6-1-2013

An Efficient Navigation-Control System for Small Unmanned Aircraft

Jonathan Alejandro Girwar-Nath
University of Denver

AN EFFICIENT NAVIGATION-CONTROL SYSTEM FOR SMALL UNMANNED AIRCRAFT

A Thesis

Presented to

the Faculty of Engineering and Computer Science

University of Denver

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

---

by

Jonathan Alejandro Girwar-Nath

June 2013

Advisor: Kimon P. Valavanis, Ph.D. and Matthew J. Rutherford, Ph.D.

Author: Jonathan A. Girwar-Nath
Title: AN EFFICIENT NAVIGATION-CONTROL SYSTEM FOR SMALL UNMANNED AIRCRAFT
Advisor: Kimon P. Valavanis, Ph.D. and Matthew J. Rutherford, Ph.D.
Degree Date: June 2013

Abstract

Unmanned Aerial Vehicles have been research in the past decade for a broad range of tasks and application domains such as search and rescue, reconnaissance, traffic control, pipe line inspections, surveillance, border patrol, and communication bridging.

This work describes the design and implementation of a lightweight Commercial-Off-The-Shelf (COTS) semi-autonomous Fixed-Wing Unmanned Aerial Vehicle (UAV). Presented here is a methodology for System Identification utilizing the Box-Jenkins model estimator on recorded flight data to characterize the system and develop a mathematical model of the aircraft. Additionally, a novel microprocessor, the XMOS, is utilized to navigate and maneuver the aircraft utilizing a PD control system.

In this thesis is a description of the aircraft and the sensor suite utilized, as well as the flight data and supporting videos for the benefit of the UAV research community.

Acknowledgements

I would like to express my sincere appreciation to all of those who gave me the opportunity to complete this thesis. I acknowledge the University of Denver for their continued financial support in educating me with the highest level of excellence.

I would like to thank my advisors, Dr. Kimon Valavanis, Dr. Matthew Rutherford for always continuing to support and encourage me to complete my work. Additionally, I would like to thank Dr. Michail Kontitsis for his support and assistance in completing this thesis project.

Additionally, I would like to thank the faculty and staff Department of Electrical and Computer Engineering at the University of Denver, as well as the Unmanned Systems Group; Allistair Moses, Gonçalo Martins, Chris Brune, Kostas Kanistras, Danielle Sartori, the Denver RC Eagles for letting me use their flying field at Cherry Creek State Park and my flight instructor, Larry Everett Jr., who has passed away during the writing of this thesis, for teaching me how to fly and always being there when I needed it.

Finally I would like to thank my parents Rajendranauth and Carmen for their love and support and Sangeeta for her love, strength and support throughout the years.

I dedicate this thesis to my late grandfather, Juan Antonio Roa. No hay un día que pasa que no pienso en ti. Siempre te amare mi querido viejo.

Table of Contents

List of Figures

List of Tables

Chapter 1 – Introduction

1.1 Motivation

The presence and use of Unmanned Aerial Vehicles (UAVs) has increased considerably in the past decade. This is due to their cost effectiveness and adaptability; significant decreases in operational costs and human risk have been made. The capabilities of UAVs are rapidly altering; this is attributed to advancements in power distribution, design and construction, and sensor cost and effectiveness. With these changes, the design of a UAV should be adaptable to a variety of mission capabilities.

Unlike conventional aircraft flown today, the dynamics and kinematics of UAVs are not readily available, or fully documented. Compared to full-scale aircraft, UAVs fly at relatively low speeds, are highly nonlinear, and tend to display time varying properties. UAVs are exposed to unpredictable external disturbances because of uncertainties in the environment. Controller design and accuracy is critical in developing a successful UAV. Therefore the need exists for a robust and adaptive flight controller in which issues such as non-linearity and flight uncertainty must be addressed for the development of a flight controller.

The Unmanned Systems Laboratory at the University of Denver in Colorado known as DU^{2}SL is in the development of ground, aerial and surface unmanned systems. Predominantly using commercial-of-the-shelf (COTS) platforms due to the relatively initial cost, repair downtime in addition to proven, reliable airframes. Researchers at DU^{2}SL design and develop control systems for various UAVs such as helicopters, quad rotors and fixed-wing for various or specific applications. The work presented in this thesis is for a fixed-wing UAV. Topics and disciplines learned involved physics, aerospace, electronics, mechanics, and computer science to name a few.

1.2 Problem Statement

This research focuses on the ability of a fixed-wing unmanned aircraft to fly autonomously. The aircraft should be able to recover and stabilize itself during flight in both calm and windy flight conditions. It is also imperative that the UAV be reconfigurable to additional sensors based on requirements to complete a task. The flight-controller for the unmanned system should be adaptable to variations in the aircraft's dynamics due to external disturbances as previously mentioned.

1.3 Proposed Solution

The scope of this project is to develop a fully functional autonomous fixed-wing vehicle with an integrated navigation controller. The autopilot is integrated into the UAV so that both the information gathered from pilot commands and the movement related to

those commands feed into the controller for adaptability. This adaptability allows for a plug and play solution for various fixed-wing vehicles. The proposed solution objectives are:

- Obtain an aircraft with the necessary flexibility and flight stability for data collection.
- Analyze the needs for aerial vehicle to accommodate hardware integration with respect to weight and power constraints.
- Strategically integration hardware to preserve the aircraft's flight characteristics, maneuverability and dynamics.
- Develop identification techniques based on intelligent systems to learn about the behavior of the UAV in cruise (trim) flight conditions
- Design an intelligent controller based on the identification to mimic human pilot commands.

1.4 Contributions

The proposed solution presented in this thesis is the design and implementation of a fixed-wing Unmanned Aerial Vehicle (UAV) capable of autonomous navigation. This work benefits all areas of research involving UAV vehicles including control development, and system integration. Major contributions include:

- *Development of a platform from concept to final:* This proposed solution is a ground-up approach that spans all encompassing disciplines of Mechatronic

Systems Engineering. This includes considerations in development phase to include modifications to the aircraft to maintain proper data collection and autonomous flight.

- *Specifying a design methodology that enables a far-reaching extent of development*: The work explained in this thesis is designed to provide development opportunities for various areas of unmanned system by utilizing hardware and software that is designed for easy modification, replacement or removal without the need to restructure the system.
- *Enabling the design and implementation of an autonomous fixed-wing aircraft with the simple study of various disciplines*: This is accomplished by specifying the design details from the concept through completion of the aircraft including hardware selection, integration, software design and testing.
- *Delivering a system that operates without the need of advanced knowledge*: From conception, this work is designed to be reproducible without the need for external information. To follow this standard, Commercial-Off-The-Shelf (COTS) hardware is utilized as much as possible.

1.5 Thesis Outline

This thesis consists of seven chapters each explaining the methodology used in the development of an autonomous fixed-wing vehicle. In Chapter 2 are background information and literature reviews where a comparative study in related fields of research

is explored and contrasted with the goal of this project. The literature review is composed of Currently Developed Autonomous Aircraft, Navigation Techniques, Control Systems, and System Identification. The scope of the literary review is to gain knowledge of the work that other researchers and institutions have completed in the past and build on their strengths.

In Chapter 3 the UAV platform is described in detail. First is an analysis of different aircraft that can serve as the UAV platform. The UAV platform should be easy to replace/repair in the event of aircraft loss, and be capable to hold all the necessary sensors for unmanned operation. Next, an explanation of sensors on the UAV that include an Inertial Measurement Unit, a multi-core microcontroller, a GPS module and a telemetry (flight data) module for recording is given. Additionally, Chapter 3 includes a sub-section dedicated to the microcontroller implemented in this UAV. Also described are the sensor assembly and software integration requirements for the aircraft. Finally, an overview of the System Identification process and procedure is presented.

The initial phase in developing navigation, guidance and control strategies for a UAV is to develop suitable dynamic models. Deriving the nonlinear equations is the focus in the first part of Chapter 4. The equations of motion that are appropriate for control design are found in the latter part of the chapter.

The first part of Chapter 4 derives the expressions for the kinematics and the dynamics of a rigid body where Newton's laws are applicable. The emphasis in this section is defining the relations between position and velocities, (the kinematics) and the relation between forces and moments and the momentum (dynamics). The second part of

Chapter 4 combines the relations to create the complete nonlinear equations of motion. The notation and coordinate frames utilized in this chapter are typical in aeronautics literature.

Since the equations of motion for a UAV are both complex and coupled, designing controllers based on them is challenging and requires a more direct approach. Here, the equations of motion are linearized and decoupled for control system design.

The dynamics for fixed-wing aircraft can be decomposed into two motions: longitudinal and lateral. Longitudinal motion includes airspeed, pitch angle, and altitude, while lateral motion includes roll and heading angles. Force and moment equilibrium, in flight dynamics is called trim. The linear models are derived with respect to an equilibrium (trim) condition.

Chapter 5 merges the information and knowledge Chapters 3 and 4 to discuss the Identified UAV model, where "System identification is the determination on the basis of observation of input and output of a system within a specified class of systems to which the system under test is equivalent." – Zadeh [33]. The aircraft sensors are assembled and integrated into the system along with the software necessary to unify all components.

Chapter 6 displays the results obtained from the system identification in the previous chapter and the results in the form of flight data obtained from autonomous flight. Finally Chapter 7 concludes the thesis.

Chapter 2 – Literature Review

Prior to this thesis, the need to understand, develop and design an autonomous vehicle is necessary since a suitable platform did not exist at the university of Denver. In this section various aircraft that are currently flying autonomously are reviewed.

Stancliff et al [1] at the University of Florida found the ability to abstract human skill into computational models, capable of realistic emulation of dynamic human behavior and utilize them in an aerial vehicle. They chose a fixed-wing aircraft as a platform based on having a greater payload range, more stability, and operation with fewer and simpler control mechanisms. Subsequently, a fixed-wing aircraft requires a less expensive radio system and less skill to operate.

Stancliff et al made decisions on which platform to use, the commercial off the shelf Sig Kadet Senior is the aircraft of choice. The MotoCalc software is used to compute power usage, efficiencies and flight envelopes of the aircraft, while taking in various parameters. On board sensors include an altimeter, electronic compass, and an accelerometer, all connected to two computers, one for low level I/O and coordination of sensor information, and the other to process intelligent control behaviors. A safety system to switch between computer and human control is also integrated into the system.

[1] Has successfully modeled human control strategies and can classify all control as either continuous or discontinuous. Where steering varies continuously with sensor inputs, while accelerations vary discontinuously with sensor inputs, because of the necessary switching between the gas and brake pedals. Hence, a statistical Markovian discontinuous learning architecture [2] successfully abstracts switching control.

A control system that allows an air vehicle to localize and navigate using GPS and an AHRS from [3] utilizing noisy sensory data based on non-linear mathematical model of the aircraft is run in simulation.

The aircraft to model is an RC Kadet MKII high wing, because of inherent stability of the aircraft. The mathematical model is designed to be accurate enough to cover reasonably large flight envelopes and to incorporate varying atmospheric parameters such as wind and density variations with altitude. It is specified using the AeroSim Blockset. The model takes three controls as input: aileron, elevator, and throttle, and parameters that include the geometric design, aerodynamic stability derivatives/coefficients, and engine parameters.

The localization procedure first transforms the simulated sensory data of the airplane's body frame to the ground frame and converts the latitude and longitude data of the simulated GPS to meters so that is consistent with the velocity states and acceleration inputs. The navigation module takes in as inputs the desired location in terms of latitude and longitude, and next calculates the direction that the airplane needs to head into in order to reach that location. The assumption is that longitudinal (pitch) and lateral

dynamics (roll, yaw) of the airplane are decoupled. The flight control system is designed for mid-flight operating conditions. Therefore, three decoupled controllers are designed for altitude, airspeed and heading control.

A fuzzy controller, that determines the required angle of attack to maintain based on current airspeed is fed to a PI controller that when actuated, the relevant control surface is adjusted to achieve the required angle of attach. To prevent excessive overshoot, an anti-windup action was incorporated into the PI controller.

Various control systems exist for autonomous flight. The control system in this thesis is explained in chapters 5 and 6, however this section describes two different types of control for aircraft.

A two-module fuzzy logic controller is presented in [4] for autonomous navigation where the UAV must repeatedly fly to pre-defined waypoints, provide trajectory tracking and be able to follow a previous UAV's trajectory. The Mamdani type fuzzy law is implemented on the UAV with the dynamics readily available. This two-module is composed of an attitude module and a latitude-longitude model.

The attitude controller is based of 3 inputs: attitude error, Δ error, and airspeed. However the aircraft presented only have outputs of elevator and throttle for this module. The latitude-longitude controller has the heading error, and Δ error as inputs and the roll angle as outputs (assumed to be the aileron). Membership functions that pertain to both fuzzy controllers are derived. The controllers and airplane are simulated in matlab. A complete experiment is performed, however no mention of climate maneuverability or rudder is performed.

A Nonlinear Model Predictive Control (NMPC) control action from [5] is used to solve an optimal control problem over a horizon of N steps into the future, where the first control input is used as the current time step, and the rest is saved as an initial condition for the online optimization. The advantage is its ability to handle constraints on the inputs and states.

For the purposes of this paper, the NMPC is used to design a high level tracking controller for a small fixed-wing UAV, where the closed-loop performance is analyzed throughout the hardware in the loop simulations (HILSs) of a UAV tracking multiple line segments as well as a single line. The mathematical dynamics of the UAVs with low-level control are modeled using a kinematic model in a 2-D horizontal plane, where the velocity is constant

The development of the NMPC for the fixed-wing UAV is based upon receding-horizon control by using multiple line segments descried in spatial coordinate systems, the construction of final weight for stability, and a phase portrait.

Real-time performance on a real UAV system was tested with HILSs and the piccolo autopilot system. According to the paper, this proved that the performance of the system was good enough for practical applications.

In the period before Kalman, people in the statistics and time series communities developed the system identification process. A claim can be made that it has its roots in the work of Gauss (1809) and Fisher (1912). Graphical methods based on step or impulse responses were used for system identification. However, those methods were less precise

because there were no tools to estimate the model errors. Moreover, the system identification process is harder because scientists are unable to manipulate the closed loop data. System identification grew after 1960 and the start of the state space era where the model-based prediction and control theory is developed. It was at that time that the Kalman filter replaced Wiener's and the LQG control is introduced.

Nowadays, system identification becomes a tool of importance in the process of designing engineering systems and especially such complicated systems as aircrafts. The objective of this chapter is to present the state of the art in system identification and point out the issues that still exist in the process of designing the mathematical models for aircraft systems in system identification.

At the University of Minnesota [6] frequency domain system identification was used on a low-cost fixed-wing Ultra Stick 25e flight test vehicle. During that procedure, wind tunnel tests and simplifying assumptions are used to calculate the coefficients of that structure. The frequency domain method is used to obtain the parametric model of the aircraft, a method that was applied previously [7] on small-scale rotorcrafts. According to the author, that method provides the advantage to have direct control over the frequency ranges that will be modeled and also lends itself to a linear control design that simplifies the process. The equations of motion of the aircraft were broken down in the longitudinal and lateral motion. All coefficient measures and the preliminary analysis are done in a wind tunnel and the state and input matrices for both motions are calculated.

A model structure for the aircraft dynamics is derived, albeit a lot of assumptions for simplifications are made. Based on the results, this method proves that with its accuracy, it is able to predict the primary dynamic modes of the system.

Researchers at the University of Chungnam in Korea [8] present an approach to identify the aerodynamic parameters of an unmanned aerial vehicle (UAV) that is equipped with an automatic flight control system. The flight test data are carefully acquired by considering consistency and reproducibility. A constrained parameter optimization algorithm is used to find the system model for both longitudinal and lateral/directional flight modes for a large flight speed envelope.

Iliff, 1989, extensively used the (Maximum Likelihood Estimation) MLE method that is based on maximizing the likelihood function; lliff and Maine, 1984 and their study results can be used effectively in the aerodynamic parameter estimation. The supplementary method applied is based on the linearization of the conventional Kalman Filter to nonlinear problems. The states on this method should also be estimated as a function of the selected parameters but the aerodynamic parameters should be estimated as well. The author claims that this method is very sensitive to the initial estimates, and in some cases, a discrepancy in initial guess may lead to divergence of the estimation, which was the case of this research. A mathematical model DATCOM based on stability and control is used and the initial values for later/directional identification are derived from the said model.

At Purdue University of Indianapolis [9] research is done on state space nonlinear identification where the purpose and goal of the research is to solve the motion control problem for advanced aircraft via the development of straightforward identification and control methods to achieve requirements for real-time implementation. In order to control the aircraft, analyze the aircraft dynamics and performance, perform diagnostics and failure detection, and accommodate the failure through control reconfigurations, etc., the real time identification of nonlinear flight dynamics for an aircraft needs to be solved. The focus is to obtain the real-time motion control of aircraft as approached via identification, control redesign and reconfiguration.

To solve the problem, an identification method is applied to locate the unknown bounds of nonlinear dynamic systems. The applied control method is developed from a viewpoint of reducing the computational complexity utilizing the Hamilton-Jacobi framework. The aircraft models were generated using the Lagrange equations of motion, where a multivariable mathematical model maps the aerodynamics into a set of differential equations. From the parameter set, the mathematical model of the aircraft is a subset of the aircraft model itself. The unknown parameters of the aircraft model described by a nonlinear differential equation, are identified by a bounded controller are obtained by minimizing the Hamilton-Jacobi-Bellman equation. It needs to be mentioned that during the process from a real-time perspective, a control-oriented nonlinear identification and constrained optimization is not achieved.

The main contribution of this paper, as claimed by the authors [10], is giving more realistic inputs using the adaptive autopilot that control the aircraft trajectory. In this paper, system identification is used in both lateral and longitudinal dynamics for an ARF-60 UAV. Autoregressive with exogenous method (ARX) for linear estimation and Hammerstein-Wiener method for nonlinear estimation is used.

A specific trajectory is given as an input to the system, where an input trajectory should give the parameters a sufficient excitation that guarantees enough experimental data so that the system identification will be able to take the given data and create its model. By using the ARX method, an estimation match of 43% is achieved that shows that the estimated model using this method is not reliable. On the other hand, the H-W method gives a much better estimation in the transient and sudden changes in the steady state. The difference between the original output and the estimated output for elevator increased to 82%. Although, the throttle input signal did not show major improvement in the nonlinear model, the range of matching between the actual system and the estimated model has increased up to 90% for the elevator signal. It is shown that the lateral model performs well for everything but the lateral throttle parameter. Finally, disturbance effect is studied for both models and it is shown that disturbance is added to the signal and the increase of the disturbance increases the matching difference between the original and the estimated model.

After the control selection and system identification process, navigation techniques can be implemented to further test the robustness of the system. While the techniques involve multiple aircraft, the ideology behind them can be utilized for a single vehicle.

The use of multiple UAV for a single purpose; to confuse anti-aircraft software or human operators is discussed in [11]. The approach is based on having multiple UAVs and N targets. When a location of desire is found M UAVs will simultaneously arrive at a target while not violating aircraft velocity and turn rate constraints. The architecture is made up of 3 components: target manager, path planner and intercept manager to generate waypoints paths for the UAV. To communicate with multiple UAVs at the same time a global communication manager is also implemented. To arrive at a target a Voroni diagram is implemented where a threat cost and length cost is assigned, and additionally Eppstein's k-best algorithm is implemented. Although the target manager is tasked to find the minimal path length, the total cost rises when M UAVs are utilized with munitions to N targets. Utilizing the intercept manager to synchronize the UAVs to arrive simultaneously becomes a challenge when the UAVs are spread further than desired based on environmental conditions such as thermal pockets and wind gusts. Assuming all UAVs are the same construction the trajectory generator finds time-parameterized trajectories while consistent attitude maneuvers are constrained by the heading rate. However different path lengths as mentioned can exist for each UAV in the team.

The desire to reduce the time & requirement of human operators allowing for an increase in reconnaissance capability while reducing risk and cost is approached in [12]. Furthermore the ability to control multiple vehicles with little human interaction/personnel is also investigated. The purpose is to have the UAV track an assigned ground target with little or no human intervention utilizing a generic fixed–wing UAV equipped where real-time telemetry is obtained by the onboard piccolo autopilot. The proposed algorithm generates control for the turn rate and camera angle. A major constraint is the target cannot be faster than the UAV's speed. For this, the target tracking is dependent on a camera gimbal that must keep track of the aircraft regardless of position & orientation. The closed loop gimbal control is coupled with the world model and geographically tracks the target in the earth. To estimate future telemetry, a Kalman filter is implemented, thus the estimate to the camera angles is also estimated. Navigation is controlled by a single radius parameter, where the UAV loops around the target based on it's heading and trajectory, creating target trail points and looping to the points by calculating the turn rate. The events were simulated however no aircraft specifications were given, based on the information provided, a lightweight UAV would need to be implemented, specifically with larger ailerons.

Chapter 3 – Development, Instrumentation and Modeling of a UAV platform

3.1 Introduction

The literature review in Chapter 2 exhibits the main aspects of fixed-wing UAVs, from conventional and intelligent controllers to simulation techniques for model verification to real fight testing.

This chapter discusses process in developing a UAV from the ground-up and the composition of equipment that is necessary for a UAV to collect flight data and fly autonomously after modeling and analysis of obtained flight data. These three components maintain a relationship that utilizes all aspects of Mechatronic Systems Engineering. Simply defined, Mechatronic Systems Engineering is the academic integration of mechanical, electrical, computer engineering and computer science, and not limited to aspects of physics and mathematics and other scientific disciplines. For this chapter, Section 3.2 describes the development phase where the needs and requirements are established and research is conducted. In section 3.3 the instrumentation of the UAV is explained and the process of integrating mechanical systems such as motors with electrical systems such as microcontrollers begins. Section 3.4 explains the rules and regulations associated with flying an Unmanned Aerial Vehicle. Section 3.5 combines the

research and instrumentation to obtain a useable model platform for data collection and autonomous flight with section 3.6 briefly describing the procedure for system identification.

3.2 UAV Platform Requirements

Initially only one fixed-wing platform existed at DU2SL, a Carl Goldberg MiG-27 constructed of foam, Figure 3.1. Originally donated from the Army Research Laboratory in Abdereen, MD where it was used as a target drone. The aircraft was rather large and with a wingspan of 168.5cm and is driven by a two-stroke .91 cc glow-fuel engine. Since this aircraft lacked an abundance of spare parts and vibrations from the large motor can introduce noise into the sensors. Since [13] performed further development outside the scope of this thesis, there existed a need to conduct research to find a suitable a fixed-wing aircraft.

The new aircraft must fit within the requirements for this project. The aircraft must be Commercial-Off-The-Shelf and have an abundance of spare parts that are easily accessible. The aircraft needs to be flyable by operators of all skill levels, meaning, the aircraft should not be advanced to the point where multiple years of flight experience is required. To protect and increase the lifespan of the sensors and hardware, the aircraft should be an electric type and be adaptable to nature where it can sustain the varying weather in Denver, CO. Finally; the aircraft should be ready to fly in general without the need of additional equipment, aside from gluing and recharging batteries.

Since a requirement of the aircraft is that it must be electric, a choice exists on the battery chemistry with regards to the battery. In terms of battery power, two types of batteries can be found in RC aircraft, Lithium-Polymer (LiPo) and Lithium-Ion (Li-ion). Both batteries have advantages and disadvantages, however LiPo batteries are the choice batteries for this project, more specifically a 3-cell, 11.1v with a Milliamp Hour rating of 4200 to maintain longer flights.

With a wide variety of COTS fixed-wing aircraft available, two novice or trainer aircraft are analyzed; the e-Flite Apprentice 15e and the Horizon Hobby Super Cub, shown in Figure 3.2 and 3.3 respectively.

The two aircraft mentioned above are flown tele-operated from the ground using a radio transmitter. However are not able to transmit data to conduct experiments for research, modeling and controller design out of the box. The need exists for embedded systems and sensors to gather data to develop a UAV. Later the same embedded system is used as the flight controller for the autonomy for the vehicle.

Figure 3.1 Flight ready Carl Goldberg MiG 27

Two initial advantages of these aircraft are the use of non-72 MHz traditional rc aircraft control, allowing for less interference from cross talk or conflicting frequencies. The airplanes are categorized as ARF (Almost Ready to Fly), meaning all the necessary components to fly are included in the box, requiring only minor assembly prior to initially flying (i.e. gluing, fastening parts, etc.). Both airplanes are comparable in price and specifications as indicated below in Table 3.1 for the Super Cub and the Apprentice [14] [15].

Hobby Zone Super Cub Specifications	**e-Flite Apprentice 15e Specifications**
480Kv brushed motor	15 Size 950Kv brushless motor w/30A ESC
27MHz 3-channel transmission	2.4 GHz 5-channel transmission
9x6 prop size	11x8 prop size
Length 825cm (nose to tail)	Length 106cm (nose to tail)
Flying weight 715g	Flying Weight 1135g
Power 11.1v	Power 11.1v

Table 3.1 Aircraft specifications: Super Cub vs. Apprentice

Figure 3.2 Hobby Zone Super Cub
Source: Hobby Zone

Figure 3.3 e-Flite Apprentice 15e
Source: e-Flite RC

Since the Super Cub is tele-operated via a 27MHz transmitter and receiver with only 3 controllable surfaces: elevator, rudder and throttle, and the e-Flite Apprentice 15e model has 4 control surfaces and is operated using a 2.4GHz transmitter, the Apprentice 15e is the choice vehicle for this thesis. The aircraft's wingspan is 1475mm and has a flight weight of 1.13 kg with no sensors installed.

3.3 UAV Instrumentation

Since the choice aircraft is the Apprentice 15e, the aircraft needs proper instrumentation in order to fly autonomously. The instrumentation is not however limited to electrical equipment, t to mechanical components as well. This section describes all the mechanical and electrical instrumentation necessary for the aircraft to fly autonomously

There are four parts to all standard configuration aircraft, Motor, Fuselage, Wing, and Tail. For the Apprentice aircraft, the motor supplied is an e-Flite Brushless Out-runner, 840Kv motor; it is comparable to a .15 size glow engine motor. In order to maintain within the requirements, additional motors are purchased, however due to the fact the aircraft will be carrying extra weight (less than 1kg), a more powerful motor may be necessary. A more suitable motor is an Exceed-RC Monster Power 15 950 kV motor, at a weight of 170 grams. Both the e-Flite motor and Monstor Power motor are out runner type motors, where the magnetic core is stationary, and the outside shell rotates. This is contrary to an in-runner motor where the outer shell remains stationary and its metal inner core rotates.

Due to a mounting difference, a custom made bracket is needed to mount the motor to the firewall. The bracket proved to be more stable for the aircraft and reduce vibrations on the fuselage, this improving the flight data obtained.

Another aspect of instrumentation is the physics of flight and how the deflection angles generate lift of the aircraft. More of the kinematics and dynamics are in Chapter 4. For the aircraft all four control surfaces are needed to generate lift. The throttle speed needs to be between 80% and 90% full throttle, to generate forward velocity, utilizing the rudder for heading correction, next the full elevator deflection is applied along with the aileron to lift the aircraft into the air. Detailed movement measurements are taken from the airplane's ailerons (left and right), elevator, and rudder. The measurements below represent the angle deflection from center position.

Elevator:	Up (stick down) = 11°	Down (stick up) = 12°
Rudder:	Left (stick left) = 20°.	Right (stick right) = 17°
Aileron (left):	Up (stick right) = 20°.	Down (stick left) = 18°
Aileron (right):	Up (stick left) = 16°.	Down (stick right) = 16°

3.3.1 Inertial Measurements Unit and Attitude and Heading Reference System

An Inertial sensor measures both rotational and linear movement without the need to reference an external coordinate system. Inherent in these sensors is a gyroscope, used to measure rotation rates, and an accelerometer to measure acceleration. Commonly, IMUs are constructed of micro electro-mechanical systems (MEMS) and consist of 3 gyroscopes and 3 accelerometers all respectively placed orthogonally to obtain measurements in 3 axes commonly referred to as X, Y, and Z or in Euler angles as Yaw, Pitch, and Roll. The most effective installation location to obtain flight data and control the aircraft in autonomous mode is as close to the vehicles center of gravity as possible. This placement tends to negate rotational body effects that may be experienced by the device.

An Attitude and Heading Reference System (AHRS) is commonly available as a MEMS device and contains 3 gyroscopes, 3 accelerometers and 3 magnetometers all respectively orthogonal to collect data on all three axes. The fundamental difference between an IMU and an AHRS is the addition of an on-board processing system. Essentially, an AHRS provides solved attitude and heading solutions where as an IMU would have to deliver collected sensor data to an additional (sometimes external) device to obtain a solution. Kalman filtering is typically implemented to compute the solution

from these multiple sources. AHRS differ from traditional navigation systems by attempting to estimate only attitude (i.e. roll, pitch, yaw, also known as heading) states, rather than attitude, position and velocity.

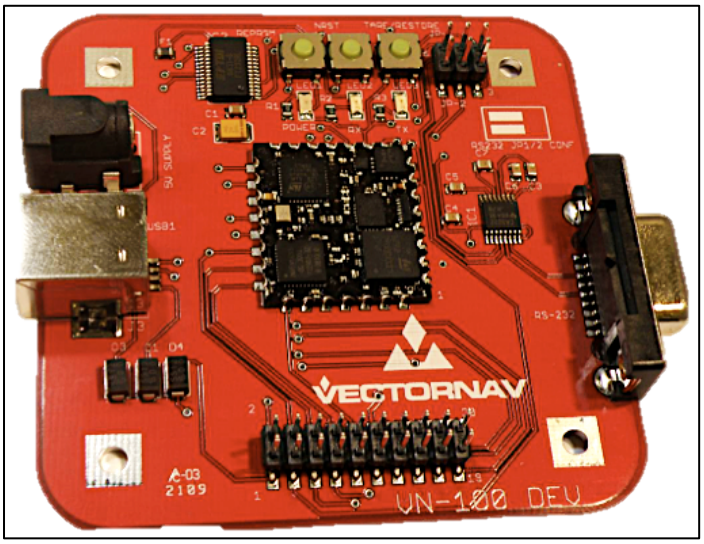


Figure 3.4 Vectornav VN-100 Development board
Source: Vectornav UM100 manual

In this project, a Vectornav VN100 Development board shown in Figure 3.4 is used. These development boards provide direct serial output pin headers on the board itself. The device's dimension is 75.5x75.8x12.2mm, weighs 40g and has an input power range of 3.3 – 5v. The VN100 has three rate gyroscopes, three accelerometers, three magnetometers and onboard Kalman filtering, making it an AHRS. The data rate of the device is 50Hz and it is set to output Euler angles in full-scale range of ±180 degrees. Interface to the device can be over RS232, USB or UART with a maximum baud rate of

921,600 baud; 115,200 baud is used for this project. One advantage of this unit versus others is the proprietary development and interface environment called ‘sensor explorer’ that allows a user to calibrate or adjust output parameters directly on the device. [17]

3.3.2 Microcontrollers

A microcontroller is a small computer-on-a-chip that is used to process data or manipulate electronic devices. With respect to the aforementioned AHRS, all of the process and decision-making happens inside the microcontroller, making it the ‘brain’ of the UAV. Typical microcontrollers contain an integrated CPU, RAM or ROM and I/O interfaces, as well as COM ports, inter-integrated circuit (I^2C) ports, analog to digital (A2D) and digital to analog (D2A) converter channels.

3.3.2.1 XMOS XC-2 microcontroller

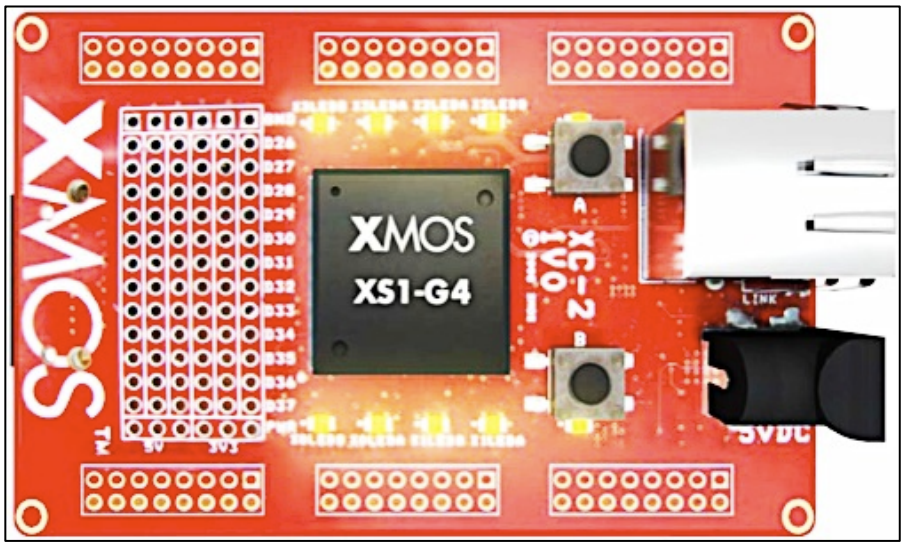


Figure 3.5 XMOS XC-2 Ethernet embedded system
Source: XMOS

Installed alongside the AHRS, an XMOS XC-2 Ethernet Kit microcontroller, shown in Figure 3.5, is used with the apprentice UAV. The XC-2 is an 84-pin user I/O 4-core microcontroller, measuring 84x54mm, running a XS1-G4 4-Xcore, 100MHz microchip, capable of 400 million instructions per second (MIPS) and 64Kb of RAM per core. Programming for the XC-2 card is done through a J-TAG device called an X-Tag that has a USB interface, with a derivative of the C language called XC utilizing an Eclipse-based environment called the XMOS Development Environment (XDE) supplied by XMOS. [18]

The XC-2 and AHRS is fastened on a non-ferrous magnetic plate that is mounted inside the fuselage of the aircraft. Large nylon screws are installed and glued into the fuselage frame of the aircraft to reduce movement and disturbances seen by the XMOS and especially the AHRS. An advantage of using the XMOS over other microcontrollers is the ability to run multiple tasks independently. For example, autonomous code can run on one core and mathematical calculations on another, and the result directly communicated with the core can operate and maneuver a vehicle in real time. Essentially each task can run independently and if necessary more tasks can be added by adding additional cores.

3.3.2.2 XMOS devices

XMOS processing is event-driven and not dependent on the conventional interrupt schemes found in most microcontrollers. As mentioned in the previous section, each XCore can provide up to 100 MIPS per core for deterministic real-time tasks, which has up to 8 threads, each having access to 16 dedicated registers and 64KBytes of shared SRAM. Each XCore has up to 64 I/O pins (model dependent) that can provide 10ns-timing resolution, allowing the programmer to create software-defined peripherals, such as UART or PWM.

From a software development standpoint, the implementation and enabling of concurrent tasks, real-time control, hardware timers and ports is programmed using the XC language. XC provides the ability to combine clean, high- level modules with efficient and safe hardware I/O, if necessary, logic and basic general-purpose algorithmic code can be written and implemented in C/C++ and utilized in the XC environment.

The advantage of XMOS scalability manifests in real-time robotics applications. With the popularity of unmanned systems increasing, and the need for vehicle platforms to perform beyond their respective design parameters, this architecture allows for the easy integration of additional sensors, actuators, and other components. An XMOS-based robotic system's advantage is fundamentally any available general-purpose I/O (GPIO) pin can be used for equipment interfaces. [19]

3.3.3 Servos

The most common actuation device for any radio-controlled vehicle or UAV is a servo-motor, shown in Figure 3.6. This device is typically controlled with a pulse width modulation (PWM) signal. The angular rotation of the servo is directly transferred to its respective (connected) control surface through the use of mechanical linkages.

Figure 3.6 Standard aircraft servo
source: spektrum-rc

For typical fixed-wing aircraft there are three control surfaces: aileron, elevator, and rudder, along with the throttle. In electric vehicles the PWM for the throttle is received from the transmitter and is sent to an Electronic Speed Controller (ESC) and then to the motor to obtain the desired speed.

3.3.4 Electronic Speed Controller

All electric motor applications must have some form of speed control, in the early times of RC aircraft the speed of the motor was controlled via servo, however, modern electronic speed controllers integrate a Field Effect Transistor (FET) to maintain a fixed resistance from the aircraft's initial power on. Additionally modern ESCs incorporate a Battery Eliminator Circuit and a small microprocessor for regulating the voltage from the battery pack to the receiver, and additionally not allowing the motor to function if power is applied while the RC Transmitter is in a high throttle position. [34] The Apprentice 15e contains a 30Amp ESC that delivers 3.5A continuous current and a 5V Switch-Mode BEC Circuit that drives power to the receiver and servos. [15]

3.3.5 Global Positioning System

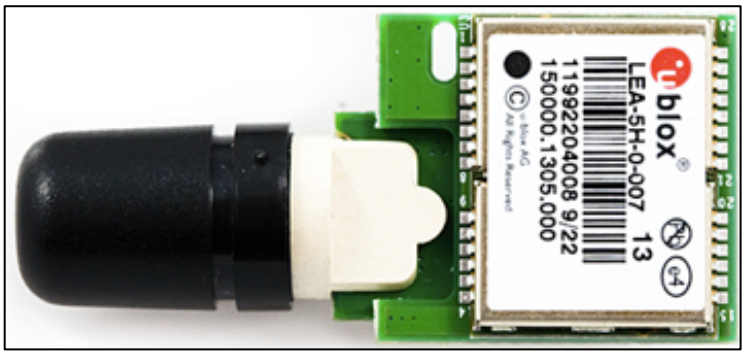


Figure 3.7 u-Blox LEA-5H GPS module
source: sparkfun

For waypoint navigation, the implementation of a GPS receiver is critical. The Apprentice UAV utilizes a u-Blox LEA-5H GPS unit, shown in Figure 3.7. This device is a 50-channel receiver and is 47.1 x 22.9mm weighing in at 16g. Communication to this

device is over UART at a rate 57600 baud with a 4Hz update rate. One advantage of this module is its interface environment, the u-Blox u-Center that allows for direct adjustments to the device. For the purposes of this project, the GPS device was not utilized as it was beyond the scope of the project, however the device remains installed on the aircraft [21] [22] [23].

3.3.6 Flight Data Collection

For the Apprentice UAV, the autopilot consists of the XMOS microcontroller, and Vectornav IMU. Data is collected onboard the aircraft on a micro SD memory card, and verified as a post flight process. The data-logging device, from OpenLog, is a small 4x15x19mm, and communicates directly over UART with the XMOS. The flight data collection process was repeated several times with varying weather conditions. [24] [25]

3.4 Flight rules, regulations, location

Fixed-wing aircraft require a runway to take off and land, as well as a designated RC park to fly legally. The American Modelers Association (AMA) governs designations of approved RC parks. The AMA is the governing body that coordinates with the United States FAA to make sure model aircraft pilots are in compliance with Federal and State regulations regarding the National Airspace. Membership and registration to the AMA is required for all pilots because it provides insurance in the event of bodily injury or property damage. [16]

The nearest RC park within proximity of the University of Denver campus is located at the Cherry Creek State Park. The Cherry Creek Flight Club (known as the Denver RC Eagles) maintains the runway and field. A flight instructor is necessary for new pilots.

The training consists of take offs from all four directional runways, landing on all four runway directions, coordinated mid air left and right turning with both aileron and rudder, understanding the flight ceiling of 121.92m (400ft), and flight borders around the pit and road areas.

3.5 Sensor Assembly, Modeling and Software Integration

The standard of control of a hobby grade aircraft is accomplished by utilizing a 2.4GHz Remote Control system sending a Pulse Width Modulation signal to the respective servo. A servo under power in neutral position has a PWM in milliseconds of 1500, the minimum pulse width is 1000ms and the maximum is 2000ms. To correctly identify the pulse train of the RC receiver, a logic analyzer is used and sorted in time order sequence. The Apprentice aircraft has a Spektrum AR500 receiver, and sends a pulse train order of Aileron, Elevator, Rudder, Throttle, and Gear to the microprocessor.

In addition to servo commands, the XMOS microprocessor receives x-axis, y-axis, z-axis position data from the AHRS and generates the necessary output servo commands to maintain flight. The AHRS serially transmits data to the XMOS in a scaled degrees form -18000 to +180000 for all the Euler angles. This is because the XMOS does

not support floating-point numbers, for example. a roll value of +133.85 degrees would display as 13385 on the console terminal. This integration of hardware and visual cueing is utilized throughout this project.

An added benefit in the Apprentice aircraft is the extra space in the fuselage where the servos are mounted. Given the sensitivity of the sensors, mainly the AHRS, a wooden plate is cut to fit inside the Apprentice's fuselage. Additionally, to remove any magnetic interference, nylon screws and standoffs are fastened from the XMOS and AHRS into the wooden plate. The wooden plate is stabilized in the fuselage by wooden standoffs that are held in place by the foam construction of the fuselage. With time, however the wood plate begun to fail and new plate is needed. Version 2 of the mounting plate utilizes a non-Ferro magnetic plate with 2cm diameter holes throughout the plate, allowing the sensors to be closely wired and easily removable or replaceable if necessary. The fuselage standoffs are replaced with longer 10cm nylon screws and nuts and glued into the fuselage for a more stable mounting plate.

The sensor suite that includes the microcontroller, AHRS, flight logger, and servos underneath is installed in the fuselage of the aircraft. Figure 3.8 shows the inside of the fuselage with the sensor suite installed and mounted. The GPS receiver is mounted between the aircraft cowl and leading edge of the wing. The weight of the sensor suite including the non-Ferro magnetic plate is 180 grams, thus bringing the overall weight of the aircraft, including the ~~l~~arger electric motor to 3kg.

To deliver power to the components, a power bus is installed from the Electronic Speed Controller to a Medusa Research voltage regulator. The voltage regulator, along with an external on/off switch ensures the XMOS receives the necessary 5V at 1A to power up from the source. From the XMOS microcontroller board, the servos, Vectornav AHRS, data logger, and GPS device receive their respective power. The external switch, mentioned earlier, powers off the sensor suite to prevent heat buildup when not in use.

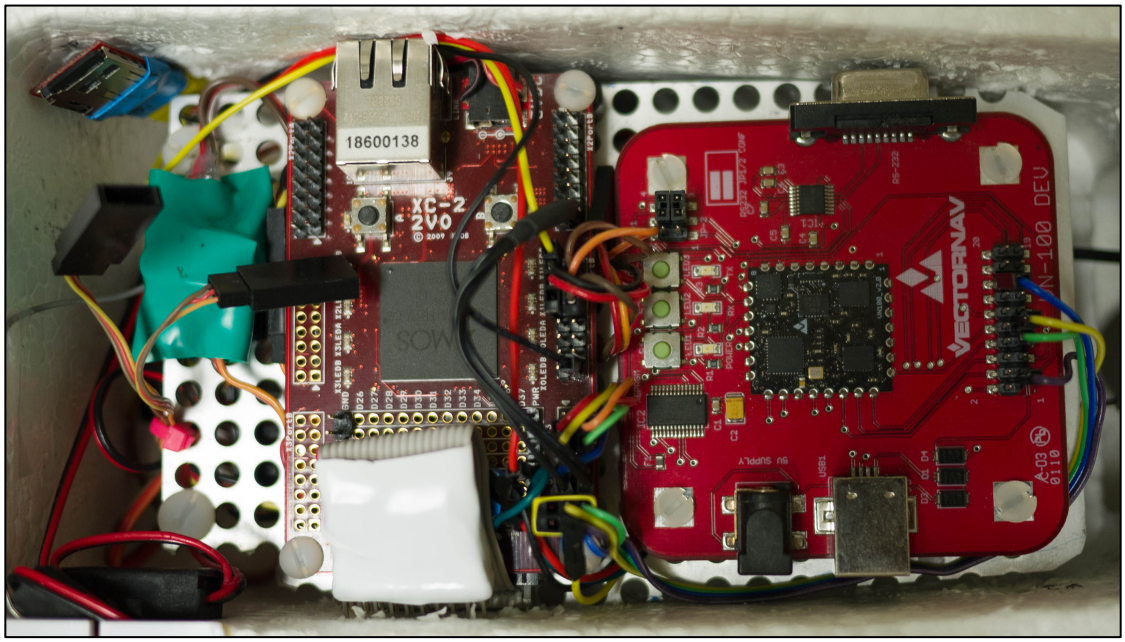


Figure 3.8 Inside the UAV Fuselage with XMOS, AHRS, OpenLog installed

As previously mentioned, all the sensors are directly connected to the XMOS. Figure 3.9 shows the block diagram of how and where each sensor and device is connected. For this platform, the XMOS is utilizing 7 channels and running 8 tasks concurrently through 12 Input-Output (I/O) lines.

3.6 UAV System Identification Procedure

The remainder of this thesis is focused on obtaining data from trim (cruise) flight for system identification. The purpose of system identification is outlined more in Chapter 5. However, the flowchart in Figure 3.10 shows the system identification process for this project.

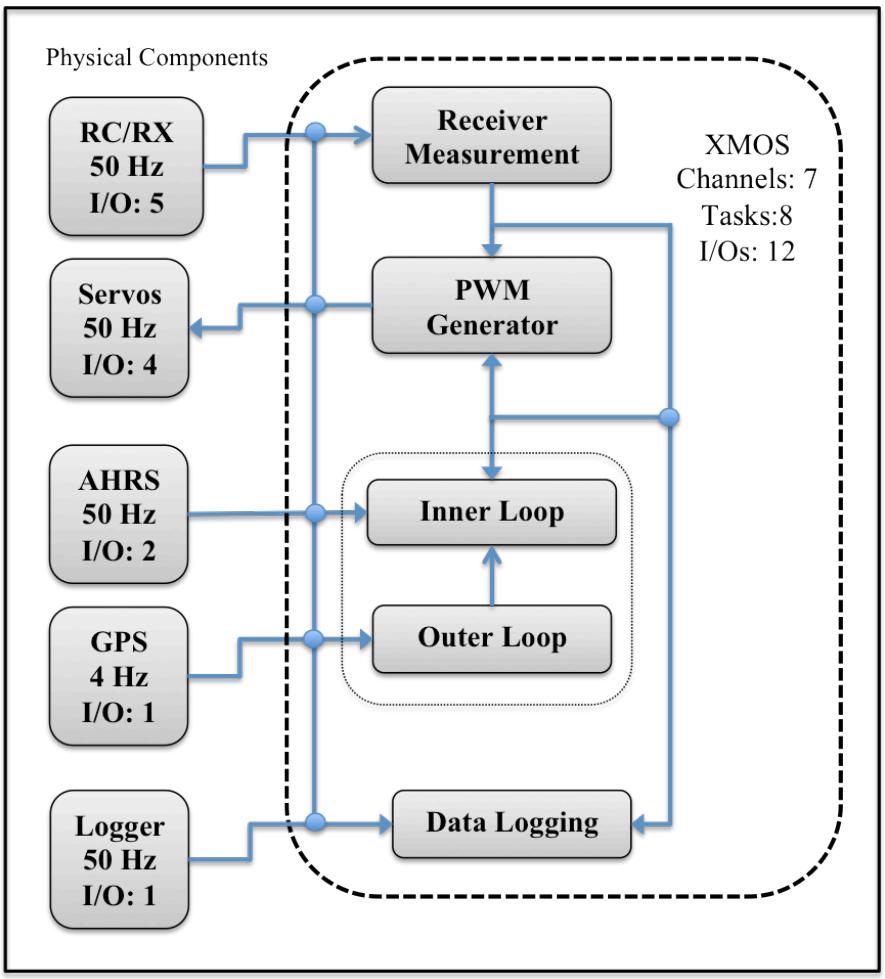


Figure 3.9 Block diagram of the UAV XMOS software.

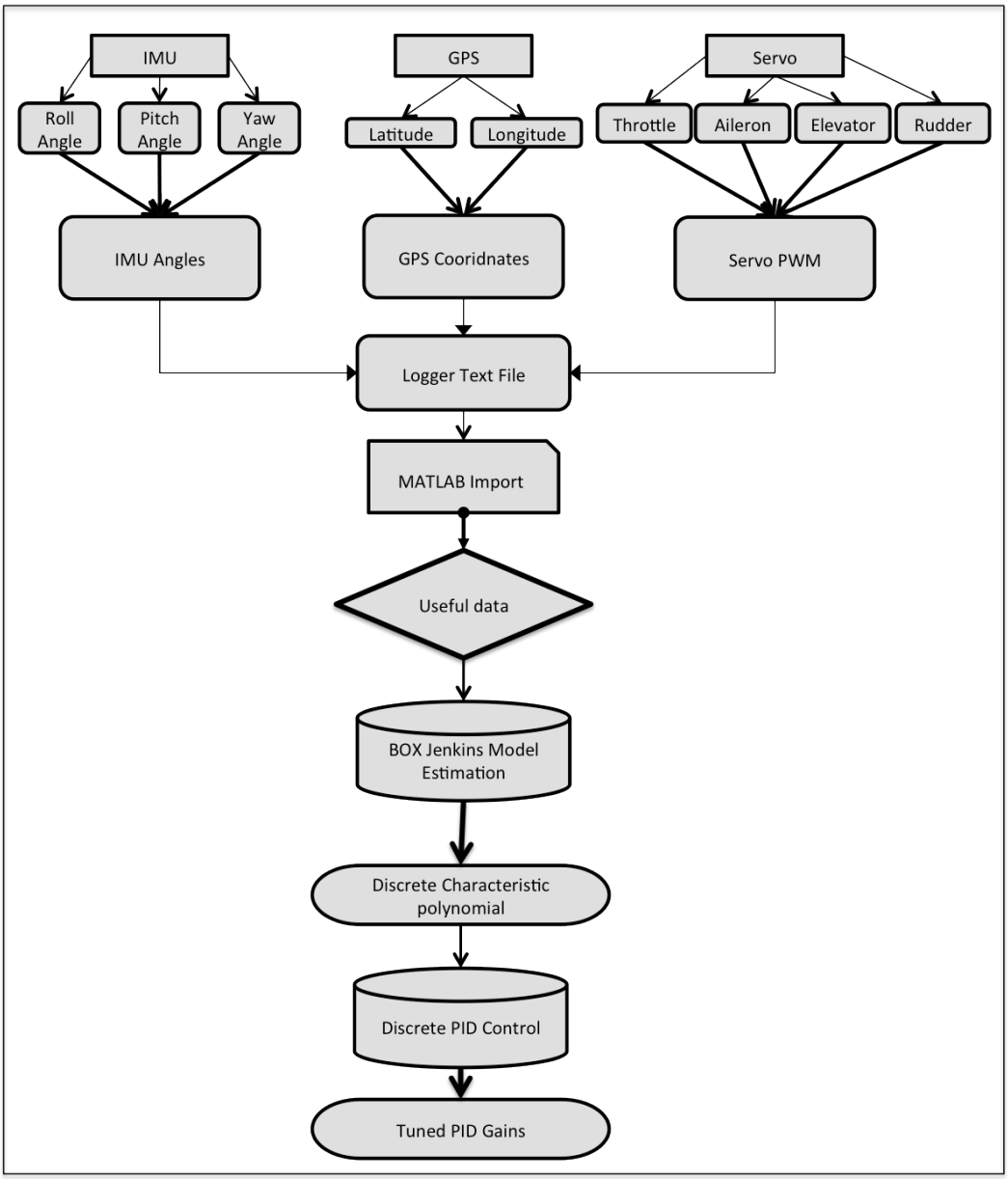


Figure 3.10 System Identification Process

Chapter 4 – UAV Kinematics, Dynamics and Equations of Motion

The study of aircraft kinematics and dynamics is important to understand when it comes to UAVs. However, the study of how the aircraft is oriented with respect to the Earth is equally critical when developing an Unmanned Aerial Vehicle. In this chapter, various coordinate systems are used to describe the position and orientation of the aircraft, and the transformation between these coordinate systems, as well as the Kinematics and Dynamics of an aircraft are explored. The text in this chapter is based on R. Beard and T. McLain [26], for further details and information related to UAV dynamics and simulation, please refer to the citation.

The use of different coordinate systems is necessary because Newton's equations of motion are derived to a fixed inertial reference, and because it is easy to describe aerodynamics forces and torques in a body fixed reference frame. The two coordinate frames can be transformed from one to another via rotation and translation.

4.1 Rotation Matrices

The first step in describing rotation matrices is to start by taking into account the two coordinate frames, shown in Figure 4.1, named F^0 and F^1, where frame F^0 is $\mathbf{i}^0$, $\mathbf{j}^0$, $\mathbf{k}^0$ and frame F^1 is $\mathbf{i}^1$, $\mathbf{j}^1$, $\mathbf{k}^1$, and both vector sets are mutually exclusive.

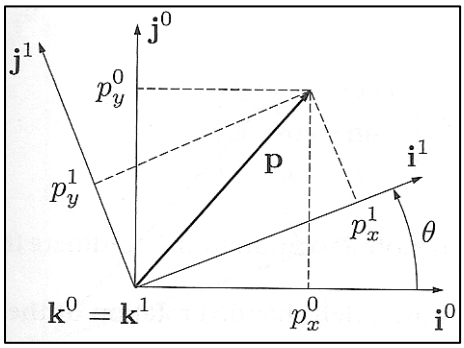


Figure 4.1. Rotation in 2-dimensional space

The vector **p** is expressed in frame F^0 as $\mathbf{p} = p_x^0\mathbf{i}^0 + p_y^0\mathbf{j}^0 + p_z^0\mathbf{k}^0$ and alternatively as $\mathbf{p} = p_x^1\mathbf{i}^1 + p_y^1\mathbf{j}^1 + p_z^1\mathbf{k}^1$ for frame F^1. Since **p** in $F^0 = \mathbf{p}$ in F^1, the expression becomes $p_x^1\mathbf{i}^1 + p_y^1\mathbf{j}^1 + p_z^1\mathbf{k}^1 = p_x^0\mathbf{i}^0 + p_y^0\mathbf{j}^0 + p_z^0\mathbf{k}^0$, and the dot product yields $\mathbf{p}^1 = \begin{pmatrix} p_x^1 \\ p_y^1 \\ p_z^1 \end{pmatrix} = \begin{pmatrix} (\mathbf{i}^1\cdot\mathbf{i}^0) & (\mathbf{i}^1\cdot\mathbf{j}^0) & (\mathbf{i}^1\cdot\mathbf{k}^0) \\ (\mathbf{j}^1\cdot\mathbf{i}^0) & (\mathbf{j}^1\cdot\mathbf{j}^0) & (\mathbf{j}^1\cdot\mathbf{k}^0) \\ (\mathbf{k}^1\cdot\mathbf{i}^0) & (\mathbf{k}^1\cdot\mathbf{j}^0) & (\mathbf{k}^1\cdot\mathbf{k}^0) \end{pmatrix} \begin{pmatrix} p_x^0 \\ p_y^0 \\ p_z^0 \end{pmatrix}$ in matrix form.

Using the notation of $\mathcal{R}_0^1$ to denote the rotation from frame F^0 to frame F^1, an equation from figure 4-1 can be derived and expressed as: $\mathbf{p}^1 = \mathcal{R}_0^1\mathbf{p}^0$. [4-1]

By way of rotation of the system about a particular axis yields a unique result for each axis, a x-axis right-handed rotation yields $\mathcal{R}_0^1 = \begin{pmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$, for the y-axis it

becomes $\mathcal{R}_0^1 = \begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix}$, and about the z-axis, $\mathcal{R}_0^1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{pmatrix}$

The $\mathcal{R}_0^1$ matrices above are orthonormal.

4.2 UAV Coordinate Frames

There are several coordinate systems that exist for UAVs, which describe their dynamic behavior. Seven frames are explored in thesis: the inertial, vehicle, vehicle-1, vehicle-2, body, stability, and wind. The inertial and vehicle frames differ from the others because they are related by translation, where as next three are related by rotation. The attitude of the aircraft is described by Euler Angles (roll, pitch and yaw) for the vehicle-1, vehicle-2 and body frames for relative orientations, and the angle of attack and sideslip angle compose the relative orientation for the body, and stability and wind coordinate frames.

The inertial coordinate system is an earth-fixed coordinate system with its origin at a defined location. The inertial frame is commonly referred to as the north-east-down system, where the unit vector $\mathbf{i}^i$ is directed north, $\mathbf{j}^i$ is east and $\mathbf{k}^i$ is directed towards Earth, down. The inertial frame is shown as $\mathcal{F}^i$.

The vehicle frame is almost identical to the inertial frame with respect to where the unit vectors are pointing, however the difference exists in the origin, where it is now at the center of gravity of the UAV. The Vehicle frame is denoted as $\mathcal{F}^v$ and is shown in Figure 4.2.

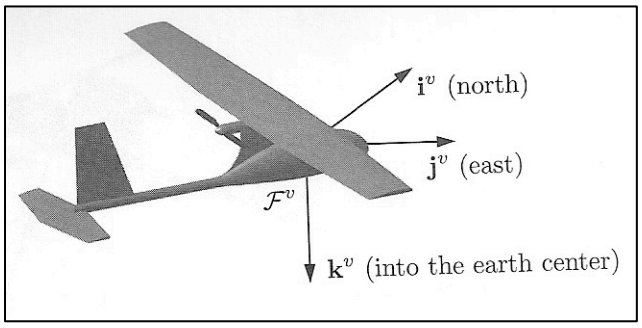


Figure 4.2. Vehicle coordinate frame

As mentioned earlier, the vehicle-1 ($\mathcal{F}^{v1}$), vehicle-2 ($\mathcal{F}^{v2}$) and body ($\mathcal{F}^b$) frame are related to the Vehicle Frame $\mathcal{F}^v$ except that each differs by a rotation of an Euler Angle.

The vehicle-1 frame, shown as $\mathcal{F}^{v1}$ in Figure 4.3 is the right-handed (positive) rotation about $\mathbf{k}^v$ via the yaw Euler angle (ψ), where the north-east-down system changes the unit vector $\mathbf{i}^{v1}$ directed north out of the nose of the aircraft, $\mathbf{j}^{v1}$ is east out of the right wing and $\mathbf{k}^v = \mathbf{k}^{v1}$ pointing down towards the Earth. As shown in Figure 4.4, the frame $\mathcal{F}^{v2}$ is the vechle-1 frame rotation of the pitch angle (θ) about the $\mathbf{j}^{v1}$ axis, with each unit vector is still pointing in its respective direction. The final rotation is in the body frame $\mathcal{F}^b$ and is the rotation of the $\mathcal{F}^{v2}$ frame about the vector $\mathbf{i}^{v2}$ by the roll angle (ϕ), shown in Figure 4.5.

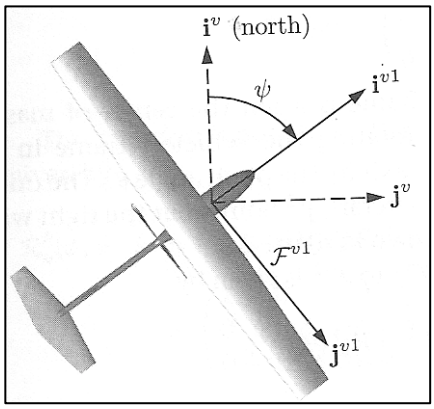


Figure 4.3 Vehicle-1 frame

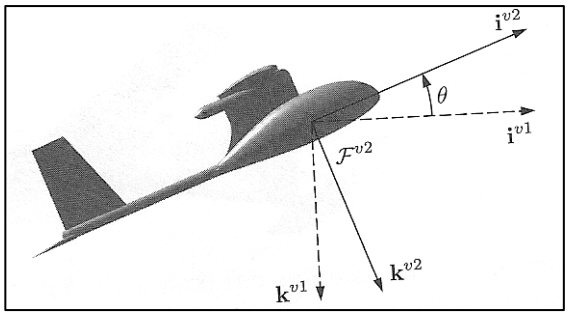


Figure 4.4. Vehicle-2 frame

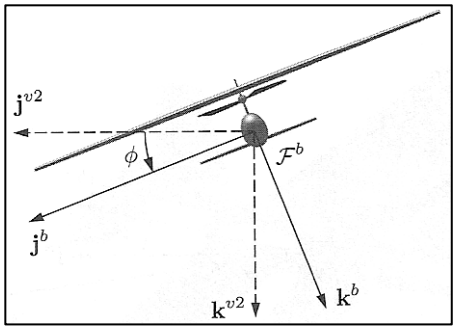


Figure 4.5. The body frame

With the rotations of the Euler angles defined, the transformation from frame to frame is shown below.

From $\mathcal{F}^v$ to $\mathcal{F}^{v1}$: $\mathbf{p}^{v1} = \mathcal{R}_v^{v1}(\psi)\mathbf{p}^v$, where $\mathcal{R}_v^{v1}(\psi) = \begin{pmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix}$

From $\mathcal{F}^{v1}$ to $\mathcal{F}^{v2}$: $\mathbf{p}^{v2} = \mathcal{R}_{v1}^{v2}(\theta)\mathbf{p}^{v1}$, where $\mathcal{R}_{v1}^{v2}(\theta) = \begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix}$

From $\mathcal{F}^{v2}$ to $\mathcal{F}^{b}$: $\mathbf{p}^{b} = \mathcal{R}_{v2}^{b}(\phi)\mathbf{p}^{v2}$, where $\mathcal{R}_{v2}^{b}(\phi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{pmatrix}$

Making the overall transformation from $\mathcal{F}^{v}$ to $\mathcal{F}^{b}$ as a combination of rotations

$$\mathcal{R}_{v}^{b}(\phi,\theta,\psi) = \mathcal{R}_{v2}^{b}(\phi)\mathcal{R}_{v1}^{v2}(\theta)\mathcal{R}_{v}^{v1}(\psi) \qquad [4\text{-}2]$$

$$= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix} \begin{pmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\cos\theta \\ \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\cos\theta \end{pmatrix} \qquad [4\text{-}3]$$

The angle of attack, α, is in the stability frame $\mathcal{F}^{s}$ is the positive angle with respect to the airspeed vector, $\mathbf{V}_a$ that is necessary to generate lift for an aircraft, and is defined as a left-handed rotation about $\mathbf{j}^{b}$ so that $\mathbf{i}^{s}$ is aligned with the projection of $\mathbf{V}_a$. The wind frame, $\mathcal{F}^{w}$, is the angle between $\mathbf{V}_a$ and the $\mathbf{i}^{b},\mathbf{j}^{b},\mathbf{k}^{b}$ plane is called the side-slip angle denoted by β, where by performing a right-hand rotation about $\mathbf{k}^{s}$ where the unit vector $\mathbf{i}^{w}$ is aligned with $\mathbf{V}_a$. Figure 4.6 shows the angle of attack and Figure 4.7 shows the side-slip angle.

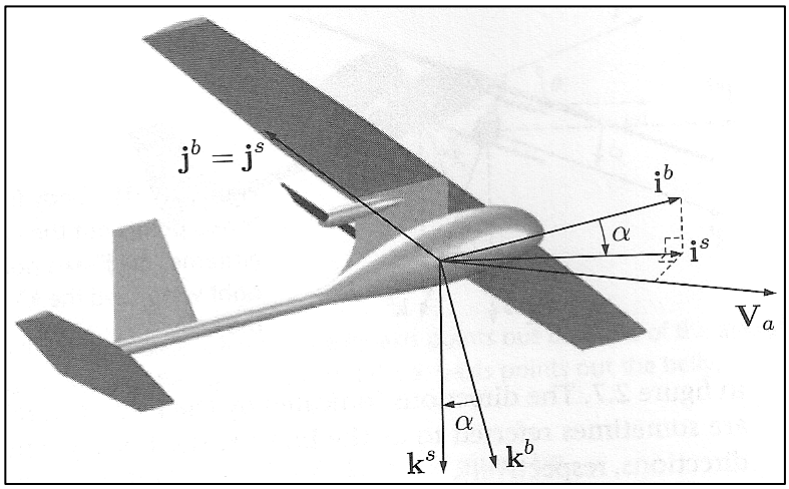


Figure 4.6 The stability frame showing the angle of attack

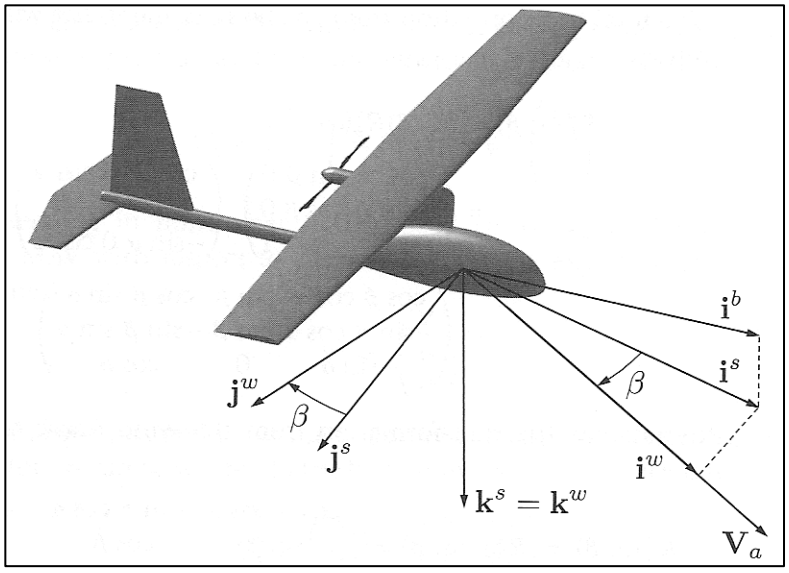


Figure 4.7 The wind frame showing the side-slip angle

With the rotations of the Euler angles defined, the transformation from frame to frame is shown below.

From $\mathcal{F}^b$ to $\mathcal{F}^s$: $\mathbf{p}^s = \mathcal{R}_b^s(\alpha)\mathbf{p}^b$, where $\mathcal{R}_b^s(\alpha) = \begin{pmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{pmatrix}$

From $\mathcal{F}^s$ to $\mathcal{F}^w$: $\mathbf{p}^w = \mathcal{R}_s^w(\beta)\mathbf{p}^s$, where $\mathcal{R}_s^w(\beta) = \begin{pmatrix} \cos\beta & \sin\beta & 0 \\ -\sin\beta & \cos\beta & 0 \\ 0 & 0 & 1 \end{pmatrix}$

Makin the overall transformation from $\mathcal{F}^b$ to $\mathcal{F}^w$ as a combination of rotations

$$\mathcal{R}_b^w(\alpha,\beta) = \mathcal{R}_s^w(\beta)\mathcal{R}_b^s(\alpha) = \begin{pmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{pmatrix}\begin{pmatrix} \cos\beta & \sin\beta & 0 \\ -\sin\beta & \cos\beta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} \cos\beta\cos\alpha & \sin\beta & \cos\beta\sin\alpha \\ -\sin\beta\cos\alpha & \cos\beta & -\sin\beta\sin\alpha \\ -\sin\alpha & 0 & \cos\alpha \end{pmatrix}$$

During the developing the dynamic equation of motion for a UAV, the inertial forces experienced by the UAV are reliant on velocities and accelerations relative to a fixed reference frame. However, the aerodynamic forces depend on the velocity of the airframe relative to the surrounding air. Since wind is almost always present with UAVs distinctions between air speed and ground speed are necessary. The velocity with respect to the surrounding air $\mathbf{V}_a$ represents the former, and the latter by the velocity with respect to the inertial frame $\mathbf{V}_g$. Air speed and ground speed related by the expression.

$$\mathbf{V}_a = \mathbf{V}_g - \mathbf{V}_w \qquad [4\text{-}4]$$

Here, $\mathbf{V}_w$ is the wind velocity relative to the inertial frame.

In the body frame, the UAV velocity, $\mathbf{V}_g$, can be expressed in terms of components along the $\mathbf{i}^b$, $\mathbf{j}^b$, and $\mathbf{k}^b$ axes as $\mathbf{V}_g^b = \begin{pmatrix} u \\ v \\ w \end{pmatrix}$ where $\mathbf{V}_g^b$ is the velocity of the UAV with respect to the inertial frame, as expressed in the body frame. Additionally, when the north, east and down components of the wind as w_n, w_e, and w_d respectively, the expression for wind velocity in the body frame becomes

$$\mathbf{V}_w^b = \begin{pmatrix} u_w \\ v_w \\ w_w \end{pmatrix} = \mathcal{R}_w^b(\phi,\theta,\psi) \begin{pmatrix} w_n \\ w_e \\ w_d \end{pmatrix}$$

Since airspeed vector $\mathbf{V}_a$ is the velocity of the UAV with respect to the wind, the expression can be written as $\mathbf{V}_a^w = \begin{pmatrix} V_a \\ 0 \\ 0 \end{pmatrix}$.

For the body-frame components of the airspeed vector, u_r, v_r, and w_r the body frame can be written as $\mathbf{V}_a^b = \begin{pmatrix} u_r \\ v_r \\ w_r \end{pmatrix} = \begin{pmatrix} u - u_w \\ v - v_w \\ w - w_w \end{pmatrix}$

In the development of a UAV, u_r, v_r, and w_r are used to calculate aerodynamic forces and moments acting on the UAV. The wind velocity components u_w, v_w, and w_w usually come from wind model as inputs to the equations of motion. And the body-frame

velocity components u, v, and w are available from the solution of the equation of motion. The combinations of these are the airspeed vector body-frame components in terms of the airspeed magnitude angle of attack and sideslip angle as.

$$\mathbf{V}_a^b = \begin{pmatrix} u_r \\ v_r \\ w_r \end{pmatrix} = \mathcal{R}_w^b \begin{pmatrix} V_a \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} \cos\beta\cos\alpha & \sin\beta & \cos\beta\sin\alpha \\ -\sin\beta\cos\alpha & \cos\beta & -\sin\beta\sin\alpha \\ -\sin\alpha & 0 & \cos\alpha \end{pmatrix} \begin{pmatrix} V_a \\ 0 \\ 0 \end{pmatrix},$$

$$\text{implying that } \begin{pmatrix} u_r \\ v_r \\ w_r \end{pmatrix} = V_a \begin{pmatrix} \cos\alpha\cos\beta \\ \sin\beta \\ \sin\alpha\sin\beta \end{pmatrix} \qquad [4\text{-}5]$$

Since the aerodynamic forces are more commonly expressed in terms of $\mathbf{V}_a, \alpha,$ and β.

$$\begin{aligned} V_a &= \sqrt{u_r^2 + v_r^2 + w_r^2} \\ \alpha &= \tan^{-1}\left(\frac{w_r}{u_r}\right) \\ \beta &= \sin^{-1}\left(\frac{v_r}{\sqrt{u_r^2 + v_r^2 + w_r^2}}\right) \end{aligned} \qquad [4\text{-}6]$$

4.3 Kinematics and Dynamics

The equations of motion for a UAV are composed of twelve state variables, listed in Table 4.1, three position states and three velocity states associated with the translational motion of the UAV. Additionally there are three angular position and three angular velocity states associated with the rotational motion.

Name	Description
p_n	Inertial north position of the UAV along the $\mathbf{i}^i$ in the $\mathcal{F}^i$
p_e	Inertial east position of the UAV along the $\mathbf{j}^i$ in the $\mathcal{F}^i$
p_d	Inertial down position of the UAV along the $\mathbf{k}^i$ in the $\mathcal{F}^i$
u	Body frame velocity along $\mathbf{i}^b$ in $\mathcal{F}^b$
v	Body frame velocity along $\mathbf{j}^b$ in $\mathcal{F}^b$
w	Body frame velocity along $\mathbf{k}^b$ in $\mathcal{F}^b$
ϕ	Roll angle with respect to $\mathcal{F}^{v2}$
θ	Pitch angle with respect to $\mathcal{F}^{v1}$
ψ	Yaw angle with respect to $\mathcal{F}^{v}$
p	Roll rate measured along $\mathcal{F}^b$
q	Pitch rate measured along $\mathcal{F}^b$
r	Yaw rate measured along $\mathcal{F}^b$

Table 4.1 State variables for equations of motion

In Figure 4.8, the state variables are shown schematically, where the north-east-down position of the UAV (p_n, p_e, p_d) are defined relative to the inertial frame, $h = -p_d$ may be used to denote the attitude. The linear velocities (u,v,w) and angular velocities (p, q, r) of are defined with respect to the body frame, and the Euler angles (roll, pitch, yaw) defined with respect to the vehicle-2, vehicle-1, and vehicle frames respectively. Angular rates (p,q,r) are not simply time derivatives of the attitude angles (ϕ, θ, ψ) however, $p = \dot{\phi}$, $q = \dot{\theta}$, and $r = \dot{\psi}$ only at the instant that $\phi = \theta = 0$. Generally, the angular rates (p, q, r) are function of the time derivatives of the attitude angles ($\dot{\phi}, \dot{\theta}, \dot{\psi}$) and the angles ϕ and θ.

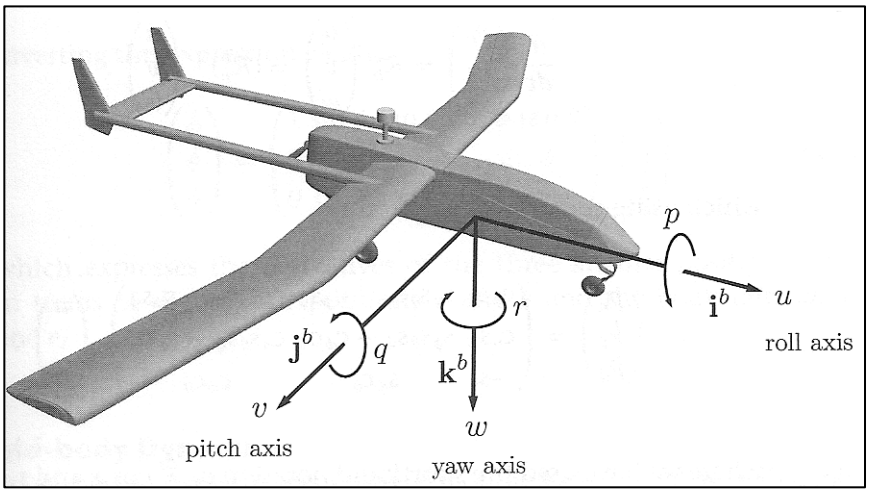


Figure 4.8 Definition of axes of motion

Translation velocity is frequently expressed in terms of the velocity components. Here, the components u,v,w correspond to the inertial velocity of the vehicle projected onto the $\mathbf{i}^b,\mathbf{j}^b,\mathbf{k}^b$ axes, respectively. However, relating the translational velocity and position requires differentiation and a rotational transformation, and when rotating equation 4-2 yields

$$\begin{pmatrix} \dot{p}_n \\ \dot{p}_e \\ \dot{p}_d \end{pmatrix} = \begin{pmatrix} \cos\theta\cos\psi & \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi \\ \cos\theta\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{pmatrix} \begin{pmatrix} u \\ v \\ w \end{pmatrix}.$$ This

is kinematic relation relates the derivative of position to velocity, where forces or accelerations are not considered.

4.3.1 Kinematics

There exists a complex relationship between ϕ, θ, ψ and p, q, r. This is because angular rates are defined in the body frame $\mathcal{F}^b$, and angular positions are defined in three different coordinate frames: roll from $\mathcal{F}^{v2}$ to $\mathcal{F}^b$ about the $\mathbf{i}^{v2} = \mathbf{i}^b$ axis; pitch from $\mathcal{F}^{v1}$ to $\mathcal{F}^{v2}$ about the $\mathbf{j}^{v1} = \mathbf{j}^{v2}$; and yaw from $\mathcal{F}^v$ to $\mathcal{F}^{v1}$ about the $\mathbf{k}^v = \mathbf{k}^{v1}$ axis.

Following the proper rotational transformation, the body frame angular rates can be expressed as the derivatives of the Euler angles:

$$\begin{pmatrix} p \\ q \\ r \end{pmatrix} = \begin{pmatrix} \dot{\phi} \\ 0 \\ 0 \end{pmatrix} + \mathcal{R}_{v2}^{b}(\phi) \begin{pmatrix} 0 \\ \dot{\theta} \\ 0 \end{pmatrix} + \mathcal{R}_{v2}^{b}(\phi)\mathcal{R}_{v1}^{v2}(\theta) \begin{pmatrix} 0 \\ 0 \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{pmatrix} \begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix}$$

Where inverting the expressions gives

$$\begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \dfrac{\sin\phi}{\cos\theta} & \dfrac{\cos\phi}{\cos\theta} \end{pmatrix} \begin{pmatrix} p \\ q \\ r \end{pmatrix} \qquad [4\text{-}7]$$

Which expresses the derivatives of the three angular position states in terms of the angular positions of roll, pitch and the body rates p, q, and r.

4.3.2 Rigid-Body Dynamics

To derive the dynamic equations of motion for the UAV, Newton's second law is applied to the translational degrees of freedom and next to the rotational degrees of freedom. An additional assumption is a flat Earth model, which is appropriate for small

and miniature air vehicles. Although motion is referenced to a fixed frame, it can be expressed using vector components associates with other frames.

Using the UAV velocity vector $\mathbf{V}_g$, commonly expressed in the body frame as $\mathbf{V}_g^b = \begin{pmatrix} u & v & w \end{pmatrix}^T$, where $\mathbf{V}_g^b$ is the velocity of the UAV with respect to the ground as expressed in the body frame.

4.3.2.1 – Translational Motion

Newton's second law applied to a body undergoing translational motion can be stated as $\mathrm{m}\frac{d\mathbf{V}_g}{dt_i} = \mathbf{f}$ [4-8]

where m is the mass of the UAV, $\frac{d\mathbf{V}_g}{dt_i}$ is the time derivative in the inertial frame, and $\mathbf{f}$ is the sum of all external forces acting on the UAV including gravity, aerodynamic forces, and propulsion forces. The derivative of velocity taken in the inertial frame expressed as both the derivative in the body frame and the angular velocity is

$$\frac{d\mathbf{V}_g}{dt_i} = \frac{d\mathbf{V}_g}{dt_b} + \omega_{b/i} \times \mathbf{V}_g \qquad [4\text{-}9]$$

Where $\omega_{b/i}$ is the angular velocity of the UAV with respect to the inertial frame.

Since the aircraft is maneuvering in flight, the differentiated combination of equations 4.8 and 4.9 allow Newton's second law to be

$$\mathrm{m}\left(\frac{d\mathbf{V}_g^b}{dt_b} + \omega_{b/i}^b \times \mathbf{V}_g\right) = \mathbf{f}^b \qquad [4\text{-}10]$$

Where $\mathbf{V}_g^b = (u,v,w)^{\mathrm{T}}$ and $\omega_{b/i}^b = (p,q,r)^{\mathrm{T}}$. The vector $\mathbf{f}^b$ represents the sum of the externally applied forces and is defined in terms of its body-frame components as $\mathbf{f}^b = (f_x, f_y, f_z)^{\mathrm{T}}$.

The expression $\frac{d\mathbf{V}_g^b}{dt_b}$ is the rate of change viewed by an observer on the moving body. Since u,v,w are the instantaneous projections of $\mathbf{V}_g^b$ onto the $\mathbf{i}^b, \mathbf{j}^b, \mathbf{k}^b$ axes, it follows that $\frac{d\mathbf{V}_g^b}{dt_b} = (u,v,w)^{\mathrm{T}}$

Furthermore, by expanding the cross product and rearranging terms, equation 4-10 yields

$$\begin{pmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{pmatrix} = \begin{pmatrix} ru - qw \\ pv - ru \\ qw - pv \end{pmatrix} + \frac{1}{m}\begin{pmatrix} f_x \\ f_y \\ f_z \end{pmatrix} \qquad [4\text{-}11]$$

4.3.2.2 – Rotational Motion

For rotational motion, Newton's second law states where h is the angular momentum in vector form and m is the sum of all externally applied moments that $\frac{d\mathbf{h}}{dt_i} = \mathbf{m}$, Provided that moments are summed about the center of gravity of the UAV. The derivative of angular momentum taken in the inertial frame can be expanded as

$$\frac{d\mathbf{h}}{dt_i} = \frac{d\mathbf{h}}{dt_b} + \omega_{b/i} \times \mathbf{h} = \mathbf{m}, \text{ and as } \frac{d\mathbf{h}^b}{dt_b} + \omega_{b/i}^b \times \mathbf{h}^b = \mathbf{m}^b \qquad [4\text{-}12]$$

in the body frame.

Angular momentum is defined as the product of the inertia matrix **J** and the angular velocity vector $\mathbf{h}^b = \mathbf{J}\omega_{b/i}^b$ for a rigid body. Where **J** is

$$\begin{pmatrix} \int (y^2+z^2)d\mathrm{m} & -\int xyd\mathrm{m} & -\int xzd\mathrm{m} \\ -\int xyd\mathrm{m} & \int (x^2+z^2)d\mathrm{m} & -\int yzd\mathrm{m} \\ -\int xzd\mathrm{m} & -\int yzd\mathrm{m} & \int (x^2+y^2)d\mathrm{m} \end{pmatrix} = \begin{pmatrix} J_x & -J_{xy} & -J_{xz} \\ -J_{xy} & J_y & -J_{yz} \\ -J_{xz} & -J_{yz} & -J_z \end{pmatrix} \qquad [4\text{-}13]$$

The moments of inertia are the diagonal terms, and the products of inertia non-diagonal terms. The moments of inertia are measures of the aircraft's tendency to oppose acceleration about a specific axis of rotation.

A resultant of the integrations in equation 4-13, **J** is constant when viewed from the body frame, hence, $\frac{d\mathbf{J}}{dt_b}$=0. Taking derivatives and substituting, equation 4-12 becomes: $\mathbf{J}\frac{d\omega_{b/i}^b}{dt_b} + \omega_{b/i}^b \times \mathbf{J}\omega_{b/i}^b = \mathbf{m}^b$ [4-15]

Since $\frac{d\omega_{b/i}^b}{dt_b}$ is the rate of change of the angular velocity, p,q,r are the instantaneous projections of $\omega_{b/i}^b$ onto the $\mathbf{i}^b,\mathbf{j}^b,\mathbf{k}^b$ axes, following that

$$\dot{\omega}_{b/i}^b = \frac{d\omega_{b/i}^b}{dt_b} = \begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{pmatrix}$$

And by rearranging terms: $\dot{\omega}_{b/i}^b = \mathbf{J}^{-1}\left[-\omega_{b/i}^b \times \mathbf{J}\omega_{b/i}^b + \mathbf{m}^b\right]$ [4-16]

Because aircraft symmetry about the plane spanned by $\mathbf{i}^b$ and $\mathbf{k}^b$, it is safe to assume that $J_{xy} = J_{yz} = 0$ resulting in $\mathbf{J} = \begin{pmatrix} J_x & 0 & -J_{xz} \\ 0 & J_y & 0 \\ -J_{xz} & 0 & J_z \end{pmatrix}$, and following the assumption, the inverse of $\mathbf{J}$ is given by $\mathbf{J} = \begin{pmatrix} \frac{J_x}{\Gamma} & 0 & \frac{J_{xz}}{\Gamma} \\ 0 & J_y & 0 \\ \frac{J_{xz}}{\Gamma} & 0 & \frac{J_z}{\Gamma} \end{pmatrix}$ where $\Gamma = J_x J_z - J_{xz}^2$.

Defining the components of the externally applied moment about the $\mathbf{i}^b, \mathbf{j}^b, \mathbf{k}^b$ axes as $\mathbf{m}^b = (l, m, n)^{\mathrm{T}}$, equation 4-16 can be written in component form as

$$\begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{pmatrix} = \begin{pmatrix} \frac{J_z}{\Gamma} & 0 & \frac{J_{xz}}{\Gamma} \\ 0 & \frac{1}{J_y} & 0 \\ \frac{J_{xz}}{\Gamma} & 0 & \frac{J_x}{\Gamma} \end{pmatrix} \left[\begin{pmatrix} 0 & r & -q \\ -r & 0 & p \\ q & -p & 0 \end{pmatrix} \begin{pmatrix} J_x & 0 & -J_{xz} \\ 0 & J_y & 0 \\ -J_{xz} & 0 & J_z \end{pmatrix} \begin{pmatrix} p \\ q \\ r \end{pmatrix} + \begin{pmatrix} l \\ m \\ n \end{pmatrix} \right]$$

$$= \begin{pmatrix} \Gamma_1 pq - \Gamma_2 qr \\ \Gamma_5 pr - \Gamma_6 (p^2 - r^2) \\ \Gamma_7 pq - \Gamma_1 qr \end{pmatrix} + \begin{pmatrix} \Gamma_3 l - \Gamma_4 n \\ \frac{1}{J_y} m \\ \Gamma_4 l - \Gamma_8 n \end{pmatrix} \qquad [4\text{-}17]$$

,where Γ_1 through Γ_8 are in the table below.

$\Gamma_1 = \frac{J_{xz}(J_x - J_y + J_z)}{\Gamma}$	$\Gamma_2 = \frac{J_z(J_z - J_y) + J_{xz}^2}{\Gamma}$	$\Gamma_3 = \frac{J_z}{\Gamma}$	$\Gamma_4 = \frac{J_{xz}}{\Gamma}$
$\Gamma_5 = \frac{J_z - J_x}{\Gamma}$	$\Gamma_6 = \frac{J_{xz}}{J_y}$	$\Gamma_7 = \frac{(J_x - J_y)J_x + J_{xz}^2}{\Gamma}$	$\Gamma_8 = \frac{J_x}{\Gamma}$

Table 4.2. Γ substitution table for J

Table 4-3 below, summarizes the kinematics and dynamics for the 6 Degree of Freedom, 12-state model for a UAV

$$\begin{pmatrix} \dot{p}_n \\ \dot{p}_e \\ \dot{p}_d \end{pmatrix} = \begin{pmatrix} \cos\theta\cos\psi & \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi \\ \cos\theta\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{pmatrix} \begin{pmatrix} u \\ v \\ w \end{pmatrix} \quad [4.18]$$

$$\begin{pmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{pmatrix} = \begin{pmatrix} ru - qw \\ pv - ru \\ qw - pv \end{pmatrix} + \frac{1}{m} \begin{pmatrix} f_x \\ f_y \\ f_z \end{pmatrix} \quad [4.19] \qquad \begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \frac{\sin\phi}{\cos\theta} & \frac{\cos\phi}{\cos\theta} \end{pmatrix} \begin{pmatrix} p \\ q \\ r \end{pmatrix} \quad [4.20]$$

$$\begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{pmatrix} = \begin{pmatrix} \Gamma_1 pq - \Gamma_2 qr \\ \Gamma_5 pr - \Gamma_6 (p^2 - r^2) \\ \Gamma_7 pq - \Gamma_1 qr \end{pmatrix} + \begin{pmatrix} \Gamma_3 l - \Gamma_4 n \\ \frac{1}{J_y} m \\ \Gamma_4 l - \Gamma_8 n \end{pmatrix} \quad [4.21]$$

Table 4.3. Kinematics and dynamics for the 6 Degree of Freedom, 12-state model

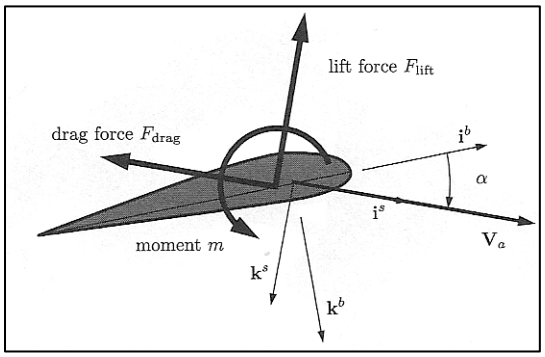
4.4 Forces and Moments

The objective of this section is to describe the forces and moments that act on a UAV. Following [27], the assumption that the forces and moments are primarily due to three sources: gravity, aerodynamics, and propulsion are made. Letting $\mathbf{f}_g$ be the force due to gravity, $(\mathbf{f}_a, \mathbf{m}_a)$ be the forces and moments due to aerodynamics, and $(\mathbf{f}_p, \mathbf{m}_p)$ be the forces and moments due to propulsion, $\mathbf{f} = \mathbf{f}_g + \mathbf{f}_a + \mathbf{f}_m + \mathbf{f}_p$ and $\mathbf{m} = \mathbf{m}_a + \mathbf{m}_g + \mathbf{m}_p$, where $\mathbf{f}$ is the total force action on the airframe and $\mathbf{m}$ is the total moment action on the airframe.

The effect of the Earth's gravitational field on a UAV can be modeled as a force proportional to the mass action on the center of gravity. This force acts in the $\mathbf{k}^i$ direction and is proportional to the mass of the UAV by the gravitational constant g in the vehicle frame $\mathcal{F}^v$; the gravity force action on the center of the mass is given by $\mathbf{f}_g^v = \begin{pmatrix} 0 \\ 0 \\ \mathrm{m}g \end{pmatrix}$.

When applying Newton's second law discussed earlier, forces along the axes in the body frame are summed together. Therefore, a transformation from the gravitational force into body-frame components gives $\mathbf{f}_g^b = \mathcal{R}_v^b \begin{pmatrix} 0 \\ 0 \\ \mathrm{m}g \end{pmatrix} = \begin{pmatrix} -\mathrm{m}g\sin\theta \\ \mathrm{m}g\cos\theta\sin\phi \\ \mathrm{m}g\cos\theta\cos\phi \end{pmatrix}$. Since the gravity force acts through the center of mass of the UAV, no moments are produced by gravity.

As a UAV passes through the air, a pressure distribution is generated around the UAV body. The strength and distribution of the pressure action on the UAV is a function of the airspeed, the air density, and the shape and attitude of the UAV. Accordingly, the dynamic pressure is given by $\frac{1}{2}\rho V_a^2$, where ρ is the air density, and V_a^2 is the speed of the UAV through the surrounding air mass. Here, the effect of the pressure with a combination of forces and a moment is characterized. As shown in Figure 4.9 the lift, drag and moment are commonly expressed as



$$F_{lift} = \frac{1}{2}\rho V_a^2 S C_L(\alpha, q, \delta_e$$

$$F_{drag} = \frac{1}{2}\rho V_a^2 S C_D(\alpha, q, \delta_e) \qquad [4\text{-}22]$$

$$m = \frac{1}{2}\rho V_a^2 S c C_m(\alpha, q, \delta_e)$$

Figure 4.9 Pressure distribution modeling

Where the C_L, C_D, C_m are nondimensional aerodynamic coefficients, S is the platform area of the UAV wing and c is the mean chord of the UAV wing. The effects of the α angle and β angle are considered; the angular rates p,q,r and the deflection of control surfaces on the aerodynamic coefficients. It is common to decompose the aerodynamic forces and moments into two groups: longitudinal and lateral. The longitudinal forces and moments act in the $\mathbf{i}^b$-$\mathbf{k}^b$ plane, also called the pitch plane. They include forces in the $\mathbf{j}^b$ and $\mathbf{k}^b$ directions (caused by lift and drag) and the moment about the $\mathbf{j}^b$ axis. The lateral forces and moments include the force in the $\mathbf{j}^b$ direction and the moments about the $\mathbf{i}^b$ and $\mathbf{k}^b$ axes.

In Chapter 3, the control surfaces for an aircraft were briefly explained. This section further defines the control surfaces of the aircraft. The Apprentice aircraft follows the 'standard' fixed-wing aircraft configuration, where the control surfaces are the aileron, elevator and rudder. In Figure 4.10 aileron deflection is denoted by δ_a, elevator deflection by δ_e, and rudder deflection by δ_r. The positive direction of a control surface

deflection can be determined by applying the right-hand rule to the hinge axis of the control surface. Since the aileron is a composite deflection, it is defined by $\delta_a = \frac{1}{2}\left(\delta_{a-\text{left}} - \delta_{a-\text{right}}\right)$.

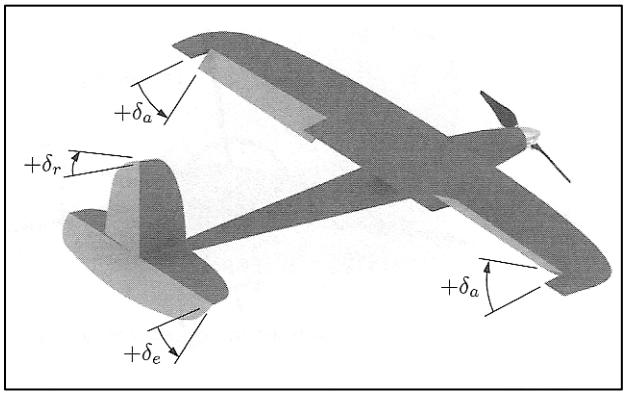


Figure 4.10 Standard configuration aircraft surfaces

4.4.1 Longitudinal Aerodynamics

Longitudinal aerodynamic forces and moments, also known as the pitch plane cause motion in the $\mathbf{i}^b$-$\mathbf{i}^k$ plane. Where the aerodynamic forces and moment, which are most familiar, are lift, drag, and pitching moment. Following the definition, the lift and drag forces are aligned with the axes of the stability frame, as show in Figure 4.10. Then when represented as a vector, the pitching moment also aligns with the $\mathbf{j}^s$ axis of the stability frame. Thus the lift, drag forces, and pitching moment are all influenced by the angle of attack, and additionally, the pitch rate q and the elevator deflection δ_e also influences the longitudinal forces and moment. Based on this, the equation for lift, drag

and pitching moment to express this function dependence on α, q, δ_e can be written as

$$F_{lift} = \frac{1}{2}\rho V_a^2 S C_L(\alpha, q, \delta_e)$$

$$F_{drag} = \frac{1}{2}\rho V_a^2 S C_D(\alpha, q, \delta_e)$$

$$m = \frac{1}{2}\rho V_a^2 S c C_m(\alpha, q, \delta_e)$$

Applying a Taylor series approximation on the lift force can be written as

$$F_{lift} = \frac{1}{2}\rho V_a^2 S\left[C_{L0} + \frac{\partial C_L}{\partial \alpha}\alpha + \frac{\partial C_L}{\partial q}q + \frac{\partial C_L}{\partial \delta_e}\delta_e\right] \quad [4\text{-}23]$$

and then adjusted to $F_{lift} = \frac{1}{2}\rho V_a^2 S\left[C_{L0} + C_{L\alpha}\alpha + C_{Lq}\frac{c}{2V_a}q + C_{L\delta e}\delta_e\right]$ [4-24]

since the units of q are in radians per second. The Taylor series approximation for F_{drag} and m are below

$$F_{drag} = \frac{1}{2}\rho V_a^2 S\left[C_{D0} + C_{D\alpha}\alpha + C_{Dq}\frac{c}{2V_a}q + C_{D\delta e}\delta_e\right] \quad [4\text{-}25]$$

$$m = \frac{1}{2}\rho V_a^2 S c\left[C_{m0} + C_{m\alpha}\alpha + C_{mq}\frac{c}{2V_a}q + C_{m\delta e}\delta_e\right] \quad [4\text{-}26]$$

4.4.2 Lateral Aerodynamics

The lateral aerodynamic force and moments cause translational motion in the lateral direction along the UAV axis as well as rotational motions in roll and yaw that result in directional changes in the flight path of the UAV. The lateral aerodynamics are most significantly influenced by the sideslip angle β. They are also influenced by the roll rate p, the yaw rate r, the deflection of the aileron δ_a, and the deflection of the rudder

δ_r. Denoting the lateral force as f_y and the roll and yaw moments as *l* and *n* respectively

$$f_y = \frac{1}{2}\rho V_a^2 S C_Y(\beta, p, r, \delta_a, \delta_r)$$
$$l = \frac{1}{2}\rho V_a^2 S b C_l(\beta, p, r, \delta_a, \delta_r)$$
$$n = \frac{1}{2}\rho V_a^2 S b C_n(\beta, p, r, \delta_a, \delta_r)$$

Where C_Y, C_l and C_n are nondimensional aerodynamic coefficients and b is the wingspan of the aircraft. Similar to the longitudinal aerodynamic forces and moments, the coefficients C_Y, C_l, and C_n are non-linear in their constitutive parameters $\beta, p, r, \delta_a, \delta_r$. These nonlinear relationships however are difficult to characterize. Further, linear aerodynamic models yield acceptable accuracy in most applications and provide valuable insights into the dynamic stability of the aircraft. Following the approach used in the previous section to produce the linear longitudinal aerodynamic models: first-order Taylor series approximation and nondimensionalization of the aerodynamic coefficients. Using this approach, linear relationships for lateral force, roll moment, and yaw moment are given by

$$f_y = \frac{1}{2}\rho V_a^2 S\left[C_{Y0} + C_{Y\beta}\beta + C_{Yp}\frac{b}{2V_a}p + C_{Yr}\frac{b}{2V_a}r + C_{Y\delta a}\delta_a + C_{Y\delta r}\delta_r\right] \quad [4\text{-}27]$$

$$l = \frac{1}{2}\rho V_a^2 S\left[C_{l0} + C_{l\beta}\beta + C_{lp}\frac{b}{2V_a}p + C_{lr}\frac{b}{2V_a}r + C_{l\delta a}\delta_a + C_{l\delta r}\delta_r\right] \quad [4\text{-}28]$$

$$n = \frac{1}{2}\rho V_a^2 Sc\left[C_{n0} + C_{n\beta}\beta + C_{np}\frac{b}{2V_a}p + C_{nr}\frac{b}{2V_a}r + C_{n\delta a}\delta_a + C_{n\delta r}\delta_r\right] \quad [4\text{-}29]$$

These forces and moments are aligned with the body axes of the aircraft and do not

require a rotational transformation to be implemented in the equations of motion. The coefficient C_{Y0} is the value of the lateral force coefficient C_Y when $\beta = p = r = \delta_a = \delta_r = 0$. For aircraft that are symmetrical about the $\mathbf{i}^b$-$\mathbf{k}^b$ plane, C_{Y0} is typically zero. The coefficients C_{l0} and C_{n0} are defined similarly and are also typically zero for symmetric aircraft.

Aerodynamic coefficients $C_{m\alpha}, C_{l\beta}, C_{n\beta}, C_{mq}, C_{lp}$ and C_{nr} are referred to as stability derivatives because their values determine the static and dynamic stability of the UAV. Static stability deals with the direction of aerodynamic moments as the UAV is perturbed away from its nominal flight condition. If the moments tend to restore the UAV to its nominal flight condition, the UAV is said to be statically stable. Most aircraft are designed to be statically stable. The coefficients of $C_{m\alpha}, C_{l\beta}$, and C_{nr} determine the static stability of the UAV. They represent the change in the movement coefficients with respect to changes in the direction of the relative airspeed, as represented by α and β. $C_{m\alpha}$ is referred to as the longitudinal static stability derivative. For the UAV to be statically stable, $C_{m\alpha}$ must be less than zero.

$C_{l\beta}$ is called the roll static stability derivative and is typically associated with dihedral in the wings. For static stability in roll $C_{l\beta}$ must be negative as it results in rolling moments that roll the UAV away from the direction of sideslip, thereby driving the sideslip angle β to zero.

$C_{n\beta}$ is referred to as the yaw static stability derivative, where if the aircraft is statically stable in yaw, it naturally points into the wind. The value of $C_{n\beta}$ is heavily influenced by the design of the tail of the aircraft. The larger the tail and the further the tail is aft of the center of gravity, the larger $C_{n\beta}$ is. For the UAV to be stable in yaw, $C_{n\beta}$ must be positive, implying for a positive sideslip angle, a positive yawing moment is introduced. This yawing moment physically yaws the UAV into the direction of the relative airspeed, driving the sideslip angle to zero.

Dynamic stability deals with the dynamic behavior of the airframe in response to disturbances. If a disturbance is applied to the UAV, the UAV is said to be dynamically stable if the response of the UAV damps out over time. C_{mq} is referred to as the pitch damping derivative, C_{lp} is called the roll damping derivative, and C_{nr} is the yaw damping derivative. Each of these damping derivatives is usually negative, meaning that a moment is produced that opposes the direction of motion, thus damping the motion. The aerodynamic coefficients $C_{m\delta e}, C_{l\delta_a}$, and $C_{n\delta r}$ are associated with the deflection of control surfaces and are referred to as the primary control derivatives. They are primary because the moments produced are the intended result of the specific control surface deflection.

For more detailed information please refer to [26], as this text is based on the source.

Chapter 5 – The Derived UAV Model

5.1 System Identification

Aircraft system identification, as Figure 5.1 demonstrates, is a rigorous, multi-step process that analyzes the data gathered during a flight and produces a mathematical description of the vehicle. One of the problems that is often observed in system identification is defining the input signal for the system. The challenge of system identification is to predict the parameters of the system and construct a model that can estimate the system behavior under both normal inputs and disturbance.

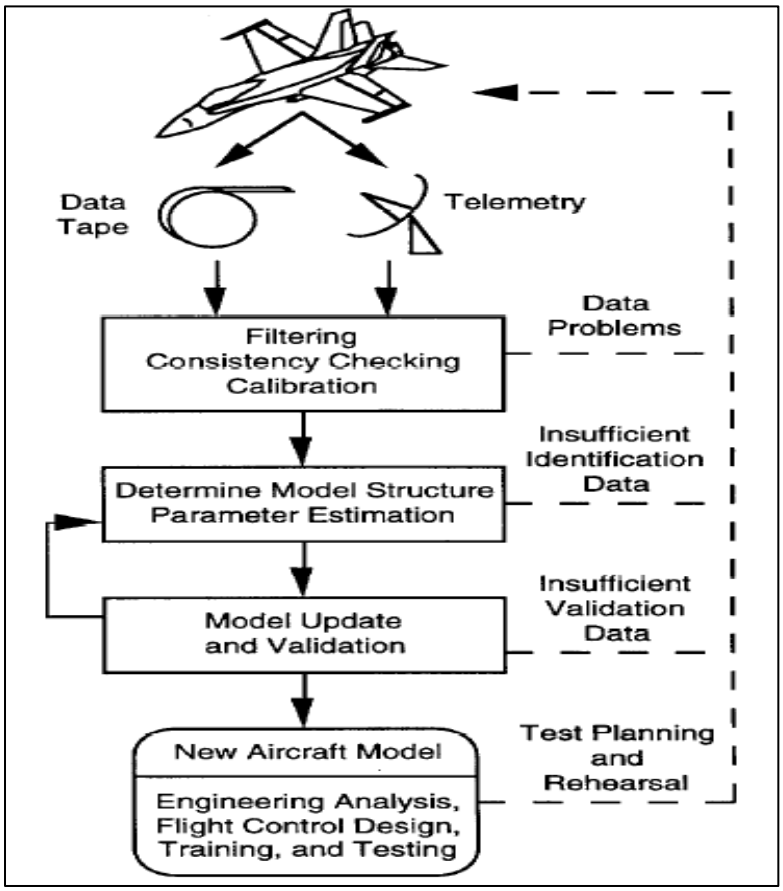


Figure 5.1 The Aircraft System Identification process. [28]

Two modeling approaches that are used for the fixed-wing aircrafts exist. The first is called physical modeling, where the dynamics and aerodynamics are used in order to model the vehicle dynamics from the ground up, and the second is by using system identification where experimental data is recorded during flight tests and requires computation tools (mathematical models). For this project, the second modeling approach is used.

5.2 Methodology

This chapter describes basic principles of mathematical modeling and the models available for system identification. In simple words, a mathematical model is a set of equations that describe the behavior of a system in general. These models are not anything more than differential equations, however the complexity comes from the way the dynamical models are developed. Due to phenomena that the designer is not aware of, a precise and perfect model of a physical system is be extremely difficult to achieve and thus model errors or model uncertainties on a system may occur. However, a model that describes a part of the reality is very useful for analysis and design. The method of mathematical modeling that is based on experimental data is called system identification and in this chapter those models are explained and reviewed.

5.3a Autoregressive-Moving Average (ARMA)

Autoregressive-Moving Average (ARMA) models are widely applicable in parameterization. It is an accurate and straightforward method for estimating transfer function coefficients. These models are flexible in use by allowing the order of an ARMA model to increase; any linear series model with described accuracy can be approximated. ARMA modeling is a technique that fits a discrete transfer function of specified order to data by using the least squares method. If the transfer function parameters of the system are determined, an equivalent continuous transfer function can be obtained.

All physical systems are nonlinear to an extent and a system is called nonlinear if the Input-Output (I/O) steady state relation is not linear. The use of nonlinear models is very important because of the huge operating range they cover. They are able to describe the system behavior in a much larger operating region than the linear models do. Due to the complexity of the dynamical behavior of the nonlinear model structures, the most common characteristic and most difficult task in nonlinear system identification that occurs at the same time is dimensionality. Linear and nonlinear identification do not differ much. Linear identification methods can be compared with the roots of a tree and the nonlinear methods are the part that ends at or starts from the roots [29]. The model structure and the establishment of the parameterization are the key parts of the nonlinear identification that makes it crucial and complicated at the same time.

5.3b Box-Jenkins

The Box-Jenkins (BJ) approach is used during the system identification process on this project. Box-Jenkins is an extension of the abovementioned ARMA, and is applicable to a wide variety of statistical modeling situations. It provides a convenient framework that allows an analyst to think about the data, and to find an appropriate statistical model that can be used to help answer relevant questions about the data. This modeling procedure involves a three-stage process of model selection, parameter estimation and model checking. When applying a BJ model, those three stages of model development should be followed. The first step is the identification of the form of the model that fits the given data. At the second stage, the estimation process, the model parameters are calculated by using the maximum likelihood method and finally, the

model is checked for possible inadequacies [30]. If the model is found to be inadequate, it is necessary to go back to the second step and try to identify a better model. Once the model has been selected, estimated and checked, the transfer function of the part that fits best is noted down in order to continue the process of the model design. Figure 5.2 shows an example of the Box-Jenkins model estimator in SIMULINK, and Figure 5.3 is the tracking of the estimator with an Input and Output.

The choice of the Box-Jenkins model estimator as a method of system identification stems from [32], where the methodology is used for a LQG controller. This thesis carries on the ideology of [31] and expands and implements it using classical PID control. Additionally, a the Box-Jenkins model estimator is used for a miniature helicopter at DU^{2}SL as a proof of concept vehicle, the work present here is more proof that the Box-Jenkins model estimator can be used for system identification for different types of vehicles.

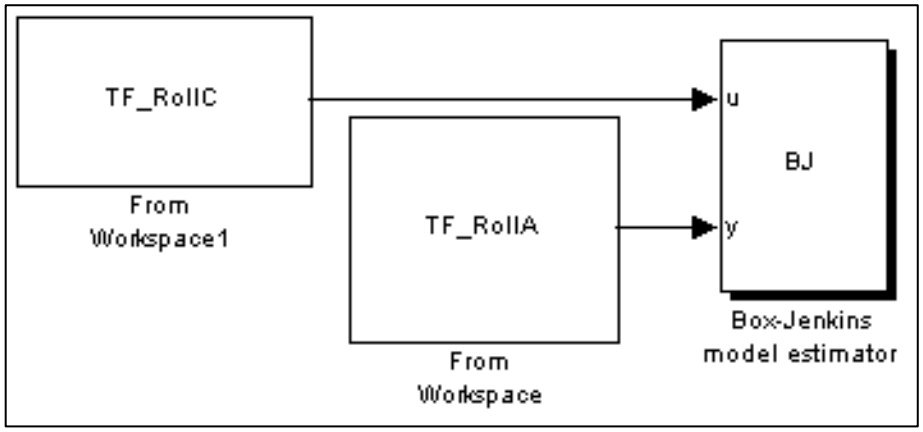


Figure 5.2: Box-Jenkins model estimator Simulink Blocks
u is the input (PWM control values), and y is the output (IMU angles)

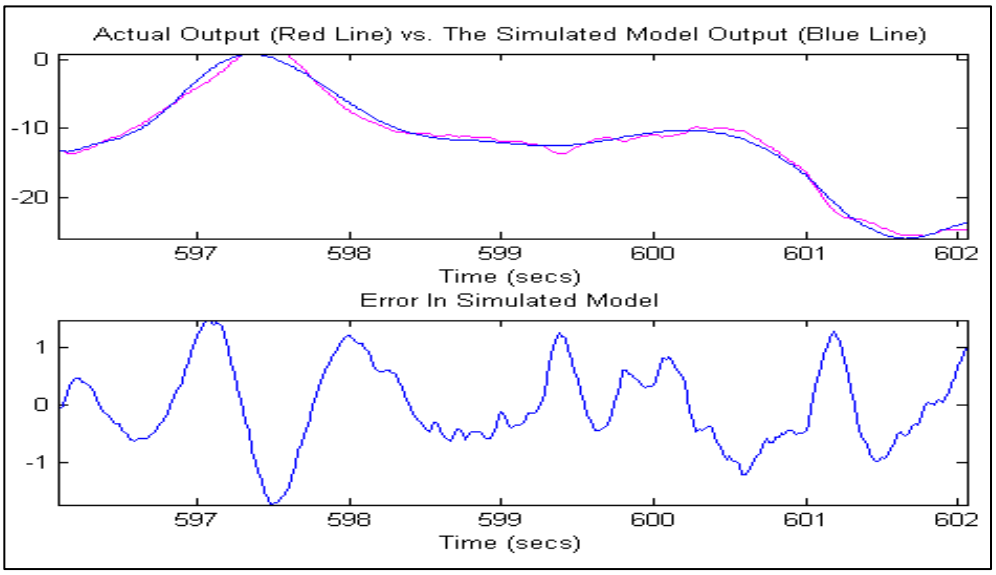


Figure 5.3: Box-Jenkins model estimator tracking the output

5.4 Data Procedure

The first step during the system identification process is to obtain the data through flight experiments. This process allows the determination of complete and accurate models. For this project, data from cruise (trim) flight is observed. Each set of data contains the aspects of the aircraft that are needed for the system identification and those are the following:

- A time constant (every 20 ms new data are obtained)
- Aileron and Roll
- Elevator and Pitch
- Rudder and Yaw
- Additional data collected is Throttle and Safety Switch position (if necessary)

After the data are obtained, the analysis continues where the flight data is separated into three parts for Yaw, Pitch and Roll. A set of inputs and outputs is created for each of the Euler angles. The inputs are the measurements of the PWM and the outputs are the

angular values recorded through the IMU of the movement of the aircraft. The data separation is necessary to identify the characteristics of the system, where a characteristic transfer function from each set should be identified. The system identification toolbox from SIMULINK, the Box-Jenkins model estimator, is used in to derive the transfer function that describes the system. This process eliminates the errors and the assumptions that need to be considered, which leads to a better and more accurate model.

Finally, the last step regarding the data analysis is to calculate the gains of the system. The controller that is utilized is the Discrete PI because it is a simple controller to implement and can reach the requirements that are needed for the system to be controlled; this is further explored in Chapter 6.

5.5 Data Analysis – Cruise Flight

As previously mentioned, the recorded data is separated into three parts: yaw, pitch and roll for an airplane in cruise (trim) flight. While processing the data in MATLAB, the Box-Jenkins estimator provides a good correlation of the data on the interval 428.8 – 430.8 s. No data was filtered through a high-pass filter, as it would introduce a phase shift that could not be accounted for in the programming, thus, the Euler angles are raw values obtained from the aircraft.

5.5a Aileron – Roll Data Analysis

The aileron pilot command stick deflection and observed roll angle is shown in Figure 5.4a. The pilot transmitter is highly sensitive and hence the jump in the

numeration of PWM from 1546 – 1550. As stated in Chapter 3, a servo at neutral position has a PWM of 1500, for the aileron, a trim adjustment was made to level the aircraft at 1547.

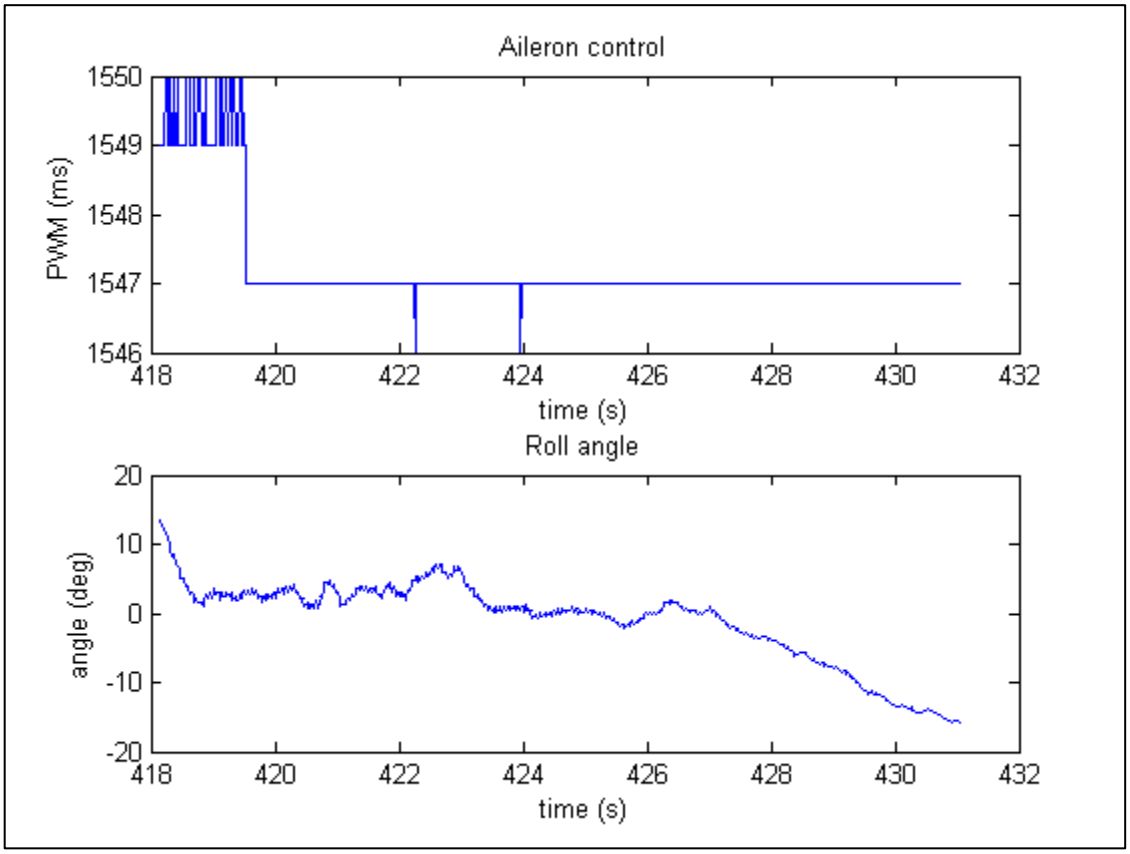


Figure 5.4a: Aileron pilot command from transmitter and roll angle from IMU

In the angle subplot, the aircraft experiences a natural deflection of approximately 14° on each side. This can be attributed to wind encountered throughout the flight envelope. The data from Figure 5.4a is processed in the Box-Jenkins Model Estimator (BJME) and its result is shown in Figure 5.4b. According to the data from the BJME, the aircraft experienced a deflection of approximately 8° in 2 seconds. The resulting transfer function for the BJME model at the interval 428.8 – 430.8 is shown in equation 5-1.

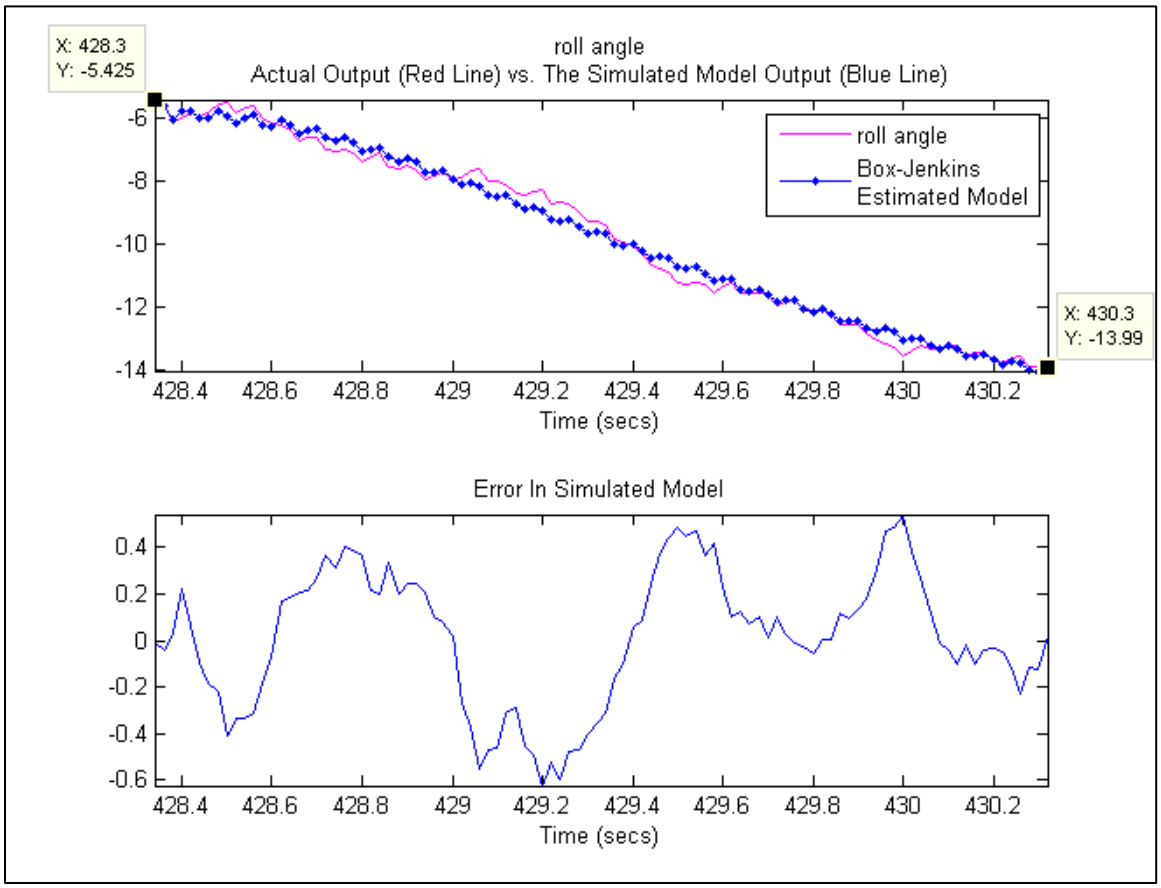


Figure 5.4b: Box-Jenkins aileron PWM to roll angle estimated model & noise model

$$\frac{-6.4548e^{-5}\ z^3 - 6.4548e^{-5}\ z^2 - 6.4548e^{-5}\ z}{z^3 - 0.88204\ z^2 - 0.31501\ z - 0.21397}$$

Equation 5-1: Roll Transfer Function

5.5b Elevator – Pitch Data Analysis

In this subsection the elevator pilot command and measured pitch angle is presented in Figure 5.5a. As previously mentioned, the pilot transmitter is highly sensitive, and the digital treading (or jitter) of the transmitter moved the signal between PWM measurements of 1560 and 1561 throughout the flight envelope. The adjusted trim from 1500 for the elevator is 1561; this is due to the position of the servo rod that

connects the servo to the armhole on the physical control surface. As shown in the angle subplot, a less than 10° pitch deflection was experienced throughout the flight envelope. It should be noted that because the airplane is flown under trim conditions, no adjustments are made to the control surface, therefore the wind 'carries' the aircraft, this leads to the variations in pitch experienced by the aircraft.

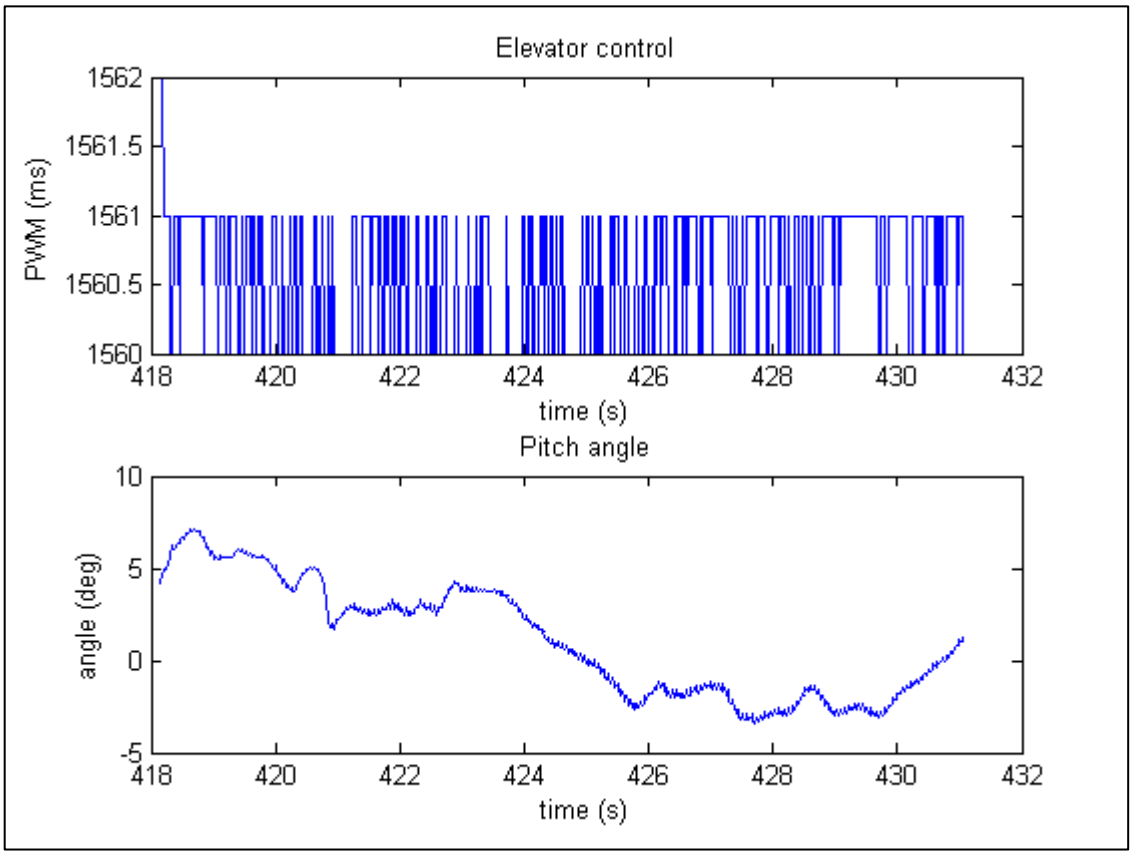


Figure 5.5a: Elevator pilot command from transmitter and pitch angle from IMU

The pitch data from the cruise flight is processed in the BJME and shown in Figure 5.5b. According to the data from the BJME, the aircraft experienced a deflection of approximately 1° across the 2 second measurement window. The transfer function from the BJME for elevator-roll is found in equation 5-2.

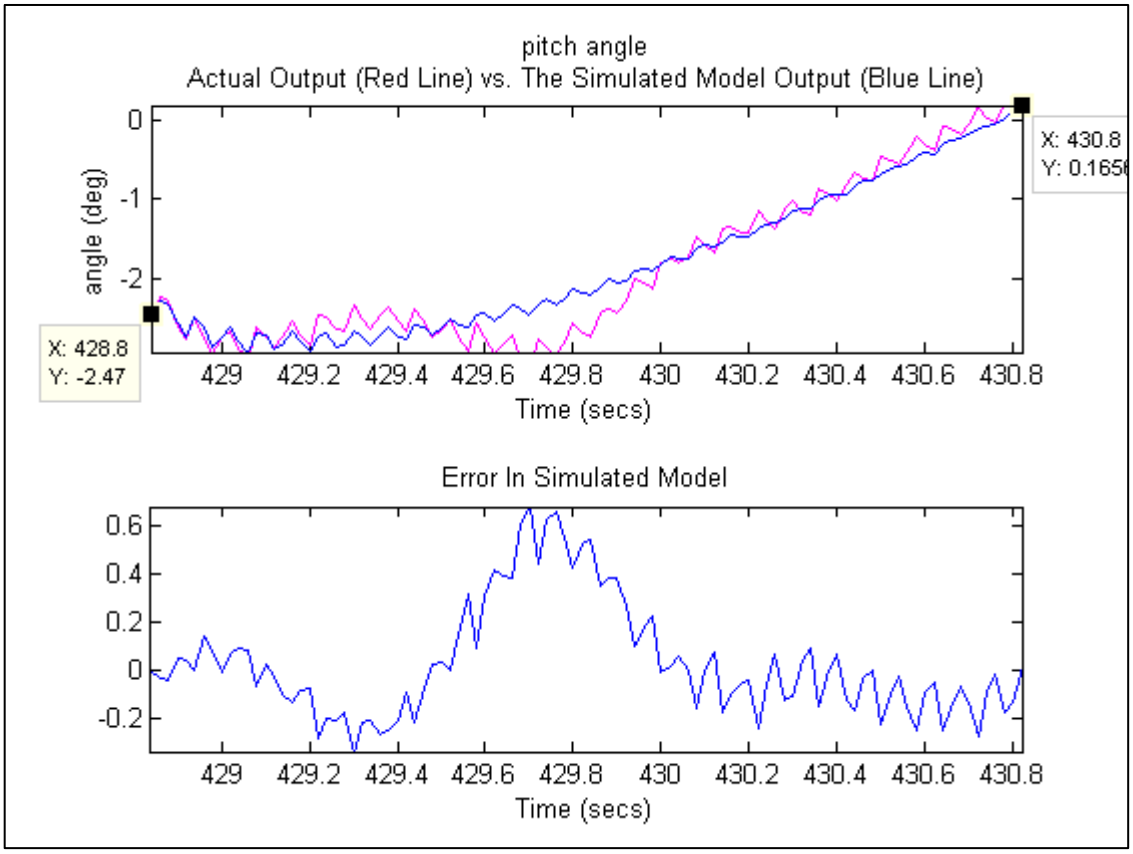


Figure 5.5b: Box-Jenkins elevator PWM to pitch angle estimated model & noise model

$$\frac{0.033799\,z^3 - 0.034608\,z^2 + 0.00083221\,z}{z^3 - 0.91735\,z^2 - 0.6149\,z - 0.53287}$$

Equation 5-2: Pitch Transfer Function

5.5c Rudder – Yaw Data Analysis

Finally, the rudder command and yaw angle are shown in Figure 5.6a. The rudder PWM had very little signal deviation during the flight envelope. The adjusted trim from the neutral position for the rudder is 1486. Unlike the previous two angles, the rudder experiences the most deflection against wind. Because of the lightweight characteristic of the UAV, it is susceptible to curving into the wind, hence the 85° rotational movement.

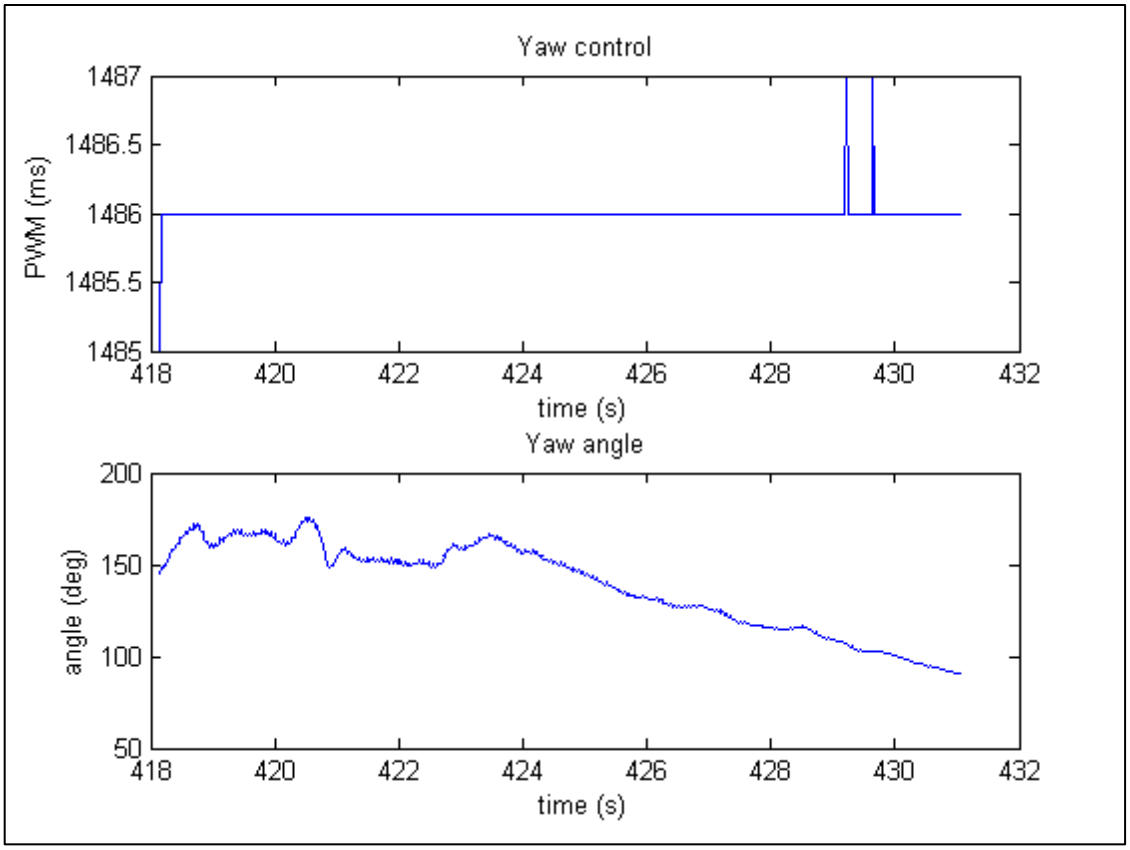


Figure 5.6a: Rudder pilot command from transmitter and yaw angle from IMU

The best estimated model seen by the Box-Jenkins is shown in Figure 5.5b. The yaw angle simulated output was the best tracked amongst the three angles. During the 428.8 – 430.8 interval, the aircraft was considerably stable with the exception of the 2.53° decline seen in a 0.4 second frame (429.4 – 428.8). The resulting transfer function for the yaw angle is shown in equation 5-3.

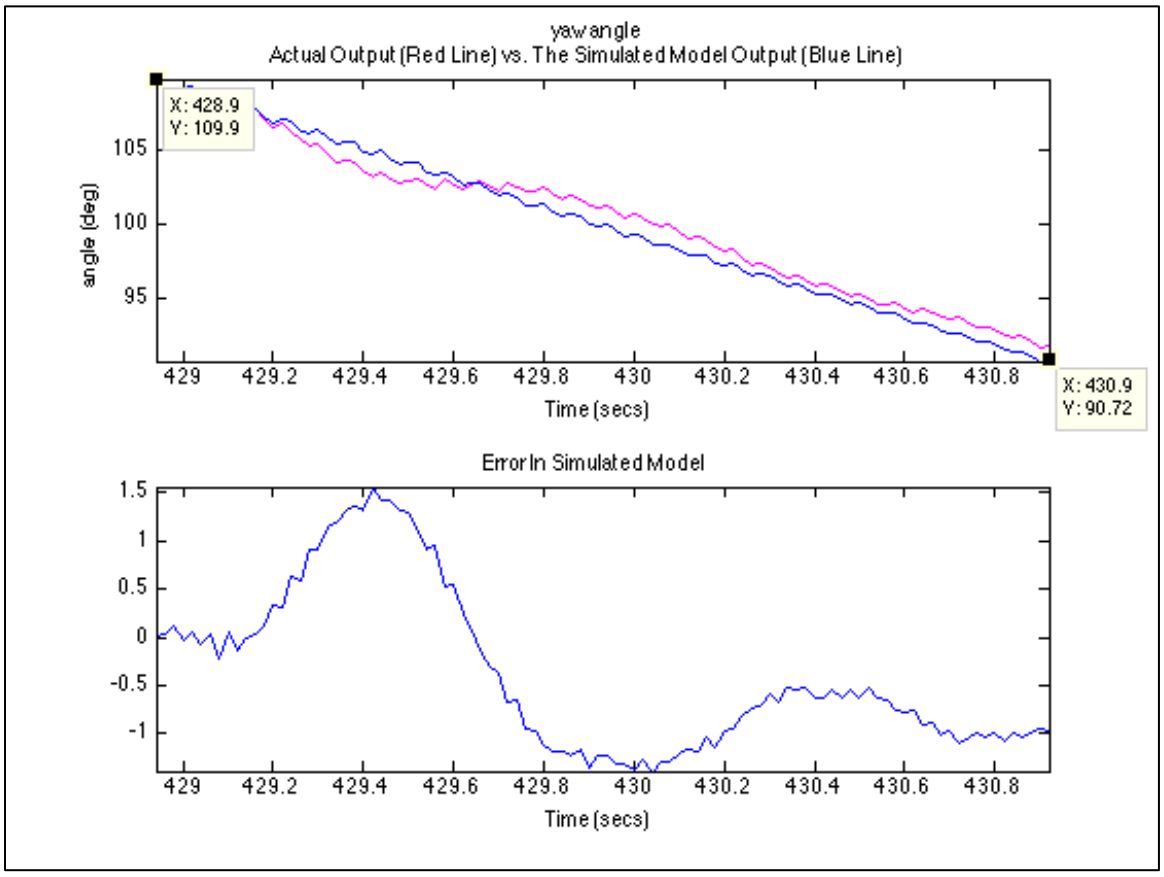


Figure 5.6b: Box-Jenkins rudder PWM to yaw angle estimated model & noise model

$$\frac{-0.025741\ z^3 + 0.036907\ z^2 - 0.012966\ z}{z^3 - 0.57012\ z^2 - 0.5938\ z - 0.95394}$$

Equation 5-3: Yaw Transfer Function

5.6 Transfer Function Summary

The following section is a summary of the transfer functions found for the system. These transfer functions are using in the next chapter to obtain the gains for the PID controller in order to fly the aircraft autonomously.

Parameter/Surface	Transfer Function
Roll/Aileron	$\frac{-6.4548e^{-5}\ z^3 - 6.4548e^{-5}\ z^2 - 6.4548e^{-5}\ z}{z^3 - 0.88204\ z^2 - 0.31501\ z - 0.21397}$
Pitch/Elevator	$\frac{0.033799\ z^3 - 0.034608\ z^2 + 0.00083221\ z}{z^3 - 0.91735\ z^2 - 0.6149\ z - 0.53287}$
Yaw/Rudder	$\frac{-0.025741\ z^3 + 0.036907\ z^2 - 0.012966\ z}{z^3 - 0.57012\ z^2 - 0.5938\ z - 0.95394}$

Table 5.1 summarized table of transfer functions obtained from the Box-Jenkins Model Estimator

Chapter 6 – Results

6.1 Calculated Results

The final steps in the system identification process are verifying that the transfer functions are stable and ensuring that they accurately represent the system. The Box-Jenkins Model Estimator outputs a transfer function in discrete-time. Henceforth, all procedures, tests, and verifications must be in discrete time.

The discrete-time PID controller block along with the discrete transfer function block is loaded into a SIMULINK model. Figure 6.1 is a consolidated image of the PID implementation. With this implementation, the transfer function for each Euler angle is implemented to its respective block. The simulation is run and the PID gains are obtained using the PID auto-tune functionality in SIMULINK; however, this does not always yield a desired result. The gains are manually tuned to achieve a desired settling and rise time.

In this chapter, a comparison is shown for non-optimal gains and optimal gains based on the flight data obtained for each Euler angle. It should be noted here that the actual parameters obtained are that of a PI controller, since the derivative term for each Euler angle is zero. Henceforth, PI controller is utilized, although the derivative term is shown, it its not incorporated into the final controller. Supplemental video documentation of the flights can be seen at the following URLs

Roll: http://youtu.be/KEPsIkfrX0g
Pitch: http://youtu.be/Q9v2ta-JhTk
Roll/Pitch: http://youtu.be/mxjHMtdGRzc http://youtu.be/rQSbfD-k9Fk
Roll/Yaw: http://youtu.be/14skYIiX2wQ http://youtu.be/14skYIiX2wQ
Roll/Pitch/Yaw: http://youtu.be/Ha8pzzw-5_g

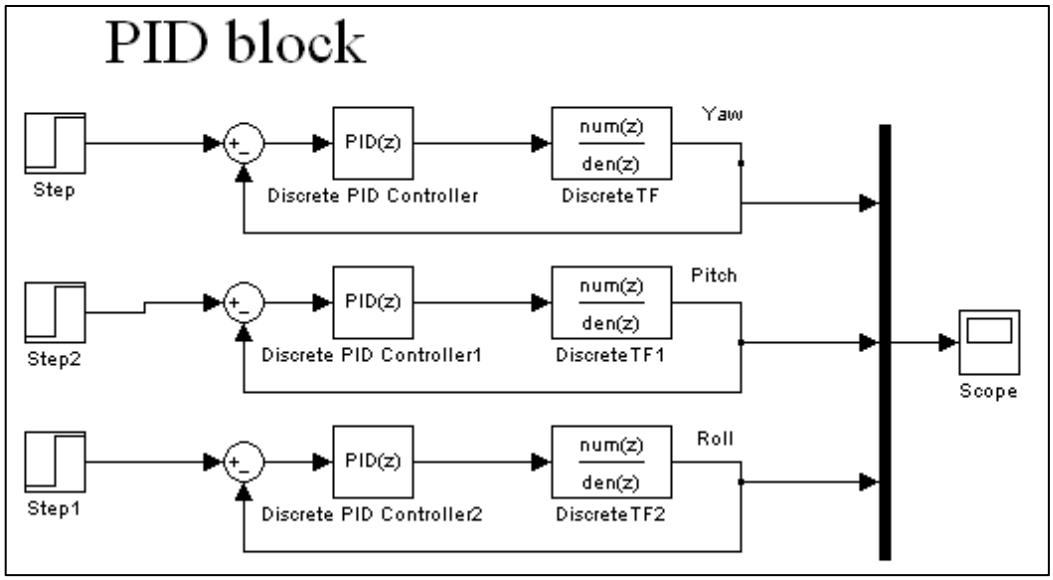


Figure 6.1 PID Block

6.1 Roll-Aileron

The first angle that is experimented on is the Roll angle, since correction is easily visible from the ground. As shown in Figure 6.2a the non-optimal solution seems to have an acceptable rise time and settling time, 3.02s and 4.62s respectively. After adjusting the flight parameters it is discovered that by reducing the gains the autopilot task would be able to calculate the correct error and adjust the roll angle from the setpoint as necessary. Figure 6.2b shows the tuned optimal solution that provides better flight results, albeit having a 180s settling time and a 100s rise time.

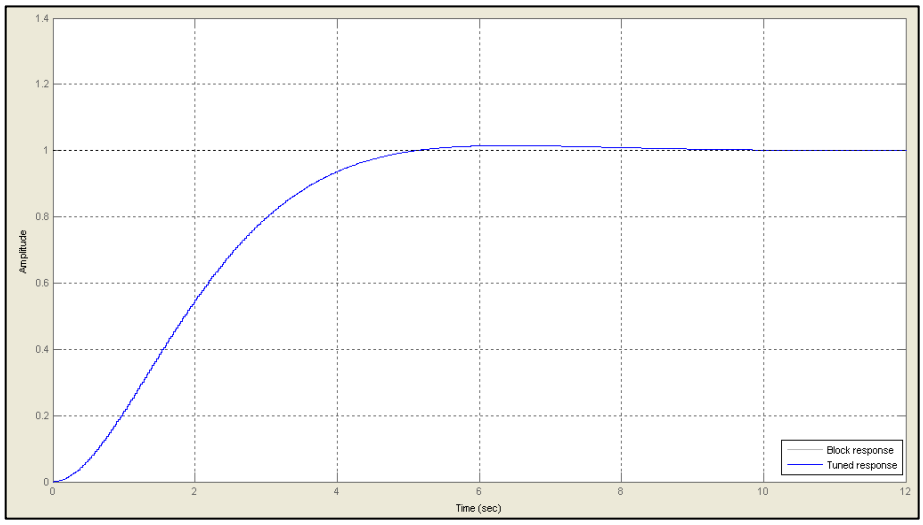


Figure 6.2a: PID Step Function for non-optimal roll transfer function

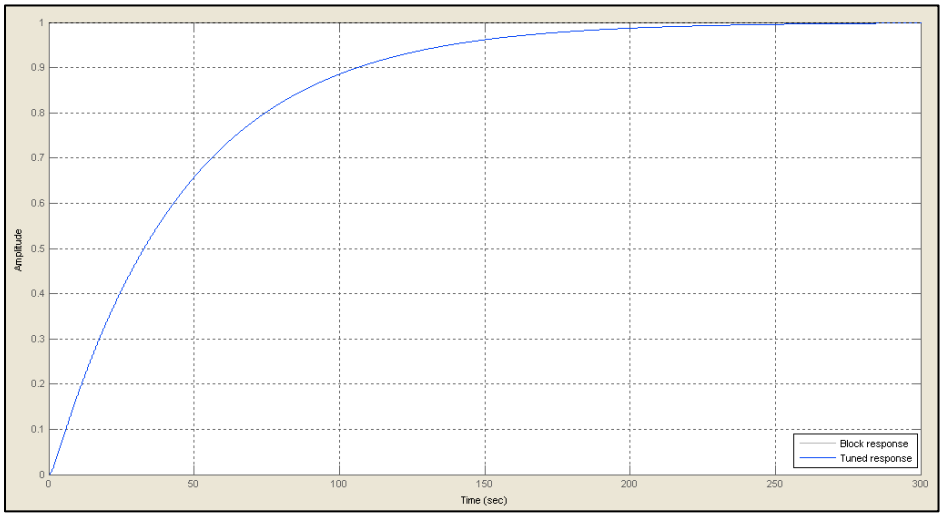


Figure 6.2b: PID Step Function for optimal roll transfer function

The obtained optimal PID gains for the transfer function in Equation 5.1 are shown in Table 6.1 along with the non-optimal for comparison.

Parameter	Non-optimal	Optimal
P	-0.49118	-0.018669
I	-49.1184	-1.8669
D	0	0
N	100	100
Rise Time (sec)	3.02	100
Settling Time (sec)	4.62	0180
% Overshoot	1.53%	0%
Peak	1.02	0.999

Table 6.1 Roll PI parameters

To test and verify that the transfer function is stable, the root locus of the controller and plant is taken and as Figure 6.3 shows, the transfer function is marginally stable for discrete time and is within the unit circle.

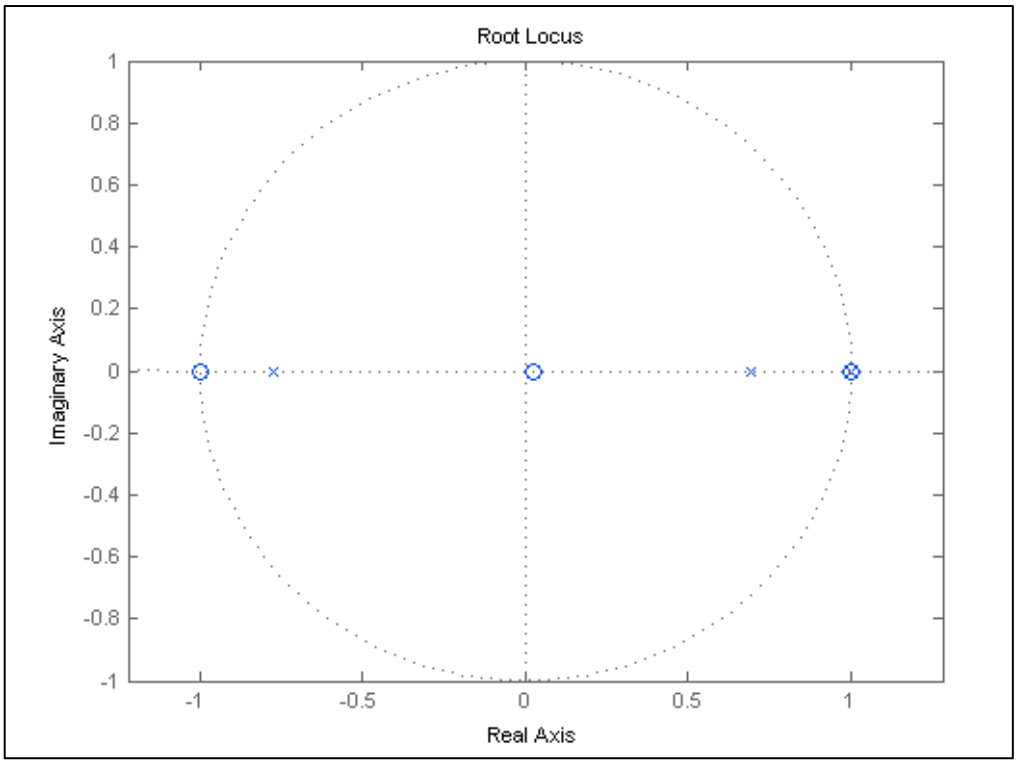


Figure 6.3: Root locus for roll PID

Wing rock is encountered during flight (visible in the video footage), due to the PI controller function calculating rapidly and the error growing and attempting to stabilize itself to the setpoint. The flight data from the five autonomous points as observed from the video are shown below

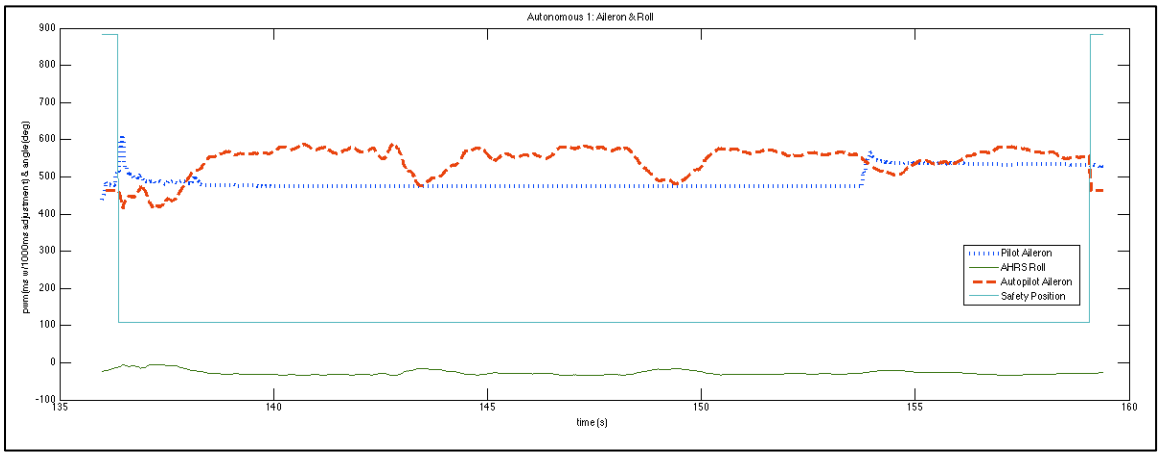


Figure 6.4a: Aileron/Roll Data for first autonomous flight

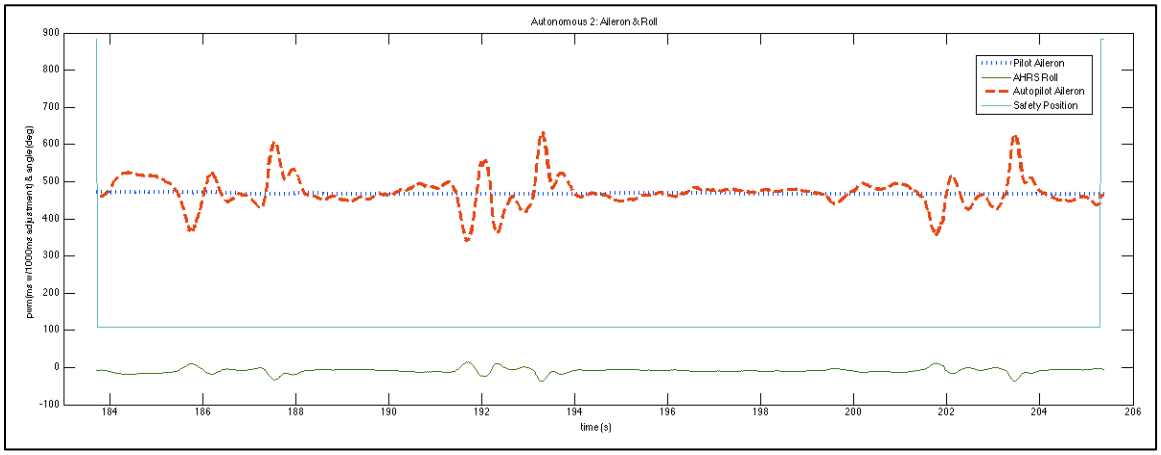


Figure 6.4b: Aileron/Roll Data for second autonomous flight

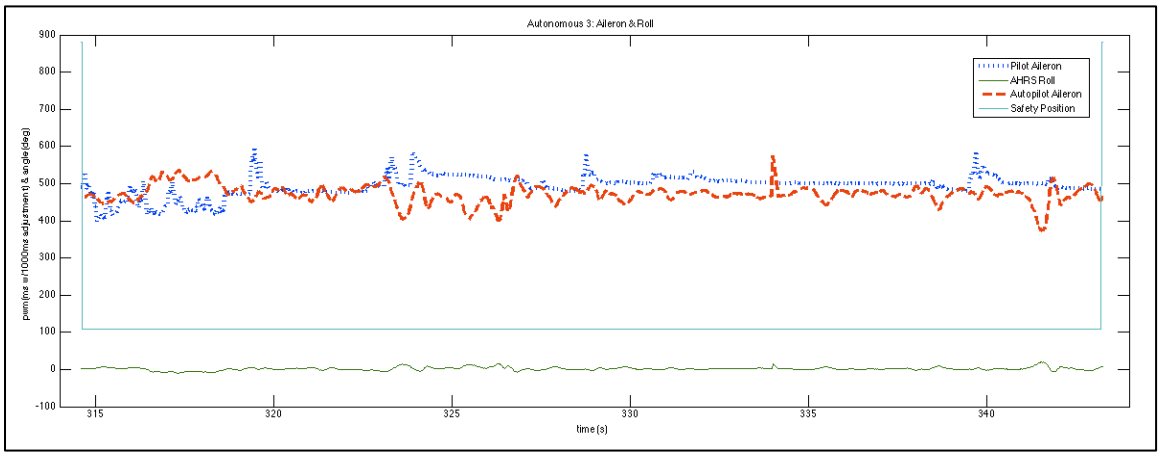


Figure 6.4c: Aileron/Roll Data for third autonomous flight

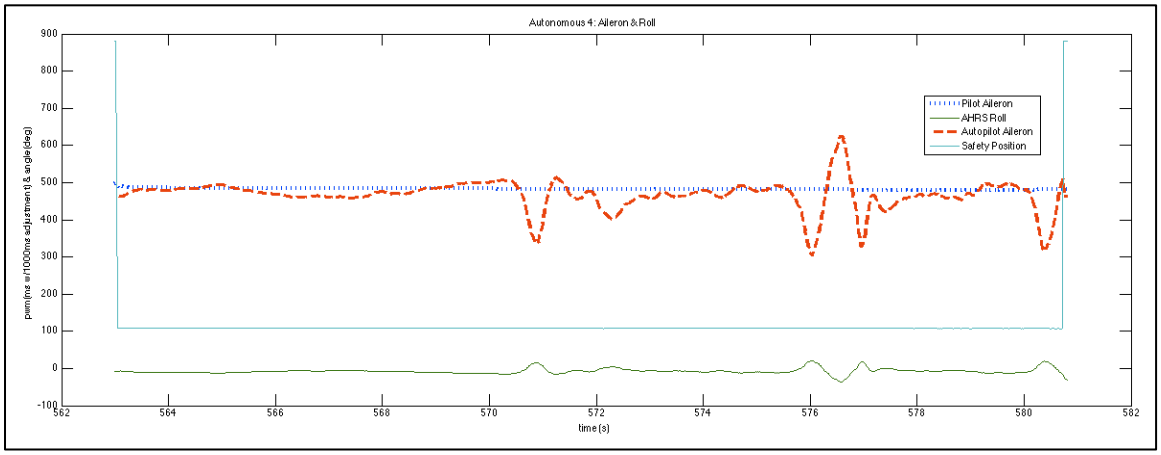


Figure 6.4d: Aileron/Roll Data for fourth autonomous flight

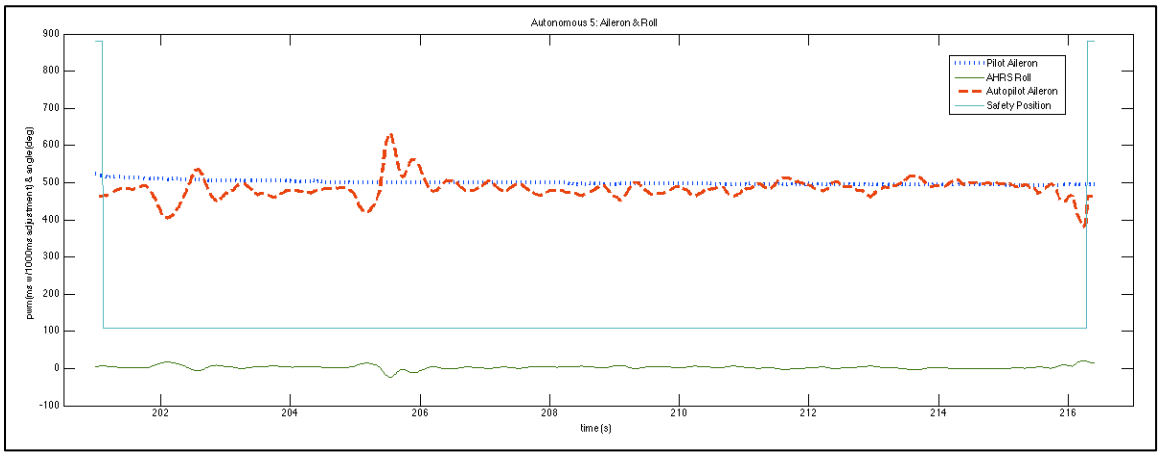


Figure 6.4e: Aileron/Roll Data for fifth autonomous flight

6.2 Pitch-Elevator

The pitch Euler angle that is controlled by the elevator is the second parameter tested. The step function for the non-optimal PID controller is shown in Figure 6.5a although the gains found appeared to yield good results, the flight proved differently, with Figure 6.5b showing the optimal trans for the transfer function from Equation 5.2, with a settling time of 87.9s and a rise time of 32.4s compared to the non-optimal 0.26s rise time and 0.44s settling time

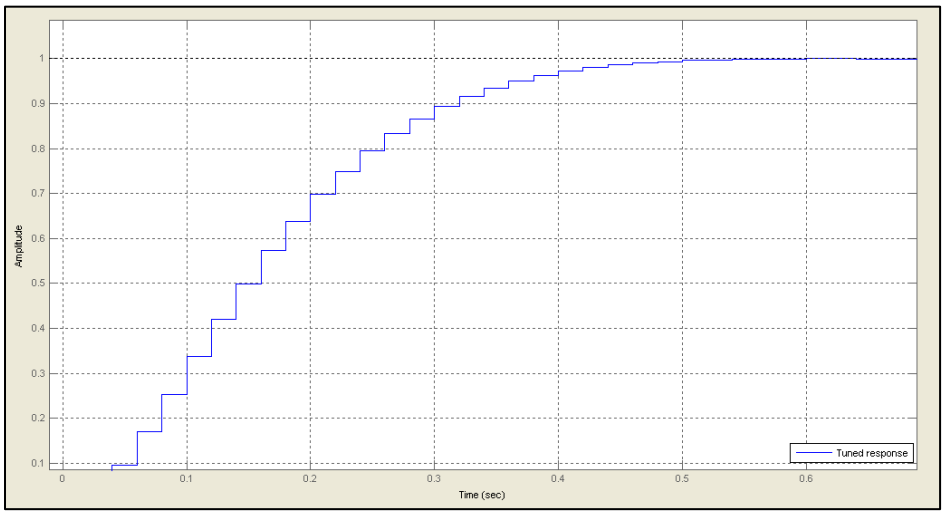


Figure 6.5a: Discrete PID Step Function for non-optimal pitch transfer function

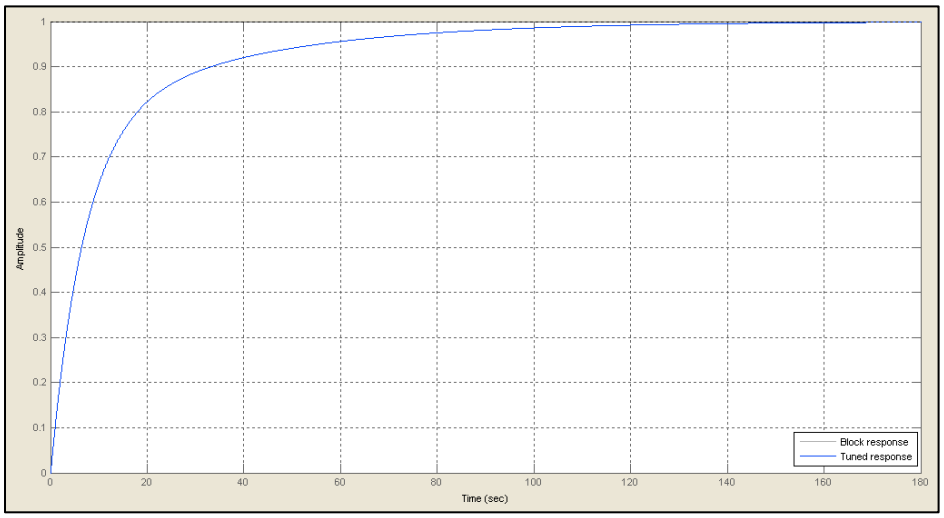


Figure 6.5b: Discrete PID Step Function for optimal pitch transfer function

Following tuning, the optimal PI gains are shown in Table 6.2 with the corresponding flight data in Figure 6.7a-e. The flight data shows the aircraft attempting to maintain the pitch within the 2 degrees of error allotted.

Parameter	**Non-optimal**	**Optimal**
P	0.99174	0.019389
I	99.1739	1.9389
D	0	0
N	100	100
Rise Time (sec)	0.26	32.4
Settling Time (sec)	0.44	87.9
% Overshoot	0%	0%
Peak	1	0.999

Table 6.2: Pitch PI parameters

To further test and verify the optimal transfer function is stable, the root locus of the controller and plant is taken and, as Figure 6.6 shows, marginally stable with a pole on the unit circle.

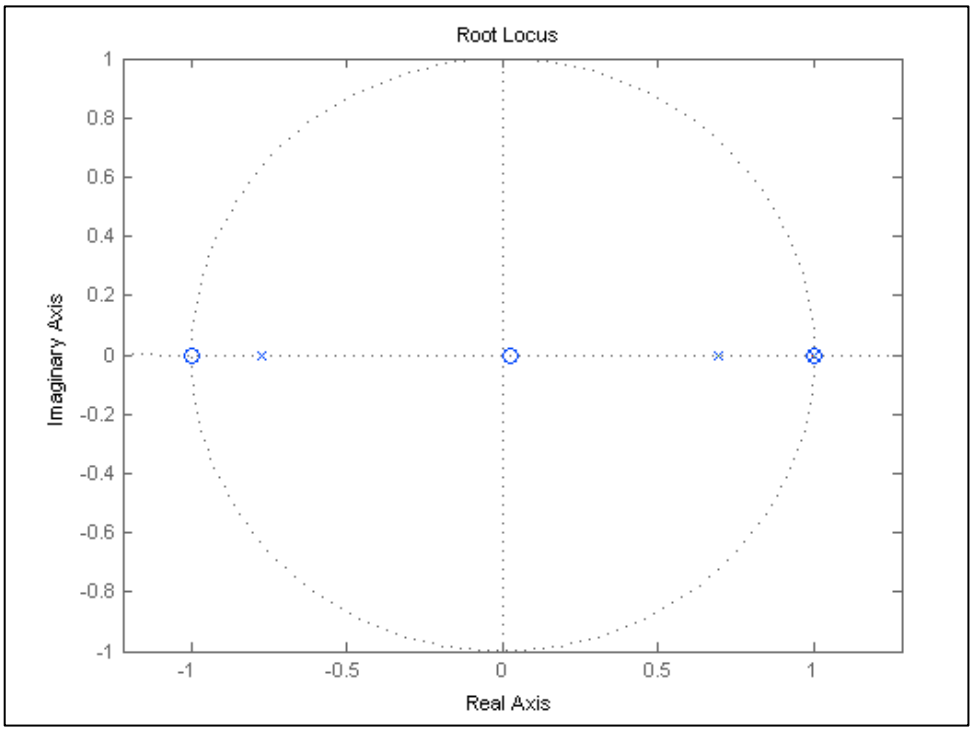


Figure 6.6: Root locus for pitch PI

Based on the optimal parameters, the flight data is shown below in Figure 6.7a through Figure 6.7e

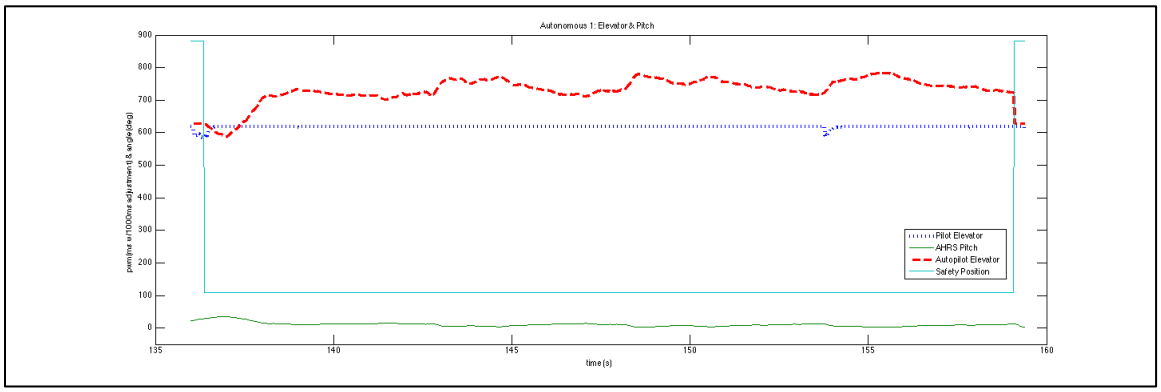


Figure 6.7a: Elevator/Pitch Data for first autonomous flight

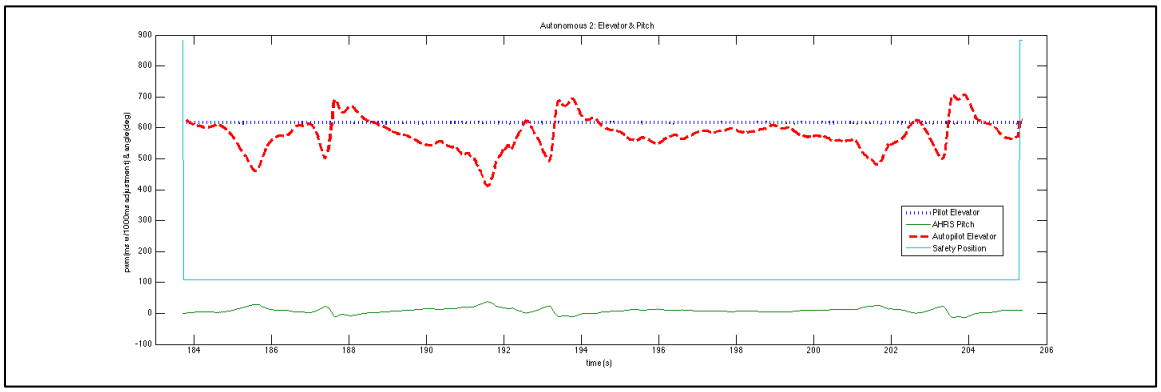


Figure 6.7b: Elevator/Pitch Data for second autonomous flight

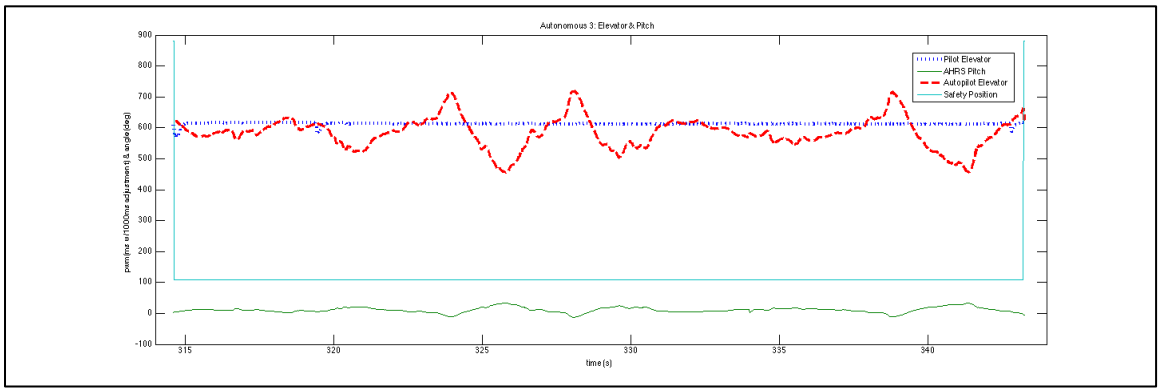


Figure 6.7c: Elevator/Pitch Data for third autonomous flight

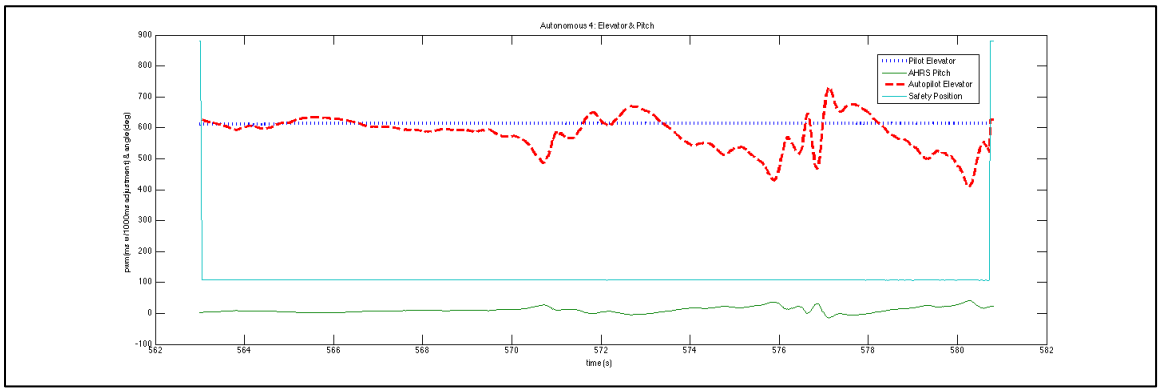


Figure 6.7d: Elevator/Pitch Data for fourth autonomous flight

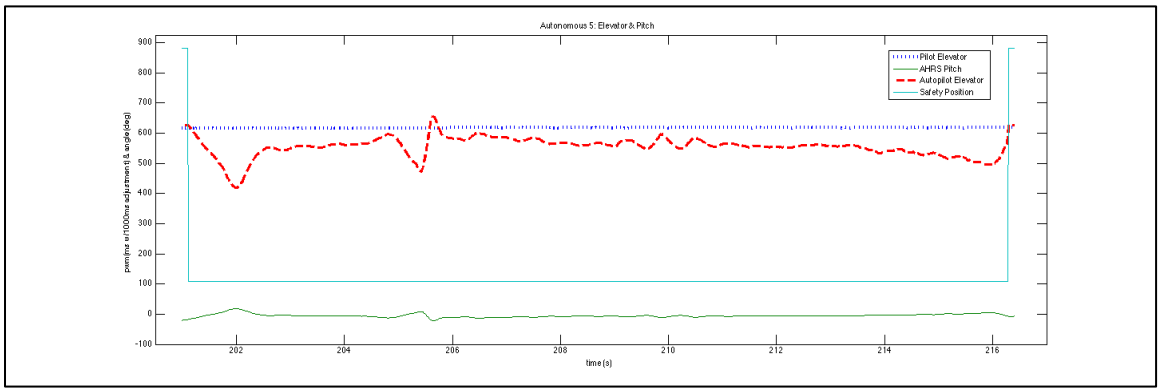


Figure 6.7e: Elevator/Pitch Data for fifth autonomous flight

6.3 Yaw-Rudder

The final parameter under Discrete PID testing is the yaw. For the auto-correction autopilot, the yaw serves as a directional heading hold, and due to the sideslip angle, mentioned in Chapter 4, a 10 degree window is incorporated into the yaw PI controller. The reason for the large window is to prevent rudder rock as experienced by the non-optimal gains with the rise time of 0.42s and a settling time of 0.72s. The non-optimal and optimal results are shown in Figure 6.8a and Figure 6.8b respectively with both gains in Table 6.3.

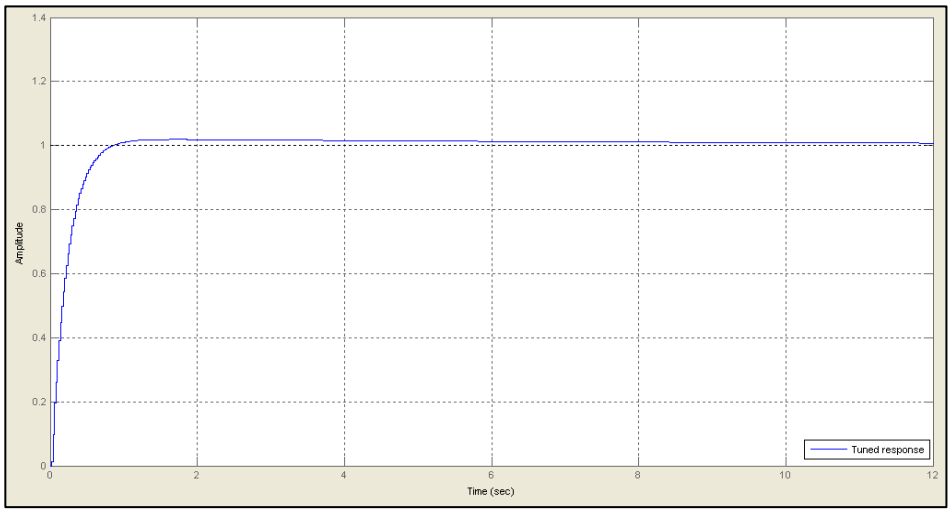


Figure 6.8a: PID Step Function characteristics for yaw transfer function

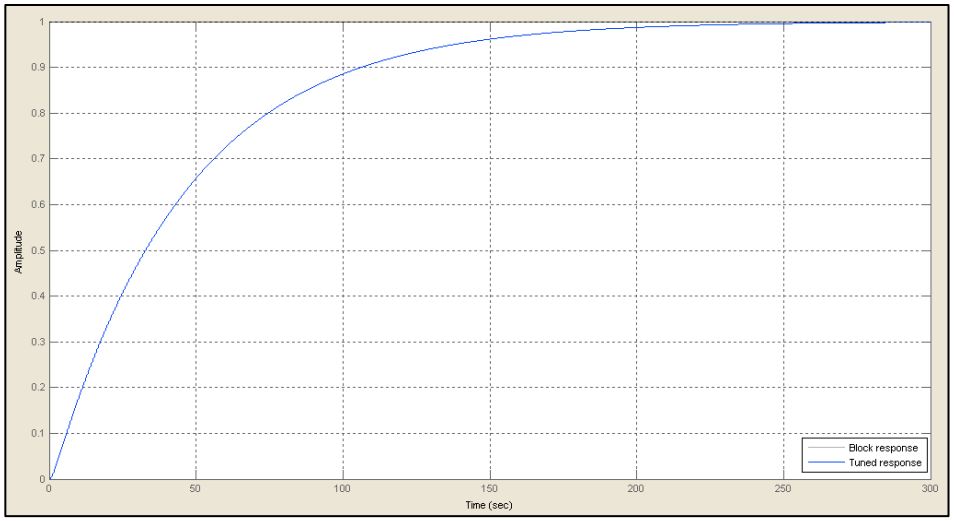


Figure 6.8b: PID Step Function characteristics for yaw transfer function

Parameter	Non-Optimal	Optimal
P	-0.71375	-0.018552
I	-71.3749	-1.8552
D	0	0
N	100	100
Rise Time (sec)	0.42	44.7
Settling Time (sec)	0.72	80.1
% Overshoot	1.90%	1.90%
Peak	1.02	1.02

Table 6.3: Yaw PI parameters

The final verification parameters are the root locus for the optimal yaw PI gains. From Figure 6.9, the yaw results are marginally stable, but acceptable for heading hold.

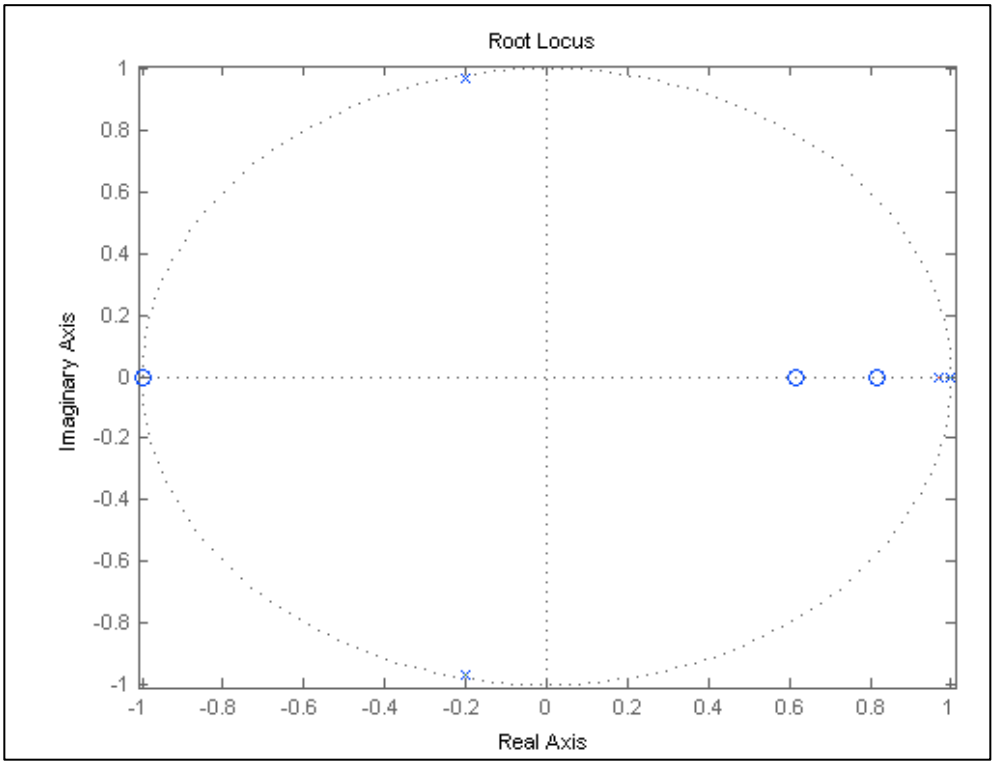


Figure 6.9: Root locus for yaw PID

The flight data from the five autonomous points as observed from the video are shown below. The heading hold seems to do well in the second flight, Figure 6.10b. It should be noted for the flight data shown in Figure 6.10b through Figure 6.10e, the gaps in data are due to the aircraft turning 180°, because of this turn the PWM values swing from one extreme to another. The video displays the aircraft adjusting and attempting to hold heading correctly.

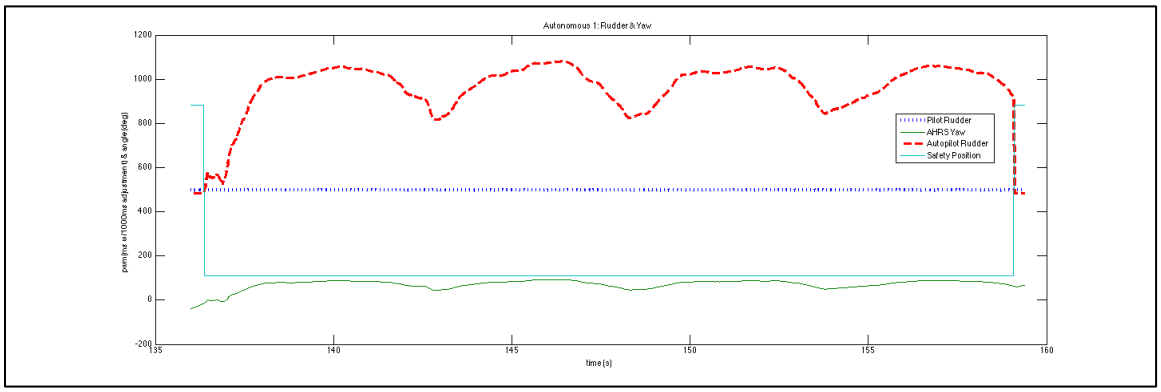


Figure 6.10a: Rudder/Yaw Data for first autonomous flight

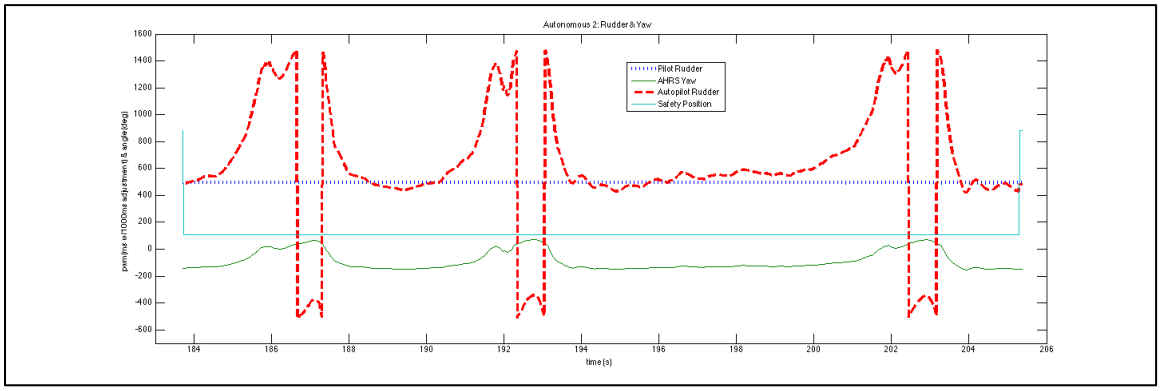


Figure 6.10b: Rudder/Yaw Data for second autonomous flight

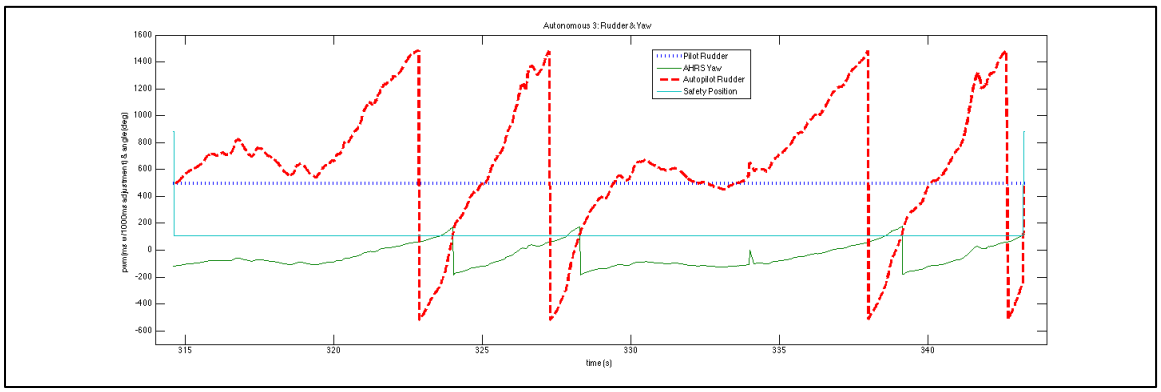


Figure 6.10c: Rudder/Yaw Data for third autonomous flight

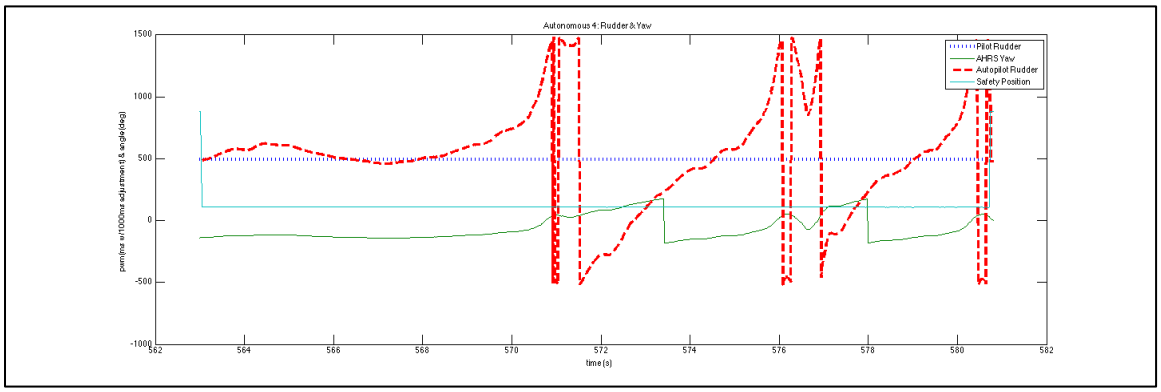


Figure 6.10d: Rudder/Yaw Data for fourth autonomous flight

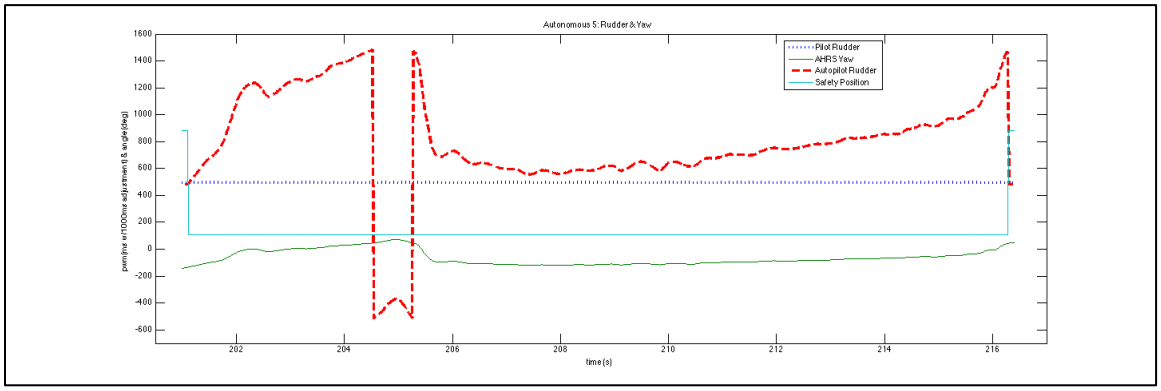


Figure 6.10e: Rudder/Yaw Data for fifth autonomous flight

Figure sets 6.4a-e, 6.7a-e, and 6.10a-e display the flight data of the aircraft once the optimal gains are introduced into the system. For the non-optimal gains, as seen in the Roll video, the Pitch video, the Roll/Pitch video, and the Roll/Yaw video, please refer to Appendix E for the non-optimal data, as well as and the entire flight envelope for the optimal flight.

Chapter 7 – Conclusions & Future Work

7.1 Conclusion

This research presented the design and implementation of a semi-autonomous fixed-wing aircraft designed by DU^2SL. This vehicle is designed to support future modification of both hardware and software elements as considered necessary by the user. This is made possible through the use, design and implementation of modularized hardware, software and the use of COTS components. This vehicle has preformed over twenty semi-autonomous flights and represents a baseline for development toward a fully autonomous platform. It's distinction in comparison to other aircraft capable of autonomous flight lies both in its lightweight foam construction and embedded system, where commands are processed and driven to the necessary surfaces for control and stability. It is this researchers hope that this work will provide others in the field an initial step so that they may continue to contribute to the evolving and expanding technology.

7.2 Future Work

The DU^{2}SL fixed-wing aircraft has demonstrated the ability to autonomously maintain a desired heading based on a pilot command. For this aircraft, the possible future work is boundless. Work can range from full automation, vision and controller development, to mechanical enhancements, software development and aircraft upgradeability for instance.

Full automation for this aircraft would involve taking the work presented in this thesis a step further by incorporating the GPS module and developing outer loop control. With outer loop control in place, the aircraft can fly to a particular waypoint and preform navigation tasks as directed by the operator. A multitude of possibilities present themselves once full automation for an aircraft is reached.

The multidisciplinary research of vision can also be incorporated into the aircraft. From here, different camera types, shapes and sizes can be mounted to the aircraft and collect data for topics such as obstacle avoidance, surveillance, visual inspection. Additionally, vision research can lead to further development of navigation of the aircraft solely by vision.

To further enhance autonomy of this aircraft more efficient controllers can be developed including, but not limited to Fuzzy Logic or Adaptive Control. These techniques can advance the aircraft intelligence and provide a substantial contribution to the field.

Mechanical enhancements and software development go hand in hand as they may reach a point where there is nothing more to add on to a system, or there simply exists a need to simply move to a different, more powerful platform for autonomous flight whether it incorporates vision or not.

Finally an upgrade to the aircraft or the possibility of moving to a completely different platform is also considerable.

References

[1] S.B. Stancliff, J.L. Laine, and M.C. Nechyba, "Learning to Fly: Design and Construction of an Autonomous Airplane," 1999 Florida Conference on Recent Advances in Robotics. (http://cimar.me.ufl.edu/FLA99/session4/flying.pdf)

[2] M. C. Nechyba and Y. Xu, "On Discontinuous Human Control Strategies," Proc. *IEEE Int. Conference on Robotics and Automation*, vol. 3, pp. 2237-43, 1998.

[3] A. Awan, S. A. Zaheer et al, "Autonomous control for a Kadet MKII RC airplane," Proc. *IEEE International Conference on Emerging Technologies*, pp. 50-55, 2008.

[4] L. Doitsidis and K. P. Valvanis, "A framework for fuzzy logic based UAV navigation and control," Proc. of the 2004 IEEE Int. Conf. on Robotics & Automation, pp. 4041-4046, New Orleans, USA, 2004.

[5] Y. Kang and J. K. Zaheer, "Linear Tracking for a Fixed-Wing UAV Using Nonlinear Model Predictive Control," Proc. *IEEE transactions on Control Systems Technology*, vol 15, no 5, 2009.

[6] A. Dorobantu, A. M.Murch, B. Mettler and G. J.Balas, "Frequency Domain System Identification for a Small,Low-Cost,Fixed-Wing UAV" University of Minnesota, Minneapolis.

[7] C. Theodore and M. Tischler "Rapid Frequency-Domain Modeling Methods for Unmanned Aerial Vehicle Flight Control Applications," *Journal of Aircraft*,Vol. 41, No.4, 2004

[8] J. Suk, Y. Lee, S. Kim, H. Koo and J. Kim "System Identification and Stability Evaluation of an Unmanned Aerial Vehicle from Automated Tests." Chungnam National University 2003

[9] K. R. Dunipace, S. E. Lyshevski, and R. D.Colgren "State-Space Nonlinear Identification and Tracking Control of Aircraft: A Motion Control Problem." Department of Electrical Engineering, Purdue University at Indianapolis 1998.

[10] B. Khalil and A. Yesildirek (2008) “System Identification of UAV under an Autopilot Trajectory Using ARX and Hammerstein-Wiener Methods” American University of Sharjah 2010.

[11] R. W. Beard, T. W. McLain, M. Goodrich, and E. P. Anderson, “Coordinated target assignment and intercept for unmanned air vehicles,” *IEEE Transactions on Robotics and Automation*, vol. 18, pp. 911–922, December 2002.

[12] F. Rafi, S. Khan, K. Shafiq, and M. Shah. “Autonomous target following by unmanned aerial vehicles.” Computer Vision Lab, Department of Computer Science, University of Central Florida, Orlando, FL

[13] M. Abdulrahim, “Flight Dynamics and Control of an Aircraft With Segmented Control Surfaces.” *AIAA Southeastern Regional Student Conference*. March 2003.

[14] Unknown “ SuperCub Instruction Manual” Hobbyzone, Horizon Hobby Inc., Champaign, Il, 2006 ‘http://www.horizonhobby.com/products/super-cub-dsm-rtf-HBZ7380’

[15] Unknown, “Apprentice RTF Assembly Manual” e-Flite, Horizon Hobby Inc., Champaign, Il, 2008 ‘http://www.e-fliterc.com/Products/Default.aspx?ProdID=EFL2725’

[16] Website. Academy of Model Aeronautics, Muncie, IL. ‘http://www.modelaircraft.org’

[17] Unknown. “VN-100 Development Board User Manual”, 2009, VectorNav Technologies ‘http://www.vectornav.com/index.php?option=com_content&view=article&id=14&Itemid=16’

[18] XMOS Ltd., “XC-2 Hardware Manual” version 1.4, 09/07/2009, XMOS Ltd, King Street, Bristol, United Kingdom. http://www.xmos.com/products/development-kits/xc-2-ethernet-kit

[19] G. Martins, A. Moses, M. Rutherford, and K. Valavanis “Enabling Intelligent Unmanned Vehicles through XMOS Technology” The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology. January 2011.

[20] Website. Spektrum Rc “http://www.spektrumrc.com/Products/Default.aspx?ProdID=SPMSA6110”

[21] Website. Sparkfun Electronics GPS receiver http://www.sparkfun.com/products/9436

[22] Unknown, “SL1206 Active GPS Antenna Product Specification” Sarantel Ltd., Wellingborough, United Kingdom. ‘http://www.sarantel.com/products/sl1206’

[23] Website. “LEA-5 module Series” U-blox AG, Switzerland ‘http://www.u-blox.com/en/lea-5h.html’

[24] Website. Sparkfun Electronics OpenLog DEV-09530 ‘https://www.sparkfun.com/products/9530’

[25] Website – Opensource & code sharing. Github ‘https://github.com/sparkfun/OpenLog’

[26] R. Beard and T McLain “Small Unmanned Aircraft Theory and Practice.” New Jersey: Princeton University Press, 2012

[27] M. V. Cook, “Flight Dynamics Principles. New York: John Wiley & Sons, 1997

[28]. D. J. Linse “Aircraft System Identification using Integrated Software Tools”

[29]. F.Guo. “A new Identification Method for Wiener and Hammerstein Systems” 2004.

[30]. K. W. Hipel, A. I. McLeod, W. C. Lennox “Advances in Box- Jenkins Modeling 1. Model construction”1977

[31] K. Vu. ‘Optimal setting for discrete PID controller’. *Pmc. IEEE,* 1992, vol 139, p. 31-40

[32] K.Vu, P. Tessier, and G.A. Dumont, ”Box-Jenkins Model LQG Controller: Design and Performance” *IEEE Proceedings-Control Theory and Applications*, Vol.148, No. 5, 419-429.

[33] L.A. Zadeh “From circuit theory to system theory”. — *Proc. IRE*, Vol. 50, pp. 856–865. c 1961

[34] S. Vorkoetter “An Electronic Speed Control Primer” *Sailplane & Electric Modeler Magazine* 3o Sep, 1997. n. pag. Print

Appendix

XMOS XC-2 Auxiliary characteristics

The XMOS XC-2 is an Event-Driven Processor development card intended for designing Ethernet-based products such as audio/video bridging applications and industrial control systems. It comprises a single XS1-G4 device, 10/100-BASE-T Ethernet, 4Mbits SPI flash memory, 10 LEDs and two press-buttons. I/O expansion areas are provided for connecting additional components, and an XSYS/XTAG connector can The devices dimensions are 86 x 54mm, with mounting holes measuring 3mm in diameter.

The XS1-G4 microprocessor consists of four XCores, each comprising an event-driven multi-threaded processor with integrated general-purpose I/O pins and 64 KBytes of on-chip RAM. The onboard crystal oscillator clocks the microprocessor at 25MHz, and each processor is clocked at 400MHz, however the I/O ports at 100MHz, by an on-chip phase-locked loop (PLL).

As stated above, 10 user LEDs are available to that can be driven by software, similarly two push-button switches whose states can be sampled at any time also by software. The I/O pins of three of the processors are brought out to expansion areas on

both sides of the card. These areas have 0.1” pitch through-plated holes and are suitable for use with IDC headers. Each expansion header provides a bank of 12 I/O pins. The Prototyping area is a plated area for adding components to the card.

The XC-2 requires connection from an external 5V power supply. The on-board regulator can supply both 5V and 3.3V.

The XMOS XTAG connector converts between XSYS and USB 2.0, allowing the XC-2 to be connected to most PCs Since no UART hardware is provided, two UART pins are mapped to ports [Douglas Watt, XMOS LTD . XC---2 Hardware Manual. Version 1.4, 2009].

Vectornav IMU/AHRS auxiliary characteristics

The VN-100 DEV is a development board for VN-100 attitude and heading reference module. It is designed to provide easy access to the module for development purposes. USB and RS-232 interfaces are provided, along with a header that includes full prototyping access to the VN-100.

The board can be powered in three ways, by the USB host computer, a 5 V adapter, or through the 20-pin header. A red LED indicates the power status of the board. The USB jack is a USB-B port. When connected to a PC the VN-100 DEV board will power up and a RED status LED will be illuminated.

The DB-9 port provides access to RS-232 level logic. The RS-232 line driver supports speeds up to 1Mbit. The RS-232 interface requires the VN-100 DEV board to be powered to function properly. The power can be supplied thru the USB port or thru the 5V power jack.

This is a standard 5V power jack. This powers the board when using the RS-232 interface.

Indicator lights are red when power is supplied to the board. Green flashing lights when data is being transferred across USB. However, when using the RS-232 logic interface, neither the RX nor the TX LED on the silkscreen will flash. This is normal behavior and does not indicate that the board is not receiving/transmitting data.

Push buttons S1 is only used to enable the VN-100 Boot mode. S2 is used for resetting the VN-100. S3 is normally used to tare the VN-100. Jumpers JP1 toggle the TX (data leaving the VN-100 module) UART data line. With the jump between pin 1 and pin 2 the TX signal is routed to the RS-232 interface. Setting the jump to pin 2 and pin 3 enables the TX at the USB interface. JP2 toggles the RX (data entering the VN-100 module) UART data line. With the jump between

The J5 header provides a complete interface to all the used pins on the VN-100 module.

Pin headers for the UM100 perform different operations; Table A-1 lists the assignments of the 20 headers found on the UM100 Development board.

Pin number		Pin number	
1	Vcc 3.3v - 5v	11	Not Used
2	REPRGM	12	DR_INT
3	TARE/RESTORE	13	Not Used
4	NRST	14	GND
5	ENABLE	15	SPI_SCK
6	Not Used	16	GND
7	TX	17	SPI_MOSI
8	SPI_CS	18	GND
9	RX	19	SPI_MISO
10	Not Used	20	GND

Table A-1. VectorNav UM100 Development board pin header assignments

Ublox LEA-5H auxiliary characteristics

The LEA-5 module series is a family of stand-alone GPS receivers featuring the high performance u-blox 5 positioning engine. These versatile receivers feature an extensive and flexible range of functionality, connectivity and cost savings options.

RECEIVER TYPE:

50 Channels. GPS L1 frequency, C/A Code. GALILEO Open Service L1 frequency
TIME TO FIRST FIX: Cold Start 29s
SENSITIVITY: Tracking & Navigation -160dBm. Reacquisition -160dBm. Cold Start -144 dBm.
HORIZONTAL POSITION ACCURACY: Autonomous < 2.5m
ACCURACY OF TIME PULSE SIGNAL: RMS 30 ns.
MAX NAVIGATION UPDATE RATE: 4HZ.
VELOCITY ACCURACY: 0.1m/s
OPERATIONAL LIMITS: Altitude 50000m. Velocity 500 m/s

GALILEO: The u-blox 5 is a GNSS chip that receives and tracks GPS and GALILEO signals simultaneously, enhancing accuracy and coverage.

f

PROTOCOL: NMEA and UBX. Both protocols are available on UART, USB, DDC and SPI.

LEA-5 modules include one configurable UART interface for serial communication.

LEA-5 modules provide a USB version 2.0 FS (Full Speed, 12Mbit/s) interface as an alternative to the UART.

The GPGGA message structure is used in this thesis; Table A-2 describes the GPGGA message.

Name	Example Data	Description
Sentence Identifier	$GPGGA	Global Positioning System Fix Data
Time UTC	170834	17:08:34 UTC
Latitude, Cardinal Direction	4124.8963, N	41d 24.8963' N or 41.248963
Longitude, Cardinal Direction	08151.6838, W	81d 51.6838' W or -081.516838
Fix Quality:		
If 0 = Invalid. If 1 = GPS fix. If 2 = DGPS fix	1	Data is from a GPS fix
Number of Satellites	5	5 Satellites are in view
Horizontal Dilution of Precision (HDOP)	1.5	Relative accuracy of horizontal position
Altitude	280.2, M	280.2 meters above mean sea level
Height of geoid above WGS84 ellipsoid	-34.0, M	-34.0 meters
Time since last DGPS update	blank	No last update
DGPS reference station id	blank	No station id
Checksum	*75	Checks for transmission errors

Table A-2. GPGGA Message Structure

Xbee 900MHZ long range modem characteristics.

Although not incorporated into the final UAV design, the XBee modems assisted in early telemetry readings prior to the OpenLog device. The XBee modules are long-range 900 MHz multipoint RF modules that are ideal for applications requiring low latency and predictable communication timing. Providing quick, robust communication in point-to-point, peer-to-peer, and multipoint/star configurations, XBee multipoint products enable robust end-point connectivity with ease. Whether deployed as a pure cable replacement for simple serial communication, or as part of a more complex hub-and-spoke network of sensors,

XBee multipxoint RF modules maximize wireless performance and ease of development. Table A-3 highlights the specifications of an XBee module.

RF Data Rate	10 kbps / 9.6 kbps
Indoor/Urban Range	Up to 1200 ft (370 m)
Outdoor/RF Line-of-Sight Range	Up to 6 mi (9.6 km)
Transmit Power	100 mW (+20 dBm)
Receiver Sensitivity (1% PER)	-106 dBm
Serial Data Interface	3.3V CMOS UART (5V Tolerant)
Configuration Method	AT Commands
Frequency Band	902 MHz to 928 MHz
Interference Immunity	FHSS (Frequency Hopping Spread Spectrum)
Serial Data Rate	1200 bps - 57.6 kbps
ADC Inputs	None
Digital I/O	None
Antenna Options	Wire Whip, U.FL, RPSMA

Table A-3. XBee Specifications