ and $\hat{r}_{2,2}(\cdot,\cdot)$ denote estimates for $r_{2,1}(\cdot,\cdot)$ and $r_{2,2}(\cdot,\cdot)$, respectively. We also can denote $r_{2,1}(t, v(t)) = \eta_1(t)\hat{r}_{2,1}(t, v(t))$ and $r_{2,2}(t, v(t)) = \eta_2(t)\hat{r}_{2,2}(t, v(t))$. Since $r_{2,1}(\cdot,\cdot)$ is in the denominator of $\phi_{\text{ref}}(\cdot)$, we must determine if there are any flight conditions in which $r_{2,1}(t, v(t)) = 0$, $(t, v(t)) \in [t_0, \infty) \times \mathbb{R}^3$. To this goal, let $r_{2,1}(t, v(t)) = A(t, v(t)) + B(t, v(t))$, $(t, v(t)) \in [t_0, \infty) \times \mathbb{R}^3$, where

$$\begin{aligned} A(t, v(t)) &\triangleq (u_1(t) - L(t, v)\sin\alpha(t) + D(t, v)\cos\alpha(t))\cos\theta(t), \\ B(t, v(t)) &\triangleq (L(t, v)\cos\alpha(t) + D(t, v)\sin\alpha(t))\sin\theta(t). \end{aligned}$$

As shown in Figure 3.2, $r_{2,1}(\cdot,\cdot)$ denotes the sum of the aerodynamic and thrust forces acting along the X axis. Therefore, assuming that the UAV does not move in the plane containing the Y and Z axes, $r_{2,1}(t, v(t)) \neq 0$, $(t, v(t)) \in [t_0, \infty) \times \mathbb{R}^3$.

To generate a reference roll angle which can handle uncertainties in the system's parameters, we use a linear-quadratic baseline controller augmented by an MRAC law with disturbance rejection. A proof of this control technique is provided in [36, pp. 303 – 311]. It follows from (3.33) and (3.32), that the $y_A(\cdot)$ dynamics can be expressed in the form

$$\dot{x}(t) = Ax(t) + B\Lambda[\hat{\phi}_{\text{ref}}(t) + \Theta^{\mathrm{T}}\Phi(t) + d(t)], \qquad x(t_0) = x_0, \qquad t \geq t_0, \tag{3.34}$$

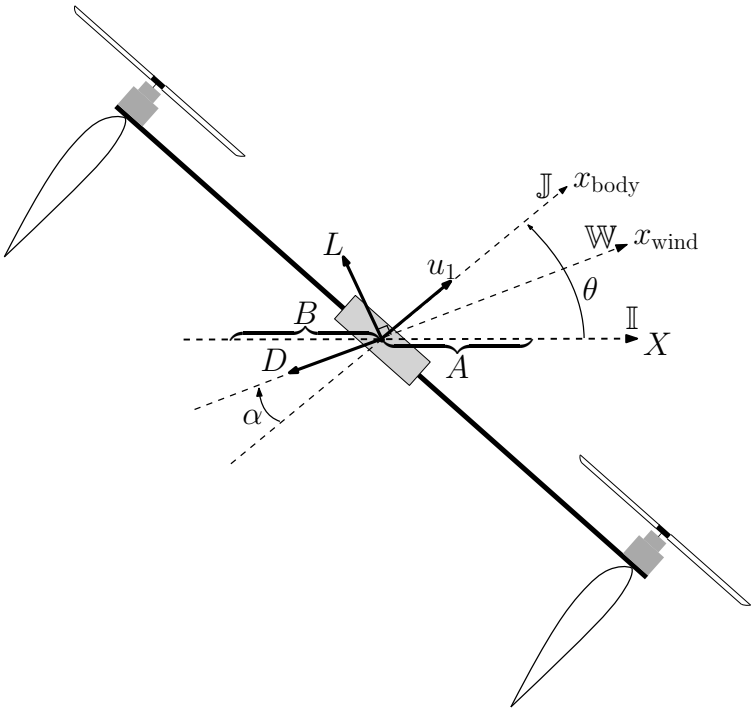

Figure 3.2: Visual depiction of $r_{2,1}(\cdot,\cdot)$

$$\text{with } x(t) = \begin{bmatrix} \int_{t_0}^{t} y_A(\tau)\mathrm{d}\tau \\ y_A(t) \\ \dot{y}_A(t) \end{bmatrix}, A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \Lambda \triangleq \frac{\hat{\eta}_1}{m},$$

$$\Theta = 1, \tag{3.35}$$

$$\Phi(t) = -\hat{r}_{2,2}(t, v(t))\psi(t), \tag{3.36}$$

$$d(t) = (\eta_1(t) - \hat{\eta}_1)\hat{\phi}_{\text{ref}} + \frac{\eta_2(t)}{\eta_1(t)}\hat{r}_{2,2}\psi, \tag{3.37}$$

$$\hat{\phi}_{\text{ref}}(t) = u_{bl}(t) + u_{ad}(t), \tag{3.38}$$

$$u_{bl}(t) = -K_x^{\mathrm{T}} x(t), \tag{3.39}$$

$$u_{ad}(t) = -\hat{K}_u^{\mathrm{T}}(t)u_{bl}(t) - \hat{\Theta}\Phi(t), \tag{3.40}$$

$\hat{\eta}_1$ denotes a time-invariant estimate of $\eta_1(t)$, $e_y(t) \triangleq y_A(t) - y_{A,\text{ref}}(t)$ denotes the *state error*,

$K_x^{\mathrm{T}} = R^{-1}B_y^{\mathrm{T}}P_y$ denotes the *LQR-baseline controller gain*, $R \in \mathbb{R}$ is symmetric, positive-definite, and user-defined, $P_y \in \mathbb{R}^{3\times 3} > 0$ is the symmetric, positive definite solution of the algebraic Riccati equation

$$0 = A^{\mathrm{T}}P_y + P_yA + Q_y - P_yB_yR^{-1}B^{\mathrm{T}}P, \tag{3.41}$$

$Q_y \in \mathbb{R}^{3\times 3} > 0$ is symmetric, positive-definite, and user-defined,

$$\dot{\hat{K}}_u(t) = \Gamma_u u_{bl} e_y^{\mathrm{T}}(t) P_{\mathrm{ref}} B_y, \qquad \hat{K}_u(t_0) = \hat{K}_{u,0}, \tag{3.42}$$

$$\dot{\hat{\Theta}}(t) = \Gamma_\Theta \Phi(t) e_y^{\mathrm{T}}(t) P_{\mathrm{ref}} B_y, \qquad \hat{\Theta}(t_0) = \hat{\Theta}_0, \tag{3.43}$$

$\Gamma_u \in \mathbb{R}^{3\times 3}$, $\Gamma_\Theta \in \mathbb{R}$ are symmetric, positive-definite, and denote the user-defined *rates of adaptation*, $P_{\mathrm{ref}} \in \mathbb{R}^{3\times 3} > 0$ is the symmetric, positive definite solution of the algebraic Lyapunov equation

$$A_{\mathrm{ref}}^{\mathrm{T}}P_{\mathrm{ref}} + P_{\mathrm{ref}}A_{\mathrm{ref}} = -Q_{\mathrm{ref}}, \tag{3.44}$$

and $Q_{\mathrm{ref}} \in \mathbb{R}^{3\times 3} > 0$ is symmetric, positive-definite, and user-defined. This adaptive control technique will generate a reference roll angle that will drive the system back to the inertial X axis despite the uncertainties in the system's parameters.

Once the reference roll $\phi_{\mathrm{ref}}(\cdot)$ angle has been established, we must command the roll angle $\phi(\cdot)$ of the vehicle to follow its reference angle. To this goal, we will apply the PID control law

$$u_2(t) \triangleq K_{p,\phi}e_\phi(t) + K_{i,\phi}\int_{t_0}^{t} e_\phi(\tau)\mathrm{d}\tau + K_{d,\phi}\dot{e}_\phi(t), \qquad t \geq t_0, \tag{3.45}$$

where $K_{p,\phi} > 0$, $K_{i,\phi} > 0$, and $K_{i,\phi} > 0$ denote the user-defined proportional, integral and

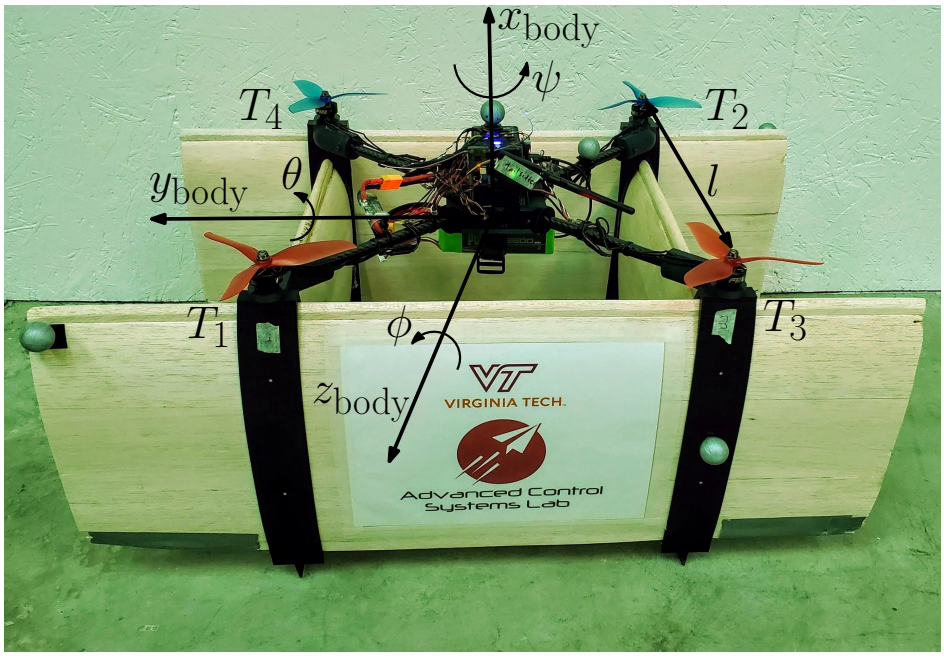

Figure 3.3: Body reference frame and motor numbering for the tailsitter UAV

derivative gains, respectively, and $e_\phi(t) \triangleq \phi(t) - \phi_{\text{ref}}(t)$.

Similarly, it follows from (2.7) that $u_4(t)$, $t \geq t_0$, denotes the control input for the yaw dynamics of the vehicle $\psi(\cdot)$, which is given by

$$u_4(t) \triangleq K_{p,\psi} e_\psi(t) + K_{i,\psi} \int_{t_0}^{t} e_\psi(\tau) \mathrm{d}\tau + K_{d,\psi} \dot{e}_\psi(t), \qquad t \geq t_0, \tag{3.46}$$

where $K_{p,\psi} > 0$, $K_{i,\psi} > 0$, and $K_{d,\psi} > 0$ denote the user-defined proportional, integral, and derivative gains, respectively, which will steer the yaw angle of the tailsitter to zero, and $e_\psi(t) \triangleq \psi(t) - \psi_{\text{ref}}(t)$.

3.7 Realization of Required Force and Moments

In this section, we compute the thrust force each motor must produce to realize the desired control inputs. Analyzing the UAV's geometry from Figure 3.3, we deduce that

$$\begin{bmatrix} T_1(t) \\ T_2(t) \\ T_3(t) \\ T_4(t) \end{bmatrix} \triangleq \begin{bmatrix} \frac{1}{4} & -\frac{1}{2l} & \frac{1}{2l} & \frac{1}{4k_T} \\ \frac{1}{4} & \frac{1}{2l} & -\frac{1}{2l} & \frac{1}{4k_T} \\ \frac{1}{4} & \frac{1}{2l} & \frac{1}{2l} & -\frac{1}{4k_T} \\ \frac{1}{4} & -\frac{1}{2l} & -\frac{1}{2l} & -\frac{1}{4k_T} \end{bmatrix} \begin{bmatrix} \frac{-u_z(t,v)}{\sin\theta(t)} \\ u_2(t) \\ u_{rot}(t) \\ u_4(t) \end{bmatrix}, \qquad t \geq t_0, \tag{3.47}$$

where $-\frac{u_z(\cdot,\cdot)}{\sin\theta(\cdot)}$ denotes the required thrust force along the $x_{\text{body}}(\cdot)$ axis given the desired control input along the Z inertial axis and the current pitch angle, $T_1(\cdot)$, $T_2(\cdot)$, $T_3(\cdot)$, and $T_4(\cdot)$ denote the thrust produced by each of the propellers, $k_T > 0$ denotes the *drag coefficient of the propellers*, and $l > 0$ denotes the length between the propellers along the $z_{\text{body}}(\cdot)$ axis. We recognize that $\frac{-u_z(\cdot,\cdot)}{\sin\theta(\cdot)}$ will become undefined when the pitch angle is 0 degrees. Logically, if we were pitched all the way to 0 degrees, no amount of force along the $x_{\text{body}}(\cdot)$ axis would satisfy our desired force in the Z inertial axis since the motors would be oriented perpendicular to the Z inertial axis. For this reason, and due to our symmetric airfoil design discussed in Chapter 5, we choose to retain a small pitch angle during horizontal flight tests. Equation (3.47) provides a static relation between control inputs and propellers' thrust force. This relation can be replaced by a dynamic relation that accounts for the motors' dynamics.

Applying these thrust forces, the resulting forces in the inertial reference frame $\mathbb{I}$ and

pitching moment about the $y_{\text{body}}(\cdot)$ axis are given by

$$F_x^{\mathcal{I}}(t) \triangleq (T_a(t) + T_b(t)) \cos\theta(t), \qquad t \geq t_0, \tag{3.48}$$

$$F_z^{\mathcal{I}}(t) \triangleq -(T_a(t) + T_b(t)) \sin\theta(t), \tag{3.49}$$

$$M_y(t) \triangleq \frac{l}{2}(T_b(t) - T_a(t)), \tag{3.50}$$

respectively, where $T_a(t) \triangleq T_2(t) + T_4(t)$ and $T_b(t) \triangleq T_1(t) + T_3(t)$. We then apply the control inputs to our equations of motion, given by (2.9), (2.10), and (2.13), to determine the translational and rotational accelerations of the tailsitter UAV.

3.8 Conclusion

In this chapter, we designed a control algorithm that will enable a tailsitter UAV to transition between vertical and horizontal flight regimes. To begin, we reviewed the current literature on controlling a tailsitter UAV and detailed many different control methods applied to tailsitter UAVs. From this, we found that tailsitter UAV control is a relatively novel research topic and early tailsitter UAV were controlled by a pilot rather than an autonomous algorithm. Some of the first tailsitter UAVs to be controlled autonomously were the T-wing developed at the University of Sydney, which used gain scheduled controllers, and the tailsitter developed at the University of Compiegne, which used feedback linearization. Many of the other papers reviewed used variations of PID controllers for the tailsitter UAV. We then proceeded to introduce the fundamentals of MRAC and showed the adaptive control law (3.3), which is a function of the adaptive gains. Following from our $z_A(\cdot)$ equation of motion (2.10) we performed feedback linearization and derived our ideal adaptive control law (3.20). A similar process was performed for our pitch dynamics that resulted in the

ideal adaptive control law for $\theta(\cdot)$, (3.29). Next, we developed a controller for the outer loop to determine the control input $u_2(\cdot)$ necessary to maintain the desired $y_A(\cdot)$ position during all flight regimes, (3.45). These four control inputs are then multiplied by the inverse of the mixer, given in (3.47), to determine the necessary thrusts of each propeller to follow the desired reference trajectories, which completes the control loop. In this next chapter, we will outline the numerical simulations conducted to test our control law.

Chapter 4

Numerical Validation of the Proposed Control Systems

4.1 Introduction

In this chapter, the control system presented in Chapter 3 will be tested through numerical simulations. These simulations will ensure that the controller is effective and allow us to begin tuning the adaptive rates and other user-defined parameters underlying the adaptive laws. First, we will overview the user-defined trajectory chosen for these simulations. Following that, the results of the numerical simulations performed in the Matlab environment are presented. Successively, we present the results of the outer loop tests conducted in the Matlab environment. These simulations are an important stepping stone to the real flight tests presented in Chapter 5.

4.2 Reference Trajectories

To perform flight simulations, reference trajectories have been outlined by employing quintic polynomials [41, pp. 191-192]. This method allows us to produce reference trajectories that meet user-defined constraints on the initial and final positions, velocities, and

accelerations. Moreover, this technique guarantees that the reference acceleration is continuously differentiable at all time and hence, the reference trajectory the UAV follows is sufficiently smooth.

To compute reference trajectories as quintic polynomials, a set of coefficients $a = [a_0, a_1, a_2, a_3, a_4, a_5]^{\mathrm{T}} \in \mathbb{R}^6$ is computed offline as the solution of the following linear algebraic equation

$$\begin{bmatrix} 1 & t_0 & t_0^2 & t_0^3 & t_0^4 & t_0^5 \\ 0 & 1 & 2t_0 & 3t_0^2 & 4t_0^3 & 5t_0^4 \\ 0 & 0 & 2 & 6t_0 & 12t_0^2 & 20t_0^3 \\ 1 & t_f & t_f^2 & t_f^3 & t_f^4 & t_f^5 \\ 0 & 1 & 2t_f & 3t_f^2 & 4t_f^3 & 5t_f^4 \\ 0 & 0 & 2 & 6t_f & 12t_f^2 & 20t_f^3 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} = \begin{bmatrix} q_0 \\ v_0 \\ \alpha_0 \\ q_f \\ v_f \\ \alpha_f \end{bmatrix}, \tag{4.1}$$

where t_0, $t_f \in \mathbb{R}$ denote the initial and final times, respectively, $t_0 \neq t_f$, q_0, $q_f \in \mathbb{R}$ denote the initial and final positions, respectively, v_0, $v_f \in \mathbb{R}$ denote the initial and final velocities, respectively, and a_0, $a_f \in \mathbb{R}$ denote the initial and final accelerations, respectively. Then, the user-defined trajectory along a given direction is calculated as

$$q(t) = a^{\mathrm{T}}[1, t, t^2, t^3, t^4, t^5]^{\mathrm{T}}, \qquad t \geq t_0, \tag{4.2}$$

$$\dot{q}(t) = a^{\mathrm{T}}[0, 1, 2t, 3t^2, 4t^3, 5t^4]^{\mathrm{T}}, \tag{4.3}$$

$$\ddot{q}(t) = a^{\mathrm{T}}[0, 0, 2, 6t, 12t^2, 20t^3]^{\mathrm{T}}. \tag{4.4}$$

This strategy will be applied to generate user-defined trajectories for both the vertical position $z_A(\cdot)$ and the pitch angle $\theta(\cdot)$ of the tailsitter UAV. Having set a reference trajectory, our MRAC controller for $z_A(\cdot)$ and $\theta(\cdot)$, along with our outer loop controller, is then used to

determine the necessary control inputs for the actual system to track the reference system.

4.3 Numerical Simulations in the Matlab Environment

To test the proposed control system, we first developed a simulation in Matlab. This simulation employed the built-in *ode23* function in Matlab to apply the Runge-Kutta scheme and integrate our vehicle equations of motion (2.9), (2.10), and (2.13), the reference model dynamics (3.16), (3.26), and the adaptive laws (3.21), (3.22), (3.23), and (3.30) at every time step. To evaluate the robustness of our controller, we performed two sets of tests considering that the estimated values of the vehicles mass, lift coefficient, drag coefficient, planform area, and the air's density are 90% and 50% of their true vales, which were measured either directly or through high-fidelity computer aided design (CAD) models. These tests allowed us to verify the controllers' ability to stabilize the system despite these parametric uncertainties. For the vehicle parameters and controller gains used during the Matlab simulations, see Appendix A.

Since the scope of this thesis is to design control algorithms to allow a successful transition between vertical and horizontal flight regimes, our simulations focused on this scenario. The user-defined trajectory strategy for the vertical position expressed in meters is as follows

$$z_{\text{user}}(t) = \begin{cases} \frac{1}{2}t^2, & 0 \leq t < 1, \\ q_1(t), & 1 \leq t < 4, \\ 3, & 4 \leq t < 12, \\ q_2(t), & 12 \leq t < 15, \\ q_3(t), & 15 \leq t < 19, \\ 0, & 19 \leq t, \end{cases} \tag{4.5}$$

where $q_1(t) = q(\cdot)$ given by (4.2), with $q_0 = 0.5\text{m}$, $v_0 = 1\text{m/s}$, $a_0 = 1\text{m/s}^2$, $q_f = 3\text{m}$, $v_f = 0\text{m/s}$, $a_f = 0\text{m/s}^2$, $q_2(t) = q(\cdot)$ given by (4.2), with $q_0 = 3\text{m}$, $v_0 = 0\text{m/s}$, $a_0 = 0\text{m/s}^2$, $q_f = 3.5\text{m}$, $v_f = 0\text{m/s}$, $a_f = 0\text{m/s}^2$, and $q_3(t) = q(\cdot)$ given by (4.2), with $q_0 = 3.5\text{m}$, $v_0 = 0\text{m/s}$, $a_0 = 0\text{m/s}^2$, $q_f = 0\text{m}$, $v_f = 0\text{m/s}$, $a_f = 0\text{m/s}^2$. The user-defined trajectory strategy for the pitch angle expressed in degrees is as follows

$$\theta_{\text{user}}(t) = \begin{cases} 90, & 0 \leq t < 1, \\ q_4(t), & 1 \leq t < 4, \\ 5, & 4 \leq t < 12, \\ q_5(t), & 12 \leq t < 15, \\ 90, & 15 \leq t, \end{cases} \tag{4.6}$$

where $q_4(t) = q(\cdot)$ given by (4.2), with $q_0 = 90$ degrees, $v_0 = 0$ degrees/s, $a_0 = 0$ degrees/s^2, $q_f = 5$ degrees, $v_f = 0$ degrees/s, $a_f = 0$ degrees/s^2, and $q_5(t) = q(\cdot)$ given by (4.2), with $q_0 = 5$ degrees, $v_0 = 0$ degrees/s, $a_0 = 0$ degrees/s^2, $q_f = 90$ degrees, $v_f = 0$ degrees/s, $a_f = 0$ degrees/s^2.

The user-defined trajectories (4.5) and (4.6) capture the following mission scenario: our simulations began with the vehicle sitting on its tail, and we instructed the tailsitter UAV to accelerate vertically for 1s before commencing the transition to the horizontal flight regime. The transition to the horizontal flight regime then lasted for 4s before we instructed the vehicle to remain at its current altitude of 3m for 7s of horizontal flight. We then instructed the vehicle to transition back to vertical flight, and during this transition we allowed the vehicle to increase its altitude to 3.5m, which created a smooth transition. The vehicle then landed following a quintic polynomial trajectory for the vertical position $z_A(\cdot)$ to ensure a soft landing.

Figures 4.1 and 4.3 show the UAV's actual trajectory and pitch angle, the reference trajec-

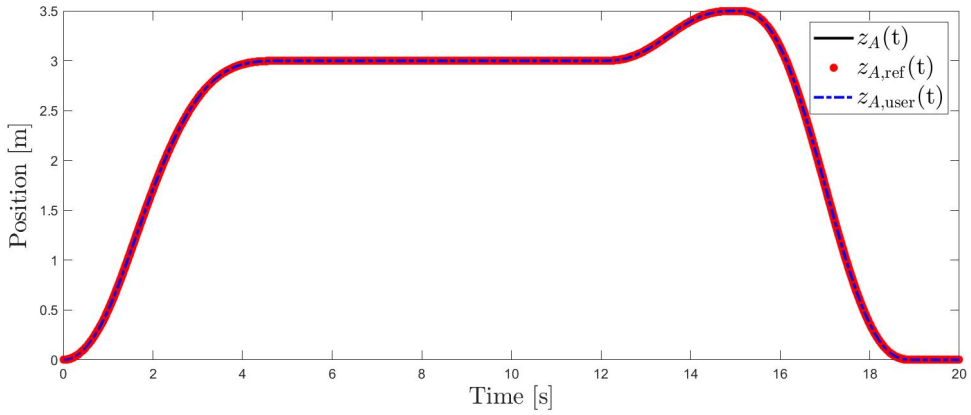

Figure 4.1: Matlab simulation of transition for $z_A(\cdot)$ with 90% confidence in system's parameters and using the proposed MRAC laws.

tory and pitch angle given by (3.16) and (3.26), respectively, and the user-defined trajectory and pitch angle given by (4.5) and (4.6), respectively, assuming a 90% confidence level on all parametric uncertainties. As shown by these figures, the tailsitter UAV successfully transitioned to the horizontal flight regime, maintained horizontal flight, and transitioned back to the vertical flight regime using the proposed control architecture. The trajectory tracking error $e_z(\cdot)$ was identically zero throughout the simulation, as shown in Figure 4.2. The maximum pitch angle trajectory tracking error $e_\theta(\cdot)$ was 0.65 degrees, as shown in Figure 4.4. It can also be seen that the reference trajectory was able to follow the user-defined trajectory with great accuracy in both cases. The

The plot for the vertical velocity, the vertical reference velocity, and the user-defined vertical velocity is shown in Figure 4.5. This plot shows that the vehicle was able to achieve the reference and user-defined velocities throughout the entire simulation of the transition between vertical and horizontal flight regimes. The pitch angular velocity during the transition between horizontal and vertical flight regimes is shown in Figure 4.6. This plot shows that the pitch rate of the vehicle was able to perfectly follow the reference and user-defined angular velocities.

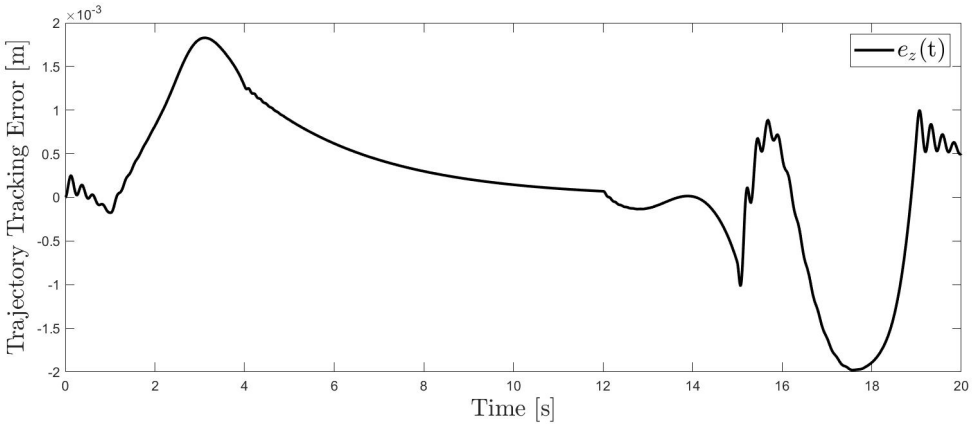

Figure 4.2: Matlab simulation of transition for $e_z(\cdot)$ with 90% confidence in system's parameters and using the proposed MRAC laws.

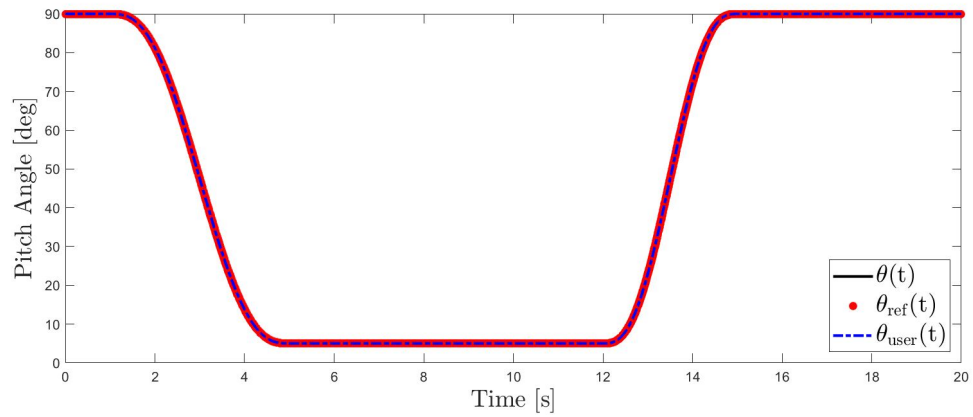

Figure 4.3: Matlab simulation of transition for pitch angle with 90% confidence in system's parameters and using the proposed MRAC laws.

While we leave the translational motion along the $x_A(\cdot)$ direction uncontrolled, it is important to observe the resulting motion in the X inertial direction. Figure 4.7 shows the the trajectory profile that the vehicle undertook over these simulations. The vehicle travelled 120m in the X inertial direction, which will be acceptable for our real flight tests because of the size of the outdoor drone cage in which the flight tests will be performed.

The final plot of significance from the Matlab simulation using the MRAC controller is the plot of the thrust force needed to meet the reference trajectory, shown in Figure 4.8.

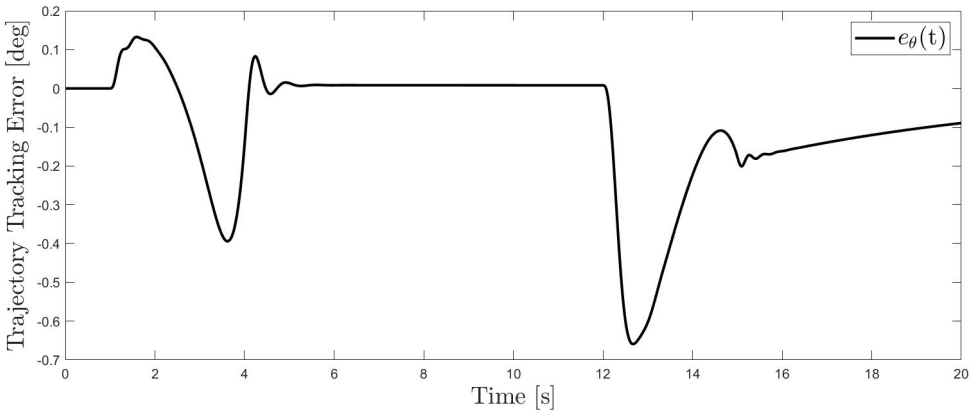

Figure 4.4: Matlab simulation of transition for pitch angle error with 90% confidence in system's parameters and using the proposed MRAC laws.

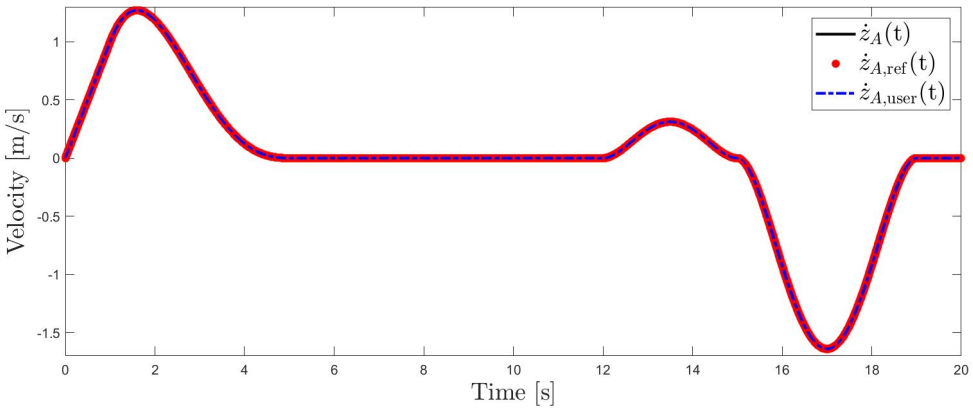

Figure 4.5: Matlab simulation of transition for $\dot{z}_A(\cdot)$ with 90% confidence in system's parameters and using the proposed MRAC laws.

This result was critical in choosing the motor and propeller combination that would be used for our tailsitter UAV, as will be discussed in Chapter 5. This plot shows that the maximum force required from our motors was about 33N, while the force required to maintain horizontal flight was about 25N. This plot also shows our individual $T_A(\cdot)$ and $T_B(\cdot)$ required to make the transition from vertical to horizontal flight regimes and vice versa.

The same transition maneuver was simulated in Matlab using a PID controller. The PID controller was manually tuned by first varying the proportional gain to achieve sustained

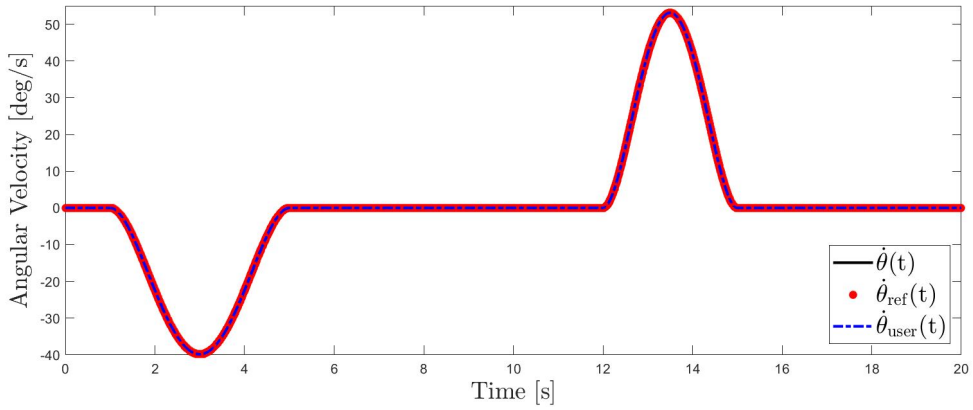

Figure 4.6: Matlab simulation of transition for pitch angular velocity with 90% confidence in system's parameters and using the proposed MRAC laws.

oscillation. Subsequently, the derivative gain was tuned to dampen the system's response. Finally, the integral gain was tuned to reduce the steady state error. It is worthwhile to recall alternative techniques to tune a PID controller such as Ziegler-Nichols, lead, lag, and lead-lag to name a few [42]. For details on the gains used perform these simulations, see Appendix A. As shown in Figures 4.9–4.12, the tracking errors are considerably larger. Therefore, the benefits of the PID controller of lower computational time do no justify the use of these controllers. As shown in Figures 4.11 and 4.12, the pitch angle converged to the reference pitch angle with an overshoot of 2 degrees, however controller failed to guarantee satisfactory performance in forward flight. The largest trajectory tracking error for $z(\cdot)$ during this simulation was 0.55m. This result confirms the need for an adaptive controller to be use for a nonlinear system undergoing a complex maneuver.

To further show the need for an adaptive controller to control the transition between vertical and horizontal flight regimes of a tailsitter UAV, we conducted Matlab tests with lower confidence in our system's parameters. For these simulations, we lowered the estimated parameter values to be 50% of the actual parameter values, while leaving the other simulation parameters unchanged. This was a useful test because while some system param-

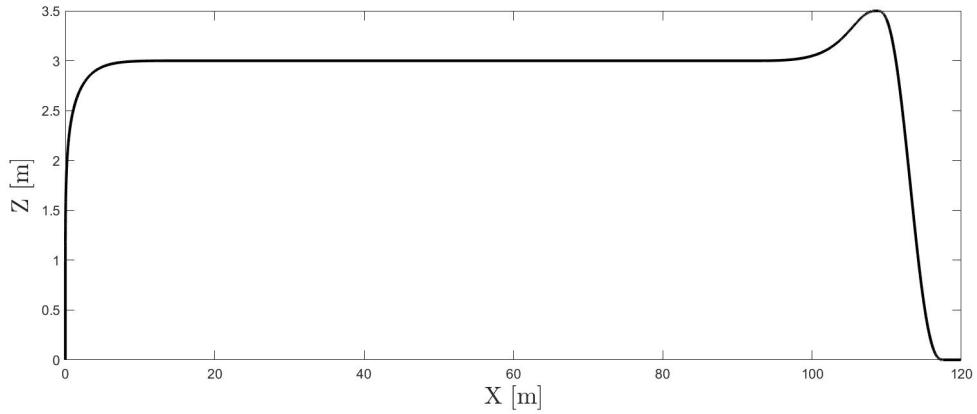

Figure 4.7: Position of the UAV in the X-Z plane over a vertical take off, horizontal flight, and a vertical landing maneuver with 90% confidence in system's parameters and using the proposed MRAC laws.

eters, such as the mass, were easily determined on the actual vehicle, other parameters, such as the aerodynamic coefficients, were more difficult to determine accurately. It is shown in Figures 4.13 and 4.14 that the MRAC controller was able to overcome these uncertainties and stabilize the system's vertical position $z_A(\cdot)$ with a maximum trajectory tracking error of 0.18m during transition flight.

The simulation with 50% confidence in the system parameters was run again, but with the PID controller. Figures 4.15 and 4.16 show that the PID controller was unable to stabilize the system during the transition flight and resulted in a maximum trajectory tracking error of 1m and a steady state error of 0.4m during horizontal flight. This is due to the lower robustness levels of the PID control technique, once again confirming the need for an adaptive controller to be used.

The final simulations completed in the Matlab environment were tests of the outer loop control architecture. As previously stated, the purpose of the outer loop was to ensure the vehicle retains its position on the X inertial axis by controlling the roll angle of the vehicle. For all of the previous simulations in Matlab, we set the initial position for $y_A(t_0) = 0$ and

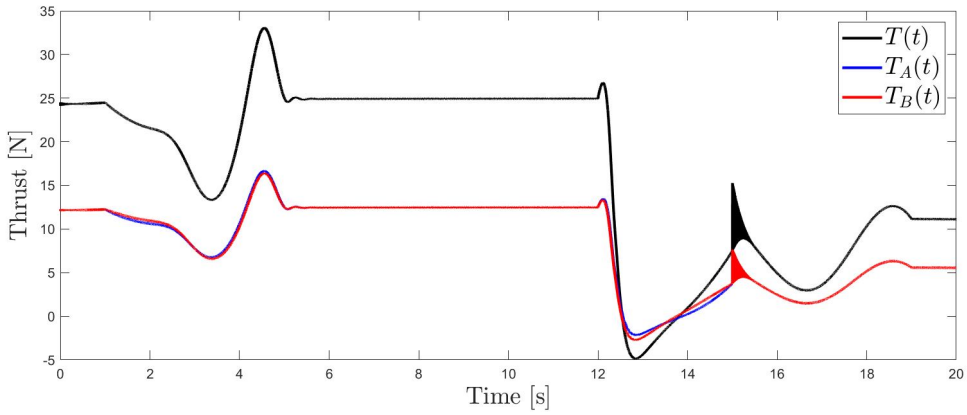

Figure 4.8: Matlab simulation of transition forces with 90% confidence in system's parameters and using the proposed MRAC laws.

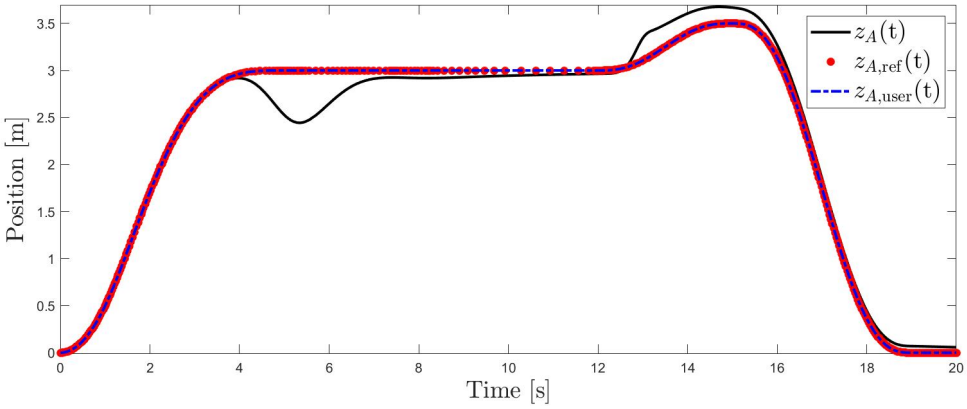

Figure 4.9: Matlab simulation of transition for $z_A(\cdot)$ with 90% confidence in system's parameters and using a PID control law.

since there were no external disturbances, the vehicle was able to hold that position through the entire flight. In this case, the outer loop was never actually used, which would also be the case for real flight tests if we flew in ideal conditions. Since we anticipated that we will not be able to operate the vehicle in ideal conditions, we designed some simulations in Matlab to test the outer loop controller. For the first test, the vehicle had an initial starting position of $y_A(t) = 1\text{m}$, $t = 0\text{s}$, and we observed the outer loop controller's ability to steer the vehicle back to the inertial X axis. The $z_A(\cdot)$ and $\theta(\cdot)$ trajectories during these simulations were

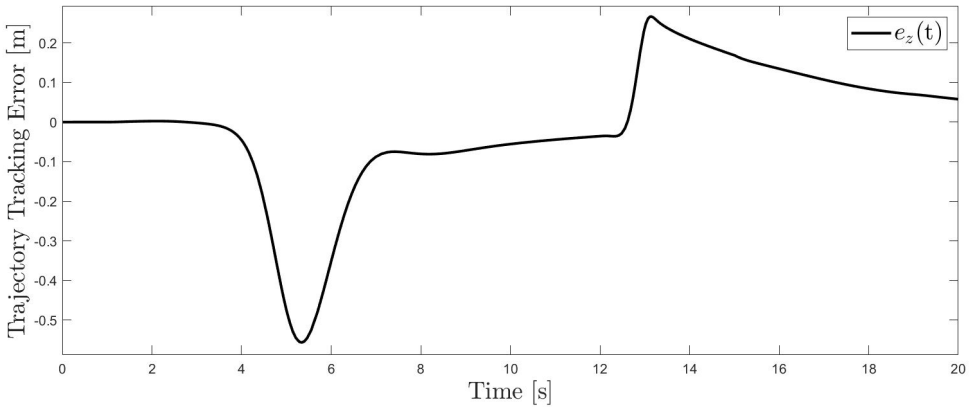

Figure 4.10: Matlab simulation of transition for $e_z(\cdot)$ with 90% confidence in system's parameters and using a PID control law.

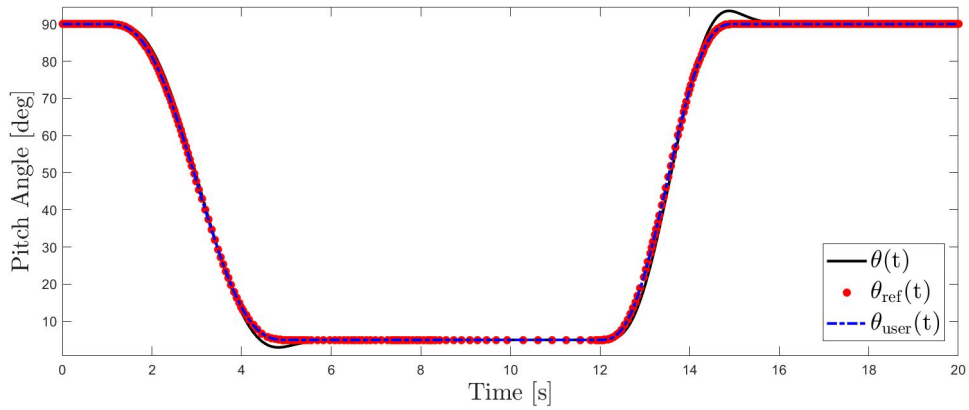

Figure 4.11: Matlab simulation of transition for pitch angle with 90% confidence in system's parameters and using a PID control law.

the same as the previous simulations. It is shown in Figure 4.17 that the vehicle was able to return to $y_A(\cdot) = 0$ within 4s during the simulation. Since we are indirectly controlling the $y_A(\cdot)$ position with the roll angle of the vehicle, we did not expect fast response times during simulations or real flight tests.

The roll angle and reference roll angle which drove the vehicle back to the inertial X axis is shown in Figure 4.18. The roll angle of the vehicle was able to follow the reference roll angle, guiding the vehicle back to $y_A(t) = 0$. We set the reference roll angle to saturate

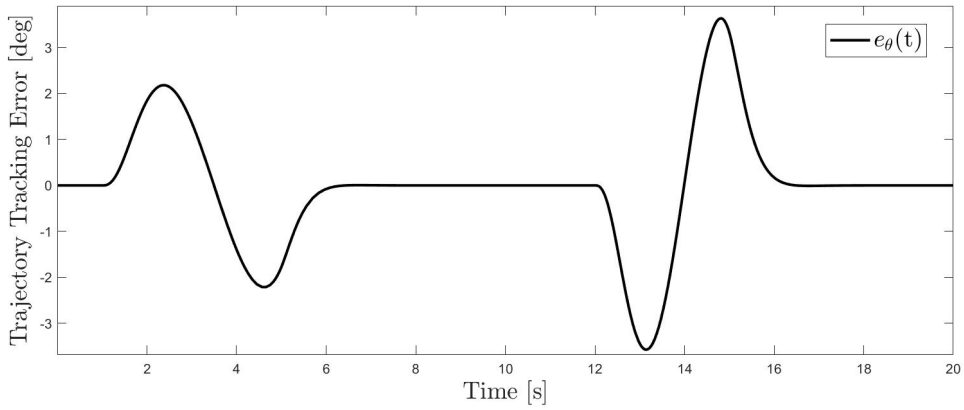

Figure 4.12: Matlab simulation of transition for pitch angle error with 90% confidence in system's parameters and using a PID control law.

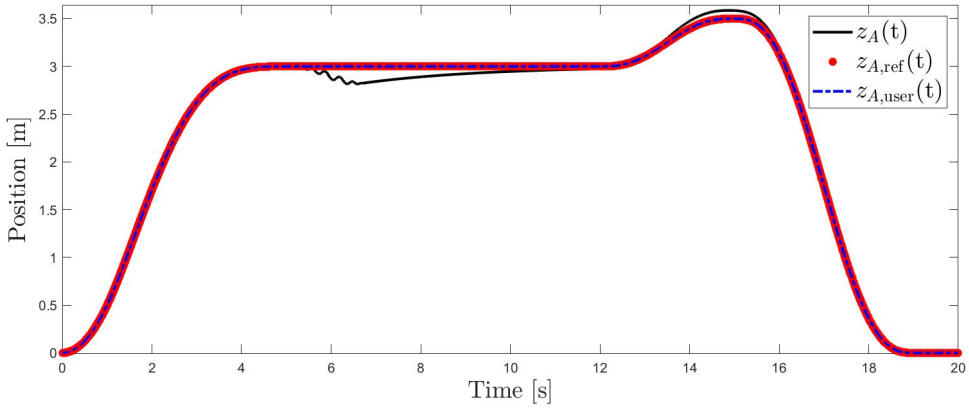

Figure 4.13: Matlab simulation of transition for $z_A(\cdot)$ with 50% confidence in system's parameters and using the proposed MRAC laws.

at 12 degrees during both simulation and flight test. This result validates our outer loop controllers ability to return the vehicle to the inertial X axis in the case that the vehicle takes off with an unwanted initial position.

The second simulation test of the outer loop incorporated an unknown external disturbance acting on the vehicle during the horizontal flight regime. We modeled an unknown external disturbance that would apply a specified amount of force in Newtons to the $y_A(\cdot)$ dynamics of the vehicle during the horizontal flight regime. We found this to be a realistic

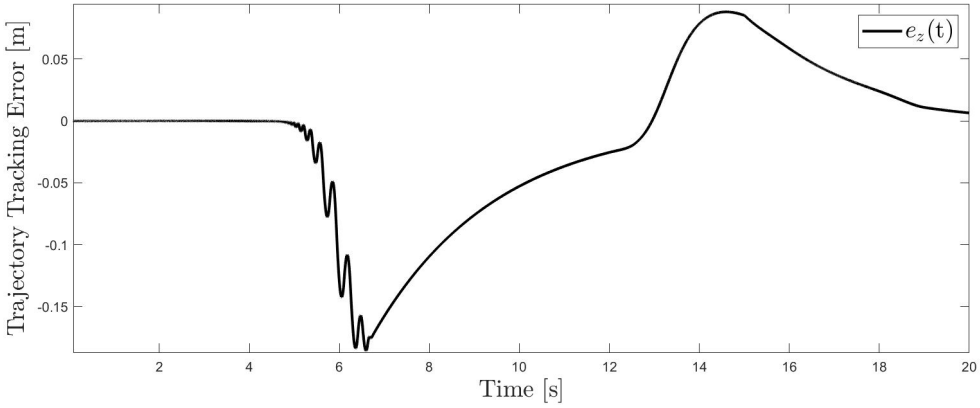

Figure 4.14: Matlab simulation of transition for $e_z(\cdot)$ with 50% confidence in system's parameters and using the proposed MRAC laws.

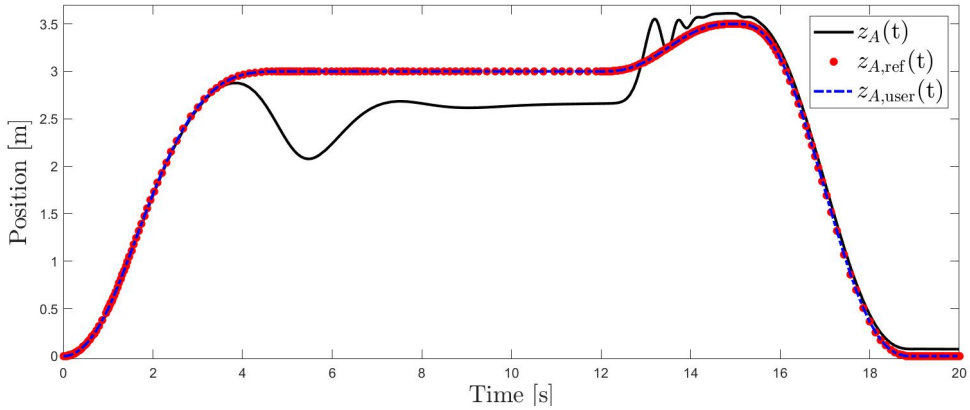

Figure 4.15: Matlab simulation of transition for $z_A(\cdot)$ with 50% confidence in system's parameters and using a PID control law.

situation during real flight tests, since the vehicle occasionally would experience gusts of wind during flight. That was the type of situation that our outer loop controller was designed to handle during flight tests. Figure 4.19 shows the external disturbance applied to the vehicle between 5s and 10s with a maximum magnitude of 2N.

It is shown in Figure 4.20 that the vehicle was able to return to the X axis while only deviating 0.25m despite the unknown force applied to the vehicle. The corresponding roll angle and reference roll angle that drove the vehicle back to the X axis are shown in Figure

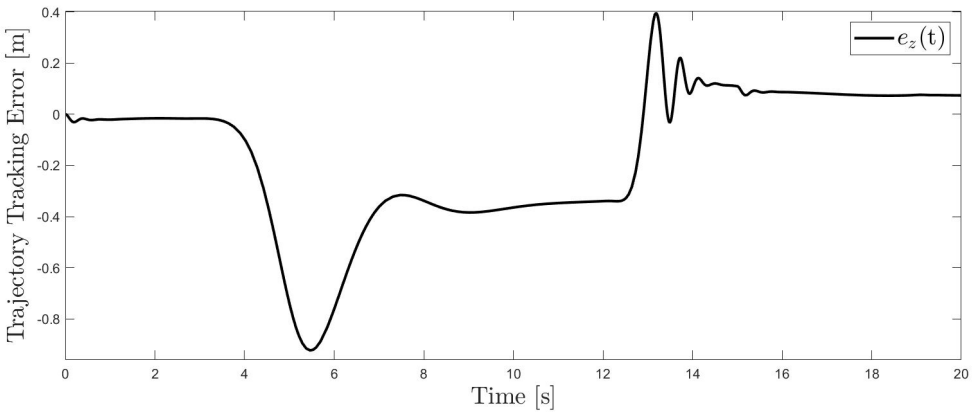

Figure 4.16: Matlab simulation of transition for $e_z(\cdot)$ with 50% confidence in system's parameters and using a PID control law.

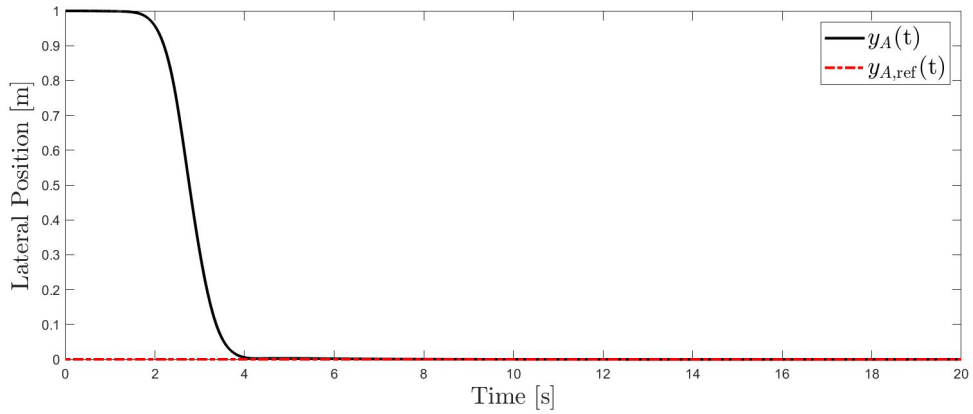

Figure 4.17: Matlab simulation of $y_A(\cdot)$ for test of outer loop controller when an initial condition was present.

4.21. The combination of the two tests conducted in Matlab for the outer loop control of the vehicle validate the outer loop controller in realistic flight conditions.

4.4 Conclusion

In this chapter, we evaluated our control algorithms using a Matlab simulation. For these simulations, we employed a quintic polynomial user-defined trajectory for the vertical

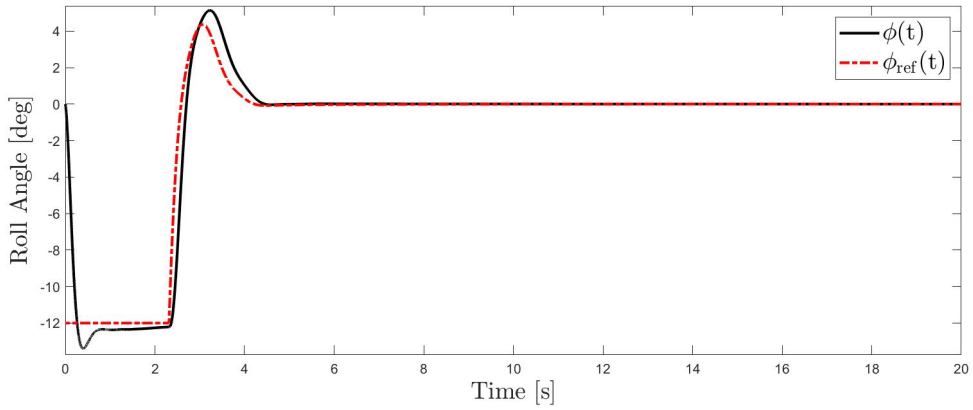

Figure 4.18: Matlab simulation of roll angle for test of outer loop controller when an initial condition was present.

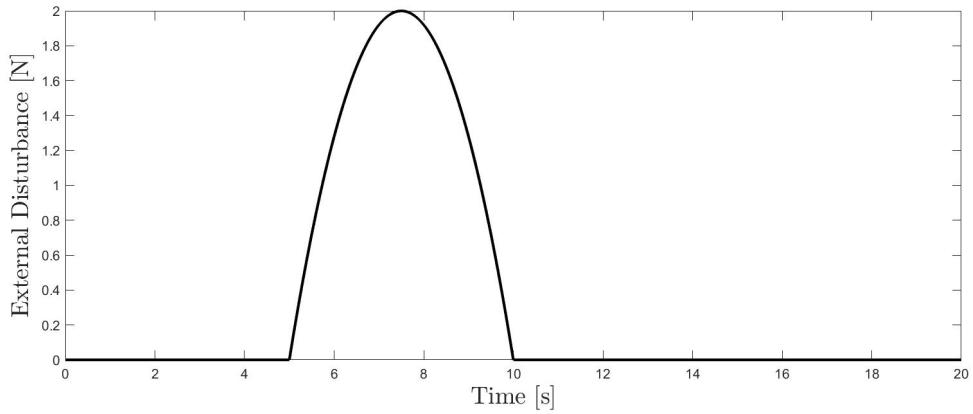

Figure 4.19: External disturbance applied during Matlab simulation testing the outer loop.

position $z_A(\cdot)$ and pitch angle $\theta(\cdot)$. The Matlab simulation showed a successful transition between vertical and horizontal flight regimes for both the vertical position $z_A(\cdot)$ and pitch angle $\theta(\cdot)$ with 10% uncertainty in the vehicle's parameters. These simulations also showed the corresponding distance the vehicle traveled in the X inertial direction. We were also able to determine the thrust from the motors that was necessary to complete these flights, which was used to choose a motor and propeller combination. Next, we evaluated both our MRAC controller and the baseline controller's ability to control the vehicle when there was a 50% uncertainty in the vehicle's parameters and showed much better results with the MRAC

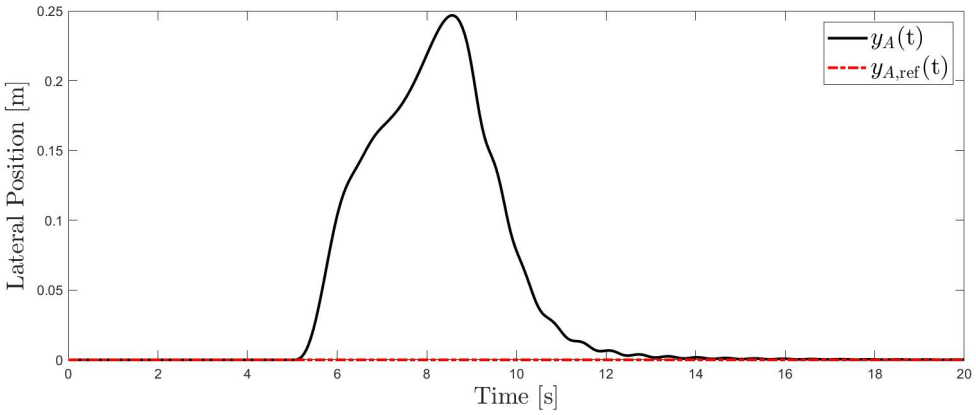

Figure 4.20: Matlab simulation of $y_A(\cdot)$ for test of outer loop controller with an unknown external disturbance.

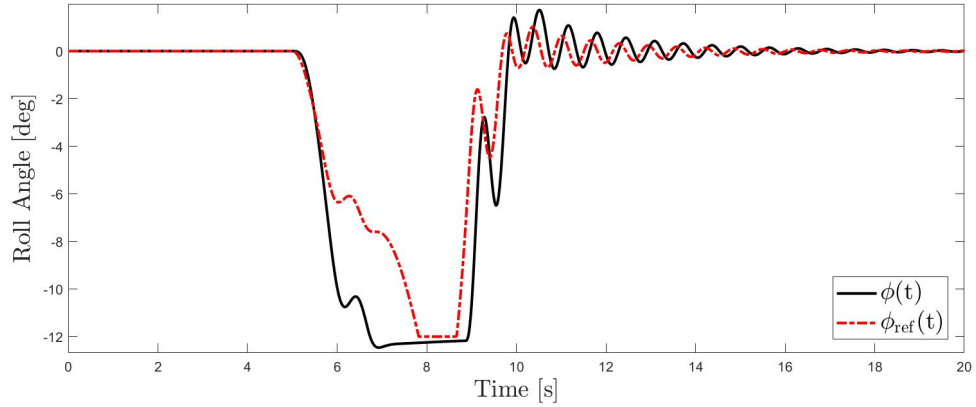

Figure 4.21: Matlab simulation of roll angle for test of outer loop controller with an unknown external disturbance.

controller, which confirmed the need for an adaptive controller for the tailsitter UAV. Finally, we showed sufficient Y position control from the outer loop in the presence of both incorrect starting position and unknown external disturbance, which confirmed our symmetric flight assumption. In the next chapter, we will begin by detailing the design and fabrication of a prototype tailsitter UAV and following that we will show the results of the real flight tests performed with our tailsitter UAV.

Chapter 5

Flight Test

5.1 Introduction

In this chapter, we will explore the design, fabrication, and flight tests that were conducted to test the MRAC controller designed in Chapter 3. Indoor flight tests have been performed in the drone cage the Advanced Control Systems Lab (ACSL). This facility was instrumental in the fabrication, hover testing, and maneuvers at small pitch angles of the tailsitter UAV. Once we were ready to move on to transition testing, outdoor flight tests were performed at the Virginia Tech Drone Park, Figure 5.1. These facilities have larger flight areas, which was necessary to complete a transition between vertical and horizontal flight regimes. Furthermore, the Drone Park allows us to perform outdoor tests without complying with FAA part 107 rules [43].

Figure 5.1: VT Drone Park [10]

We will begin this chapter by detailing the design decisions that were made and the process of designing the tailsitter UAV. Following that, we will show how the prototype tailsitter UAV was fabricated and assembled at the ACSL. Next, we outline the hover tests that were completed, both indoor and outdoor. Finally, we will discuss the transition test flights and the results of these tests.

5.2 Tailsitter UAV Design

As with any aircraft, the aerodynamic and structural design decisions are critical to the performance of the vehicle. The airfoil shape, wingspan, chord, and wing spacing for a biplane are all design decisions that must be carefully researched to achieve our desired performance. The choice of a motor and propeller combination and frame design are also critical steps during the design phase.

The first design decision that was made in regards to our tailsitter UAV was the configuration of the vehicle. As discussed in Chapter 1, we chose the quad-rotor biplane configuration because of its ability to transition between vertical and horizontal flight regimes using a differential in thrusts between the top two and bottom two propellers. We also chose this configuration to allow for future collaborations with the University of Maryland and the Army Research Lab. We decided to not include any control surfaces on the vehicle to reduce the weight and mechanical complexity. Finally, we chose to include two stabilizing wings to act as rudders and prevent unwanted drift.

The most pressing concern with a VTOL tailsitter UAV design was the transition period between vertical and horizontal flight regimes. During this period, the wings will be experiencing stall, which is when the angle of attack is greater than the specified stall angle for the airfoil. When stall occurs, it causes separation in airflow along the top of the airfoil,

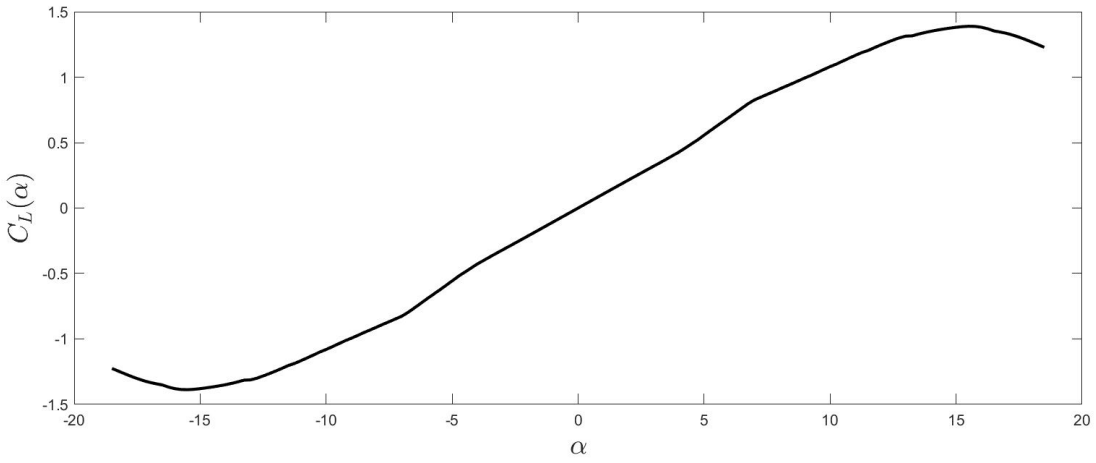

Figure 5.2: Plot of lift coefficient versus angle of attack for the NACA-0012 airfoil [11].

which results in increased drag and decreased lift [44]. Therefore, our goal was to choose an airfoil that has a larger stall angle of attack so that our vehicle will spend less time in the stall region during our transition tests. Another consideration when choosing an airfoil for our tailsitter UAV was the lift coefficient at zero angle of attack. A symmetric airfoil will not generate any lift at zero angle of attack while an asymmetric airfoil will generate lift at zero angle of attack. For our tailsitter UAV, we chose a symmetric airfoil because we do not want to generate any lift when we are accelerating in the inertial Z axis before we begin transition. During this period of acceleration at the beginning of our transition flight tests, our angle of attack will be 0 and if we had an asymmetric airfoil, we would generate unwanted lift. Deciding on a symmetric airfoil also means that during our horizontal flights, we needed to maintain a small pitch angle, while our flight angle is zero, to maintain a small angle of attack so we can generate the necessary lift. With all these considerations taken into account, we chose the NACA-0012 symmetric airfoil [45]. A plot of the lift coefficient versus the angle of attack of the airfoil is shown in Figure 5.2. This plot shows zero lift coefficient at zero angle of attack, a linear increase in lift coefficient as angle of attack increases, and a stall angle of attack of 15 degrees.

Once the airfoil was chosen, the next step was determining the dimensions of the wing to provide the necessary planform area to generate enough lift. In our Matlab simulations, we were able to determine that a planform area of $0.5m^2$ generated the necessary lift to maintain horizontal flight given our airfoil shape, estimated vehicle mass, and maximum motor thrust. Therefore, we choose to have two wings with a 0.75m wingspan and a 0.28m chord. This wingspan and chord results in an aspect ratio of 2.7, which falls into the range of aspect ratios that the researchers in [24] recommend for tailsitter UAV vehicles. These researchers also recommend that the spacing between the wings be at least 1.5 times the chord, so we chose to have our wings spaced 0.5m apart. This means that our stabilizing wings had a 0.5m wingspan and had the same chord as our main wings.

One assumption that was made with regards to the dynamics of the vehicle was the rigid body assumption. This means that we had to make the entire vehicle as rigid as possible to prevent deformation during flight. We accomplished this by designing the main structure of the vehicle with 0.5 inch carbon fiber rods, which are particularly strong and stiff. The four carbon fiber rods were connected in the middle of the vehicle with a 3D printed plastic piece that also acted as the mounting point for the ODROID computer, Pixhawk autopilot sensor board, and the battery. At the other end of the carbon fiber rods were the 3D printed mounts that connect to the motors, the main wings, and the stabilizing wings. The 3D printed mounts for the wings also acted as the landing gear of the vehicle, so we did not have to land on our wings. We designed this vehicle so the 3D printed mounts acted as the weak link in the event of a crash, and therefore they would break first and prevent anything important from breaking. All of the components of this vehicle were designed in Solidworks and the final model is shown in Figure 5.3.

The choice of motors and propellers used for our tailsitter UAV was a crucial decision. This choice can be circular because stronger motors require a larger batter, which in turn

Figure 5.3: Solidworks CAD model of our tailsitter UAV

requires additional lift and thrust force. To solve this issue, we set the vehicle mass in our Matlab simulator to the value we desired and then let the simulation results show us the amount of thrust we need, as discussed in Chapter 4. From our Matlab simulations, we determined that the maximum thrust that we would need would be 33 N. This means that each motor would have to be rated for at least 0.84kg of thrust. We ended up choosing the T-motor FP60PRO IV V2.0 KV1750 motors with 6 inch (15.2cm), 3 blade propellers that have 4.8 inches (12.2cm) of pitch. These motors provided about 1.1kg of thrust at 75% throttle which met our needs. We then found a battery that would provide the necessary capacity, while staying in the given weight limit. The motors we chose required 24 V, so we chose the PULSE Ultra Power Series 6S LiPo battery because it had 2600 mAh of capacity and weighed 410 g. This combination of motors, propellers, and battery met our thrust needs while staying within the given weight limit of 2.6kg.

5.3 Tailsitter UAV Fabrication

Once we had designed the tailsitter UAV, we moved on to the fabrication of the prototype vehicle. The first parts of the tailsitter UAV to be fabricated were the wings. The foam core of the wings was given by the WY-4 symmetric wings by Eureka Aircraft [46]. We chose to use foam for the core of the UAV's wings to guarantee low weight, and then we sheeted the wings with 1/16" balsa wood sheets to add stiffness. To sheet the wings, multiple sheets of balsa wood were glued together and left to dry for 24 hours. We then used the larger balsa sheets, clamped them to the wings, and left them for 24 hours to allow the glue to set; see Figure 5.4. The addition of the balsa wood gave the wings greater stability in all directions, which improved aerodynamic properties and prevented the wings from breaking on crashes. Once the sheeting was complete, the leading edge of the wing was attached and sanded to shape using a sanding block. The leading edge was made of a 1/2" × 1/2" balsa wood bar and was also attached using wood glue and left to dry for 24 hours; see Figure 5.5. This process was completed for all four wings and then the stabilizing wings were cut to size using a band saw.

Figure 5.4: Foam core wing with balsa sheeting clamped for drying

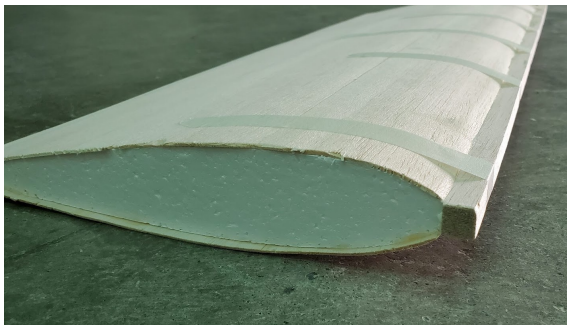

Figure 5.5: Leading edge of wing drying before sanding to correct shape

The next parts to be fabricated were the 3D printed parts. The ACSL is equipped with

a QIDI X-max 3D printer capable of printing with a wide range of materials. We had to evaluate the trade offs of printing with PC (Polycarbonate) with carbon fiber inlaid and PETG (Polyethylene Terephthalate with Glycol). The PC material was stronger and lighter, but it was more expensive and took longer to print. Ultimately, we choose to use the PC material for the motor mounts, where the strength was needed the most, and to use PETG for the wing mounts and center piece of the drone. These parts were designed to be easily replaced upon a crash and extras were printed in preparation. Figure 5.6 shows one of the wing mounts that had been printed on the QIDI X-max. Each of these wing mounts took about 30 hours to print and used our entire print space.

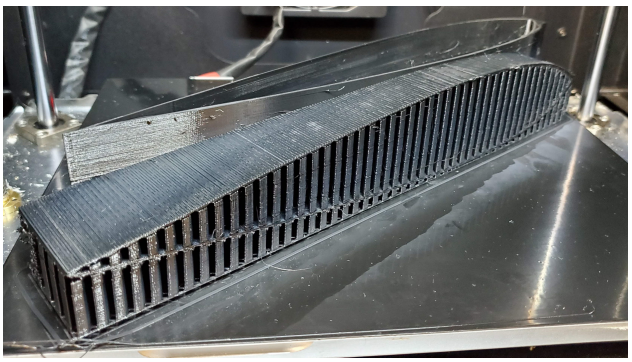

Figure 5.6: Wing mount printed with PETG in the QIDI X-max 3D printer

The final step of the fabrication for the tailsitter UAV was the assembly. To begin, holes were drilled in the carbon fiber rods with a drill press so the rods could be attached to the 3D printed center piece with M3 bolts. The same process had to take place on the other ends of the carbon fiber rods for the motor mounts. For each motor, four M3 bolts were fed from the bottom side of the wing mount, through the motor mount, and into the motor, clamping the three parts together. The stabilizing wings had to already be in place during this step. Next, the motors were soldered to their ESCs (Electronic Speed Controller) and then the ESCs were soldered to the power distribution board, which was on the center piece of the vehicle. The power distribution board connected to the battery and supplied power

for the motors, Pixhawk autopilot, and ODROID. The ODROID and Pixhawk autopilot were mounted above the power distribution board on the center piece of the vehicle and the battery was mounted on the bottom of the center piece. The main wings of the vehicle were then slid into place. Some pictures of the assembly process are shown in Figure 5.7 and a picture with the labeled components of the tailsitter UAV is shown in Figure 5.8. The modularity and simplicity of design of this prototype vehicle allowed for quick repairs and opportunities for future innovation.

Figure 5.7: Assembly process of the ACSL tailsitter UAV

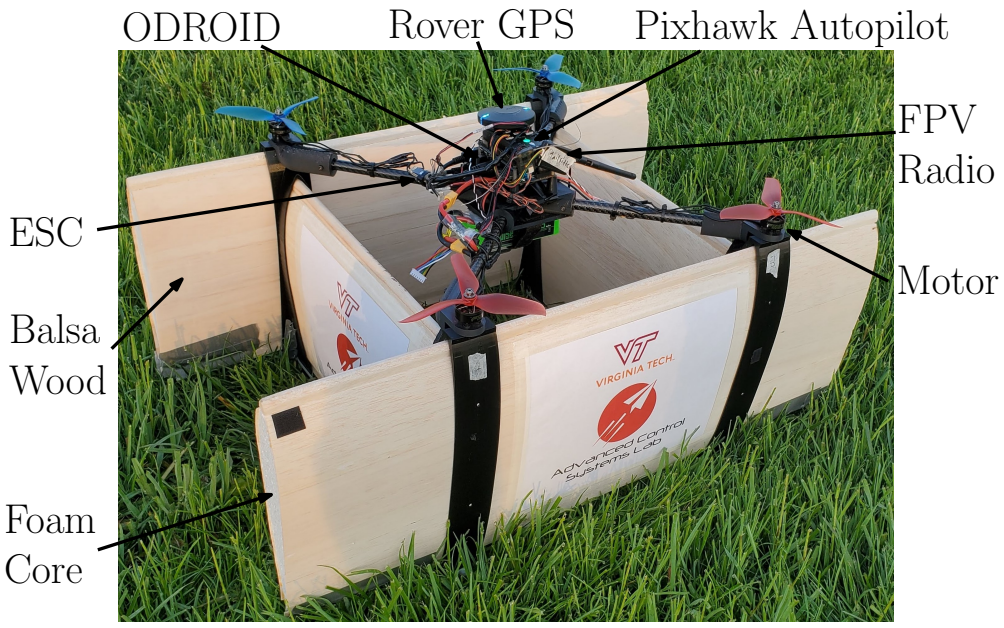

Figure 5.8: Labeled Components on ACSL tailsitter UAV

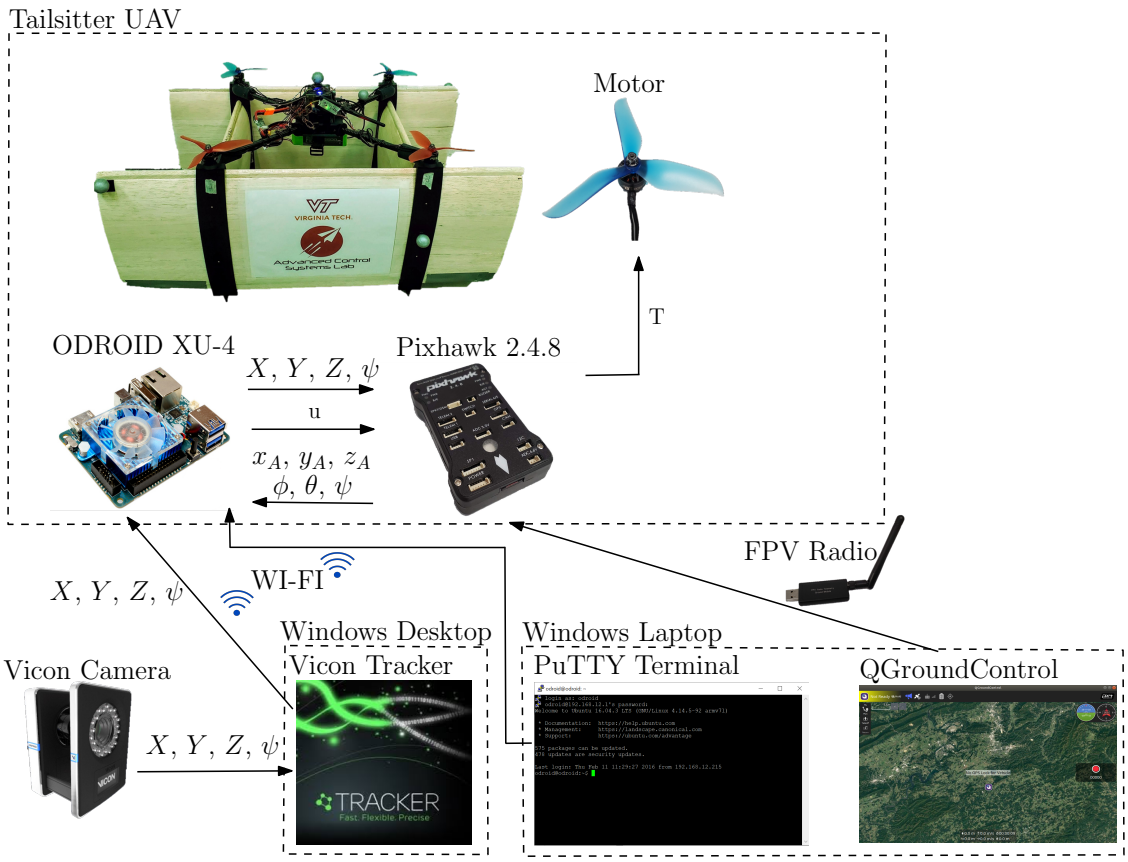

Figure 5.9: Flow of data for indoor flight tests

5.4 Hover Flight Tests

Once the vehicle had been assembled, we began testing. To ensure the vehicle is ready for transition tests, it must first be tuned by conducting indoor and outdoor hover tests. The indoor tests were completed at the ACSL in the indoor drone cage. This drone cage is fitted with six Vicon V6 motion capture cameras [47] that provided accurate feedback on the position and yaw angle of the vehicle. The pitch and roll angles were measured by the onboard Pixhawk autopilot since it has a greater measurement accuracy for these angles than the Vicon cameras. The data from the Vicon cameras was streamed over WI-FI from a desktop computer to the onboard ODROID. The ODROID then sent the data to the Pixhawk autopilot to be merged through a Kalman filter with its own estimates and

returned altitude and attitude values to the ODROID. The ODROID then sent commands to the Pixhawk autopilot, which in turn controls the motor thrusts. A visual depiction of the data flow for indoor and outdoor flight tests is shown in Figure 5.9. To provide a WI-FI signal that would work for indoor and outdoor tests, the ODROID was set up to be an access point and broadcast its own WI-FI signal.

To verify the effectiveness of the proposed UAV and its control system, we first performed autonomous hover tests. The user-defined trajectory strategy for the vertical position $z_A(\cdot)$ expressed in meters for these hover tests was

$$z_{\text{user}}(t) = \begin{cases} 0, & 0 \leq t < 15, \\ \frac{t-15}{3}, & 15 \leq t < 18, \\ 1, & 18 \leq t < 37, \\ -\frac{t-37}{3}, & 37 \leq t < 40. \end{cases} \tag{5.1}$$

The user-defined trajectory strategy for the pitch angle remains vertical at 90 degrees throughout the simulation. The simulation started at 15s, to be consistent with the real flight tests warm-up period, and the vehicle was commanded to go from $z_A(t) = 0$ to $z_A(t) = 1$ over the course of the next 3s, remain at that height until 37s had passed, and then descend back towards the ground over the next 3s where the simulation would end at 40s. One hundred and sixty indoor hover tests were performed to tune gains and verify the effectiveness of the UAV and of the control system's design. The gains chosen for indoor and outdoor hover tests are shown in Appendix A. Figures 5.10 and 5.11 show the trajectory and trajectory tracking error of the vehicle during the hover test. The results of this test show a maximum trajectory tracking error of 0.15m during the transient period and 0.05m during the steady state.

To confirm our pitch controller's ability to hold the vehicle's vertical orientation during

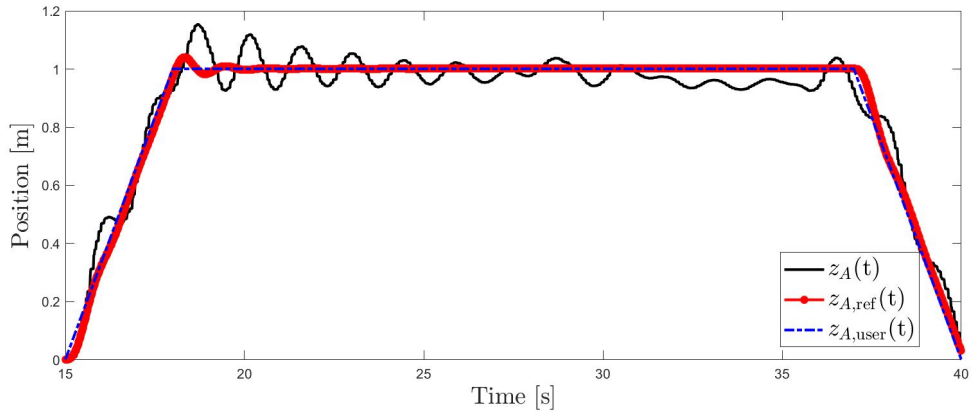

Figure 5.10: UAV's altitude during indoor hover tests.

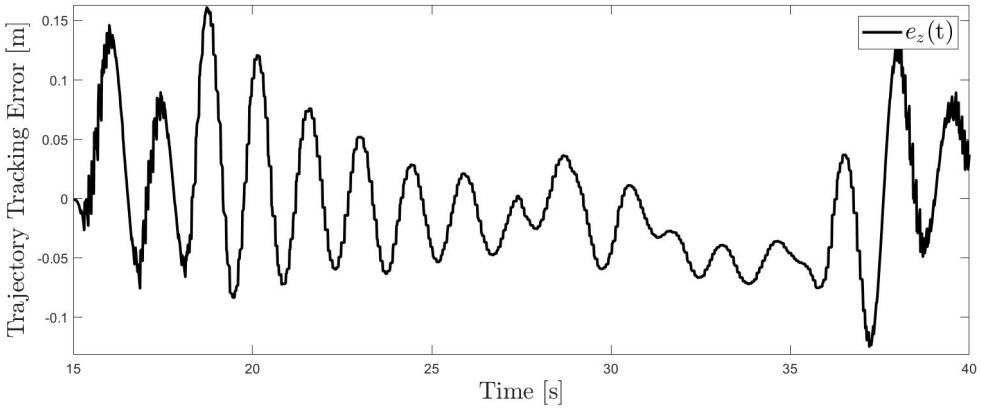

Figure 5.11: UAV's altitude trajectory tracking error during indoor hover tests.

hover, we observed the trajectory and trajectory tracking error for the pitch angle; see Figures 5.12 and 5.13, respectively. These plots show trajectory tracking errors of 2.3 degrees during the transient period, and 0.5 degrees during the steady-state period.

We also evaluated our outer loop controller's ability to maintain its Y inertial position. For these flight tests, we set the desired $y(\cdot)$ position to a constant equal to the UAV's initial position. During the hover test, our outer loop controller always tried to steer the vehicle back to that initial position. As shown in Figure 5.14, the vehicle deviated 0.2m from its desired position during the transient period and stayed within 0.05m of its desired position

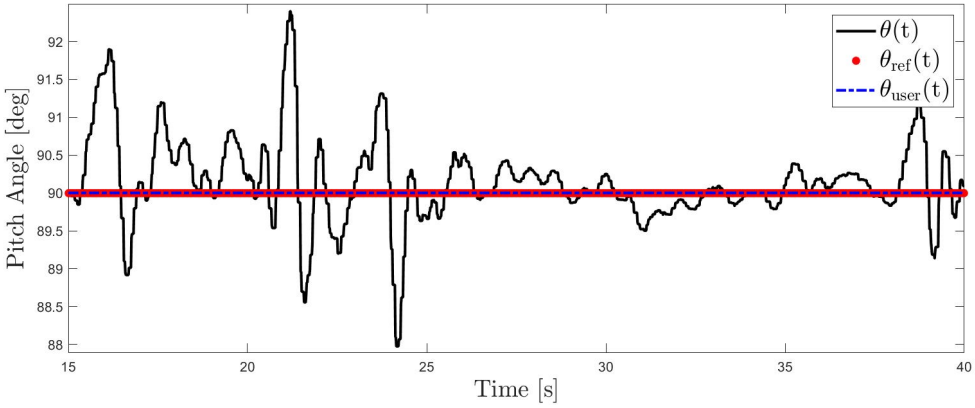

Figure 5.12: UAV's pitch angle during indoor hover tests.

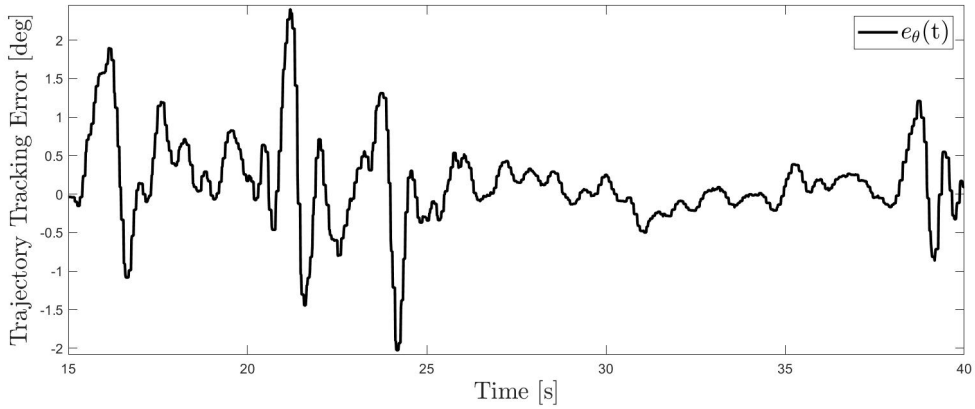

Figure 5.13: UAV's pitch angle trajectory tracking error during indoor hover tests.

during the steady-state period. The reference roll angle and actual roll angle of the vehicle during this test is shown in Figure 5.15. The vehicle's roll angle was able to stay within 1 degree of the desired roll angle during the steady state hover period. These results validate our outer loop controller's ability to maintain the vehicle's $y(\cdot)$ position. The outer loop was tested more extensively during the outdoor hover tests.

We then moved on to outdoor hover tests at the Virginia Tech Drone Park. For these tests, the UAV's position ws estimated by the Here+ differential RTK (Real-Time Kinematic) GPS system, which was chosen for its compatibility with the Pixhawk autopilot and its .025m

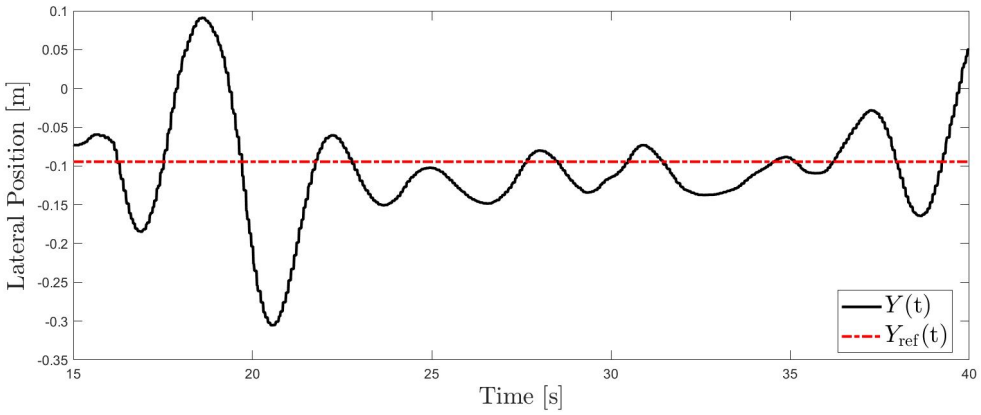

Figure 5.14: Outer loop indoor hover tests results for $y_A(\cdot)$ position.

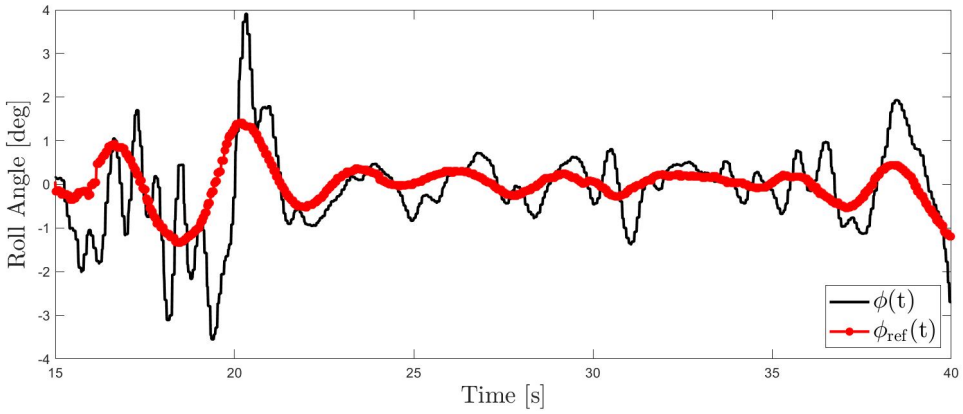

Figure 5.15: Outer loop indoor hover flight tests results for roll angle.

precision [48]. This system comprises a base station GPS which connected to the laptop running QGroundControl, and a rover GPS unit installed on the vehicle and connected to the Pixhawk autopilot. For increased accuracy over conventional GPS systems, this differential RTK GPS relied on the relative position between the airborne and the fixed units for position estimates. The roll and pitch angles were still given by the Pixhawk autopilot, as they were in the indoor flights, while the yaw angle was measured by the magnetometer in the rover GPS unit. This data flow is shown in Figure 5.16, which shows the only difference between the indoor and outdoor tests being the method of altitude and yaw feedback; recall Figure 5.9.

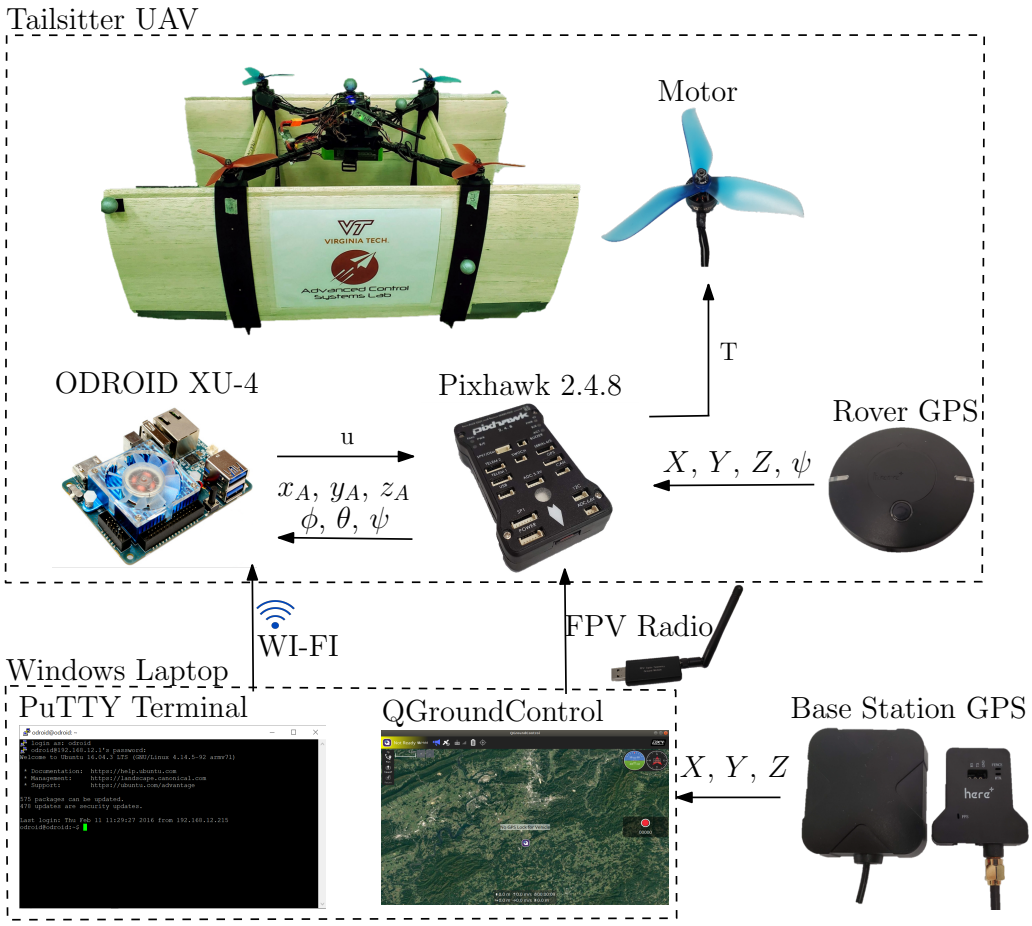

Figure 5.16: Flow of data for outdoor flight tests

After having calibrated the RTK GPS, outdoor hover tests were performed. For these tests, the vehicle was commanded to go to 2.5m of altitude in 5s, hold that altitude for 15s, and then return to its starting position to complete test within 40s. The user-defined trajectory strategy for the vertical position $z_A(\cdot)$ expressed in meters for these outdoor hover tests is

$$z_{\text{user}}(t) = \begin{cases} 0, & 0 \leq t < 15, \\ \frac{t-15}{2}, & 15 \leq t < 20, \\ 2.5, & 20 \leq t < 35, \\ -\frac{t-35}{2}, & 35 \leq t < 40. \end{cases} \tag{5.2}$$

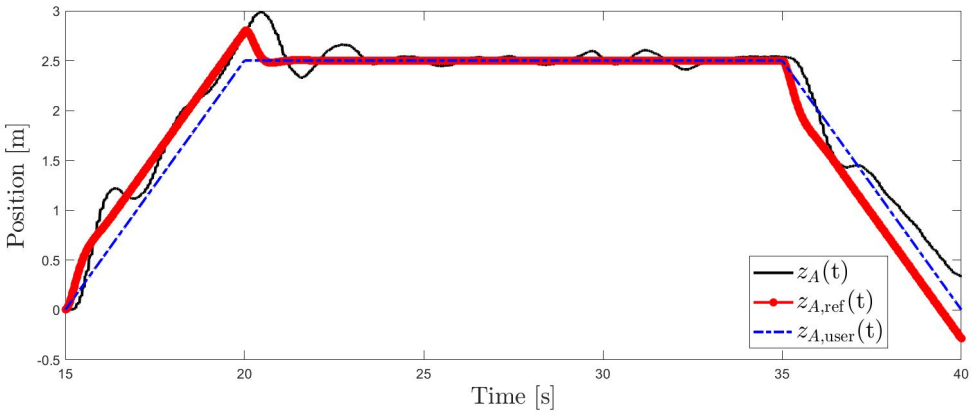

Figure 5.17: UAV's altitude during outdoor hover tests.

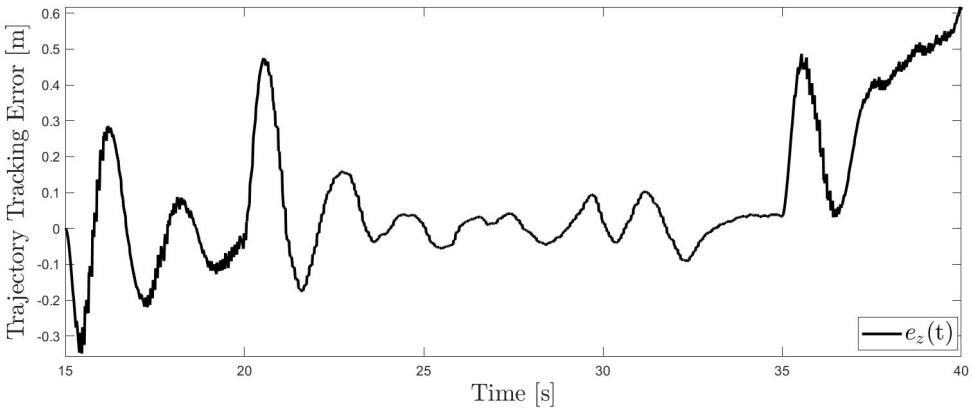

Figure 5.18: UAV's altitude trajectory tracking error during outdoor hover tests.

Figures 5.17 and 5.18 show the UAV's vertical position and the corresponding trajectory tracking error, respectively, using the proposed control architecture. The maximum trajectory tracking error during the transient period was 0.5m, and during the steady state period, the trajectory tracking error stayed within 0.1m.

5.5 Transition Flight Tests

In this section, we will discuss the results of the flight tests from take off to landing employing the proposed control architecture. In the previous section, we verified our vehicle's ability to hold a stable hover. The logical next step was to attempt to transition the vehicle and fly horizontally. For this transition flight, The user-defined trajectory strategy for the vertical position expressed in meters is as follows

$$z_{\text{user}}(t) = \begin{cases} \frac{1}{2}t^2, & 15 \leq t < 17, \\ q_1(t), & 17 \leq t < 20, \\ 4, & 20 \leq t < 21, \\ q_2(t), & 21 \leq t < 24, \\ q_3(t), & 24 \leq t < 27, \\ 0, & 27 \leq t, \end{cases} \tag{5.3}$$

where $q_1(t) = q(\cdot)$ given by (4.2), with $q_0 = 2\text{m}$, $v_0 = 2\text{m/s}$, $a_0 = 1\text{m/s}^2$, $q_f = 4\text{m}$, $v_f = 0\text{m/s}$, $a_f = 0\text{m/s}^2$, $q_2(t) = q(\cdot)$ given by (4.2), with $q_0 = 4\text{m}$, $v_0 = 0\text{m/s}$, $a_0 = 0\text{m/s}^2$, $q_f = 4.5\text{m}$, $v_f = 0\text{m/s}$, $a_f = 0\text{m/s}^2$, $q_3(t) = q(\cdot)$ given by (4.2), with $q_0 = 4.5\text{m}$, $v_0 = 0\text{m/s}$, $a_0 = 0\text{m/s}^2$, $q_f = 0\text{m}$, $v_f = 0\text{m/s}$, $a_f = 0\text{m/s}^2$. The user-defined trajectory strategy for the pitch angle expressed in degrees is as follows

$$\theta_{\text{user}}(t) = \begin{cases} 90, & 15 \leq t < 17, \\ q_4(t), & 17 \leq t < 20, \\ 10, & 20 \leq t < 21, \\ q_5(t), & 21 \leq t < 24, \\ 90, & 24 \leq t, \end{cases} \tag{5.4}$$

Figure 5.19: Stages of a transition flight between vertical and horizontal flight regimes [12].

where $q_4(t) = q(\cdot)$ given by (4.2), with $q_0 = 90$ degrees, $v_0 = 0$ degrees/s, $a_0 = 0$ degrees/s^2, $q_f = 10$ degrees, $v_f = 0$ degrees/s, $a_f = 0$ degrees/s^2, $q_5(t) = q(\cdot)$ given by (4.2), with $q_0 = 10$ degrees, $v_0 = 0$ degrees/s, $a_0 = 0$ degrees/s^2, $q_f = 90$ degrees, $v_f = 0$ degrees/s, $a_f = 0$ degrees/s^2. This trajectory strategy same as for the Matlab simulations presented in Section 4.3, but the horizontal flight period was shortened to account for the size of the Drone Park and the range of the connection between the vehicle and the base station. Using this trajectory strategy, we were able to observe the behavior of the vehicle during the transition between vertical and horizontal flight regimes, which was the goal of this thesis. A series of images showing the take off, transition to the horizontal flight regime, horizontal flight, and transition back to the vertical flight regime are shown in Figure 5.19.

For our transition tests, the vertical position $z_A(\cdot)$ and its respective trajectory tracking error are shown in Figures 5.20 and 5.21. During the transition from the vertical to the horizontal flight regime the maximum trajectory tracking error was 0.4m and during the horizontal flight regime, the maximum trajectory tracking error was 0.2m. The maximum trajectory tracking error during the whole flight was 1m, which occurred during the transition back to the vertical flight regime before landing. During this period, the vehicle's motors are once again facing in the vertical direction and therefore the vehicle will naturally gain altitude. This is a common result among the literature for tailsitter UAV's and some researchers have focused their work on reducing this altitude gain during the transition back

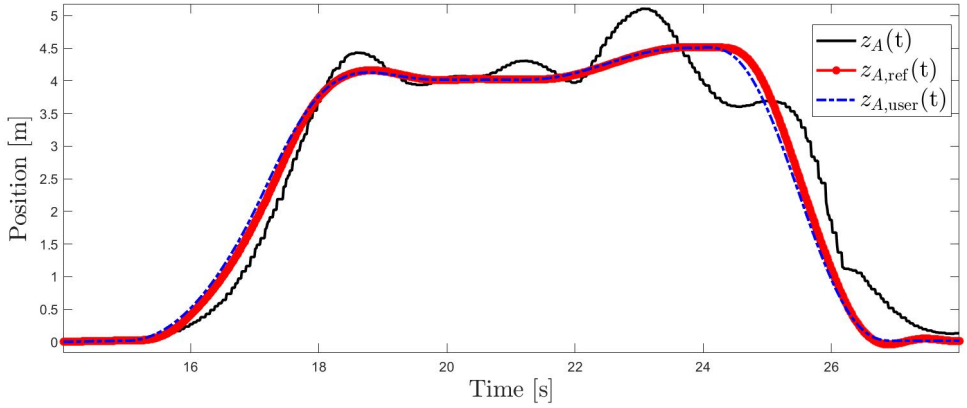

Figure 5.20: UAV's altitude during outdoor transition tests.

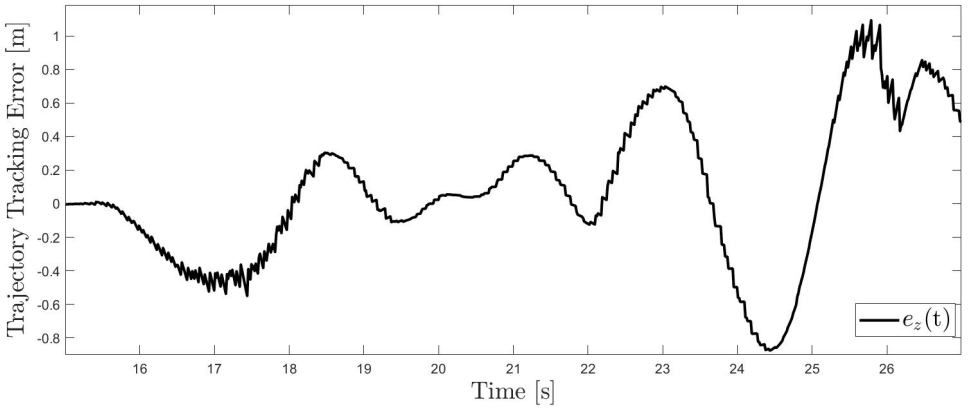

Figure 5.21: UAV's altitude trajectory tracking error during outdoor transition tests.

to the vertical flight regime [6].

The results for the pitch angle and the pitch angle trajectory tracking error during the transition flight tests are shown in Figures 5.22 and 5.23, respectively. These plots show a successful transition both to and from the horizontal flight regime. Figure 5.23 shows a maximum trajectory tracking error of about 25 degrees during this flight. This result showed the ability for the vehicle to transition to the horizontal flight regime, fly under the power of the wings, and transition back to the vertical flight regime. These results could be improved through the use of ailerons or elevators which would help to maintain the desired pitch angle

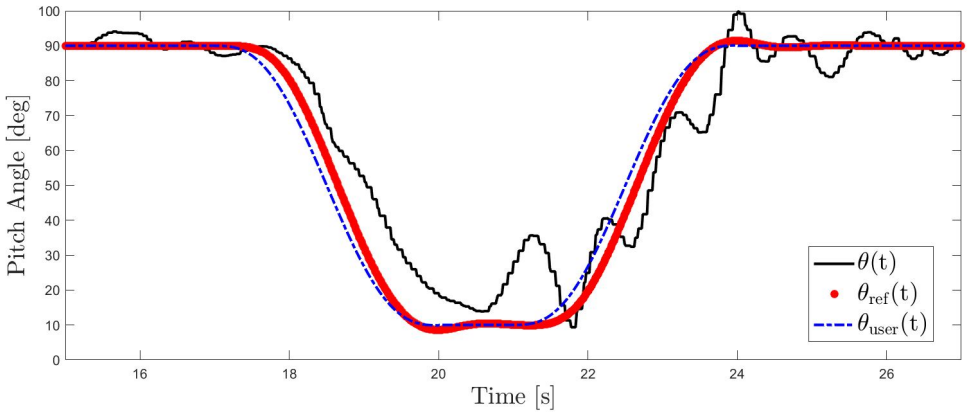

Figure 5.22: UAV's pitch angle during outdoor transition tests.

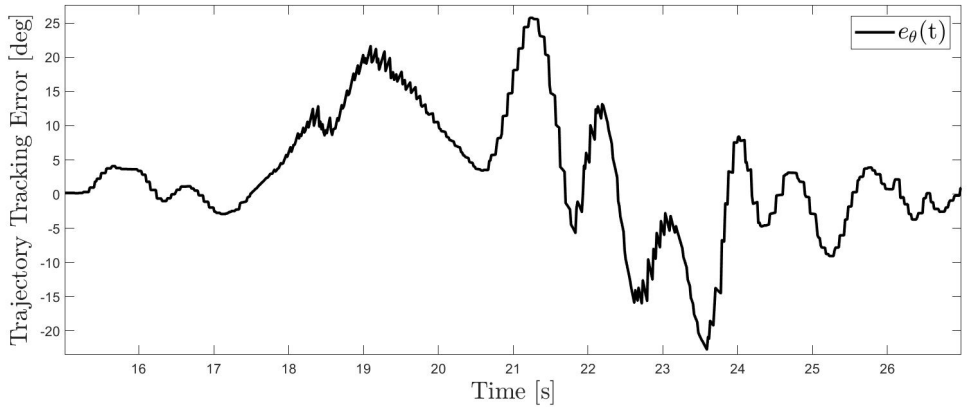

Figure 5.23: UAV's pitch angle trajectory tracking error during outdoor transition tests.

during horizontal flight.

Finally, the results for the Y position of the vehicle during this transition flight test is shown in Figure 5.24. The vehicle deviated about 3.5m from its starting position. This result shows that the vehicle was able to fly straight and provide similar results as the Matlab simulations. Since these transition tests were short, the vehicle was in the transient period the entire flight, where the outer loop showed worse performance during the hover tests completed. Implementation of control surfaces on the wings would help to provide a more steady horizontal flight. A video of a transition flight test is provided on YouTube [12].

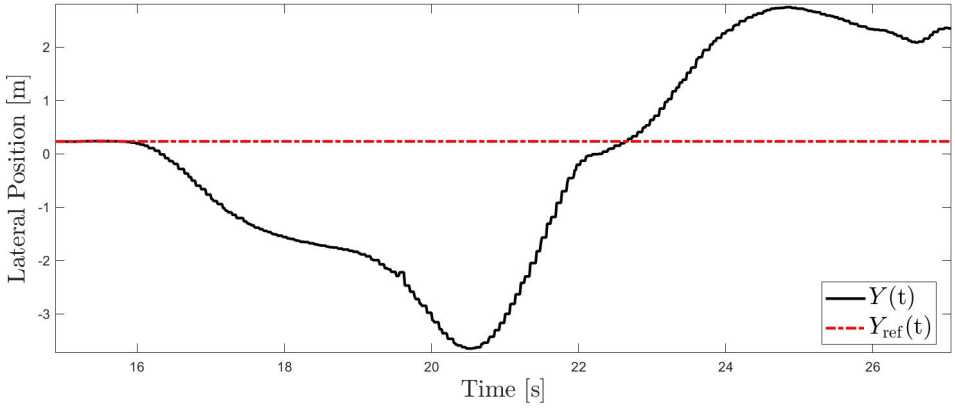

Figure 5.24: UAV's lateral position during outdoor transition tests.

5.6 Conclusions

In this chapter, we detailed the design, fabrication, and testing of the prototype tailsitter UAV. For the configuration, we chose a QRBP tailsitter with stabilizing wings and no control surfaces on the wings. For the wing design, we chose the symmetric NACA-0012 airfoil with a 0.75m wingspan and 0.28m chord, resulting in about 0.5m^2 of planform area between the two wings. The main structure of the vehicle was made of carbon fiber rods that attached in the middle with a 3D printed center piece and at the other ends to the motors and wings with 3D printed mounts. We chose T-motor FP60PRO IV V2.0 motors and 6 inch, 3 blade propellers to generate the necessary thrust, which was calculated from the Matlab simulations in Chapter 4. To fabricate this prototype vehicle, we made the wings from foam core bases and sheeted them with balsa wood. We then used PC and PETG to print the mounts for the vehicle with the 3D printer in the ACSL. Once the individual parts had been fabricated, the vehicle was assembled and the electronics were soldering, resulting in a finished prototype vehicle. Next, we discussed the process in which we completed hover

tests for the vehicle, which included piloted tests, autonomous indoor hover test using the Vicon motion capture cameras, and finally outdoor autonomous tests using the RTK GPS. Finally, we completed the transition flight test to validate our proposed control architecture. We found that the trajectory tracking for the altitude of the vehicle was very good, while the trajectory tracking of the pitch angle was satisfactory for these tests using a prototype vehicle. When the trajectory of the altitude and the pitch are observed together, they show a successful transition to the horizontal flight regime, a short but steady horizontal flight, and a transition back to the vertical flight regime.

Chapter 6

Conclusion and Future Research Directions

6.1 Summary

The goal of this thesis was to design and test a robust MRAC controller on a tailsitter UAV. More specifically, we wanted to regulate the transition between vertical and horizontal flight regimes of the vehicle. To complete this task, we developed a dynamic model of the vehicle, designed a robust controller, validated the controller in numerical simulations, designed and built a prototype tailsitter UAV, and finally performed tests flights.

To begin, we reviewed the current literature on configuration of tailsitter UAVs. Through this review, we found that there were many ways to configure a tailsitter UAV and each configuration has its own benefits. Tailsitter UAVs with a single wing, one motor, and ailerons on the wing have the simplest configuration, but have a lesser pitching moment due to the reliance on ailerons for pitching. To create a larger pitching moment, multiple motors were used away from the axis of rotation and the differential in motor thrusts created a greater pitching moment. This idea led to many different combinations of vehicle configurations that take advantage of differential thrusts pitching moment. These vehicles had either two or four motors, one or two wings, and relied on different combinations of control surfaces. A common configuration of the tailsitter UAV was the QRBP, which stood out for its agility

and its ability to generate lift with smaller wing platform areas. For these reasons, we chose to develop a QRBP for this thesis. Our vehicle had no control surfaces, four motors, two main wings, two stabilizing wings, and was connected together using a carbon fiber frame.

Before we derived the dynamic model of the tailsitter UAV, we reviewed previous vehicle's modeling strategies. One of the main areas of study for tailsitter UAV's was the aerodynamic models. Previous researchers studied everything from the blade element solution for the propellers slipstream velocity to the effects of wing spacing on aerodynamic properties for a QRBP. Some vehicles were put into wind tunnels or flow analysis software so the aerodynamic coefficients would be more accurately modeled. These extensive aerodynamic studies show the importance of taking great care to model the aerodynamics of the vehicle correctly. Some notable assumptions that were made by other researchers when deriving the dynamic model of tailsitter UAVs were the symmetric flight assumption, that the center of the body-fixed reference frame coincided with the center of the gravity of the vehicle, and that the longitudinal and lateral dynamic equations can be decoupled. Some of these assumptions would also be important for our dynamical model.

Following that, we discussed the notation that would be used in this thesis and derived the dynamic equations that would be used in this thesis starting from the general dynamic and kinematic equations of motion of an aircraft. To complete this simplification, we assumed that the lateral component of the aerodynamic force and the moment of the aerodynamic forces were zero, resulting in a side slip angle of zero at all times. This assumption allowed for the decoupling of the longitudinal and lateral equations of motion, which reduced the equations of motion to the form used in this thesis. In this section, we also discussed how the aerodynamic forces and moments would be modeled. Finally, we reviewed the notion of controllability and how it applies to our vehicle. Through this analysis, we found that our vehicle was underactuated, which was why we chose to only directly control the vehicle's

pitch angle $\theta(\cdot)$ and altitude $z_A(\cdot)$.

Once we had derived the dynamic equations, we moved on to the control of the tailsitter UAV. Through a review of the literature, we found that many early researchers controlled their vehicles using a pilot instead of an autonomous control technique. More recently, most researchers chose to control the vehicle with separate controllers for the different flight regimes, which caused inconsistencies in the control during transition due to modeling changes. The majority of previous tailsitter UAVs researched were controlled using a PID controller, or a variant of a PID controller. For our controller, we chose to use a robust adaptive control law for the vertical position $z_A(\cdot)$ and pitch angle $\theta(\cdot)$ to account for the parametric uncertainties. We began by feedback linearizing the dynamic equations derived in Chapter 2. We then applied a control algorithm called Model Reference Adaptive Control (MRAC) to the $z_A(\cdot)$ translation and $\theta(\cdot)$ rotation. Following that, we developed a control algorithm for the outer loop of the vehicle that would retain the vehicle's position on the X inertial axis. The outer loop was designed to create a reference roll angle $\phi_{\text{ref}}(\cdot)$ based on the error in the $y_A(\cdot)$ position and a PID controller would steer the vehicle's roll angle towards the reference roll angle, which will bring the vehicle back to the inertial X axis. Finally, we examined how the force of each individual motor is determined based on the desired control inputs $u(\cdot)$. This completed our control algorithm, allowing us to move on to the numerical simulations.

For the first round of testing the proposed control architecture, we developed a numerical simulation in the Matlab environment. For our user-defined trajectory, we utilized the quintic polynomials reference trajectory, which allowed the user to dictate the the initial and final position, velocity, and accelerations. From the Matlab simulations, we were able to validate the controllers designed in Chapter 3 for flight tests between vertical and horizontal flight regimes with 90% confidence in the system's parameters. Using the MRAC control laws, both

the vertical position $z_A(\cdot)$ and pitch angle $\theta(\cdot)$ were able to follow their reference trajectories and therefore the user-defined trajectories with identically zero trajectory tracking error. Using the same user-defined trajectory, but instead using the the PID controller showed worse trajectory tracking. From these simulations, we also determined the necessary thrust that our motors would need to generate for our vehicle to follow the trajectory, which was useful in the design process of the prototype vehicle. The adaptive controller was then compared to the baseline controller when there was a 50% uncertainty in the vehicle's parameters, which resulted in the adaptive controller showing better results. The results of this test further confirmed the need for the adaptive controller. Next, we tested the outer loop of the vehicle in the presence of an unknown external disturbance and observed the resulting $y_A(\cdot)$ position. We observed that the roll angle of the vehicle was able to track the reference roll angle, which resulted in satisfactory tracking of the $y_A(\cdot)$ position. Using the Matlab simulations, we were able to validate our dynamic model and control algorithms, showing we were ready to move on to real flight tests.

Once we were confident in our numerical simulation results, we moved on to the real flight tests. We had to first design and fabricate a prototype vehicle before moving on to hover tests, and finally, transition test. The prototype tailsitter UAV was designed as a quadrotor biplane with stabilizing wings and no control surfaces. We chose symmetric NACA-0012 airfoils and designed the wings to have a total planform area of 0.5m^2. The motors and propellers were chosen based on the Matlab simulations, which showed the thrust necessary for our vehicle to fly. We then fabricated the wings out of foam core and balsa wood sheets and printed the mounts using the 3D printer in the ACSL. The vehicle was then assembled and the electronics were soldered. Following that, we completed autonomous indoor hover tests in the drone cage at the ACSL before moving on to outdoor hover tests at the Drone Park on Virginia Tech's campus. These tests showed close tracking of the

altitude of the vehicle for hover tests. The outer loop was also validated during indoor hover tests by observing the vehicle's ability to retain its Y inertial position. Finally, transition tests were completed at the outdoor Drone Park. Through these tests we were able to show satisfactory trajectory tracking for both the vertical position and pitch angle. These tests further validated our proposed control architecture for the vertical position and pitch angle, which was the goal of the thesis.

6.2 Future Work

The primary area of improvement of this thesis consists in the design of a control architecture that exploits the UAV's nonlinear dynamics and regulates both the longitudinal and lateral directional dynamics without relying on the separation principle. This would re-introduce the aerodynamic side slip angle and side force and therefore re-couple the longitudinal and lateral dynamics of the tailsitter UAV. The issue of gimbal lock would then become present due to the reliance on Euler angles. This issue could be solved by instead using Euler parameters to represent the orientation of the vehicle. The outer loop of the vehicle would also have to be redesigned to account for this more complex model. While this would lead to a more complete dynamic model of the vehicle, we decided this modeling strategy was outside of the scope of this thesis. Implementation of this model in future work could lead to improved vehicle control and flight tests. Another method of improving the dynamic model of the vehicle would be more accurately modeling the aerodynamics of the vehicle. This could be completed by using both a flow analysis software such as Ansys and performing wind tunnel tests with the actual vehicle. Both these options could provide us with a more accurate model of the aerodynamic coefficients. Additionally, developing a prop-wash model for the effects of the wind created by the propellers of the vehicle would

also help to improve the aerodynamic model of the vehicle.

An additional area of possible future improvement is the hardware of the vehicle. As with any prototype vehicle, it is important to have a constantly evolving design. An interesting adaptation that could be explored for the tailsitter UAV is the integration of tilting rotors, as researched in [49]. Tilting rotors would enable the tailsitter UAV to better control the thrust and moments created by the motors during the transition between flight regimes [50]. This idea is similar to vector thrusting motors on the V-22 Osprey [14]. Another possible modification to the vehicle would be the addition of control surfaces on the wings. This addition has been widely explored in the literature on tailsitter UAVs and would provide better control over the vehicle in horizontal flight and would also help provide a larger moment during the transition between flight regimes. We could also find other ways to improve the aerodynamics of the vehicle by reducing the drag caused by flat surfaces or painting the wings to reduce their skin friction. Also, once the final design has been determined for our tailsitter UAV, some of the mounts and connecting pieces could be fabricated externally with stronger materials than can be 3D printed. One other improvement for the hardware of the vehicle could be the sensors. If we have more accurate feedback on the altitude and attitude of the vehicle, this would result in better flight tests.

As for the control of the vehicle, there are many possibilities that could be explored. One control algorithm that could be used to improve the robustness would be the σ-modification of MRAC [36, pp. 323-327] [51]. The σ-modification of MRAC allows for greater robustness against external disturbances without knowledge of the upper bound on those disturbances. Other adaptive controllers that could be implemented to the tailsitter UAV in future work include the e-modification of MRAC [52] and the Projection-Based MRAC Design [36, p. 337]. Finally, a $\mathcal{L}1$ adaptive controller could be implemented for the control of the the tailsitter UAV, which would guarantee robustness in the presence of fast adaptation [53].

Any of these control architectures would increase the robustness of our controllers and their ability to handle external disturbances.

One possible application of this vehicle is rapid payload delivery. In order to develop a payload handling system on our tailsitter UAV, both hardware and software modifications would have to be made. Some type of payload container would have to be incorporated to the design of the vehicle and studies would have to be conducted to determine the maximum payload the vehicle could handle and to optimize the placement of the payload. The addition of a payload to the vehicle would create additional unknown mass and inertia properties of the vehicle, which would have to be accounted for in the control technique. While this addition would be useful for commercial applications of the tailsitter UAV, it was outside of the scope of this thesis.

6.3 Conclusions

The goal of this thesis was to regulate the transition between vertical and horizontal flight regimes for a tailsitter UAV. Through a review of the current literature on tailsitter UAVs, we found that the application of a robust adaptive controller to a QRBP tailsitter UAV had not been explored. The control method that we employed in this thesis was Model Reference Adaptive Control, which utilized a reference system and gains that adapt based on the response of the vehicle. Through Matlab simulations we were able to validate our control formulations for our vertical position and pitch angle controllers. We also implemented an outer loop controller to ensure the vehicle is able to hold its position in the Y inertial direction and validated this control formulation in the Matlab simulations. We then designed and fabricated a prototype tailsitter UAV at the ACSL for flight testing. During indoor and outdoor hover flight tests, we were able to validate all of our control formula-

tions and begin tuning both the adaptive and baseline controller gains. We found that the major challenges during flight tests were the the aerodynamic properties of the vehicle, data feedback methods, and the external disturbance in the form of wind. Despite these challenges, we were able to demonstrate successful transition between vertical and horizontal flight regimes, validating our adaptive controllers for both the vertical position and pitch angle controllers. The performance of this vehicle met the set expectations, but as always, there are opportunities for future innovation for the tailsitter UAV. This thesis showed the successful first implementation of MRAC to the transition flight regime of a tailsitter UAV, which was validated in both simulation and flight tests.

Bibliography

[1] Boeing, "Boeing: V-22 osprey," https://www.boeing.com/defense/v-22-osprey/, 2021, accessed: 2021-04-20.

[2] X, the moonshot factory, "Wing: Transforming the way goods are transported," https://x.company/projects/wing/, 2018, accessed: 2021-04-20.

[3] R. Kirby, "Stability and control flight tests of a vertically rising airplane model similar to the lockheed XFV-1 airplane," *National Advisory Committee for Aeronautics*, 1954.

[4] M. Factory, "Lockheed XFV (salmon)," https://www.militaryfactory.com/aircraft/detail.asp?aircraft_id=1580, 2019, accessed: 2021-01-19.

[5] N. Knoebel and T. McLain, "Adaptive quaternion control of a miniature tailsitter UAV," in *American Control Conference*. IEEE, 2008, pp. 2340–2345.

[6] A. Oosedo, S. Abiko, A. Konno, and M. Uchiyama, "Optimal transition from hovering to level-flight of a quadrotor tail-sitter UAV," *Autonomous Robots*, vol. 41, no. 5, pp. 1143–1159, 2017.

[7] O. Garcia, A. Sanchez, J. Escareno, and R. Lozano, "Tail-sitter UAV having one tilting rotor: Modeling, control and real-time experiments," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 809–814, 2008.

[8] J. P. Reddinger, K. McIntosh, D. Zhao, and S. Mishra, "Modeling and trajectory control of a transitioning quadrotor biplane tailsitter," in *Vertical Flight Society 75th Annual Forum*, 2019.

[9] P. Ryseck, D. Yeo, V. Hrishikeshavan, and I. Chopra, "Aerodynamic and mechanical design of a morphing winglet for a quadrotor biplane tail-sitter," in *Proceedings of the Vertical Flight Society 8th Autonomous VTOL Symposium*, 2019, pp. 29–31.

[10] E. Nelsen, "If we build it, they will come," https://eng.vt.edu/magazine/stories/fall-2018/if-we-build-it-they-will-come.html, 2018, accessed: 2021-03-08.

[11] Airfoil Tools, "NACA 0012 airfoils (n0012-il)," http://airfoiltools.com/airfoil/details?airfoil=n0012-il, 2021, accessed: 2021-03-09.

[12] A. L'Afflitto, "Autonomous quad biplane UAV with adaptive control law - preliminary flight test results," https://www.youtube.com/watch?v=XXTcToseCFw&ab_channel=AndreaL%27Afflitto, 2021, accessed: 2021-04-28.

[13] R. Woods, "Convertiplanes and other VTOL aircraft," *SAE Transactions*, vol. 65, pp. 150–158, 1957.

[14] H. Rosenstein and R. Clark, "Aerodynamic development of the v-22 tilt rotor," in *Aircraft Systems, Design and Technology Meeting*, 1986, p. 2678.

[15] M. Irving, "DARPA's tail-down drone set for vertical launch in 2018," https://newatlas.com/darpa-tern-uav-northrop-grumman/46770/, 2016, accessed: 2021-02-09.

[16] G. Dickson, "This neighborhood north of Fort Worth receives its first drone delivery in a test," https://www.star-telegram.com/news/business/aviation/article248932299.html, 2020, accessed: 2021-02-04.

[17] P. Phillips, V. Hrishikeshavan, O. Rand, and I. Chopra, "Design and development of a scaled quadrotor biplane with variable pitch proprotors for rapid payload delivery," in *Proceedings of the American Helicopter Society 72nd Annual Forum, West Palm Beach, FL, USA*, 2016, pp. 17–19.

[18] M. Argyle, R. Beard, and S. Morris, "The vertical bat tail-sitter: Dynamic model and control architecture," in *2013 American Control Conference*, 2013, pp. 806–811.

[19] N. Silva, J. Fontes, R. Inoue, and K. Branco, "Development of a fixed-wing vertical takeoff and landing aircraft as an autonomous vehicle," in *2017 Latin American Robotics Symposium (LARS) and 2017 Brazilian Symposium on Robotics (SBR)*, 2017, pp. 1–6.

[20] L. Young, E. Aiken, J. Johnson, R. Demblewski, J. Andrews, and J. Klem, *New concepts and perspectives on micro-rotorcraft and small autonomous rotary-wing vehicles*. American Institute of Aeronautics and Astronautics Inc., 2002.

[21] R. Stone, "The t-wing tail-sitter unmanned air vehicle: From design concept to research flight vehicle," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 218, no. 6, pp. 417–433, 2004.

[22] J. Escareno, R. Stone, A. Sanchez, and R. Lozano, "Modeling and control strategy for the transition of a convertible tail-sitter UAV," in *2007 European Control Conference (ECC)*. IEEE, 2007, pp. 3385–3390.

[23] R. Chiappinelli and M. Nahon, "Modeling and control of a tailsitter UAV," in *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2018, pp. 400–409.

[24] C. Bogdanowicz, "Experimental investigation of a quad-rotor biplane micro air vehicle," Ph.D. dissertation, University of Maryland, 2015.

[25] J. McCullough and P. Oldroyd, "Passenger pod assembly transportation system," Jul. 3 2018, US Patent 10,011,351.

[26] B. Textron, "Bell APT (Autonomous Pod Transport)," https://www.bellflight.com/products/bell-apt, 2020, accessed: 2021-01-18.

[27] B. Textron, "Bell's Autonomous Pod Transport carries 110 lbs. of payload," https://news.bellflight.com/en-US/194170-bell-s-autonomous-pod-transport-carries-110-lbs-of-payload, 2020, accessed: 2021-01-18.

[28] S. Swarnkar, H. Parwana, M. Kothari, and A. Abhishek, "Biplane-quadrotor tail-sitter UAV: flight dynamics and control," *Journal of Guidance, Control, and Dynamics*, vol. 41, no. 5, pp. 1049–1067, 2018.

[29] A. L'Afflitto, R. B. Anderson, and K. Mohammadi, "An introduction to nonlinear robust control for unmanned quadrotor aircraft: How to design control algorithms for quadrotors using sliding mode control and adaptive control techniques," *IEEE Control Systems Magazine*, vol. 38, no. 3, pp. 102–121, 2018.

[30] A. L'Afflitto, *A mathematical perspective on flight dynamics and control*. Springer, 2017.

[31] B. Etkin, *Dynamics of Atmospheric Flight*. New York, NY: John Wiley & Sons, Inc, 1972.

[32] A. Bayen, I. Mitchell, M. Oishi, and C. Tomlin, "Aircraft autolander safety analysis through optimal control-based reach set computation," *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 1, pp. 68–77, 2007.

[33] R. Stengel, *Flight dynamics*. Princeton, NJ: Princeton University Press, 2015.

[34] F. Gavilan, J. Acosta, and R. Vazquez, "Control of the longitudinal flight dynamics of an UAV using adaptive backstepping," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 1892–1897, 2011.

[35] I. Fantoni and R. Lozano, *Non-linear control for underactuated mechanical systems*. Berlin, Germany: Springer Science & Business Media, 2002.

[36] E. Lavretsky and K. Wise, *Robust and adaptive control with aerospace applications*. London: Springer London, 2013.

[37] P. Whitaker, J. Yamron, and A. Kezer, *Design of model-reference adaptive control systems for aircraft*. Massachusetts Institute of Technology, Instrumentation Laboratory, 1958.

[38] B. Shackcloth and R. Chart, "Synthesis of model reference adaptive systems by liapunov's second method," *IFAC Proceedings Volumes*, vol. 2, no. 2, pp. 145–152, 1965.

[39] T. Matsumoto, K. Kita, R. Suzuki, A. Oosedo, K. Go, Y. Hoshino, A. Konno, and M. Uchiyama, "A hovering control strategy for a tail-sitter VTOL UAV that increases stability against large disturbance," in *IEEE international conference on robotics and automation*. IEEE, 2010, pp. 54–59.

[40] Z. Liu and K. Hedrick, "Dynamic surface control techniques applied to horizontal position control of a quadrotor," in *International Conference on System Theory, Control and Computing (ICSTCC)*, 2016, pp. 138–144.

[41] M. Spong, S. Hutchinson, and M. Vidyasagar, *Robot modeling and control*. John Wiley & Sons, Inc, 2006.

[42] K. Ogata, *Modern Control Engineering*. Prentice Hall, 2010.

[43] FAA, "Certificated remote pilots including commercial operators," https://www.faa.gov/uas/commercial_operators/, 2021, accessed: 2021-04-26.

[44] J. Anderson, *Fundamentals of Aerodynamics*. New York, NY: McGraw-Hill, 1991.

[45] NASA, "2dn00: 2d NACA 0012 airfoil validation case," https://turbmodels.larc.nasa.gov/naca0012_val.html, 2021, accessed: 2021-04-26.

[46] Eureka Aircraft Co., "Generic foam wing cores," http://www.eurekaaircraft.com/foamwings/genericwingcores.htm, 2021, accessed: 2021-04-26.

[47] Vicon Motion Systems Ltd UK, "Vantage | cutting edge flagship cameras by Vicon," https://www.vicon.com/hardware/cameras/vantage/, 2021, accessed: 2021-04-20.

[48] Cubepilot, "Here3 specs," https://www.cubepilot.org/#/here/here3/specs, 2021, accessed: 2021-04-26.

[49] R. B. Anderson, J. A. Marshall, and A. L'Afflitto, "Constrained robust model reference adaptive control of a tilt-rotor quadcopter pulling an unmodeled cart," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 1, pp. 39–54, 2021.

[50] G. Matteis, L. Socio, and A. Leonessa, "Optimal inverse analysis of vectored thrust maneuvering," *International Journal of Turbo and Jet Engines*, vol. 15, no. 3, pp. 165–174, 1998.

[51] P. Ioannou and B. Fidan, *Adaptive control tutorial*. SIAM, 2006.

[52] K. Narendra and A. Annaswamy, "A new adaptive law for robust adaptation without persistent excitation," *IEEE Transactions on Automatic control*, vol. 32, no. 2, pp. 134–145, 1987.

[53] N. Hovakimyan and C. Cao, *L1 adaptive control theory: Guaranteed robustness with fast adaptation*. SIAM, 2010.

Appendices

Appendix A

Values and Descriptions for Parameters and Gains

Table A.1: Values of Parameters and Gains for Simulations and Flight Tests

Parameter	Matlab Simulation	Hover Flight	Transition Flight
$\hat{m}$	2.25 & 1.25	2.5	2.5
$\hat{C}_{L,\alpha}$	0.54 & 0.3	0.6	0.6
$\hat{C}_{L,0}$	0	0	0
$\hat{C}_{D,\alpha}$	0.18 & 0.1	0.2	0.2
$\hat{C}_{D,0}$	0	0	0
$\hat{S}$	0.45 & 0.25	0.5	0.5
$\hat{\rho}$	1.17 & 0.65	1.3	1.3
$K_{p,z}$	4	2.5	2.5
$K_{d,z}$	3.6	1	1
K_0	28	35	45
K_1	10	10	12
$\Gamma_{x,z}$	$10 \cdot I_2$	$10 \cdot I_2$	$12 \cdot I_2$
$\Gamma_{r,z}$	20	15	15
$\Gamma_{\theta,z}$	$5 \cdot I_5$	I_5	$0.75 \cdot I_5$
$K_{p,\mathrm{rot}}$	2.25	2	2
$K_{d,\mathrm{rot}}$	2.1	1	1
K_2	15	1.5	2.5
K_3	1	0.6	0.8
Γ_{rot}	$30 \cdot I_2$	$2 \cdot I_2$	$1.5 \cdot I_2$
$K_{p,y}$	8	0.5	0.05
$K_{i,y}$	4.5	0.05	0
$K_{d,y}$	6.9	0.05	0.005
$K_{p,\phi}$	10	0.3	0.2
$K_{i,\phi}$	1	0.05	0
$K_{d,\phi}$	3	0.08	0.08
$K_{p,\psi}$	5	0.7	0.4
$K_{i,\psi}$	0.5	0.05	0
$K_{d,\psi}$	1	0.02	0.02
l	0.5	0.5	0.5
k_T	0.001	0.001	0.001

Table A.2: Description of Parameters and Gains for Simulations and Flight Tests

Parameter	Units	Equation	Description
$\hat{m}$	kg	(3.10)	Estimated mass
$\hat{C}_{L,\alpha}$	n/a	(3.11)	Estimated lift coefficient slope
$\hat{C}_{L,0}$	n/a	(3.11)	Estimated lift coefficient at $\alpha = 0$
$\hat{C}_{D,\alpha}$	n/a	(3.12)	Estimated lift coefficient slope
$\hat{C}_{D,0}$	n/a	(3.12)	Estimated drag coefficient slope
$\hat{S}$	m^2	(3.10)	Estimated planform area of wings
$\hat{\rho}$	kg/m^3	(3.10)	Estimated air density
$K_{p,z}$	n/a	(3.17)	Proportional gain for Z direction reference model
$K_{d,z}$	n/a	(3.17)	Derivative gain for Z direction reference model
K_0	n/a	(3.20)	Proportional gain for Z control
K_1	n/a	(3.20)	Derivative gain for Z control
$\Gamma_{x,z}$	n/a	(3.21)	Adaptive rate for $\hat{K}_{x,z}$
$\Gamma_{r,z}$	n/a	(3.22)	Adaptive rate for $\hat{K}_{r,z}$
$\Gamma_{\theta,z}$	n/a	(3.23)	Adaptive rate for $\hat{\Theta}_z$
$K_{p,\text{rot}}$	n/a	(3.27)	Proportional gain for θ reference model
$K_{d,\text{rot}}$	n/a	(3.27)	Derivative gain for θ reference model
K_2	n/a	(3.29)	Proportional gain for θ control
K_3	n/a	(3.29)	Derivative gain for θ control
Γ_{rot}	n/a	(3.30)	Adaptive rate for $\hat{K}_x$
$K_{p,y}$	n/a	(3.39)	Proportional gain for outer loop baseline control
$K_{i,y}$	n/a	(3.39)	Integral gain for outer loop baseline control
$K_{d,y}$	n/a	(3.39)	Derivative gain for outer loop baseline control
$K_{p,\phi}$	n/a	(3.45)	Proportional gain for roll control
$K_{i,\phi}$	n/a	(3.45)	Integral gain for roll control
$K_{d,\phi}$	n/a	(3.45)	Derivative gain for roll control
$K_{p,\psi}$	n/a	(3.46)	Proportional gain for yaw control
$K_{i,\psi}$	n/a	(3.46)	Integral gain for yaw control
$K_{d,\psi}$	n/a	(3.46)	Derivative gain for yaw control
l	m	(3.47)	Length between T_A and T_B
k_T	n/a	(3.47)	Propeller drag coefficient