OnBoard Parameter Identification for a Small UAV

Amanda K. McGrail

West Virginia University

Follow this and additional works at: https://researchrepository.wvu.edu/etd

Recommended Citation


https://researchrepository.wvu.edu/etd/492

This Thesis is brought to you for free and open access by The Research Repository @ WVU. It has been accepted for inclusion in Graduate Theses, Dissertations, and Problem Reports by an authorized administrator of The Research Repository @ WVU. For more information, please contact ian.harmon@mail.wvu.edu.
OnBoard Parameter Identification for a Small UAV

Amanda K. McGrail

Thesis submitted to the
Statler College of Engineering and Mineral Resources
at West Virginia University
in partial fulfillment of the requirements
for the degree of

Master of Science
in
Aerospace Engineering

Marcello Napolitano, Ph.D., Chair
Srikanth Gururajan, Ph.D.
Mario Perhinschi, Ph.D.
Brad Seanor, Ph.D.

Department of Mechanical and Aerospace Engineering

Morgantown, West Virginia
2012

Keywords: Parameter Estimation, Stability and Control Derivatives, UAV
Abstract

OnBoard Parameter Identification for a Small UAV

By: Amanda K. McGrail

One of the main research focus areas of the WVU Flight Control Systems Laboratory (FCSL) is the increase of flight safety through the implementation of fault tolerant control laws. For some fault tolerant flight control approaches with adaptive control laws, the availability of accurate post failure aircraft models improves performance. While look-up tables of aircraft models can be created for failure conditions, they may fail to account for all possible failure scenarios. Thus, a real-time parameter identification program eliminates the need to have predefined models for all potential failure scenarios. The goal of this research was to identify the dimensional stability and control derivatives of the WVU Phastball UAV in flight using a frequency domain based real-time parameter identification (PID) approach.

The data necessary for this project was gathered using the WVU Phastball UAV, a radio-controlled aircraft designed and built by the FCSL for fault tolerant control research. Maneuvers designed to excite the natural dynamics of the aircraft were implemented by the pilot or onboard computer during the steady state portions of flights. The data from these maneuvers was used for this project.

The project was divided into three main parts: 1) off-line time domain PID, 2) off-line frequency domain PID, and 3) an onboard frequency domain PID. The off-line parameter estimation programs, in both frequency domain and time domain, utilized the well known Maximum Likelihood Estimator with Newton-Raphson minimization with starting values estimated from a Least-Squares Estimate of the non-dimensional stability and control derivatives. For the frequency domain approach, both the states and inputs were first converted to the frequency domain using a Fourier integral over the frequency range in which the rigid body aircraft dynamics are found. The final phase of the project was a real-time parameter estimation program to estimate the dimensional stability and control derivatives onboard the Phastball aircraft. A frequency domain formulation of the least-squares estimation process was used because of its low computational and memory requirements and robustness to measurement noise and sensor information dropouts. Most of the onboard parameter estimates obtained converge to the values determined using the off-line parameter estimation programs (though a few typically show a bias) within four to six seconds for longitudinal estimates and four to eight seconds for the later estimates. For the experiments conducted, the real-time parameter estimates did not diverge after the conclusion of the maneuver.
This thesis is dedicated to the memory of Ella Paulman.

“All we have to decide is what to do with the time that is given to us.”

J.R.R. Tolkien
Acknowledgments

First I would like to thank my parents, James and Deborah, and my sister, Rebecca, for their endless love and support. I would not have been able to get this far without you.

I would like thank my committee chairman, Dr. Marcello Napolitano, for his support on this project and for giving me the chance to work in the Flight Control Systems Laboratory. Working with the FCSL team provided me with opportunities I would not have had otherwise.

I would like to thank the members of my committee, Dr. Srikanth Gururajan, Dr. Brad Seanor, and Dr. Mario Perhinschi. Thank you for your support of my research and education and your insights on this project.

I would like to thank all the members of the lab past and present: Dr. Yu Gu, Dr. Haiyang Chao, Frank Barchesky, Zach Merceruio, Daniele Tancredi, Matt Rhudy, Matteo Dariol, Matteo Guerra, Giovanni De Nunzio, Tanmay Mandal, and Trenton Larrabee. Thank you for all the work you do to help make flight testing possible and for all the help you have given me with this project. I would also like to thank the pilots Mike Eden, Mike Spencer, and Dave Ellis.
# Table of Contents

List of Figures viii  
List of Tables x  
List of Symbols xii  
List of Subscripts xiii  
List of Acronyms xiv  

Chapter 1: Introduction 1  
1.1 - Research Objective 1  
1.2 - Research Organization 1  
  1.2.1 - Thrust Modeling 1  
  1.2.2 - Off-line Parameter Identification 1  
  1.2.3 - Real-Time Parameter Identification 2  
1.3 - Thesis Organization 2  

Chapter 2: Literature Review 3  
2.1 - Development of Aircraft PID from Flight Data 3  
  2.1.1 - Initial Development of Computational PID Techniques 3  
  2.1.2 - Further Development of PID Techniques 4  
2.2 - Frequency Domain Based PID Research 4  
  2.3.2 - Use of Real-Time PID within a Fault Tolerant Control System 5  

Chapter 3: Experimental Set-up and Flight Testing 9  
3.1 - WVU Phastball UAV 9  
  3.1.1 - Aircraft Configuration & Geometric Properties 9  
  3.1.2 - Inertial Properties 11  
3.2 - Flight Computer and Sensor Suite 11  
  3.2.1 - Flight Computer 11  
  3.2.2 - Real Time Operating System 12  
  3.2.3 - Sensors 12
Chapter 4: Aircraft Modeling

4.1 - Coordinate Systems

4.2 - Sign Conventions

4.2.1 - Body Axis Velocities, Forces, Moments, and Flow Angles

4.2.2 - Control Surface Deflections

4.3 - Equations of Motion

4.3.1 - Force and Moment Equations

4.3.4 - Kinematic Equations

4.3.5 - Linear Aircraft Model

4.3.6 - Short Period Approximation

4.3.7 - State Space Representation

Chapter 5: Thrust Modeling

5.1 - Thrust Equation for a Ducted Fan

5.1 - Static Thrust Model

5.2 - Dynamic Thrust Model

Chapter 6: Aircraft Parameter Identification

6.1 - Parameter Identification Techniques

6.1.1 - Ordinary Least-Squares Regression

6.1.2 - Equation Error Method

6.1.3 - Maximum Likelihood Estimator

6.1.4 - Maximum Likelihood Estimator - Optimization Algorithm

6.1.5 - Output Error

6.2 - Parameter Identification Applied to the WVU Phastball

6.2.1 - System IDentification Programs for AirCraft (SIDPAC)

6.2.2 - Pre-Processing of Flight Data

6.2.3 - Time Domain Off-Line PID
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>WVU Phastball UAV</td>
<td>9</td>
</tr>
<tr>
<td>3.2</td>
<td>ARIA System</td>
<td>12</td>
</tr>
<tr>
<td>3.3</td>
<td>Google Earth view of Louis Bennet Airfield with typical flight path</td>
<td>14</td>
</tr>
<tr>
<td>5.1</td>
<td>Definition of Control Volume in a Ducted Fan</td>
<td>23</td>
</tr>
<tr>
<td>5.2</td>
<td>Static Thrust testing flow chart</td>
<td>25</td>
</tr>
<tr>
<td>5.3</td>
<td>Static Thrust Testing Set-up</td>
<td>25</td>
</tr>
<tr>
<td>5.4</td>
<td>Static Thrust versus Throttle PWM Signal</td>
<td>26</td>
</tr>
<tr>
<td>6.1</td>
<td>Comparison of Fourier Transform Methods</td>
<td>41</td>
</tr>
<tr>
<td>7.1</td>
<td>Designed Multisines</td>
<td>44</td>
</tr>
<tr>
<td>7.2</td>
<td>Pilot-in-the-loop Flowchart</td>
<td>46</td>
</tr>
<tr>
<td>7.3</td>
<td>Multisine Inputs from October 3, 2012</td>
<td>46</td>
</tr>
<tr>
<td>8.1</td>
<td>Validation - 8/2/2012 - Flight 2 - Maneuver 5</td>
<td>50</td>
</tr>
<tr>
<td>8.2</td>
<td>Validation - 8/2/2012 - Flight 2 - Maneuver 5</td>
<td>50</td>
</tr>
<tr>
<td>8.3</td>
<td>Validation - 8/2/2012 - Flight 4 - Maneuver 9</td>
<td>51</td>
</tr>
<tr>
<td>8.4</td>
<td>Validation - 8/2/2012 - Flight 4 - Maneuver 9</td>
<td>51</td>
</tr>
<tr>
<td>8.5</td>
<td>Validation - 8/2/2012 - Flight 4 - Maneuver 2</td>
<td>54</td>
</tr>
<tr>
<td>8.6</td>
<td>Validation - 8/2/2012 - Flight 4 - Maneuver 2</td>
<td>54</td>
</tr>
<tr>
<td>8.7</td>
<td>Validation - 8/2/2012 - Flight 4 - Maneuver 8</td>
<td>55</td>
</tr>
<tr>
<td>8.8</td>
<td>Validation - 8/2/2012 - Flight 4 - Maneuver 8</td>
<td>55</td>
</tr>
<tr>
<td>8.9</td>
<td>Validation - 8/2/2012 - Flight 2 - Maneuver 2</td>
<td>58</td>
</tr>
<tr>
<td>8.10</td>
<td>Validation - 8/2/2012 - Flight 2 - Maneuver 2</td>
<td>58</td>
</tr>
<tr>
<td>8.11</td>
<td>Validation - 8/2/2012 - Flight 2 - Maneuver 2</td>
<td>59</td>
</tr>
<tr>
<td>8.12</td>
<td>Validation - 8/2/2012 - Flight 2 - Maneuver 2</td>
<td>59</td>
</tr>
<tr>
<td>8.13</td>
<td>Validation - 8/2/2012 - Flight 2 - Maneuver 3</td>
<td>60</td>
</tr>
<tr>
<td>8.14</td>
<td>Validation - 8/2/2012 - Flight 2 - Maneuver 3</td>
<td>60</td>
</tr>
<tr>
<td>8.15</td>
<td>Validation - 8/2/2012 - Flight 2 – Maneuver 3</td>
<td>61</td>
</tr>
<tr>
<td>8.16</td>
<td>Validation - 8/2/2012 - Flight 2 - Maneuver 3</td>
<td>61</td>
</tr>
<tr>
<td>8.17</td>
<td>Validation - 8/2/2012 - Flight 2 - Maneuver 5</td>
<td>64</td>
</tr>
<tr>
<td>8.18</td>
<td>Validation - 8/2/2012 - Flight 2 - Maneuver 5</td>
<td>65</td>
</tr>
<tr>
<td>8.19</td>
<td>Validation - 8/2/2012 - Flight 2 - Maneuver 5</td>
<td>65</td>
</tr>
</tbody>
</table>
List of Tables

Table 3.1  Overall Dimensions of Phastball  10
Table 3.2  Inertial Properties of Phastball  11
Table 3.3  Phastball Sensors  13
Table 6.1  Convergence Criteria for MLE with Newton-Raphson Optimization  36
Table 7.1  Acceptable Amplitudes for Linear Aircraft Model  42
Table 7.2  Properties of Optimized Multisines  45
Table 8.1  Summary of PID Flights  47
Table 8.2  Number of Maneuvers Used  47
Table 8.3  Time Domain - Longitudinal Non-Dimensional Derivative Estimates  48
Table 8.4  Time Domain – Longitudinal Dynamics Characteristics  49
Table 8.5  Summary of Residuals  49
Table 8.6  Time Domain - Lateral Non-Dimensional Derivative Estimates  52
Table 8.7  Time Domain - Lateral Dynamics Characteristics  53
Table 8.8  Summary of Residuals  53
Table 8.9  Frequency Domain - Longitudinal Non-Dimensional Derivative Estimates  56
Table 8.10 Frequency Domain - Longitudinal Dynamics Characteristics  57
Table 8.11 Summary of Residuals  57
Table 8.12 Frequency Domain - Lateral Non-Dimensional Derivative Estimates  62
Table 8.13 Frequency Domain - Lateral Dynamics Characteristics  63
Table 8.14 Summary of Residuals  64
Table 8.15 Comparison of Number of Longitudinal Maneuvers Available for Estimation  69
Table 8.16 Comparison of Non-Dimensional Derivative Estimates  69
Table 8.17 Comparison of Residuals  70
Table 8.18 Comparison of Number of Lateral Maneuvers Available for Estimation  71
Table 8.19 Comparison of Non-Dimensional Derivative Estimates  71
Table 8.20 Comparison of Residuals  72
| Table 8.21 | Average Error between Real-Time Parameter Estimate and Off-line Estimates | 76 |
| Table 8.22 | Average Standard Error, Average Settling Time, and Convergence Statistics | 76 |
| Table 8.23 | Average Error between Real-Time Parameter Estimate and Off-line Estimates - Y Parameters | 82 |
| Table 8.24 | Average Error between Real-Time Parameter Estimate and Off-line Estimates - I. Parameters | 83 |
| Table 8.25 | Average Error between Real-Time Parameter Estimate and Off-line Estimates - N Parameters | 83 |
| Table 8.26 | Average Standard Error, Average Settling Time, and Convergence Statistics | 83 |
**List of Symbols**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
<th>Unit/Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>Linear Acceleration</td>
<td>Feet/Second²</td>
</tr>
<tr>
<td>A</td>
<td>State Matrix</td>
<td>---</td>
</tr>
<tr>
<td>b</td>
<td>Wing Span</td>
<td>Feet</td>
</tr>
<tr>
<td>B</td>
<td>Input Matrix</td>
<td>---</td>
</tr>
<tr>
<td>C</td>
<td>Aerodynamic Coefficient</td>
<td>---</td>
</tr>
<tr>
<td>( \overline{c} )</td>
<td>Mean Aerodynamic Chord</td>
<td>---</td>
</tr>
<tr>
<td>I</td>
<td>Moment/Product of Inertia</td>
<td>Slug-Feet²</td>
</tr>
<tr>
<td>J</td>
<td>Cost Function</td>
<td>---</td>
</tr>
<tr>
<td>m</td>
<td>Mass</td>
<td>Slug</td>
</tr>
<tr>
<td>p</td>
<td>Roll Rate</td>
<td>Degrees/Second</td>
</tr>
<tr>
<td>q</td>
<td>Pitch Rate</td>
<td>Degrees/Second</td>
</tr>
<tr>
<td>( \overline{q} )</td>
<td>Dynamic Pressure</td>
<td>Slug-Foot/Second²</td>
</tr>
<tr>
<td>r</td>
<td>Yaw Rate</td>
<td>Degrees/Second</td>
</tr>
<tr>
<td>S</td>
<td>Planform Surface Area</td>
<td>Feet²</td>
</tr>
<tr>
<td>T</td>
<td>Thrust</td>
<td>Pound Force</td>
</tr>
<tr>
<td>V</td>
<td>Velocity</td>
<td>Feet/Second</td>
</tr>
<tr>
<td>( \alpha )</td>
<td>Angle of Attack</td>
<td>Degrees</td>
</tr>
<tr>
<td>( \beta )</td>
<td>Sideslip Angle</td>
<td>Degrees</td>
</tr>
<tr>
<td>( \delta )</td>
<td>Control Surface Deflection</td>
<td>Degrees</td>
</tr>
<tr>
<td>( \phi )</td>
<td>Roll Angle</td>
<td>Degrees</td>
</tr>
<tr>
<td>( \theta )</td>
<td>Pitch Angle, Vector of Parameter Estimates</td>
<td>Degrees, Unitless</td>
</tr>
<tr>
<td>( \psi )</td>
<td>Yaw Angle</td>
<td>Degrees</td>
</tr>
<tr>
<td>( \rho )</td>
<td>Air Density</td>
<td>Slug/Feet³</td>
</tr>
</tbody>
</table>
List of Subscripts

A  Aileron
D  Drag
E  Elevator
l  Rolling Moment
L  Lift
m  Pitching Moment
n  Yawing Moment
R  Rudder
Y  Side Force
X  Force on x-axis (Body)
Z  Force on z-axis (Body)
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARIA</td>
<td>Advanced Research Integrated Avionic</td>
</tr>
<tr>
<td>CG</td>
<td>Center of Gravity</td>
</tr>
<tr>
<td>ESC</td>
<td>Electronic Speed Controller</td>
</tr>
<tr>
<td>FCSL</td>
<td>Flight Control Systems Laboratory</td>
</tr>
<tr>
<td>FTCS</td>
<td>Fault Tolerant Control Systems</td>
</tr>
<tr>
<td>HARV</td>
<td>High Angle of Attack Research Vehicle</td>
</tr>
<tr>
<td>LSE</td>
<td>Least-Squares Estimator/Estimation</td>
</tr>
<tr>
<td>MLE</td>
<td>Maximum Likelihood Estimator</td>
</tr>
<tr>
<td>NACA</td>
<td>National Advisory Committee for Aeronautics</td>
</tr>
<tr>
<td>NASA</td>
<td>National Aeronautics and Space Administration</td>
</tr>
<tr>
<td>NTSB</td>
<td>National Transportation Safety Board</td>
</tr>
<tr>
<td>OBES</td>
<td>Onboard Excitation System</td>
</tr>
<tr>
<td>PF</td>
<td>Peak Factor</td>
</tr>
<tr>
<td>PID</td>
<td>Parameter Identification</td>
</tr>
<tr>
<td>PWM</td>
<td>Pulse Width Modulated</td>
</tr>
<tr>
<td>R/C</td>
<td>Radio Control</td>
</tr>
<tr>
<td>RTAI</td>
<td>Real-Time Application Interface</td>
</tr>
<tr>
<td>RTPID</td>
<td>Real Time Parameter Identification</td>
</tr>
<tr>
<td>SIDPAC</td>
<td>System IDentification Programs for AirCRAFT</td>
</tr>
<tr>
<td>UAV</td>
<td>Unmanned Aerial Vehicle</td>
</tr>
<tr>
<td>WVU</td>
<td>West Virginia University</td>
</tr>
</tbody>
</table>
Chapter 1: Introduction

1.1 - Research Objective

Traditionally the parameter identification (PID) process takes place after flight testing with estimation algorithms applied to a complete segment of flight data. This form of parameter identification is known as off-line PID, batch processing, or post-flight analysis. It is also possible to estimate the stability and control derivatives of the aircraft in real time by updating the estimates as more flight data is made available to the estimation algorithm. This is known as real-time parameter identification (RTPID) and can occur onboard the aircraft or in simulations after flight testing.

The goal of this effort was to create a real-time parameter identification (RTPID) algorithm to run onboard the aircraft. Before developing the onboard RTPID program, several other tasks had to be completed. First, an estimate of the thrust model was developed for use in the off-line PID programs. Second, off-line PID algorithms were developed in both the time and frequency domains to serve as baseline values for the onboard real-time PID. Finally, the RTPID programs were implemented both on a desktop computer (to test the algorithm between flight testing sessions) and onboard the aircraft. The following sections discuss each phase of the project in greater detail.

1.2 - Research Organization

1.2.1 - Thrust Modeling

The non-dimensional force and moment coefficients were needed for the off-line parameter identification but were not directly measured and had to be calculated using other flight parameters such as the dynamic thrust. The dynamic thrust model was developed using the Reynolds transport theorem applied to a ducted fan and the static thrust model developed from experimental data. Chapter 5 will discuss the thrust modeling in greater detail.

1.2.2 - Off-line Parameter Identification

The off-line PID algorithm served as the baseline comparison for the real-time PID. The off-line PID algorithms used the Maximum Likelihood Estimator (MLE) with Newton-Raphson minimization to estimate the non-dimensional stability and control derivatives. The initial estimates of the non-dimensional stability and control derivatives were obtained using Least Squares Estimation (LSE). This process (LSE followed by the MLE) was applied in both the time and frequency domains. For the off-line frequency domain PID, the Fourier Transforms of the inputs, states, and outputs, were obtained using the finite Fourier integral for the frequency range in which
the rigid body motion for the Phastball occurs. The main problems encountered during the off-line time domain PID were the noise on the flow angle measurements and sensor information dropouts. Noise on the flow angle measurements and sensor dropouts were effectively filtered by using the frequency range for the rigid body dynamics when evaluating the finite Fourier integral. Chapter 6 discusses the PID algorithms in more detail.

### 1.2.3 - Real-Time Parameter Identification

The real-time parameter identification was conducted using the frequency domain formulation of the LSE. This method was chosen for its low memory requirements and computational costs as well as its ability to filter measurement noise and sensor dropouts. The onboard real-time PID program was created in Simulink® and compiled into a Linux real-time executable using Matlab®’s Real Time Workshop.

### 1.3 - Thesis Organization

Chapter 2 presents a review of literature relevant to aircraft parameter estimation, frequency domain parameter estimation, and the role of real-time parameter estimation in fault tolerant control systems. Chapter 3 presents an overview of the Phastball UAV and the flight testing operations used for this research. Chapter 4 presents an overview of the modeling of the aircraft dynamics as it applies to this project. Chapter 5 presents a summary of the static and dynamic thrust modeling. Chapter 6 presents an overview of the least-squares and maximum likelihood estimation algorithms and their application to this project. Chapter 7 discusses the design of the PID maneuvers used for this project and Chapter 8 presents the results for the off-line PID, both time and frequency domain, and the onboard PID. Finally, Chapter 9 summarizes the conclusions and suggestions for future work.
Chapter 2: Literature Review

This literature presents a brief overview of the development of modern computational PID methods. The developments and benefits of frequency domain based parameter estimation are also discussed. Finally, the role of real-time parameter estimation within a fault detection, identification, and accommodation system is discussed.

2.1 - Development of Aircraft PID from Flight Data

2.1.1 - Initial Development of Computational PID Techniques

In 1919, NACA recommended that some work at Langley Airfield be devoted to flight testing to determine the characteristics of airplanes and how they differ from results from wind tunnel testing [1]. The 20s and 30s focused mainly on the collection of data, the isolation of lateral stability derivatives, analysis of longitudinal stability, and validation of the assumption of small oscillations. By the late 1930s a library of data on the control characteristics of aircraft existed and would form the basis of further aircraft modeling research. In the 1950s methods for estimating longitudinal stability and control derivatives using linear least-squares, estimating frequency response curves, obtaining laterals stability and control derivatives from frequency response data, and estimating transfer function coefficients were proposed [1]. Analog matching, a manual curve fitting technique, which uses an analog computer to calculate the response of the model, was also a popular technique in late 50s early 60s [2]. However, analog matching was a time consuming effort, where the results varied depending on the skill of the analyst, and only allowed for the extraction of major derivatives from flight data [3].

In the early 1960s, with the start of the digital computer age, NASA Dryden engineers Lawrence W. Taylor and Kenneth W. Iliff began using known regression techniques (i.e., linear least-squares and weight least-squares) for aircraft PID with poor results due to the presence of measurement noise [2]. A. V. Balakrishan, an expert in nonlinear, iterative estimation techniques at UCLA, worked closely with Taylor and Iliff on the development of a maximum-likelihood estimation technique (MLE). Balakrishan, Iliff, and Taylor completed their first FORTRAN program (named MMLE) in 1966 and published their first paper in 1968[2]. The code was validated and modified, in the late 60s and early 70s, using flight data from a diverse group of aircraft and flight regimes [2].
2.1.2 - Further Development of PID Techniques

After the creation of the first MLE program, researches at Dryden continued to focus on time domain parameter identification. In the late 1960s, Taylor and Iliff worked on applying an Extended Kalman Filter (EKF) to the aircraft parameter identification problem but struggled with multiple minima. In the 1970s Iliff returned to Kalman filtering, this time applying a Kalman filter (not the EKF) to PID problems where state noise (such as turbulence) was present. Dryden engineer Richard Maine later added the state noise algorithm developed by Iliff into MMLE3 (which was the last version of the program) in addition to adding efficiency, readability, and user friendliness to the code. In 1985 Maine worked with fellow Dryden engineer James Murray on an interactive parameter estimation program called “pEst”. pEst does not include the state noise algorithm included with MMLE3 but modifications can be made to pEst to include the state noise algorithm.

The first parameter estimation code developed at NASA Langley was developed in 1972 by Randall Grove[1]. Other theoretical developments that occurred at Langley include input design, model structure determination, real-time parameter estimation, data compatibility analysis, parameter estimation from steady flight measurements, and parameter estimation in the frequency domain. SIDPAC, A collection of Matlab® programs for aircraft parameter estimation, was developed by Morelli in 2002 [4]. Outside of NASA, other centers of aircraft PID research include the DLR Institute of Flight Research in Germany, Delft University of Technology in the Netherlands, and the National Aerospace Laboratory in the Netherlands [2].

2.2 - Frequency Domain Based PID Research

While researches at NASA Dryden focused on time domain parameter identification, researchers at NASA Langley worked on frequency domain parameter identification techniques. Taylor worked on frequency domain PID after his move to Langley in the late 1960s followed by a resurgence in frequency domain research at NASA Langley in the late1970s/early 1980s with work from Vladislav Klein and Eugene Morelli [3]. Klein helped to lay the groundwork for modern computational PID techniques in the frequency domain with his papers on general frequency domain parameter estimation [5] and MLE parameter estimation in the frequency domain[6]. Morelli later developed a high accuracy algorithm with flexible frequency selection to transform data from the time domain to the frequency domain [7] and later used a frequency domain formulation of the least-squares estimation for real-time PID ([8], [9]). Some of the benefits of frequency domain
parameter estimation include “physical insight in terms of frequency content, direct applicability to control system design, and a smaller number of data points for parameter estimation” [10].

2.3.2 - Use of Real-Time PID within a Fault Tolerant Control System

On July 19, 1989 United Airline flight 232 (a DC-10) lost the tail engine after the fan disk broke [11]. The shrapnel from the fan disk failure severed the lines of all three hydraulic systems removing the pilots ability to move the control surfaces. The crew was able to stabilize the aircraft immediately following the engine failure. The pilots then learned to control the aircraft using only the two remaining engines and made it to Sioux Gateway Airport for an emergency landing. The aircraft crashed on the runway with about half of the passengers surviving. On April 12, 1977 Delta Flight 1080 lost control of the elevator after it became locked at 19° of deflection [12]. The pilots, who had no indication of the failure, were able to regain control of the plane and safely land. Unfortunately, flight crews are not always able to regain control of the aircraft after a loss of control event. On September 8, 1994 USAIR Flight 427 crashed on final descent into Pittsburgh International Airport. On final descent, the aircraft’s rudder rapidly moved and reached its left aerodynamic blowdown limit, most likely caused by a jam of the main rudder power control unit valve, and crashed approximately twenty seconds later[13]. The crash report issued by the NTSB states that the crew recognized that there was a failure and attempted to regain control of the aircraft, but “could not be expected to have assessed the flight control problem and then devised and executed the appropriate recover procedure . . . under the circumstances of the flight” [13].

Aviation accidents such as United Airlines 232, Delta 1080, and USAIR 427 highlight the need for fault tolerant control systems in aircraft. Real-Time parameter identification can be used for both fault identification (through the change in aerodynamic parameters) and then for fault accommodation since an “update a model is often required for indirect adaptation” [14]. There have been numerous projects using real-time parameter identification for use with in an adaptable flight control system [14-16]. Shore used recursive parameter identification to track the effects of failures, such as locked control surfaces and engine failures, and for a reconfigurable control algorithm [15]. In order to track failures, the parameter estimates must be continuously updated during flight. Two main problems arise when continuously updating parameters. The first problem is the problem of no system excitation such as during cruise. Without enough information content, PID is unsuccessful. For a PID algorithm designed for monitoring the health of the aircraft, it is desirable that the algorithm be responsive enough to accurately track rapid changes in the health of the aircraft. The problem with a responsive PID algorithm is that it is less resilient to measurement
noise. To address these two problems, Shore used the stabilized recursive least squares algorithm with forgetting factor; the cost function of the algorithm is given by Equation 2.1 where $\lambda$ is the forgetting factor and $\alpha_w$ is used to adjust the stability of the algorithm [15].

$$ J(\theta(n)) = \sum_{k=1}^{n} \lambda^{n-k} \left[ y(k) - \theta^T(n)w(k) \right]^2 + \alpha_w \| \theta(n) - \theta(n-1) \|^2$$

(2.1)

Setting the forgetting factor close to zero allows the algorithm to track rapid changes but makes it more susceptible to noise, while setting it close to one makes the algorithm resilient to noise but slow to respond to changes in the system. The second term controls the stability of the algorithm (it is tuned using $\alpha_w$) when there is little system excitation by limiting “the deviation of the current estimate from the previous estimate”. Using this method, convergence of parameter estimates was achieved in three to twelve seconds, depending on the parameter.

Hardier and Bucharles also used a real-time parameter identification program a part of a aircraft health monitoring process in a fault tolerant control system [14]. A frequency domain implementation of the output error estimation algorithm was used to estimate the stability and control derivatives from a linear aircraft model. The output error method was chosen for its ability to “process measurement noises which spoil the A/C states and thus bias the LS estimates” [14], and formulating it in the frequency domain allowed it to be applied in real time. The results were convergence of the estimated parameters in one or two iterations [14].

A collaborative study with West Virginia University and Hankuk Aviation University was conducted to study the use of an on-line parameter identification program to estimate the aircraft model after the failure of a control surface [17]. For the study the frequency domain implementation of the least squares estimation was used. A frequency domain approach was chosen for its resilience to system and measurement noise and the availability of a reliable parameter to analyze the quality of the estimate online. The control surface failure considered was a damaged and locked elevator (locked actuator and reduced control surface effectiveness). The study showed the on-line PID program was able to provide accurate estimates of the aircraft parameters after the failure. The study also showed the importance of using a short duration PID maneuver following the detection of a control surface failure to improve the accuracy of the parameter estimates that get passed to the accommodation scheme [17].

A second collaborative study with West Virginia University, Polytechnic of Torino, and the University of Perugia further investigated the use of real-time parameter identification after control
surface failures for use with in the NASA Intelligent Flight Control System (IFCS) [18]. The study specifically looked at the estimation of the non-dimensional derivatives versus the estimation of the dimensional derivatives, formulating the problem with individual control surfaces or total control surfaces, and the use of time domain derivatives versus frequency domain derivatives. For this study a formulation of the Fourier Transform Regression that allows for estimation of either the dimensional or non-dimensional derivatives was used. The failure for this study was a locked stabilator with reduced effectiveness. At nominal conditions it was found that the parameter estimates vary little between use of a single control surface or the global control surface in the formulation of the problem as the FTR method is “more robust to correlation in the input time histories” [18]. After a control surface failure, using individual control surfaces for the parameter estimation process proved to be more successful. The study also showed that while using time domain derivatives in the PID process led to shorter convergence times than using frequency domain derivatives, the end results were comparable. Finally, the study showed that the estimates of the dimensional derivatives were more accurate than the estimates of the non-dimensional derivatives; however, the estimates for the dimensional derivatives took longer to converge [18].

Possibly the most notable example of the use of a real-time parameter identification program within a fault tolerant control systems is NASA’s Intelligent Flight Control System (IFCS) project. The goal of the IFCS project is to “develop neural network-based adaptive control systems for flight” [19] and has “proposed a set of control systems, estimation algorithms, and neural networks for flight-testing in an effort to show real-time adaptation, allowing for safe, predictable control of an aircraft with uncertain aerodynamic and control derivatives” [19]. IFCS uses a stochastic optimal feedforward and feedback technique (SOFFT) based control system that uses the stability and control derivatives in addition to the aircraft states and pilot inputs to calculate surface deflections. Prior to 2003, a neural network pre-trained with wind tunnel data for the IFCS F-15 was used to provide the stability and control derivatives to the controller based on the flight conditions. To give the IFCS the ability to adapt to unknown aerodynamics, a real time parameter identification algorithm was employed. The frequency domain implementation of the equation-error method, the Fourier Transform Regression, modified to work with a time based window of data was used for the real-time parameter estimation. The use of the time based window means that the estimates are “correlated to averaged flight conditions over the data window” [19] and that there is no long term storage of the parameter estimates. “Also, the PID is only able to compute valid estimate with sufficient system excitation, so the estimates should not be used under certain conditions” [19]. A dynamic cell structure (DCS) neural network was added to the system to store the estimated
derivative corrections (difference between the online estimated parameter value and the parameter estimate from the pre-trained neural network) for the given flight conditions when the parameter estimates are valid and to recall the parameter estimates when the real-time PID estimates are not valid (such as conditions without system excitation). The implemented real-time PID algorithm “demonstrated the ability to estimate the aerodynamic stability and control derivatives in real time” [19], and the DCS was able to train and recall the derivative information. “As a complete system, the online PID algorithm and DCS have produce promising results in a closed-loop simulation environment” [19].
Chapter 3: Experimental Set-up and Flight Testing

This chapter discusses the geometric and inertial properties of the WVU Phastball UAV, the Phastball’s flight computer and sensors, and the flight testing procedures used to gather data and to test the developed onboard PID schemes.

3.1 - WVU Phastball UAV

3.1.1 - Aircraft Configuration & Geometric Properties

The WVU Phastball is a radio-controlled (R/C) aircraft designed, constructed, and instrumented by WVU researchers [20]. The FCSL currently has a fleet of three Phastball aircraft all with similar configurations and basic sensor suites. In addition to the basic sensors, two of the Phastballs have the Advanced Research Integrated Avionic (ARIA) system (see Section 3.2.1) integrated to extend the research capabilities of the aircraft. The aircraft with the ARIA system were used for this project.

Figure 3.1: WVU Phastball UAV

The Phastball is a monoplane aircraft with a T-tail configuration, tricycle landing gear, and two electric ducted fans mounted aft of the CG. Figure 3.1 shows the WVU Phastball. The main portion of the wing features a NACA 2410 airfoil and is made up of three sections: an inboard section and two identical outboard sections (the main wing utilizes winglets). The NACA 0009 was used for both the horizontal and vertical stabilizers. Phastball’s control surfaces include two elevators, two ailerons, and a rudder which are all operated by digital servos. The fuselage uses a carbon fiber and fiberglass composite with plywood bulkheads for structural rigidity. The power plant consists of two ARC 36-55-1.5 electric brushless motors with Skyworld 90 mm ducts. Each
motor and duct produces approximately 3 lbs of static thrust [21] and 1.5 lbs of dynamic thrust at the cruise throttle setting. Table 3.1 summarizes the Phastball’s overall geometric properties. Appendix A contains a more detailed description of the aircraft’s dimensions.

Table 3.1: Overall Dimensions of Phastball

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Overall Dimensions</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Length</td>
<td>7.25</td>
<td>feet</td>
</tr>
<tr>
<td>Height</td>
<td>0.90 (to top of fuselage)</td>
<td>feet</td>
</tr>
<tr>
<td></td>
<td>1.82 (to top of tail)</td>
<td></td>
</tr>
<tr>
<td>Wingspan</td>
<td>7.33</td>
<td>feet</td>
</tr>
<tr>
<td><strong>Wing Overall Dimensions</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Airfoil</td>
<td>NACA 2410</td>
<td>---</td>
</tr>
<tr>
<td>Wingspan</td>
<td>7.33</td>
<td>feet</td>
</tr>
<tr>
<td>Mean Aerodynamic Chord</td>
<td>1.08</td>
<td>feet</td>
</tr>
<tr>
<td>Planform Area</td>
<td>7.847</td>
<td>feet²</td>
</tr>
<tr>
<td><strong>Horizontal Stabilizer Dimensions</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Airfoil</td>
<td>NACA 0009</td>
<td>---</td>
</tr>
<tr>
<td>Span</td>
<td>2.83</td>
<td>feet</td>
</tr>
<tr>
<td>Mean Aerodynamic Chord</td>
<td>0.7</td>
<td>feet</td>
</tr>
<tr>
<td>Planform Area</td>
<td>2.01</td>
<td>feet²</td>
</tr>
<tr>
<td><strong>Vertical Stabilizer Dimensions</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Airfoil</td>
<td>NACA 0009</td>
<td>---</td>
</tr>
<tr>
<td>Span</td>
<td>0.92</td>
<td>feet</td>
</tr>
<tr>
<td>Mean Aerodynamic Chord</td>
<td>0.96</td>
<td>feet</td>
</tr>
<tr>
<td>Planform Area</td>
<td>1.75</td>
<td>feet²</td>
</tr>
</tbody>
</table>
3.1.2 - Inertial Properties

The main structure of the Phastball (fuselage, wing, etc) as well as its components (batteries, camera, computer, etc) were modeled in Solidworks. By assigning material properties to the all of the components, the inertial properties of the aircraft assembly were estimated by Solidworks. Because the configuration of the Phastball is continuously changing to meet various research objectives, this method of estimating the inertial properties was preferred over the bifilar pendulum method. Table 3.2 contains the estimated inertial properties of the Phastball for the flight testing days considered in this thesis.

<table>
<thead>
<tr>
<th>Flight Date</th>
<th>Phastball Platform</th>
<th>Mass (slug)</th>
<th>( I_{xx} ) (slug-ft(^2))</th>
<th>( I_{yy} ) (slug-ft(^2))</th>
<th>( I_{zz} ) (slug-ft(^2))</th>
<th>( I_{xz} ) (slug-ft(^2))</th>
</tr>
</thead>
<tbody>
<tr>
<td>8/2/2012</td>
<td>Blue</td>
<td>0.760</td>
<td>0.535</td>
<td>2.089</td>
<td>2.543</td>
<td>0.108</td>
</tr>
<tr>
<td>9/5/2012</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10/3/2012</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10/27/2012</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

3.2 - Flight Computer and Sensor Suite

3.2.1 - Flight Computer

The Phastball features the ARIA system (shown in Figure 3.2), which was specifically designed by WVU researchers for fault tolerant flight control research projects [22]. The ARIA features a stack of four printed circuit boards (PCB). The top PCB holds a compact flash card reader, GPS receiver, indicator LEDs, and micro SD data loggers. The second PCB contains the main flight computer, a Diamond System 800 MHz Athena II general purpose computer. The main flight computer has six serial ports for communication with other devices. Two of the serial ports are used for communication with a Netburner MOD 5213 microprocessor, one for communication with a GPS receiver, and the other for communication with an RF modem. The third board features the PC104 power supply and two additional serial ports. The bottom PCB serves as the main sensor interface and signal distribution center. A MOD 5213 microprocessor is integrated into the bottom PCB and is used for most low level communications including: interfacing with the IMU, reading ground pilot commands, and creating actuator commands as specified by the control laws.
3.2.2 - Real Time Operating System

The operating system used on the main computer is a Linux kernel with a Real Time Application Interface (RTAI) patch [22]. The RTAI patch allows researchers to create sensing and control algorithms in Matlab® and Simulink® that can then be compiled into real-time executables using Matlab®’s Real Time Workshop. The operating system runs from a bootable compact flash card.

3.2.3 - Sensors

The Phastball utilizes a suite of sensors to measure all relevant flight information. An IMU is incorporated onto the bottom PCB of the ARIA System to measure linear and angular accelerations. The remaining sensors, located outside of the ARIA System, include a pitot tube and angle of attack/sideslip flow vanes, which were developed in house, to measure airspeed, angle of attack, and sideslip. A commercial off-the-shelf GPS receiver is used to record positions and velocities with respect to the earth based coordinate system. For this project, due to damage received by the pitot tube during transportation and the occasional failure of the pressure sensors used for the pitot tube, the GPS velocity was the more reliable velocity measurement. On the Phastball platforms with an ARIA System, attitude information is determined using a GPS/INS sensor fusion algorithm developed by WVU researchers [22]. The GPS/INS sensor fusion algorithm is replaced by a vertical gyroscope on the Phastball platform without the ARIA system (which was not used for
this project because a flight computer was necessary). Table 3.3 details the sensors used on the Phastball.

Table 3.3 - Phastball Sensors [23]

<table>
<thead>
<tr>
<th>Sensor</th>
<th>Model Information</th>
<th>Notes</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMU</td>
<td>ADIS 16355® IMU</td>
<td>$a_x$, $a_y$, $a_z$ range: $\pm 10$ g</td>
<td>Bottom PCB of ARIA System</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$p,q,r$ range: $\pm 150$ deg/sec</td>
<td></td>
</tr>
<tr>
<td>GPS</td>
<td>Novatel OEM4</td>
<td>$x, y, z, V_x, V_y, V_z$ in Earth Based</td>
<td>Behind Nose</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Coordinate System</td>
<td></td>
</tr>
<tr>
<td>Pitot Tube</td>
<td>WVU FCSL Design</td>
<td>---</td>
<td>Nose</td>
</tr>
<tr>
<td>Angle of Attack Flow</td>
<td>WVU FCSL Design</td>
<td>---</td>
<td>Side of Nose</td>
</tr>
<tr>
<td>Vanes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Side Slip Flow Vanes</td>
<td>WVU FCSL Design</td>
<td>---</td>
<td>Bottom of Nose</td>
</tr>
<tr>
<td>Static Pressure Sensor</td>
<td>Honeywell ASCX15AN</td>
<td>Range: [0-15] PSI</td>
<td>Nose PCB</td>
</tr>
<tr>
<td>Dynamic Pressure Sensor</td>
<td>Honeywell ASCX01DN</td>
<td>Range: [0-1] PSI</td>
<td>Nose PCB</td>
</tr>
<tr>
<td>Temperature Sensor</td>
<td></td>
<td>Digital Sensor under 3.3 V Supply</td>
<td>Nose</td>
</tr>
<tr>
<td>Laser Range Finder</td>
<td></td>
<td>Precision: $\pm 1$ m</td>
<td>Nose</td>
</tr>
<tr>
<td>GPS/INS Sensor Fusion</td>
<td>WVU FCSL Design</td>
<td>Estimates of $\phi, \theta, \psi$</td>
<td>Control Scheme</td>
</tr>
</tbody>
</table>

3.2.4 - Flight Testing

The data required for this project was obtained during flight testing conducted by the FCSL team at Louis Bennet Airfield at the WVU’s Jackson’s Mill facility near Weston, West Virginia. A typical flight consisted of racetrack style circuits with two straight legs of approximately 600 meters each. Figure 3.3 shows an aerial image of the airfield with a typical mission flightpath.
During flight tests, maneuvers designed to excite the short period, dutch roll, and roll modes of the aircraft (Chapter 7 discusses maneuver design in greater detail) were implemented during the straight leg portions of the flight either by the ground pilot or by the On-board Excitation System (OBES). During the straight leg portions of the flight, the ground pilot can pass control to the OBES by flipping a control switch on the R/C transmitter. Once control has been passed to OBES, the control surface deflections for a predefined maneuver (doublet, frequency sweep, multisine, etc) are commanded by the onboard computer. Traditionally, once the control switch has been flipped, total control of the aircraft’s control surfaces is passed to OBES. However, OBES with the ground pilot in-the-loop (or simply pilot-in-the-loop) was used for this project. For pilot-in-the-loop activations, OBES adds predefined perturbation deflections, to the control surface command from the ground pilot. The benefits and execution of pilot-in-the-loop flights is discussed in more detail in Chapter 7.
Chapter 4: Aircraft Modeling

This chapter describes the various reference frames and sign conventions used for this project; a brief overview of the equations that make up the mathematical aircraft model is also presented.

4.1 - Coordinate Systems

There are five coordinate systems used for aircraft modeling: fixed earth axes, vehicle carried earth axes, body axes, stability axes, and wind axes. The origin of the earth axes is an arbitrary point on the earth's surface with positive \([X_E, Y_E, Z_E]\) pointing towards geographic north, east, and to the center of the earth respectively. The axes of the vehicle carried earth axes \([X_V, Y_V, Z_V]\) are parallel to those of the fixed earth coordinate system, but the origin of the vehicle carried earth axes is the center of gravity (CG) of the aircraft. The origin of the body axes, stability axes, and wind axes are also at the CG of the aircraft but the alignment of the axes differ. In the body coordinate system the \([X_B, Y_B, Z_B]\) axes point through the nose, the right wing, and the bottom of the aircraft respectively. The stability axes \([X_S, Y_S, Z_S]\) point in the same general directions as the body axes but are aligned with the air-relative velocity. The positive \(X_S\) is pointed forward and aligned with the air-relative velocity projection onto the aircraft XZ-plane. The positive \(Y_S\) and \(Z_S\) axes are pointed through the right wing and through the bottom of the aircraft respectively. The stability axes are fixed with respect to the aircraft and are defined at the beginning of the maneuver. The wind axes are similar to the stability axes. The origin is at the CG of the aircraft with \(X_W\) pointed forward and aligned with the air-relative velocity vector, the \(Y_W\) is pointed out the right side of the aircraft, and \(Z_W\) is pointed out the bottom of the aircraft. Unlike the stability axes, however, the wind axes are not fixed with respect to the aircraft, and the origin of the wind axes follows the trajectory of the aircraft.

4.2 - Sign Conventions

4.2.1 - Body Axis Velocities, Forces, Moments, and Flow Angles

For this project the body axis velocities and forces were oriented in accordance with the definition of the body axes coordinate system. The body axis components of the aerodynamic moments are then defined using the right hand rule. The sign convention for angle of attack and sideslip can be derived by first defining angle of attack and sideslip angle in terms of body axis velocities as shown in Equation 4.1.
\[ \alpha = \tan^{-1}\left( \frac{w}{u} \right) \]  

(4.1a)

\[ \beta = \sin^{-1}\left( \frac{v}{V} \right) \]  

(4.1b)

Because \( V \) and \( u \) are always positive, for angle of attack and sideslip angle to be positive, \( v \) and \( w \) must also be positive. If this is case, the air-relative velocity vector is coming from under the aircraft (positive angle of attack) and from the right (positive sideslip angle).

The Euler angles are the angles between the vehicle carried earth axes and the body axes and are used to rotate the vehicle carried earth axes values to the body axes. From the aircraft CG and oriented with the positive body axes, a positive roll angle is a roll to the right, a positive pitch angle is nose up, and a positive yaw angle is a rotation to the right.

### 4.2.2 - Control Surface Deflections

For this project, it is necessary to define positive individual control surface deflection and positive total control surface deflection for multiple control surfaces (ailerons and elevators for Phastball). For individual control surface deflections, positive deflection is defined as trailing edge down. A positive total control surface deflection produces a negative aerodynamic moment. For multiple control surfaces that move symmetrically (such as the elevators), the total control surface deflection is defined as the average of the individual surface deflections. A modified definition of total surface deflection is needed for control surfaces that move asymmetrically (the ailerons). To remain consistent with the definitions of positive total surface deflection and positive individual surface deflection, the total aileron deflection is defined as one-half the difference between the right and left aileron deflections as shown in Equation 4.2.

\[ \delta_A = \frac{1}{2} (\delta_{AR} - \delta_{AL}) \]  

(4.2)

### 4.3 - Equations of Motion

This section provides an overview of the equations used to model the aircraft. The nine nonlinear differential equations are presented first. Because most PID maneuvers occur around one reference condition, the equations of motion can be linearized around that reference condition. The
second half of the section presents the assumptions and simplifications made to linearize the equations of motion. Finally, the state space representation of the aircraft is presented.

### 4.3.1 - Force and Moment Equations

Applying the conservation of linear momentum to an aircraft results in Equation 4.3, the force equations in the body axes [10].

\[
m\dot{u} = m(rv - qw) + \bar{q}SC_x - mg\sin\theta + T \quad (4.3a) \\

m\dot{v} = m(pw - ru) + \bar{q}SC_y + mg\cos\theta\sin\phi \quad (4.3b) \\

m\dot{w} = m(qu - pv) + \bar{q}SC_z + mg\cos\theta\cos\phi \quad (4.3c)
\]

Airspeed, angle of attack, and sideslip angle are easily measured by the sensors on the aircraft; therefore, it is convenient to write the force equations (Equations 4.3) in terms of airspeed, angle of attack, and sideslip angle. By expressing the angle of attack, sideslip angle, and total velocity as a function of the body axes velocities (Equations 4.2), it can be shown that Equations 4.3 can be written as Equations 4.4.

\[
\dot{V} = -\frac{\bar{q}S}{m}(C_o \cos\beta - C_y \sin\beta) + \frac{T}{m}\cos\alpha\cos\beta + g\left((\cos\phi\cos\theta\sin\alpha\cos\beta + \sin\phi\cos\theta\sin\beta - \sin\theta\cos\alpha\cos\beta)\right) \\
\dot{\alpha} = -\frac{\bar{q}S}{mV\cos\beta}(C_L + q - \tan\beta(p\cos\alpha + r\sin\alpha)) + \frac{g}{V\cos\beta}(\cos\phi\cos\theta\cos\alpha + \sin\theta\sin\alpha) - \frac{T\sin\alpha}{mV\cos\beta} \\
\dot{\beta} = \frac{\bar{q}S(mV)}{mV(C_r \cos\beta + C_o \sin\beta) + p\sin\alpha - r\cos\alpha + \frac{g}{V}\cos\beta\sin\phi\cos\theta + \frac{\sin\beta}{V}\left(g\cos\alpha\sin\theta - g\sin\alpha\cos\phi\cos\theta + \frac{T\cos\alpha}{m}\right)}
\]

Applying the conservation of angular momentum to the aircraft and assuming aircraft symmetry, results in Equations 4.5, the moment equations in the body axes [10].

\[
M_x = \dot{p}I_x - \dot{r}I_{xz} + qr(I_z - I_y) - qpI_{xz} \quad (4.5a) \\
M_y = \dot{q}I_y + pr(I_x - I_z) + (p^2 - r^2)I_{xz} \quad (4.5b) \\
M_z = \dot{r}I_z - \dot{p}I_{xz} + pq(I_y - I_x) + qrI_{xz} \quad (4.5c)
\]
The body axes moments can also be expressed as a function of the aerodynamic moments (with respect to the body axes) as shown in Equations 4.6. Equations 4.6 assume the motors have a constant angular velocity during PID maneuvers; therefore, applied moments from the motors do not need to be considered.

\[
M_x = qSbC_l \quad (4.6a)
\]

\[
M_y = qScC_m \quad (4.6b)
\]

\[
M_z = qSbC_n \quad (4.6c)
\]

Equations 4.5 and 4.6 can be combined to produce Equations 4.7.

\[
\dot{p}l_x - \dot{r}l_{xz} = \bar{q}SbC_l - qr(I_z - I_y) + qp\dot{l}_{xz} \quad (4.7a)
\]

\[
\dot{q}l_y = qScC_m - pr(I_x - I_z) - (p^2 - r^2)l_{xz} \quad (4.7b)
\]

\[
\dot{r}l_z - \dot{p}l_{xz} = \bar{q}SbC_n - pq(I_y - I_x) - qr\dot{l}_{xz} \quad (4.7c)
\]

### 4.3.4 - Kinematic Equations

The kinematic equations relate body axes components of the angular velocities and the rate of change of the Euler angles. Expressing the body axes angular velocities as a coordinate system rotation of the Euler angles produces Equation 4.8. Inverting this expression produces Equation 4.9.

\[
\begin{bmatrix}
    p \\
    q \\
    r
\end{bmatrix} =
\begin{bmatrix}
    1 & 0 & -\sin\theta \\
    0 & \cos\phi & \sin\phi\cos\theta \\
    0 & -\sin\phi & \cos\phi\cos\theta
\end{bmatrix}
\begin{bmatrix}
    \dot{\phi} \\
    \dot{\theta} \\
    \dot{\psi}
\end{bmatrix} \quad (4.8)
\]

\[
\dot{\phi} = p + \tan\theta(q\sin\phi + r\cos\phi) \quad (4.9a)
\]

\[
\dot{\theta} = q\cos\phi - r\sin\phi \quad (4.9b)
\]

\[
\dot{\psi} = \frac{q\sin\phi + r\cos\phi}{\cos\theta} \quad (4.9c)
\]
4.3.5 - Linear Aircraft Model

Most PID maneuvers occur about a single reference condition and consist of small perturbations around this reference condition. For the maneuvers with this study, the reference condition is steady, wings level flight. For the Phastball at steady flight the reference flight conditions are:

- $\alpha \approx 2.5^\circ$
- $\beta = 0^\circ$
- $V \approx 95$ ft/s
- $\delta_e \approx 2^\circ$

For this reference condition, the following assumptions are made:

- Each variable can be replaced by a steady state term plus a perturbation term
- $\beta_o = 0$ (no sideslip assumption)
- $\phi_o = 0$ (wings level flight assumption)
- $p_o = q_o = r_o = 0$ (steady flight)
- $V_o \gg \Delta V \Rightarrow V \approx V_o \Rightarrow q \approx \bar{q}$
- $\Delta T = 0$ (constant power setting)
- Small angle assumption

The previous assumptions applied to Equations 4.4, 4.7, and 4.9 along with replacing the total force and moment coefficients with linear expansions produces Equations 4.10 and 4.11 (the linear longitudinal and lateral-directional dynamic equations).

\[
\Delta \dot{V} = -\bar{q} \frac{S}{m} \left( C_{DV} \frac{\Delta V}{V_o} + C_{Da} \Delta \alpha + C_{Dq} \frac{\Delta q \bar{c}}{2V_o} + C_{D\delta} \Delta \delta \right) - g \cos \gamma_o \left( \Delta \theta - \Delta \alpha \right) - \frac{T_o \sin \alpha_o}{m} \Delta \alpha 
\]

\[
\Delta \ddot{\alpha} = -\bar{q} \frac{S}{m V_o} \left( C_{L\alpha} \frac{\Delta V}{V_o} + C_{L\alpha} \Delta \alpha + C_{Lq} \frac{\Delta q \bar{c}}{2V_o} + C_{L\delta} \Delta \delta \right) + \Delta q + \frac{g \sin \gamma_o}{m V_o} \left( \Delta \theta - \Delta \alpha \right) - \frac{T_o \cos \alpha_o}{m V_o} \Delta \alpha 
\]

\[
\Delta \dot{q} = \frac{\bar{q} \bar{s} \bar{c}}{I_{yy}} \left( C_{mV} \frac{\Delta V}{V_o} + C_{mA} \Delta \alpha + C_{m\alpha} \frac{\Delta q \bar{c}}{2V_o} + C_{m\alpha} \frac{\Delta q \bar{c}}{2V_o} + C_{m\delta} \Delta \delta \right) 
\]

\[
\Delta \theta = \Delta q 
\]
\[
\Delta \beta = \frac{\bar{q}_o S}{m V_o} \Delta C_Y + \Delta p \sin \alpha_o - \Delta r \cos \alpha_o + \frac{g \cos \theta_o}{V_o} \Delta \phi
\]  
(4.11a)

\[
\Delta \dot{p} - \frac{\Delta \dot{r} I_{xx}}{I_{xx}} = \frac{\bar{q}_o S b C_l}{I_{xx}}
\]  
(4.11b)

\[
\Delta \dot{r} - \Delta \dot{p} \frac{I_{zz}}{I_{xx}} = \frac{\bar{q}_o S b}{I_{zz}} \Delta C_n
\]  
(4.11c)

\[
\Delta \dot{\phi} = \Delta p + \tan \theta_o \Delta r
\]  
(4.11d)

\[
\Delta \psi = \sec \theta_o \Delta r
\]  
(4.11e)

### 4.3.6 - Short Period Approximation

The linear longitudinal equations can be further reduced to the short period approximation by making the following assumptions:

- \( \Delta V = 0 \)
- \( \gamma_o = 0 \) (assume maneuvers occur at level flight)
- \( \frac{T_o \cos \alpha_o}{m V_o} \Delta \alpha \) is small enough to be neglected
- \( C_L = C_{L} (\alpha, q, \delta) \)
- \( C_m = C_{m} (\alpha, \alpha, q, \delta) \)

The previous assumptions applied to Equations 4.10a - 4.10c result in the short period approximation shown in Equations 4.12.

\[
\Delta \dot{\alpha} = - \frac{\bar{q}_o S}{m V_o} \left( C_{L\alpha} \Delta \alpha + C_{Lq} \frac{\Delta \bar{q} \bar{c}}{2 V_o} + C_{L\delta} \Delta \delta \right)
\]  
(4.12a)

\[
\Delta \dot{q} = \frac{\bar{q}_o S \bar{c}}{I_{yy}} \left( C_{m\alpha} \Delta \alpha + C_{m\bar{c}} \frac{\Delta \bar{\alpha} \bar{c}}{2 V_o} + C_{mq} \frac{\Delta \bar{q} \bar{c}}{2 V_o} + C_{m\delta} \Delta \delta \right)
\]  
(4.12b)
4.3.7 - State Space Representation

For modeling, simulation, and control purposes, the state space representation is often the more efficient form for modeling the system. The general form of a state space representation of a system is shown in Equations 4.13 where \( x \) is a vector of states, \( u \) is a vector of inputs, and \( y \) is vector of outputs.

\[
\begin{align*}
\dot{x} &= Ax + Bu \quad (4.13a) \\
y &= Cx + Du \quad (4.13b)
\end{align*}
\]

For the state space representation of the linear aircraft dynamics, Equations 4.10 and 4.11 are the state equations. The states also serve as the outputs of the state space system.

To simplify the state-space representation of the aircraft dynamics, it is useful to write the differential equations in terms of the dimensional stability and control derivatives as shown in Equations 4.14-4.17 (for this project the \( \dot{\alpha} \) parameters were considered a part of the \( \alpha \) parameters).

The final linear state space representation is shown in Equations 4.18.

\[
\begin{align*}
\Delta \dot{V} &= X_v \Delta V + X_a \Delta \alpha + X_q \Delta q - g \cos(\gamma_o) \Delta \theta + X_{\delta_e} \Delta \delta_E \\
\Delta \dot{\alpha} &= Z_\alpha \Delta V + Z_a \Delta \alpha + (1 + Z_q) \Delta q + \frac{g \sin(\gamma_o)}{V_o} + Z_{\delta_e} \Delta \delta_E \\
\Delta \dot{q} &= M_{\theta} \Delta V + M_a \Delta \alpha + M_q \Delta q + M_{\delta_e} \Delta \delta_E \\
X_v &= \frac{-\bar{q} S}{m V_o} C_{p_v} \quad X_a = \frac{-\bar{q} S}{m V_o} C_{p_a} \quad g \cos(\gamma_o) \quad \frac{T_c \sin(\alpha)}{m} \quad X_q = \frac{-\bar{q} S}{m V_o^2} C_{p_q} \quad X_{\delta_e} = \frac{-\bar{q} S}{m V_o} C_{p_{\delta_e}} \\
Z_\alpha &= \frac{-\bar{q} S}{m V_o} C_{l_a} \quad Z_a = \frac{-\bar{q} S}{m V_o} C_{l_a} \quad \frac{g \sin(\gamma_o)}{V_o} \quad \frac{T_c \cos(\alpha)}{m V_o} \quad Z_q = \frac{-\bar{q} S C_{l_q}}{m V_o^2} \quad L_{\delta_e} = \frac{-\bar{q} S}{m V_o} C_{l_{\delta_e}} \\
M_a &= \frac{-\bar{q} S C_{m_a}}{I_y} \quad M_a = \frac{-\bar{q} S C_{m_a}}{I_y} \quad \frac{g \sin(\gamma_o)}{2 V_o I_y} C_{m_q} \quad M_{\delta_e} = \frac{-\bar{q} S C_{m_{\delta_e}}}{I_y} \quad \frac{g \sin(\gamma_o)}{2 V_o I_y} C_{m_{\delta_e}}
\end{align*}
\]
\[ \Delta \dot{\beta} = Y_\beta \Delta \beta + \left( Y_\rho + \sin(\alpha_o) \right) \Delta p + \left( Y_\tau + \cos(\alpha_o) \right) \Delta r + \frac{g \cos \theta_o}{V_o} \Delta \phi + Y_{\delta_\alpha} \Delta \delta_A + Y_{\delta_\tau} \Delta \delta_R \quad (4.16a) \]

\[ \Delta \dot{p} - \frac{I_{xz}}{I_{zz}} \Delta \dot{r} = L_\beta \Delta \beta + L_\rho \Delta p + L_r \Delta r + L_{\delta_\alpha} \Delta \delta_A + L_{\delta_\tau} \Delta \delta_R \quad (4.16b) \]

\[ \Delta \dot{r} - \frac{I_{xz}}{I_{zz}} \Delta \dot{p} = N_\beta \Delta \beta + N_\rho \Delta p + N_r \Delta r + N_{\delta_\alpha} \Delta \delta_A + N_{\delta_\tau} \Delta \delta_R \quad (4.16c) \]

\[ \Delta \phi = \Delta p + \tan(\theta_o) \Delta r \quad (4.16d) \]

\[ \Delta \alpha = -\frac{V_o}{g} \left( Y_\beta \Delta \beta + Y_\rho \Delta p + Y_r \Delta r + Y_{\delta_\alpha} \Delta \delta_A + Y_{\delta_\tau} \Delta \delta_R \right) \quad (4.16e) \]

\[ Y_\beta = \frac{\bar{a}_o S}{m V_o^2} C_{Y_\beta} \quad Y_\rho = \frac{\bar{q}_o Sb^2}{2m V_o^2} C_{Y_\rho} \quad Y_r = \frac{\bar{g}_o Sb^2}{2m V_o^2} C_{Y_r} \quad Y_{\delta} = \frac{\bar{g}_o S}{m V_o} C_{Y_{\delta}} \quad (4.17a) \]

\[ L_\beta = \frac{\bar{a}_o Sb}{I_x} C_{L_\beta} \quad L_\rho = \frac{\bar{q}_o Sb^2}{2 V_o I_x} C_{L_\rho} \quad L_r = \frac{\bar{g}_o Sb^2}{2 V_o I_x} C_{L_r} \quad L_{\delta} = \frac{\bar{g}_o S}{I_x} C_{L_{\delta}} \quad (4.17b) \]

\[ N_\beta = \frac{\bar{a}_o S}{I_Z} C_{N_\beta} \quad N_\rho = \frac{\bar{q}_o Sb^2}{2 V_o I_Z} C_{N_\rho} \quad N_r = \frac{\bar{g}_o Sb^2}{2 V_o I_Z} C_{N_r} \quad N_{\delta} = \frac{\bar{g}_o S}{I_Z} C_{N_{\delta}} \quad (4.17c) \]

\[ \begin{bmatrix} \dot{V} \\ \dot{\alpha} \\ \dot{q} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} X_V & X_\alpha & X_q & -g \cos(\gamma_o) \\ Z_V & Z_\alpha & Z_q & \frac{g \sin(\gamma_o)}{V_o} \end{bmatrix} \begin{bmatrix} \Delta V \\ \Delta \alpha \\ \Delta q \\ \Delta \theta \end{bmatrix} + \begin{bmatrix} X_{\delta_\alpha} \\ Z_{\delta_\alpha} \\ M_{\delta_\alpha} \end{bmatrix} \{ \delta_E \} \quad (4.18a) \]

\[ \begin{bmatrix} \dot{\beta} \\ \dot{p} \\ \dot{r} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} Y_\beta & \sin(\alpha) & -\cos(\alpha) & \frac{g}{V_t} \cos(\theta) \\ L_\beta & L_\rho & L_r & 0 \\ N_\beta & N_\rho & N_r & 0 \\ 0 & 1 & \tan(\theta) & 0 \end{bmatrix} \begin{bmatrix} \Delta \beta \\ \Delta p \\ \Delta r \\ \Delta \phi \end{bmatrix} + \begin{bmatrix} 0 & Y_{\delta_\alpha} & L_{\delta_\alpha} & 0 \\ L_{\delta_\alpha} & L_{\delta_\tau} & N_{\delta_\alpha} & N_{\delta_\tau} \\ N_{\delta_\alpha} & N_{\delta_\tau} & 0 & 0 \end{bmatrix} \{ \delta_R \} \quad (4.18b) \]
This chapter describes the thrust modeling efforts for the motor and ducted fan described in Section 3.1.1. The equations necessary for modeling the thrust of a ducted fan are presented first followed by the static and dynamic thrust models developed for the Phastball.

5.1 - Thrust Equation for a Ducted Fan

The general Reynolds Transport Theorem is expressed by Equation 5.1a [24] where B is some fluid parameter and b is the amount of that parameter per unit mass. If B is the momentum of the fluid, then the Reynolds Transport Theorem can be written as shown in Equation 5.1d (if $B = MV$ then $b = V$). By assuming steady flow (so that the first term on the right hand side of Equation 5.1d can be eliminated) and by recognizing that the time rate of change of momentum for a system is equal to the force acting on the system, Equation 5.1b can be rewritten as Equation 5.2.

$$\frac{d[B_{in}]}{dt} = \frac{\partial}{\partial t} \int_{cv} \rho b dV + \int_{cs} \rho b V \cdot \hat{n} dA$$  \hspace{1cm} (5.1a)

$$B = M = mV$$  \hspace{1cm} (5.1b)

$$b = V$$  \hspace{1cm} (5.1c)

$$\frac{dM_{out}}{dt} = \frac{\partial}{\partial t} \int_{cv} \rho V dV + \int_{cs} \rho V \cdot \hat{n} dA$$  \hspace{1cm} (5.1d)

$$F = \int_{cs} \rho V \cdot \hat{n} dA$$  \hspace{1cm} (5.2)

For a ducted fan, the control volume can be defined as the volume of air directly in front of the fan and the volume of air in the duct as shown in Figure 5.1.

![Figure 5.1: Definition of Control Volume in a Ducted Fan](image)
With this control volume, Equation 5.2 can be simplified to Equation 5.3. The volumetric flow rate in the duct must be constant (Equation 5.4a) so Equation 5.3b can be furthered simplified to Equation 5.4b. Equation 5.4b shows that the thrust from a ducted fan can be defined as a function of the difference between the velocity of the air leaving the duct and the velocity of the air immediately in front of the fan.

\[ T = (\rho V_{out} A_{out}) V_{out} - (\rho V_{in} A_{in}) V_{in} \quad (5.3a) \]

\[ T = Q_{out} V_{out} - Q_{in} V_{in} \quad (5.3b) \]

\[ Q_{out} = Q_{in} \quad (5.4a) \]

\[ T = Q (V_{out} - V_{in}) \quad (5.4b) \]

During ground testing of the ducted fan and motor, it can be assumed that the inlet velocity is zero. The resulting thrust is the static thrust. In flight the inlet velocity is non-zero due to the forward velocity of the aircraft. Due to the non-zero entrance velocity, the thrust produced by the ducted fan decreases from the static condition. The net thrust (static minus the loss due to the non-zero entrance velocity) is called dynamic thrust. Equations 5.5 show the definitions of static and dynamic thrust.

\[ T_{\text{static}} = Q V_{out} \quad (5.5a) \]

\[ T_{\text{dynamic}} = Q (V_{out} - V_{in}) \quad (5.5b) \]

**5.1 - Static Thrust Model**

Static thrust testing for the Phastball was conducted by fellow FCSL student, Zach Mercerusio [21]. A stand to hold and stabilize the duct and motor was attached to a digital scale so that an increase in thrust would produce an increase in the weight read-out from the scale. The throttle input signal was sent from an R/C transmitter to an R/C receiver that was connected via a Y-harness to an electronic speed controller (ESC) and Netburner MOD5213 micro-controller. The MOD5213 recorded the throttle pulse width modulated (PWM) signal from the receiver. The ESC converted the PWM signal from the receiver to a voltage to drive the motor. The thrust read out on the scale was recorded manually. Figure 5.2 and 5.3 show the static thrust testing set-up.
Each test run “consisted of setting the throttle on the transmitter to a user defined value, and reading the corresponding force applied to the scale. This was repeated as the throttle was stepped down from full to zero throttle, so that the thrust level would not be affected by a decreased voltage in the battery as the test was performed” [21]. The results of the static thrust testing was a relationship between the throttle PWM signal and the static thrust. The relationship between throttle PWM and static thrust is given by Equation 5.6, and a plot of the static thrust versus PWM data is shown in Figure 5.4.

\[
T_{\text{static}} = 0.00022x^2 - 0.011x - 0.33
\]  \hspace{1cm} (5.6)
5.2 - Dynamic Thrust Model

By substituting Equation 5.5a into 5.5b, the dynamic thrust can be represented by Equation 5.7.

\[ T_{\text{dynamic}} = T_{\text{static}} - QV_{\text{in}} \]  

(5.7)

For this project, PID maneuvers are assumed to occur during straight and level flight so that \( \Delta V = 0 \). It is also assumed that the throttle input is constant during PID maneuver. With these two assumptions, it is possible to estimate the dynamic thrust during PID maneuvers from Equation 5.7 by using the velocity of the aircraft for \( V_{\text{in}} \) and the static thrust corresponding to the throttle PWM during PID maneuvers.

The steady state velocity during straight and level flight for the Phastball is 95 ft/sec. Using the air density, duct inlet area, and steady state velocity, an estimate for the dynamic thrust produced
by each ducted fan can be estimate. Using Equation 5.7 the dynamic thrust estimate for the Phastball during straight and level flight is 1.51 lbf per ducted fan.
Chapter 6: Aircraft Parameter Identification

This chapter describes the theory and implementation of the parameter identification techniques used in this thesis.

6.1 - Parameter Identification Techniques

6.1.1 - Ordinary Least-Squares Regression

A system that is linear in the parameters can be modeled by the output and measurement equations shown in Equations 6.1 where $y$ is the true output, $X$ is a vector of known values, $\theta$ is a vector of parameters to be estimated, $z$ is the measured output, and $v$ is measurement noise.

\[
\begin{align*}
y & = X \theta \\
z & = X \theta + v
\end{align*}
\] (6.1a, 6.1b)

For the least-squares model, the following assumptions are made about the unknown parameters and the noise:

- $\theta$ is a vector of constant unknown parameters
- $v$ is a random vector of measurement noise
- $v$ is zero mean and uncorrelated with constant variance

With these assumptions, the best estimate of $\theta$ comes from minimizing the weighted sum of differences between the measured and modeled outputs as shown in Equation 6.2.

\[
J(\theta) = \frac{1}{2} (z - X \theta)^T R^{-1} (z - X \theta)
\] (6.2)

In the ordinary least squares method, it is assumed that the weighting matrix $R$ is the identity matrix. With this assumption the cost function reduces to Equation 6.3 with the best estimate of parameters given by Equation 6.4.

\[
J(\theta) = \frac{1}{2} (z - X \theta)(z - X \theta)^T
\] (6.3)

\[
\hat{\theta} = (X^T X)^{-1} X^T z
\] (6.4)
The covariance and standard errors of the estimated parameters are given by Equations 6.5 [25].

$$\text{Cov}(\hat{\theta}) = E\left( (\hat{\theta} - \theta)(\hat{\theta} - \theta)^T \right) = \hat{\sigma}^2 \left( X^T X \right)^{-1} \begin{bmatrix} C_{ij} \end{bmatrix} \quad i,j = 1,2,...,n_p \quad (6.5a)$$

$$\hat{\sigma}^2 = \frac{(z - \hat{z})^T (z - \hat{z})}{(N - n_p)} \quad \text{where} \ \hat{z} = X\hat{\theta} \quad (6.5b)$$

$$s(\hat{\theta}_j) = \sqrt{C_{jj}} \quad j = 1,2,...,n_p \quad (6.5c)$$

The ordinary least-squares method can also be used in the frequency domain by transforming the time domain data into frequency domain data using the Fourier Transform over the frequency domain of interest. For aircraft PID the frequency domain of interest should include the dynamic modes of the aircraft. Equations 6.6 are the equations for ordinary least-squares in the frequency domain with the number of data points being the number of frequencies used in the Fourier Transform. Then the estimate of the unknown parameters can be given by Equation 6.6a [9].

$$\hat{\theta} = \left[ \text{Re}(\tilde{X}'\tilde{X}) \right]^{-1} \text{Re}(\tilde{X}'\tilde{Z}) \quad (6.6a)$$

$$\text{Cov}(\hat{\theta}) = \hat{\sigma}^2 \left[ \text{Re}(\tilde{X}'\tilde{X}) \right]^{-1} \begin{bmatrix} C_{ij} \end{bmatrix} \quad i,j = 1,2,...,n_p \quad (6.6b)$$

$$\hat{\sigma}^2 = \frac{(\tilde{z} - \hat{\tilde{z}})^T (\tilde{z} - \hat{\tilde{z}})}{(m - n_p)} \quad \text{where} \ \hat{\tilde{z}} = X\hat{\theta} \quad (6.6c)$$

$$s(\hat{\theta}_j) = \sqrt{C_{jj}} \quad j = 1,2,...,n_p \quad (6.6d)$$

The covariance and standard errors of the estimated parameters are given by Equations 6.6b-6.6d [9].

**6.1.2 - Equation Error Method**

The equation-error method is a method where the “model matches state time-derivative information from the dynamic system, rather than matching the states or outputs, as in the output-error method” [25]. For aircraft PID, the equation-error method can estimate the non-dimensional stability and control derivatives from the non-dimensional force and moment coefficient models and the dimensional stability and control derivatives from a state space model.
Equations 6.7 are the proposed models for each of the non-dimensional force and moment coefficients.

\[
C_L = C_{L_{\alpha}} \alpha + C_{L_{\delta E}} \delta E + C_{L_o} \tag{6.7a}
\]

\[
C_D = C_{D_{\alpha}} \alpha + C_{D_{\delta E}} \delta E + C_{D_o} \tag{6.7b}
\]

\[
C_m = C_{m_{\alpha}} q\varepsilon \frac{1}{2V_o} + C_{m_{\delta E}} \delta E + C_{m_o} \tag{6.7c}
\]

\[
C_Y = C_{Y_{\beta}} \beta + C_{Y_{\delta R}} \delta R \tag{6.7d}
\]

\[
C_l = C_{l_{\beta}} \beta + C_{l_{\delta A}} \delta A + C_{l_{\delta R}} \delta R + C_{l_o} \tag{6.7e}
\]

\[
C_n = C_{n_{\beta}} \beta + C_{n_{\delta A}} \delta A + C_{n_{\delta R}} \delta R + C_{n_o} \tag{6.7f}
\]

The force and moment coefficients are not directly measured and must be calculated using other flight parameters as shown in Equations 6.8. The presence of the state time derivatives in Equations 6.8 classify this problem as an equation-error problem.

\[
C_D = -C_X \cos \alpha - C_Z \sin \alpha = -\frac{ma_z - T}{qS} \cos \alpha - \frac{ma_x}{qS} \sin \alpha \tag{6.8a}
\]

\[
C_L = -C_Z \cos \alpha + C_X \sin \alpha = -\frac{ma_z}{qS} \cos \alpha + \frac{ma_x - T}{qS} \sin \alpha \tag{6.8b}
\]

\[
C_m = \frac{I_{yy}}{qSc} \left[ \dot{q} + \frac{I_{xx} - I_{zz}}{I_{yy}} (pr) + \frac{I_{xz}}{I_{yy}} (p^2 - r^2) \right] \tag{6.8c}
\]

\[
C_Y = \frac{ma_y}{qS} \tag{6.8d}
\]

\[
C_j = \frac{I_{xx}}{qSb} \left[ \dot{p} - \frac{I_{xz}}{I_{xx}} (pq + \dot{r}) + \frac{I_{zz} - I_{yy}}{I_{xx}} qr \right] \tag{6.8e}
\]

\[
C_n = \frac{I_{zz}}{qSb} \left[ \dot{r} - \frac{I_{xz}}{I_{zz}} (p \dot{q} - qr) + \frac{I_{yy} - I_{xx}}{I_{zz}} pq \right] \tag{6.8f}
\]
The unknown coefficients on the right hand side of Equations 6.7 can be estimated using the ordinary least-squares method. Equations 6.7 can be written in the form of Equation 6.9 where \( \mathbf{Z} \) is a vector containing the non-dimensional force and moment coefficients calculated using Equations 6.8, \( \boldsymbol{\theta} \) is a vector containing the non-dimensional stability and control derivatives to be estimated, \( \mathbf{X} \) is a matrix of regressors, and \( \mathbf{v} \) is a vector of errors. Equations 6.4 or 6.6a can then be used to estimate the values of the non-dimensional stability and control derivatives.

\[
\mathbf{Z} = \mathbf{X} \boldsymbol{\theta} + \mathbf{v}
\]  \(\text{(6.9)}\)

The equation error method can also be used to estimate the dimensional stability and control derivatives. For this project, this technique was used exclusively in the frequency domain. Equation 6.10a shows the state equation in the time domain. Equation 6.10b shows the state equation after applying the Fourier Transform to both sides where \( \mathbf{\tilde{x}} \) and \( \mathbf{\tilde{u}} \) represent the state and input vectors in the frequency domain.

\[
\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \quad \text{(6.10a)}
\]

\[
j \omega \mathbf{\tilde{x}} = \mathbf{A}\mathbf{\tilde{x}} + \mathbf{B}\mathbf{\tilde{u}} \quad \text{(6.10b)}
\]

If the matrices \( \mathbf{X} \) and \( \mathbf{Y} \) are defined according to Equations 6.11, Equation 6.10b can be rewritten as Equation 6.12.

\[
\mathbf{Y} = \begin{bmatrix}
  j \omega_1 \mathbf{\tilde{x}}^T(1) \\
  j \omega_2 \mathbf{\tilde{x}}^T(2) \\
  \vdots \\
  j \omega_m \mathbf{\tilde{x}}^T(m)
\end{bmatrix} \quad \text{(6.11a)}
\]

\[
\mathbf{X} = \begin{bmatrix}
  \mathbf{\tilde{x}}^T(1) & \mathbf{\tilde{u}}^T(1) \\
  \mathbf{\tilde{x}}^T(2) & \mathbf{\tilde{u}}^T(2) \\
  \vdots & \vdots \\
  \mathbf{\tilde{x}}^T(m) & \mathbf{\tilde{u}}^T(m)
\end{bmatrix} \quad \text{(6.11b)}
\]

\[
\mathbf{Y} = \mathbf{X}\boldsymbol{\theta} + \mathbf{\varepsilon} \quad \text{(6.12)}
\]
Equations 6.6 can then be used to find $\hat{\theta}$, the covariance of the estimates, and the standard errors of the estimates.

### 6.1.3 - Maximum Likelihood Estimator

The derivation of the equation for the estimates of the unknown parameters in the least-squares model started by making the following assumptions about Equations 6.1:

- $\theta$ is a vector of constant unknown parameters
- $v$ is a random vector of measurement noise
- $v$ is zero mean and uncorrelated with constant variance

The Fisher model is derived by making the following assumptions about Equations 6.1:

- $\theta$ is a vector of constant unknown parameters
- $v$ is a random vector with probability density $p(v)$

A common estimator for the Fisher model is the maximum likelihood estimator (MLE). The MLE is the value of $\theta$ that maximizes the likelihood function with the likelihood function defined as the probability ($p$) of the output $z$ given the parameters in $\theta$ as shown in Equation 6.13 [10].

$$L(z; \theta) = p(z | \theta) \quad (6.13)$$

If it is assumed that $p(z)$ is Gaussian and $v$ is normally distributed, then the likelihood function can be expressed by Equation 6.14a, and from Equation 6.14a it can be seen that $\hat{\theta}$ will minimize Equation 6.14b [10].

$$L(z; \theta) = \left[ (2\pi)^N | R \right]^{-\frac{1}{2}} \exp \left[ - \frac{1}{2} (z - X\theta)^T R^{-1} (z - X\theta) \right] \quad (6.14a)$$

$$J(\theta) = \frac{1}{2} (z - X\theta) R^{-1} (z - X\theta) \quad (6.14b)$$

The Fisher information matrix is defined by Equation 6.15 (Equation 6.15 will be used in Section 6.1.5) [10].
\[ M = E \left[ \left( \frac{\partial \ln L}{\partial \theta} \right) \left( \frac{\partial \ln L}{\partial \theta} \right)^T \right] = -E \left( \frac{\partial^2 \ln L}{\partial \theta \partial \theta^T} \right) \quad (6.15) \]

6.1.4 - Maximum Likelihood Estimator - Optimization Algorithm

The maximum likelihood estimator is found by minimizing the cost function associated with the negative log-likelihood function. While there are many optimization algorithms that could be used for this problem, the MLE with Newton-Raphson algorithm has been shown to work well for the aircraft parameter estimation problem [26, 27] and was, therefore, used for this project. The Newton-Raphson algorithm is “a root-finding algorithm that uses the first few terms of the Taylor series of a function in the vicinity of a suspected root” [28]. Equation 6.16a shows the first few terms of the Taylor series for a function about the point \( x = x_o + \Delta x \), and the root of the function can then be estimated by setting \( f(x_o + \Delta x) = 0 \). The Newton-Raphson method can be extended to find the change for the previous guess of the location of the maximum or minimum value by finding the root of the derivative of \( f(x_o + \Delta x) \) shown in Equation 6.16b [28, 29].

\[ f(x_o + \Delta x) \approx f(x_o) + f'(x_o) \Delta x \quad (6.16a) \]

\[ f'(x_o + \Delta x) \approx f'(x_o) + f''(x_o) \Delta x \quad (6.16b) \]

\[ \Delta x = -\frac{f'(x)}{f''(x)} \quad (6.16c) \]

The derivation of the parameter estimate update equation starts by considering that the cost function can be written as a function of a steady state value \( \theta_o \) plus a perturbation value \( \Delta \theta \).

Using a second-order Taylor series expansion, the cost function can be written as Equation 6.17a. In order to optimize the cost function, the expression shown in Equation 6.17b must be true [10].

\[ J(\theta_o + \Delta \theta) = J(\theta_o) + \Delta \theta^T \frac{\partial J}{\partial \theta} \bigg|_{\theta = \theta_o} + \Delta \theta^T \frac{\partial^2 J}{\partial \theta \partial \theta^T} \bigg|_{\theta = \theta_o} \Delta \theta \quad (6.17a) \]
Combining Equations 6.17a and 6.17b and solving for $\Delta \theta$ produces Equations 6.18 [10]. Equation 6.18b will be used to derive the parameter estimate update equation as it applies to the MLE in section 6.1.5.

$$\frac{\partial J}{\partial \theta} \bigg|_{\theta = \theta_0} + \frac{\partial^2 J}{\partial \theta \partial \theta^T} \bigg|_{\theta = \theta_0} \Delta \theta = 0 \quad (6.18a)$$

$$\Delta \hat{\theta} = -\left[ \frac{\partial^2 J}{\partial \theta \partial \theta^T} \bigg|_{\theta = \theta_0} \right]^{-1} \frac{\partial J}{\partial \theta} \bigg|_{\theta = \theta_0} \quad (6.18b)$$

The Newton-Raphson optimization algorithm as it applies to aircraft PID is shown in Figure 6.1. The convergence criteria are explained in section 6.1.5.

**6.1.5 - Output Error**

For aircraft PID a linear dynamic system with no process noise (as shown in Equations 6.19) is assumed.

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (6.19a)$$

$$y(t) = Cx(t) + Du(t) \quad (6.19b)$$

$$z(i) = y(i) + v(i) \quad (6.19c)$$

$$v \sim N(0,R) \quad (6.19d)$$

It can be shown that as the sampling rate increases, the probability density $p(v)$ approaches a Gaussian distribution; therefore, the likelihood function for the system described by Equations 6.19 can be written as Equation 6.20 (which was originally introduced as Equation 6.14) [10].

$$L(z;\theta) = \left[ (2\pi)^N |R| \right]^{-\frac{1}{2}} \exp \left[ -\frac{1}{2} (v)^T R^{-1} (v) \right] \quad (6.20)$$
It is computationally more efficient to minimize the negative log-likelihood function than it is to maximize the likelihood function. It can be shown that Equation 6.20 can be rewritten as Equation 6.21a. Therefore the negative log-likelihood function can be written as Equation 6.21b [10].

\[
\mathbb{L}(\mathbf{z}_N; \theta) = \prod_{i=1}^{N} \mathbb{L}(\mathbf{z}(i)|\mathbf{z}_{i-1}; \theta) \tag{6.21a}
\]

\[-\ln(\mathbb{L}(\mathbf{z}_N; \theta)) = \frac{1}{2} \sum_{i=1}^{N} \mathbf{v}^T(i) \mathbf{R}^{-1} \mathbf{v}(i) + \frac{N}{2} \ln|\mathbf{R}| + \frac{Nn}{2} \ln(2\pi) \tag{6.21b}\]

By optimizing Equation 6.21b an estimate of \( \mathbf{R} \) can be found and is expressed by Equation 6.22; and finally, the cost function for the negative log-likelihood function becomes Equation 6.23 [10].

\[
\hat{\mathbf{R}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{v}(i) \mathbf{v}^T(i) \tag{6.22}
\]

\[
\mathbf{J}(\theta) = \frac{1}{2} \mathbf{v}(i) \hat{\mathbf{R}}^{-1}(\mathbf{z} - \mathbf{X}\theta) \mathbf{v}(i) \tag{6.23}
\]

The parameter estimate update function for the system can be found by differentiating Equation 6.23 (shown in Equations 6.24) and using the results in Equation 6.18a (the result is Equation 6.25) [10]. The second term of Equation 6.24a is not used in the parameter estimate update equation for computational efficiency; the modified optimization algorithm is known as the modified Newton-Raphson algorithm or the Gauss-Newton algorithm.

\[
\frac{\partial \mathbf{J}(\theta)}{\partial \theta_j} = -\sum_{i=1}^{N} \frac{\partial \mathbf{v}^T(i)}{\partial \theta} \hat{\mathbf{R}}^{-1} \mathbf{v}(i) = -\sum_{i=1}^{N} \frac{\partial \mathbf{y}^T(i)}{\partial \theta} \hat{\mathbf{R}}^{-1} \mathbf{v}(i) \tag{6.24a}
\]

\[
\frac{\partial^2 \mathbf{J}(\theta)}{\partial \theta_j \partial \theta_k} = \sum_{i=1}^{N} \frac{\partial \mathbf{v}^T(i)}{\partial \theta_j} \hat{\mathbf{R}}^{-1} \mathbf{v}(i) \frac{\partial \mathbf{y}(i)}{\partial \theta_k} - \sum_{i=1}^{N} \frac{\partial^2 \mathbf{y}(i)}{\partial \theta_j \partial \theta_k} \hat{\mathbf{R}}^{-1} \mathbf{v}(i) \tag{6.24b}
\]

\[
\Delta \hat{\theta} = \left[ \sum_{i=1}^{N} \frac{\partial \mathbf{y}^T(i)}{\partial \theta_j} \hat{\mathbf{R}}^{-1} \mathbf{v}(i) \frac{\partial \mathbf{y}(i)}{\partial \theta_k} \right]^{-1} \left[ \sum_{i=1}^{N} \frac{\partial \mathbf{y}^T(i)}{\partial \theta_j} \hat{\mathbf{R}}^{-1} \mathbf{v}(i) \right] \tag{6.25}
\]
Using the Fisher information matrix, the maximum likelihood parameter estimate can be given by Equation 6.26 and will satisfy Equation 6.27; the sensitivity matrix and cost gradient can then be defined by Equations 6.28 [10].

\[
\hat{\theta} = \theta_o - M^{-1}_{\theta=\theta_o} \left[ \frac{\partial J(\theta)}{\partial \theta} \right]_{\theta=\theta_o} \tag{6.26}
\]

\[
Cov(\hat{\theta}) \geq M^{-1}_{\theta=\hat{\theta}} \tag{6.27}
\]

\[
S = \frac{\partial y}{\partial \theta} \tag{6.28a}
\]

\[
g_{\theta=\theta_o} = \sum_{i=1}^{N} \left[ S^T(i) \hat{R}^{-1} v(i) \right]_{\theta=\theta_o} \tag{6.28b}
\]

The stopping/convergence criteria typically used along with the values that have been found to work well for aircraft PID [10] are summarized in Table 6.1.

\begin{table}[h]
\centering
\begin{tabular}{|c|c|}
\hline
\textbf{Stopping Condition} & \textbf{Value of Stopping Condition} \\
\hline
$|\Delta \theta|$ is sufficiently small & $\left\| \frac{\hat{\theta}_k - \hat{\theta}_{k-1}}{\hat{\theta}_k - \hat{\theta}_{k-1}} \right\| < 0.001$ \\
\hline
Changes in $J(\hat{\theta})$ are sufficiently small for multiple iterations & $\left| \frac{J(\hat{\theta}_k) - J(\hat{\theta}_{k-1})}{J(\hat{\theta}_{k-1})} \right| < 0.001$ \\
\hline
Absolute values of the elements of $g$ (the cost gradient) are close to zero & $\left| \left( \frac{\partial J(\theta)}{\partial \theta_j} \right)_{\theta=\hat{\theta}_k} \right| < 0.05 \quad j = 1, 2, \ldots, n_p$ \\
\hline
\end{tabular}
\caption{Convergence Criteria for MLE with Newton-Raphson Optimization}
\end{table}
For this project all four convergence criteria summarized in Table 6.1 must be satisfied or the maximum number of iterations (500) must be reached for the optimization algorithm to end.

### 6.2 - Parameter Identification Applied to the WVU Phastball

This section describes the parameter identification methods that were applied to the WVU Phastball UAV. The parameter estimation efforts applied to the Phastball can be separated into two main categories: off-line and on-line parameter identification algorithms. The off-line PID algorithms use an entire flight segment of data and are implemented after flight testing. On-line PID algorithms are used for parameter identification onboard the aircraft during flight testing. Unlike off-line PID algorithms, on-line PID occurs with the flight data available. While on-line PID is intended for use onboard the aircraft, it can also be conducted on desktop computers for testing of new algorithms and program parameters. For this research effort, the off-line PID efforts were implementations of the MLE in both the time and frequency domains. The on-line PID effort was an implementation of the least-squares estimation in the frequency domain with a version of the algorithm implemented both onboard the Phastball and on a desktop computer (to test different versions of the program in between flight tests).

#### 6.2.1 - System IDentification Programs for AirCraft (SIDPAC)

The System IDentification Programs for AirCraft (SIDPAC) is a “collection of computer programs for aircraft system identification” [30]. SIDPAC is not an automated Matlab® program but rather a collection of Matlab® m-files dedicated to particular tasks in the aircraft parameter identification process (such as axes rotation, equation-error, and output-error PID) allowing the user to quickly assemble a complete program to meet their research needs. Many of SIDPAC’s programs are general enough to be applied to a variety of situations. For example, the output-error program can be applied to the longitudinal or lateral state space formulations of the aircraft dynamics in either the time domain or the frequency domain. It can also be used with the non-linear aircraft
model. In summary, use of SIDPAC gives the user a high degree of freedom and control when creating a PID program to meet research needs.

### 6.2.2 - Pre-Processing of Flight Data

Most of the SIDPAC m-files assume that the data is stored in a matrix named ‘fdata’ (flight data). While the SIDPAC m-files could each be modified to use data in the format of the user’s choosing, assembling the flight data into the fdata matrix is easier. After a flight test, an initial data decoding program was used to decode the data files saved by the onboard computer. When assembling fdata, the GPS velocity components are converted to body axis velocities and the total GPS velocity is used in place of true airspeed. The pitot tube has been damaged during transportation and the pressure sensors for the pitot tubes have failed occasionally making the GPS velocity the more reliable estimate during the course of this project.

The next step in pre-processing is to correct the acceleration measurements. Because the IMU is mounted forward of the CG on the Phastball, rotation about the CG results in the tangential and centripetal accelerations that are measured by the IMU. The acceleration measurements can be corrected for the position of the IMU using Equation 6.29 where \( x_a, y_a, \) and \( z_a \) are the distances from the CG to the accelerometer.

\[
g \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix}_{\text{measured}} = g \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix}_{\text{CG}} + \begin{bmatrix} (q^2 + r^2) & -(pq - \dot{r}) & -(pr + \dot{q}) \\ -(pq + \dot{r}) & (p^2 + r^2) & -(qr - \dot{p}) \\ -(pr - \dot{q}) & -(qr + \dot{p}) & (p^2 + q^2) \end{bmatrix} \begin{bmatrix} x_a \\ y_a \\ z_a \end{bmatrix} \tag{6.29}
\]

Finally, the non-dimensional force and moment coefficients are calculated using Equation 6.8.

### 6.2.3 - Time Domain Off-Line PID

To estimate the non-dimensional stability and control derivatives, the output-error algorithm described in Section 6.1.5 (and implemented in SIDPAC) is used with the starting values estimated using the equation-error algorithm described in Section 6.1.2 (algorithm also implemented in SIDPAC). The models used for the equation-error process are show in Equations 6.6 with the regressors being velocity, angle of attack, and dimensionless pitch rate for the longitudinal dynamics and sideslip angle, dimensionless roll rate, and dimensionless yaw rate for the lateral-directional dynamics. The linearized state-space models derived in Chapter 4 are used for the output-error estimation and are shown again in Equations 6.30. The states and outputs include velocity, angle of
attack, pitch rate, and pitch angle for the longitudinal dynamics and sideslip angle, roll rate, yaw rate, and roll angle for the lateral-directional dynamics.

\[
\begin{bmatrix}
\dot{V} \\
\dot{\alpha} \\
\dot{q} \\
\dot{\theta}
\end{bmatrix} =
\begin{bmatrix}
X_v & X_\alpha & X_q & -g \cos(\gamma_o) \\
Z_v & Z_\alpha & Z_q & g \sin(\gamma_o) \\
M_v & M_\alpha & M_q & 0 \\
0 & 0 & 1 & 0
\end{bmatrix}
\begin{bmatrix}
\Delta V \\
\Delta \alpha \\
\Delta q \\
\Delta \theta
\end{bmatrix}
+ \begin{bmatrix}
X_{\delta_e} \\
Z_{\delta_e} \\
M_{\delta_e} \\
0
\end{bmatrix}\delta_E \quad (6.30a)
\]

\[
\begin{bmatrix}
\dot{\beta} \\
\dot{p} \\
\dot{r} \\
\dot{\phi}
\end{bmatrix} =
\begin{bmatrix}
Y_\beta \sin(\alpha) & -\cos(\alpha) & \frac{g}{V_t} \cos(\theta) & 0 \\
L_\beta & L_p & L_r & 0 \\
N_\beta & N_p & N_r & 0 \\
0 & 1 & \tan(\theta) & 0
\end{bmatrix}
\begin{bmatrix}
\Delta \beta \\
\Delta p \\
\Delta r \\
\Delta \phi
\end{bmatrix}
+ \begin{bmatrix}
0 & Y_{\delta_e} \\
L_{\delta_e} & L_{\delta_e} \\
N_{\delta_e} & N_{\delta_e} \\
0 & 0
\end{bmatrix}\Delta \delta_A \quad (6.30b)
\]

6.2.4 - Frequency Domain Off-Line PID

The same process used for off-line time domain PID (equation error followed by output error with the same regressors, states, and outputs) was used for the off-line frequency domain PID. The only difference was that the data was first converted to the frequency domain using the finite Fourier integral (Equation 6.31a) over the frequency range of interest (0.1-1.5 Hz). It is known that the rigid body dynamics lie in the range 0.1-1.5 Hz [9] and the results from the off-line time domain PID confirmed this for the Phastball (see Chapter 8). Using this frequency range for the evaluation of the finite Fourier integral filters measurement noise and sensor information dropouts.

6.2.5 - Frequency Domain Onboard Real-Time PID

The real-time frequency PID program used the frequency domain formulation of the least-squares to estimate the dimensional stability and control derivatives (the LSE formulation for estimating the dimensional derivatives is given by Equations 6.11 - 6.12 and the estimate of the parameters is given by Equation 6.6). For the longitudinal dynamics, the short period approximation was used for the RTPID program onboard the aircraft (so that the states of interest were the angle of attack and pitch rate). The program was later expanded to use linearized longitudinal model (states of interest were velocity, angle of attack, pitch rate, and pitch angle) and was verified in
simulations on a desktop PC. For the lateral-directional dynamics, the states of interest were the sideslip angle, roll rate, yaw rate, and roll angle. The Fourier transforms of the states and control surface deflections was evaluated using the recursive Fourier transform. The Fourier integral (Equation 6.31a) can be approximated by Equation 6.31b. The discrete Fourier transform can then be expressed by Equation 6.31c. Instead of evaluating the summation at each time step as is shown in Equation 6.31c, it is more efficient to use the recursive Fourier transform which is shown by Equation 6.31d. Like the Fourier integral for the offline frequency domain PID, the recursive Fourier transform is evaluated only over the frequency range of interest (0.1-1.5 Hz).

\[
\tilde{x}(\omega) = \int_0^T x(t) e^{-j\omega t} dt \tag{6.31a}
\]

\[
\tilde{x}(\omega) \approx \Delta t \sum_{i=0}^{N-1} x(i) e^{-j\omega i \Delta t} \tag{6.31b}
\]

\[
X(\omega) = \sum_{i=0}^{N-1} x(i) e^{-j\omega i \Delta t} \tag{6.31c}
\]

\[
X_i(\omega) = X_{i-1}(\omega) + x_i e^{-j\omega i \Delta t} \tag{6.31d}
\]

OBES sends the control surface deflection commands at a rate of 50Hz; however, it is not necessary to update the recursive Fourier transform at every time step because the Nyquist frequency (25 Hz) is significantly greater than the frequency range being used for the recursive Fourier transform [9].
Updating the Fourier transform at 25Hz instead of at every time step, saves computations but does not adversely affect the Fourier transform which can be seen in Figure 6.1 which compares the Fourier transform of the elevator deflection during a multisine maneuver evaluated using the Finite Fourier Integral and the recursive Fourier transform evaluated at 25Hz.
Chapter 7: Input Design

This chapter describes the requirements of good input design for aircraft parameter identification as well as the design and implementation of the input maneuvers for the Phastball PID.

7.1 - Requirements for Input Design

“The objective of input design for dynamic model identification is to excite the dynamic system so that the data contains sufficient information for accurate modeling”[10]. When designing an input maneuver for parameter estimation, there are three main factors to be considered:

• maneuver time
• control surface to be moved
• input forms

In order for the derived linear aircraft model to be valid, the amplitudes of the flow angles, angular rates, and translational accelerations must be small. According to Klein and Morelli [10], the following constraints shown in Table 7.1 are appropriate.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Acceptable Amplitude Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha, \beta$</td>
<td>$&lt; \pm 5^\circ$</td>
</tr>
<tr>
<td>$p, q, r$</td>
<td>$&lt; \pm 20 \text{deg/sec}$</td>
</tr>
<tr>
<td>$a_x, a_y, a_z$</td>
<td>$&lt; \pm 0.3g$</td>
</tr>
</tbody>
</table>

It is also desirable to move multiple control surfaces simultaneously to reduce the amount of time required to gather data for PID. Finally, for real-parameter identification the inputs should ideally be small enough so that the aircraft response is similar to the aircraft’s response to typical turbulence [31].

7.2 - Input Maneuver for Phastball PID

Because the straight leg portion of a typical flight testing lap typically does not exceed 600 meters, the available time for a PID maneuver (including time to set-up at a steady state condition, perform the maneuver, and allowing the dynamics play-out) is approximately 12-15 seconds. This
constraint on time limits the maneuvers that can be used for PID. Doublet inputs are short enough to be executed during the straight leg, but past experience with piloted doublets on the Phastball shows that it is difficult to remain within the restrictions on angle of attack, sideslip, angular rates, and translational acceleration described previously. Separate doublets also need to be performed to model the longitudinal and lateral-directional dynamics increasing the amount of flights dedicated to PID. Multisines can be executed during straight leg of a flight and have good information content for low input amplitudes (discussed further in Section 7.3). Additionally, they can be performed on multiple surfaces simultaneously (if they are designed to be mutually orthogonal) reducing the amount of time needed for PID. For these reasons, multisines were the input used for this project.

7.3 - Multisine Design

A multisine is a sum of sinusoids with varying frequencies, amplitudes, and phase angles and can be expressed by Equation 7.1a where \( A \) is the amplitude of the component sinusoid, \( T \) is the time length of the maneuver, \( \phi \) is the phase angle for the component sinusoid, and \( M \) is the number of harmonic frequencies to be used [10]. For the multisine input given by Equation 7.1a, the peak factor can be expressed by Equation 7.1b where \( N \) is the number of points in \( u \). The peak factor (PF) is a ratio of the maximum input deflection to the input energy. Multisines with low peak factors produce good information content for small amplitudes, and small input amplitudes help ensure the aircraft will stay around the steady state condition.

\[
\begin{align*}
  u &= \sum_{k=1}^{M} A_k \cos \left( \frac{2\pi k t (i)}{T} + \phi_k \right) \\
  PF(u) &= \frac{\max(u) - \min(u)}{\sqrt{(u^T u) / N}}
\end{align*}
\]

Morelli presents a method for designing multiple orthogonal multisine inputs [31] and implements the method in a Matlab® program included with SIDPAC. The inputs created using the method presented in [31] are mutually orthogonal in both the time and frequency domains (making the inputs suitable for time or frequency domain PID methods) and have minimized peak factors. The only inputs that must be provided to the SIDPAC m-file to create multiple multisines are the
number of inputs, the time length of the maneuver, the frequency band of interest, and the maximum input amplitudes.

For the SIDPAC program that creates the orthogonal multisines, the inputs are the length of time for the maneuver, the maximum surface deflection, and the frequency range of interest. For the Phastball a time length of eight seconds was chosen so that the pilot would have enough time to ensure the aircraft was flying straight and level before implementing the maneuver. Experimentation showed that a maximum input amplitude of 2.5° was suitable. Input amplitudes smaller than 2.5° resulted in flow angle measurements that were indistinguishable from noise. A multisine with a maximum deflection of 1.5° resulted in an average signal-to-noise ratio for the angle of attack of 3 compared to the average signal-to-noise ratio of 8 from a multisine with 2.5° of deflection. The frequency range of interest was 0.1-2 Hz to cover the rigid body dynamics of the Phastball. Frequencies and phase angles for each multisine are then chosen to minimize the peak factors of the multisines. The multisine inputs designed using the SIDPAC program are shown in Figure 7.1 with the final design parameters displayed in Table 7.2.

![Multisine Inputs](image)

**Figure 7.1: Designed Multisines**
Table 7.2: Properties of Optimized Multisines

<table>
<thead>
<tr>
<th>Elevator</th>
<th>Aileron</th>
<th>Rudder</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase Angles (rad)</td>
<td>Frequencies (Hz)</td>
<td>Phase Angles (rad)</td>
</tr>
<tr>
<td>-1.121</td>
<td>0.500</td>
<td>0.703</td>
</tr>
<tr>
<td>-1.242</td>
<td>0.875</td>
<td>1.363</td>
</tr>
<tr>
<td>2.604</td>
<td>1.250</td>
<td>-2.545</td>
</tr>
<tr>
<td>0.524</td>
<td>1.625</td>
<td>1.713</td>
</tr>
<tr>
<td>2.442</td>
<td>2.000</td>
<td></td>
</tr>
</tbody>
</table>

Peak Factor: 1.213  1.213  1.112

Two factors must be considered for the implementation of the multisines: the trim control surface deflection and the wind conditions. The designed multisines are perturbation deflections to be implemented around the trim surface deflection. If the trim surface deflections for the Phastball remained constant, the multisine could be added to the trim deflection to arrive at the total control surface deflection for the maneuver. However, the trim surface deflections for the Phastball change with the wind condition and with the pilot. Wind also affects the implementation of the multisine in a second way. In a traditional OBES maneuver, the flight computer has complete control of the control surfaces. If the wind changes during the implementation of the multisine, the aircraft could drift too far from the steady state condition. Implementing the multisines as pilot-in-the-loop addresses both the changing trim conditions and the changing wind. For a pilot-in-the-loop implementation, the flight computer adds the perturbation deflection (Figure 7.1) to the control surface input command sent by the R/C transmitter (ground pilot). If the ground pilot is “hands-off”, the multisine is injected around the trim condition. If wind causes the aircraft to drift too far from straight and level flight, the R/C pilot can command small surface deflections to bring the aircraft back to straight and level. While the pilot brings the aircraft back to straight and level flight, OBES continues to implement the multisine around the R/C pilot’s input. Figure 7.2 shows a flow chart to illustrate the pilot in the loop concept.
Experience with pilot-in-the-loop multisines has shown that very little pilot input is needed to keep the Phastball straight and level during the multisine assuming nominal weather conditions. Figure 7.3 shows the designed multisine and the multisine with pilot in the loop adjustments from a flight on October 3, 2012. On that day, the wind was coming from the west which is perpendicular to the straight legs of the flight path. Adjustments to compensate for the wind can be seen on both the elevator and aileron. The bias on the rudder is the rudder trim.
Chapter 8: Results

This chapter presents the results of the off-line and onboard PID programs. The results of the programs were obtained using flight data from the four flight testing session (flight testing information summarized in Table 8.1).

8.1 - Summary of PID Maneuvers

Flight testing for the PID effort occurred from the fall of 2011 through the fall of 2012 using the two Phastballs with flight computers. Many flights for parameter estimation were flown during this time to support configuration changes of the Phastballs. The results for the most recent Phastball configuration (which has remained constant since August of 2012) are presented in this section. Table 8.1 summarizes the flights used for the most recent PID effort. Each row in the table corresponds to one flight (flights from the same day are presented in separate rows). The third column denotes the flight number, and the fourth column displays the number of OBES activations for that flight. Table 8.2 displays the number of maneuvers used for each phase of the PID effort.

Table 8.1: Summary of PID Flights

<table>
<thead>
<tr>
<th>Date</th>
<th>Plane</th>
<th>Flight Number</th>
<th>Number of OBES Activations</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>8/2/2012</td>
<td>Blue</td>
<td>2</td>
<td>5</td>
<td>Offline PID</td>
</tr>
<tr>
<td>8/2/2012</td>
<td>Blue</td>
<td>4</td>
<td>8</td>
<td>Offline PID</td>
</tr>
<tr>
<td>9/5/2012</td>
<td>Blue</td>
<td>2</td>
<td>6</td>
<td>Offline PID</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>Blue</td>
<td>2</td>
<td>7</td>
<td>Off-line, Onboard, Real Time PID (Desktop)</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>Blue</td>
<td>3</td>
<td>7</td>
<td>Off-line, Onboard, Real Time PID (Desktop)</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>Blue</td>
<td>4</td>
<td>7</td>
<td>Off-line, Onboard, Real Time PID (Desktop)</td>
</tr>
<tr>
<td>10/27/2012</td>
<td>Blue</td>
<td>1</td>
<td>7</td>
<td>Off-line, Onboard, Real Time PID (Desktop)</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 8.2: Number of Maneuvers Used

<table>
<thead>
<tr>
<th>Use</th>
<th># of Activations Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Off-line PID</td>
<td>40</td>
</tr>
<tr>
<td>Onboard PID</td>
<td>28</td>
</tr>
<tr>
<td>Real Time PID (desktop)</td>
<td>28</td>
</tr>
</tbody>
</table>

1 The Green Phastball was being used for other research objectives and was not flown for PID flights during this time period.
8.2.1 - Off-line Time Domain PID

8.2.1.1 – Longitudinal Results

The developed off-line time domain PID algorithm was run using the flight data from each of the forty PID maneuvers presented in Table 8.1. The MLE portion of the program failed due to poorly scaled matrices for twelve of the original forty maneuvers. Of the remaining twenty-eight maneuvers, ten were outliers (based on the estimates of the non-dimensional derivatives) and were eliminated. Finally, the remaining eighteen maneuvers were divided for estimation and validation. Thirteen maneuvers were used for estimation, and five maneuvers were used for validation. The estimates of the non-dimensional derivatives were averaged to arrive at the final estimate. Table 8.3 shows the estimated value of the non-dimensional parameters along with the 95% confidence interval. A weakness of estimating stability and control derivatives from flight data, is that drag coefficients are difficult to measure. The estimated drag values are presented in Table 8.3, but are the least reliable estimates.

Table 8.3: Time Domain - Longitudinal Non-Dimensional Derivative Estimates

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Nominal</th>
<th>Lower Bound</th>
<th>Upper Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{p_{\alpha}}$</td>
<td>-0.214</td>
<td>-0.506</td>
<td>0.078</td>
</tr>
<tr>
<td>$C_{p_{q}}$</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$C_{D_{\delta E}}$</td>
<td>-0.169</td>
<td>-0.485</td>
<td>0.147</td>
</tr>
<tr>
<td>$C_{L_{\alpha}}$</td>
<td>3.309</td>
<td>3.079</td>
<td>3.539</td>
</tr>
<tr>
<td>$C_{L_{q}}$</td>
<td>41.937</td>
<td>33.069</td>
<td>50.805</td>
</tr>
<tr>
<td>$C_{b_{\alpha}}$</td>
<td>1.787</td>
<td>1.604</td>
<td>1.970</td>
</tr>
<tr>
<td>$C_{m_{\alpha}}$</td>
<td>-0.446</td>
<td>-0.475</td>
<td>-0.418</td>
</tr>
<tr>
<td>$C_{m_{q}}$</td>
<td>-4.096</td>
<td>-5.621</td>
<td>-2.571</td>
</tr>
<tr>
<td>$C_{m_{\delta E}}$</td>
<td>-0.560</td>
<td>-0.590</td>
<td>-0.530</td>
</tr>
</tbody>
</table>

The state space matrices, and subsequently the eigenvalues, were derived using the final estimate of the non-dimensional derivatives. Equation 8.16 shows the state-space model and Table 8.4 shows the final eigenvalue estimate (derived from Equation 8.16) as well as the eigenvalues for the thirteen maneuvers used for estimation.
\[
\begin{bmatrix}
\dot{V} \\
\dot{\alpha} \\
\dot{q} \\
\dot{\theta}
\end{bmatrix} =
\begin{bmatrix}
0.0275 & 51.7097 & 0 & -31.5757 \\
0 & -3.2482 & 0.7628 & -0.0650 \\
0 & -16.7744 & -0.8751 & 0 \\
0 & 0 & 1 & 0
\end{bmatrix}
\begin{bmatrix}
V \\
\alpha \\
q \\
\theta
\end{bmatrix} +
\begin{bmatrix}
15.9835 \\
-1.7784 \\
-21.0543 \\
0
\end{bmatrix}
\Delta \delta_E \quad (8.16)
\]

Table 8.4: Time Domain – Longitudinal Dynamics Characteristics

<table>
<thead>
<tr>
<th>Date</th>
<th>Flight #</th>
<th>Activation #</th>
<th>Eigenvalues</th>
<th>Damping Ratio</th>
<th>Frequency (rad/sec)</th>
<th>Frequency (Hz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>8/2/2012</td>
<td>2</td>
<td>2</td>
<td>-1.12 ± 2.952i</td>
<td>0.355</td>
<td>3.157</td>
<td>0.502</td>
</tr>
<tr>
<td>8/2/2012</td>
<td>2</td>
<td>3</td>
<td>-1.566 ± 1.998i</td>
<td>0.617</td>
<td>2.538</td>
<td>0.404</td>
</tr>
<tr>
<td>8/2/2012</td>
<td>2</td>
<td>4</td>
<td>-3.038 ± 5.786i</td>
<td>0.465</td>
<td>6.535</td>
<td>1.040</td>
</tr>
<tr>
<td>8/2/2012</td>
<td>4</td>
<td>2</td>
<td>-4.461 ± 4.861i</td>
<td>0.676</td>
<td>6.598</td>
<td>1.050</td>
</tr>
<tr>
<td>8/2/2012</td>
<td>4</td>
<td>5</td>
<td>-1.784 ± 2.138i</td>
<td>0.641</td>
<td>2.784</td>
<td>0.443</td>
</tr>
<tr>
<td>9/5/2012</td>
<td>2</td>
<td>5</td>
<td>-4.104 ± 5.347i</td>
<td>0.609</td>
<td>6.740</td>
<td>1.073</td>
</tr>
<tr>
<td>9/5/2012</td>
<td>2</td>
<td>6</td>
<td>-2.029 ± 1.732i</td>
<td>0.761</td>
<td>2.668</td>
<td>0.425</td>
</tr>
<tr>
<td>9/5/2012</td>
<td>2</td>
<td>8</td>
<td>-1.859 ± 1.356i</td>
<td>0.808</td>
<td>2.301</td>
<td>0.366</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>2</td>
<td>4</td>
<td>-2.937 ± 4.266i</td>
<td>0.567</td>
<td>5.179</td>
<td>0.824</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>3</td>
<td>4</td>
<td>-2.981 ± 5.221i</td>
<td>0.496</td>
<td>6.013</td>
<td>0.957</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>3</td>
<td>6</td>
<td>-2.026 ± 2.074i</td>
<td>0.699</td>
<td>2.899</td>
<td>0.461</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>4</td>
<td>3</td>
<td>-1.905 ± 2.104i</td>
<td>0.671</td>
<td>2.838</td>
<td>0.452</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>4</td>
<td>4</td>
<td>-1.717 ± 1.712i</td>
<td>0.708</td>
<td>2.425</td>
<td>0.386</td>
</tr>
<tr>
<td><strong>Final Estimate:</strong></td>
<td></td>
<td></td>
<td><strong>-2.003 ± 3.003i</strong></td>
<td><strong>0.555</strong></td>
<td><strong>3.609</strong></td>
<td><strong>0.574</strong></td>
</tr>
</tbody>
</table>

Table 8.5 shows the mean residual and the standard deviation of the residual for velocity, angle of attack, pitch rate, pitch angle, and acceleration from the validation maneuvers.

Table 8.5: Summary of Residuals

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mean Residual</th>
<th>Residual Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>V</td>
<td>-4.62E-02</td>
<td>2.08E-01</td>
</tr>
<tr>
<td>( \alpha )</td>
<td>-2.65E-03</td>
<td>3.09E-03</td>
</tr>
<tr>
<td>q</td>
<td>7.87E-03</td>
<td>7.69E-03</td>
</tr>
<tr>
<td>( \theta )</td>
<td>6.67E-02</td>
<td>8.25E-02</td>
</tr>
<tr>
<td>( a_z )</td>
<td>-1.57E-16</td>
<td>2.90E-16</td>
</tr>
</tbody>
</table>

Figures 8.1 -8.4 show the measured flight data with the modeled outputs for two of the validation maneuvers.
Figure 8.1 Validation - 8/2/2012 - Flight 2 - Maneuver 5

Figure 8.2 Validation - 8/2/2012 - Flight 2 - Maneuver 5
Figure 8.3: Validation - 8/2/2012 - Flight 4 - Maneuver 9

Figure 8.4: Validation - 8/2/2012 - Flight 4 - Maneuver 9
8.2.1.2 – Lateral Results

The developed off-line time domain PID algorithm was run using the flight data from each of the forty PID maneuvers presented in Table 8.1. The MLE portion of the program failed due to poorly scaled matrices for seventeen of the original forty maneuvers. Of the remaining twenty-three maneuvers, six were outliers (based on the estimates of the non-dimensional derivatives) and were eliminated. Finally, the remaining seventeen maneuvers were divided for estimation and validation. Thirteen maneuvers were used for estimation, and four maneuvers were used for validation. The estimates of the non-dimensional derivatives were averaged to arrive at the final estimate of the non-dimensional derivatives. Table 8.6 shows the estimated value of the non-dimensional parameters along with the 95% confidence interval.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Nominal</th>
<th>Lower Bound</th>
<th>Upper Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{Y\beta}$</td>
<td>-0.271</td>
<td>-0.344</td>
<td>-0.197</td>
</tr>
<tr>
<td>$C_{Y\varphi}$</td>
<td>0.217</td>
<td>-0.354</td>
<td>0.789</td>
</tr>
<tr>
<td>$C_{Y\delta}$</td>
<td>0.078</td>
<td>0.001</td>
<td>0.156</td>
</tr>
<tr>
<td>$C_{l\beta}$</td>
<td>-0.028</td>
<td>-0.038</td>
<td>-0.018</td>
</tr>
<tr>
<td>$C_{l\varphi}$</td>
<td>-0.157</td>
<td>-0.219</td>
<td>-0.095</td>
</tr>
<tr>
<td>$C_{l\delta}$</td>
<td>0.100</td>
<td>0.010</td>
<td>0.191</td>
</tr>
<tr>
<td>$C_{l\varphi}$</td>
<td>-0.066</td>
<td>-0.085</td>
<td>-0.047</td>
</tr>
<tr>
<td>$C_{l\delta}$</td>
<td>0.013</td>
<td>0.003</td>
<td>0.023</td>
</tr>
<tr>
<td>$C_{n\beta}$</td>
<td>0.047</td>
<td>0.036</td>
<td>0.058</td>
</tr>
<tr>
<td>$C_{n\varphi}$</td>
<td>-0.083</td>
<td>-0.151</td>
<td>-0.014</td>
</tr>
<tr>
<td>$C_{n\delta}$</td>
<td>-0.226</td>
<td>-0.325</td>
<td>-0.127</td>
</tr>
<tr>
<td>$C_{n\varphi}$</td>
<td>-0.009</td>
<td>-0.032</td>
<td>0.013</td>
</tr>
<tr>
<td>$C_{n\delta}$</td>
<td>-0.020</td>
<td>-0.034</td>
<td>-0.005</td>
</tr>
</tbody>
</table>

The state space matrices, and subsequently the eigenvalues, were derived using the final estimate of the non-dimensional derivatives. Equation 8.17 shows the state-space model and Table 8.7 shows the final eigenvalue estimate (derived from Equation 8.17) as well as the eigenvalues for the thirteen maneuvers used for estimation.
\[
\begin{bmatrix}
\dot{\beta} \\
\dot{p} \\
\dot{r} \\
\dot{\phi}
\end{bmatrix} =
\begin{bmatrix}
-0.2692 & 0.0511 & -0.9903 & 0.3382 \\
-25.9967 & -6.2392 & 3.5113 & 0 \\
8.7436 & -0.9330 & -1.6817 & 0 \\
0 & 1 & 0.0527 & 0
\end{bmatrix}
\begin{bmatrix}
\beta \\
p \\
r \\
\phi
\end{bmatrix} +
\begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\delta_a \\
\delta_r
\end{bmatrix}
\] (5.17)

Table 8.7: Time Domain - Lateral Dynamics Characteristics

<table>
<thead>
<tr>
<th>Date</th>
<th>Flight #</th>
<th>Activation #</th>
<th>Eigenvalues – Dutch Roll</th>
<th>Damping Ratio</th>
<th>Frequency (rad/sec)</th>
<th>Frequency (Hz)</th>
<th>Eigenvalues - Roll</th>
</tr>
</thead>
<tbody>
<tr>
<td>8/2/2012</td>
<td>2</td>
<td>2</td>
<td>-0.955 ± 3.633i</td>
<td>0.254</td>
<td>3.756</td>
<td>0.598</td>
<td>-6.256</td>
</tr>
<tr>
<td>8/2/2012</td>
<td>2</td>
<td>4</td>
<td>-1.128 ± 3.83i</td>
<td>0.282</td>
<td>3.992</td>
<td>0.635</td>
<td>-4.726</td>
</tr>
<tr>
<td>8/2/2012</td>
<td>2</td>
<td>5</td>
<td>-1.119 ± 3.621i</td>
<td>0.295</td>
<td>3.790</td>
<td>0.603</td>
<td>-8.187</td>
</tr>
<tr>
<td>8/2/2012</td>
<td>4</td>
<td>3</td>
<td>-1.039 ± 3.915i</td>
<td>0.257</td>
<td>4.050</td>
<td>0.645</td>
<td>-7.972</td>
</tr>
<tr>
<td>8/2/2012</td>
<td>4</td>
<td>5</td>
<td>-1.143 ± 4.493i</td>
<td>0.247</td>
<td>4.636</td>
<td>0.738</td>
<td>-6.653</td>
</tr>
<tr>
<td>8/2/2012</td>
<td>4</td>
<td>7</td>
<td>-1.344 ± 4.314i</td>
<td>0.297</td>
<td>4.519</td>
<td>0.719</td>
<td>-6.071</td>
</tr>
<tr>
<td>8/2/2012</td>
<td>4</td>
<td>9</td>
<td>-1.124 ± 3.812i</td>
<td>0.283</td>
<td>3.974</td>
<td>0.632</td>
<td>-9.176</td>
</tr>
<tr>
<td>9/5/2012</td>
<td>2</td>
<td>3</td>
<td>-1.0699 ± 4.4161i</td>
<td>0.23545</td>
<td>4.544</td>
<td>0.723</td>
<td>-8.006</td>
</tr>
<tr>
<td>9/5/2012</td>
<td>2</td>
<td>5</td>
<td>-1.2288 ± 3.6867i</td>
<td>0.3162</td>
<td>3.886</td>
<td>0.618</td>
<td>-6.221</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>2</td>
<td>3</td>
<td>-0.8268 ± 4.024i</td>
<td>0.201</td>
<td>4.108</td>
<td>0.658</td>
<td>-7.699</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>2</td>
<td>6</td>
<td>-1.2364 ± 3.8226i</td>
<td>0.30773</td>
<td>4.018</td>
<td>0.639</td>
<td>-8.586</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>3</td>
<td>4</td>
<td>-0.9946 ± 3.7504i</td>
<td>0.25633</td>
<td>3.880</td>
<td>0.617</td>
<td>-7.926</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>4</td>
<td>4</td>
<td>1.296 ± 4.42i</td>
<td>0.276</td>
<td>4.599</td>
<td>0.732</td>
<td>-7.077</td>
</tr>
<tr>
<td>Final Estimate</td>
<td></td>
<td></td>
<td><strong>-0.7179 ± 3.750</strong></td>
<td><strong>0.188</strong></td>
<td><strong>3.818</strong></td>
<td><strong>0.6077</strong></td>
<td><strong>-6.781</strong></td>
</tr>
</tbody>
</table>

Table 8.8 shows the mean residual and the standard deviation of the residual for the sideslip angle, roll rate, yaw rate, roll angle, and acceleration from the validation maneuvers.

Table 8.8: Summary of Residuals

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mean Residual</th>
<th>Residual Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\beta)</td>
<td>-1.73E-03</td>
<td>1.79E-03</td>
</tr>
<tr>
<td>(p)</td>
<td>-2.33E-03</td>
<td>1.03E-02</td>
</tr>
<tr>
<td>(r)</td>
<td>1.08E-04</td>
<td>6.07E-03</td>
</tr>
<tr>
<td>(\phi)</td>
<td>-2.61E-02</td>
<td>2.68E-02</td>
</tr>
<tr>
<td>(a_y)</td>
<td>-1.40E-17</td>
<td>3.25E-17</td>
</tr>
</tbody>
</table>

Figures 8.5-8.8 show the measured flight data with the modeled outputs for two of the validation maneuvers.
Figure 8.5: Validation - 8/2/2012 - Flight 4 - Maneuver 2

Figure 8.6: Validation - 8/2/2012 - Flight 4 - Maneuver 2
Figure 8.7: Validation - 8/2/2012 - Flight 4 - Maneuver 8

Figure 8.8: Validation - 8/2/2012 - Flight 4 - Maneuver 8
8.2.2 - Off-line Frequency Domain PID

8.2.2.1 – Longitudinal Results

The developed off-line frequency domain PID algorithm was run using the flight data from each of the forty PID maneuvers presented in Table 8.1. The MLE portion of the program failed due to poorly scaled matrices for six of the original forty maneuvers. Of the remaining thirty-four maneuvers, eleven were outliers (based on the estimates of the non-dimensional derivatives) and were eliminated. Finally, the remaining twenty-three maneuvers were divided for estimation and validation. Twelve maneuvers were used for estimation, and eleven maneuvers were used for validation. The estimates of the non-dimensional derivatives were averaged to arrive at the final estimate of the non-dimensional derivatives. Table 8.9 shows the estimated value of the non-dimensional parameters along with the 95% confidence interval. A weakness of estimating stability and control derivatives from flight data, is that drag coefficients are difficult to measure. The estimated drag values are presented in Table 8.9, but are the least reliable estimates.

Table 8.9: Frequency Domain - Longitudinal Non-Dimensional Derivative Estimates

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Nominal</th>
<th>Lower Bound</th>
<th>Upper Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{D_{\alpha}}$</td>
<td>0.086</td>
<td>-0.05</td>
<td>0.222</td>
</tr>
<tr>
<td>$C_{D_{\phi}}$</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$C_{D_{q}}$</td>
<td>-0.203</td>
<td>-0.69</td>
<td>-0.337</td>
</tr>
<tr>
<td>$C_{L_{\alpha}}$</td>
<td>4.582</td>
<td>4.228</td>
<td>4.937</td>
</tr>
<tr>
<td>$C_{L_{q}}$</td>
<td>-14.206</td>
<td>-32.360</td>
<td>3.949</td>
</tr>
<tr>
<td>$C_{m_{\alpha}}$</td>
<td>0.383</td>
<td>0.023</td>
<td>0.743</td>
</tr>
<tr>
<td>$C_{m_{\phi}}$</td>
<td>-0.313</td>
<td>-0.359</td>
<td>-0.267</td>
</tr>
<tr>
<td>$C_{m_{q}}$</td>
<td>-13.184</td>
<td>-15.953</td>
<td>-10.415</td>
</tr>
<tr>
<td>$C_{m_{\phi q}}$</td>
<td>-0.775</td>
<td>-0.831</td>
<td>-0.718</td>
</tr>
</tbody>
</table>

The state space matrices, and subsequently the eigenvalues, were derived using the final estimate of the non-dimensional derivatives. Equation 5.18 shows the state-space model and Table 8.10 shows the final eigenvalue estimate as well as the eigenvalues for the twelve maneuvers used for estimation.
\[
\begin{bmatrix}
\dot{V} \\
\dot{\alpha} \\
\dot{q} \\
\dot{\theta}
\end{bmatrix} =
\begin{bmatrix}
0.3881 & 23.3029 & 0 & -31.5757 \\
0 & -4.5157 & 1.0804 & -0.0650 \\
0 & -11.7676 & -2.8168 & 0 \\
0 & 0 & 1 & 0
\end{bmatrix}
\begin{bmatrix}
V \\
\alpha \\
q \\
\theta
\end{bmatrix} +
\begin{bmatrix}
19.2229 \\
-0.3807 \\
-29.1149 \\
0
\end{bmatrix}
\Delta \delta_e \quad (5.18)
\]

\textbf{Table 8.10: Frequency Domain - Longitudinal Dynamics Characteristics}

<table>
<thead>
<tr>
<th>Date</th>
<th>Flight #</th>
<th>Activation #</th>
<th>Eigenvalues</th>
<th>Damping Ratio</th>
<th>Frequency (rad/sec)</th>
<th>Frequency (Hz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>8/2/12</td>
<td>2</td>
<td>4</td>
<td>-4.13 ± 2.21i</td>
<td>0.88228</td>
<td>4.68522</td>
<td>0.74568</td>
</tr>
<tr>
<td>8/2/12</td>
<td>4</td>
<td>3</td>
<td>-3.93 ± 3.35i</td>
<td>0.7395</td>
<td>5.3177</td>
<td>0.84634</td>
</tr>
<tr>
<td>9/5/12</td>
<td>2</td>
<td>3</td>
<td>-4.33 ± 3.36i</td>
<td>0.7895</td>
<td>5.47861</td>
<td>0.87195</td>
</tr>
<tr>
<td>9/5/12</td>
<td>2</td>
<td>5</td>
<td>-4.16 ± 3.55i</td>
<td>0.76076</td>
<td>5.47109</td>
<td>0.87075</td>
</tr>
<tr>
<td>9/5/12</td>
<td>2</td>
<td>8</td>
<td>-2.96 ± 3.46i</td>
<td>0.65008</td>
<td>4.55709</td>
<td>0.72528</td>
</tr>
<tr>
<td>10/3/12</td>
<td>2</td>
<td>1</td>
<td>-3.51 ± 4.18i</td>
<td>0.64354</td>
<td>5.4591</td>
<td>0.86884</td>
</tr>
<tr>
<td>10/3/12</td>
<td>2</td>
<td>3</td>
<td>-5.50 ± 2.83i</td>
<td>0.8879</td>
<td>6.15206</td>
<td>0.97913</td>
</tr>
<tr>
<td>10/3/12</td>
<td>2</td>
<td>4</td>
<td>-3.68 ± 3.08i</td>
<td>0.76704</td>
<td>4.79758</td>
<td>0.76356</td>
</tr>
<tr>
<td>10/3/12</td>
<td>2</td>
<td>6</td>
<td>-4.39 ± 2.19i</td>
<td>0.8946</td>
<td>4.90709</td>
<td>0.78099</td>
</tr>
<tr>
<td>10/3/12</td>
<td>3</td>
<td>7</td>
<td>-3.81 ± 4.23i</td>
<td>0.66952</td>
<td>5.69699</td>
<td>0.9067</td>
</tr>
<tr>
<td>10/3/12</td>
<td>4</td>
<td>4</td>
<td>-4.57 ± 4.05i</td>
<td>0.74878</td>
<td>6.1086</td>
<td>0.97221</td>
</tr>
<tr>
<td>10/3/12</td>
<td>4</td>
<td>6</td>
<td>-4.63 ± 3.62i</td>
<td>0.78817</td>
<td>5.88008</td>
<td>0.93584</td>
</tr>
</tbody>
</table>

\textbf{Final Estimate:} \(-3.68 ± 3.48i\) \begin{bmatrix} 0.71 \\ 5.12 \end{bmatrix} \begin{bmatrix} 0.82 \end{bmatrix}

Table 8.11 shows the mean residual and the standard deviation of the residual for velocity, angle of attack, pitch rate, pitch angle, and acceleration from the validation maneuvers.

\textbf{Table 8.11: Summary of Residuals}

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mean Residual</th>
<th>Residual Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>V</td>
<td>-3.70E+00</td>
<td>3.64E+00</td>
</tr>
<tr>
<td>(\alpha)</td>
<td>-4.54E-04</td>
<td>9.48E-04</td>
</tr>
<tr>
<td>q</td>
<td>8.17E-03</td>
<td>5.24E-03</td>
</tr>
<tr>
<td>(\theta)</td>
<td>6.44E-02</td>
<td>9.03E-02</td>
</tr>
<tr>
<td>(a_z)</td>
<td>1.06E-16</td>
<td>1.48E-16</td>
</tr>
</tbody>
</table>

Figures 8.9-8.16 show the measured flight data with the modeled outputs for two of the validation maneuvers. Figures 8.9, 8.10, 8.13, and 8.14 show the results in the frequency domain, and Figures 8.11, 8.12, 8.15, and 8.16 show the results in the time domain.
Figure 8.9: Validation - 8/2/2012 - Flight 2 - Maneuver 2

Figure 8.10: Validation - 8/2/2012 - Flight 2 - Maneuver 2
Figure 8.11: Validation - 8/2/2012 - Flight 2 - Maneuver 2

Figure 8.12: Validation - 8/2/2012 - Flight 2 - Maneuver 2
Figure 8.13: Validation - 8/2/2012 - Flight 2 - Maneuver 3

Figure 8.14: Validation - 8/2/2012 - Flight 2 - Maneuver 3
Figure 8.15: Validation - 8/2/2012 - Flight 2 - Maneuver 3

Figure 8.16 Validation - 8/2/2012 - Flight 2 - Maneuver 3
8.2.2.2 – Lateral-Directional Results

The developed off-line frequency domain PID algorithm was run using the flight data from each of the forty PID maneuvers presented in Table 8.1. The MLE portion of the program failed due to poorly scaled matrices for four of the original forty maneuvers. Of the remaining thirty-six maneuvers, ten were outliers (based on the estimates of the non-dimensional derivatives) and were eliminated. Finally, the remaining twenty-six maneuvers were divided for estimation and validation. Twenty maneuvers were used for estimation, and six maneuvers were used for validation. The estimates of the non-dimensional derivatives were averaged to arrive at the final estimate of the non-dimensional derivatives. Table 8.12 shows the estimated value of the non-dimensional parameters along with the 95% confidence interval.

Table 8.12: Frequency Domain - Lateral Non-Dimensional Derivative Estimates

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Nominal</th>
<th>Lower Bound</th>
<th>Upper Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{Y\beta}$</td>
<td>-0.276</td>
<td>-0.299</td>
<td>-0.254</td>
</tr>
<tr>
<td>$C_{Y\gamma}$</td>
<td>0.058</td>
<td>-0.097</td>
<td>0.213</td>
</tr>
<tr>
<td>$C_{Y\delta_{sl}}$</td>
<td>0.045</td>
<td>0.019</td>
<td>0.071</td>
</tr>
<tr>
<td>$C_{i\beta}$</td>
<td>-0.024</td>
<td>-0.026</td>
<td>-0.022</td>
</tr>
<tr>
<td>$C_{i\gamma}$</td>
<td>-0.173</td>
<td>-0.183</td>
<td>-0.163</td>
</tr>
<tr>
<td>$C_{i\delta}$</td>
<td>0.109</td>
<td>0.095</td>
<td>0.122</td>
</tr>
<tr>
<td>$C_{i\delta_{A}}$</td>
<td>-0.072</td>
<td>-0.076</td>
<td>-0.069</td>
</tr>
<tr>
<td>$C_{i\delta_{R}}$</td>
<td>0.012</td>
<td>0.009</td>
<td>0.014</td>
</tr>
<tr>
<td>$C_{\eta\beta}$</td>
<td>0.050</td>
<td>0.049</td>
<td>0.052</td>
</tr>
<tr>
<td>$C_{\eta\gamma}$</td>
<td>-0.088</td>
<td>-0.099</td>
<td>-0.078</td>
</tr>
<tr>
<td>$C_{\eta\delta}$</td>
<td>-0.145</td>
<td>-0.158</td>
<td>-0.132</td>
</tr>
<tr>
<td>$C_{\eta\delta_{A}}$</td>
<td>-0.018</td>
<td>-0.021</td>
<td>-0.014</td>
</tr>
<tr>
<td>$C_{\eta\delta_{R}}$</td>
<td>-0.018</td>
<td>-0.021</td>
<td>-0.016</td>
</tr>
</tbody>
</table>

The state space matrices, and subsequently the eigenvalues, were derived using the final estimate of the non-dimensional derivatives. Equation 8.19 shows the state-space model and Table 8.13 shows the final eigenvalue estimate as well as the eigenvalues for the twelve maneuvers used for estimation.
Table 8.13: Frequency Domain - Lateral Dynamics Characteristics

<table>
<thead>
<tr>
<th>Date</th>
<th>Flight #</th>
<th>Activation #</th>
<th>Eigenvalues – Dutch Roll</th>
<th>Damping Ratio</th>
<th>Frequency (rad/sec)</th>
<th>Frequency (Hz)</th>
<th>Eigenvalues - Roll</th>
</tr>
</thead>
<tbody>
<tr>
<td>8/2/2012</td>
<td>2</td>
<td>2</td>
<td>-1.097 ± 3.679i</td>
<td>0.286</td>
<td>3.839</td>
<td>0.611</td>
<td>-4.613</td>
</tr>
<tr>
<td>8/2/2012</td>
<td>2</td>
<td>3</td>
<td>-1.252 ± 3.372i</td>
<td>0.348</td>
<td>3.596</td>
<td>0.572</td>
<td>-3.584</td>
</tr>
<tr>
<td>8/2/2012</td>
<td>2</td>
<td>4</td>
<td>-1.069 ± 3.974i</td>
<td>0.260</td>
<td>4.115</td>
<td>0.655</td>
<td>-6.880</td>
</tr>
<tr>
<td>8/2/2012</td>
<td>4</td>
<td>2</td>
<td>-0.86 ± 3.949i</td>
<td>0.212</td>
<td>4.041</td>
<td>0.643</td>
<td>-7.849</td>
</tr>
<tr>
<td>8/2/2012</td>
<td>4</td>
<td>3</td>
<td>-0.992 ± 3.879i</td>
<td>0.248</td>
<td>4.004</td>
<td>0.637</td>
<td>-7.980</td>
</tr>
<tr>
<td>8/2/2012</td>
<td>4</td>
<td>4</td>
<td>-1.014 ± 4.370i</td>
<td>0.226</td>
<td>4.486</td>
<td>0.714</td>
<td>-8.167</td>
</tr>
<tr>
<td>8/2/2012</td>
<td>4</td>
<td>8</td>
<td>-0.880 ± 3.427i</td>
<td>0.249</td>
<td>3.539</td>
<td>0.563</td>
<td>-7.524</td>
</tr>
<tr>
<td>8/2/2012</td>
<td>4</td>
<td>9</td>
<td>-0.956 ± 3.841i</td>
<td>0.242</td>
<td>3.959</td>
<td>0.630</td>
<td>-8.493</td>
</tr>
<tr>
<td>9/5/2012</td>
<td>2</td>
<td>5</td>
<td>-1.34 ± 3.882i</td>
<td>0.327</td>
<td>4.107</td>
<td>0.654</td>
<td>-6.762</td>
</tr>
<tr>
<td>9/5/2012</td>
<td>2</td>
<td>7</td>
<td>-1.045 ± 4.123i</td>
<td>0.246</td>
<td>4.253</td>
<td>0.677</td>
<td>-8.008</td>
</tr>
<tr>
<td>9/5/2012</td>
<td>2</td>
<td>8</td>
<td>-0.789 ± 3.190i</td>
<td>0.240</td>
<td>3.286</td>
<td>0.523</td>
<td>-5.811</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>2</td>
<td>2</td>
<td>-0.740 ± 4.051i</td>
<td>0.179</td>
<td>4.117</td>
<td>0.655</td>
<td>-7.268</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>2</td>
<td>4</td>
<td>-0.600 ± 3.613i</td>
<td>0.164</td>
<td>3.662</td>
<td>0.583</td>
<td>-7.569</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>2</td>
<td>6</td>
<td>-0.899 ± 3.909i</td>
<td>0.224</td>
<td>4.011</td>
<td>0.638</td>
<td>-8.479</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>3</td>
<td>3</td>
<td>-0.745 ± 3.218i</td>
<td>0.226</td>
<td>3.303</td>
<td>0.526</td>
<td>-8.324</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>3</td>
<td>5</td>
<td>-0.283 ± 4.182i</td>
<td>0.068</td>
<td>4.192</td>
<td>0.667</td>
<td>-8.474</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>3</td>
<td>7</td>
<td>-0.357 ± 4.094i</td>
<td>0.087</td>
<td>4.109</td>
<td>0.654</td>
<td>-9.801</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>4</td>
<td>2</td>
<td>-0.976 ± 4.112i</td>
<td>0.231</td>
<td>4.226</td>
<td>0.673</td>
<td>-7.588</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>4</td>
<td>4</td>
<td>-0.496 ± 3.931i</td>
<td>0.125</td>
<td>3.962</td>
<td>0.631</td>
<td>-10.785</td>
</tr>
<tr>
<td>10/3/2012</td>
<td>4</td>
<td>6</td>
<td>-0.977 ± 3.936i</td>
<td>0.241</td>
<td>4.055</td>
<td>0.645</td>
<td>-7.296</td>
</tr>
</tbody>
</table>

Final Estimate: -0.795 ± 3.761i

Table 8.14 shows the mean residual and the standard deviation of the residual for the sideslip angle, roll rate, yaw rate, roll angle, and acceleration from the validation maneuvers.
Table 8.14: Summary of Residuals

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mean Residual</th>
<th>Residual Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta$</td>
<td>-2.53E-03</td>
<td>1.85E-03</td>
</tr>
<tr>
<td>$p$</td>
<td>-1.18E-03</td>
<td>6.78E-03</td>
</tr>
<tr>
<td>$r$</td>
<td>1.22E-03</td>
<td>3.15E-03</td>
</tr>
<tr>
<td>$\phi$</td>
<td>-2.47E-02</td>
<td>3.43E-02</td>
</tr>
<tr>
<td>$a_y$</td>
<td>5.96E-18</td>
<td>1.67E-17</td>
</tr>
</tbody>
</table>

Figures 8.17-8.24 show the measured flight data with the modeled outputs for two of the validation maneuvers. Figures 8.17, 8.18, 8.21, and 8.22 show the results in the frequency domain, and Figure 8.19, 8.20, 8.23, and 8.24 show the results in the time domain.

Figure 8.17 Validation - 8/2/2012 - Flight 2 - Maneuver 5
Validation – Frequency Domain Results

Validation – Time Domain Results

Figure 8.18: Validation - 8/2/2012 - Flight 2 - Maneuver 5

Figure 8.19: Validation - 8/2/2012 - Flight 2 - Maneuver 5
Validation – Time Domain Results

Validation – Frequency Domain Results

Figure 8.20 Validation - 8/2/2012 - Flight 2 - Maneuver 5

Figure 8.21: Validation – 9/5/2012 - Flight 2 - Maneuver 6
Figure 8.22: Validation – 9/5/2012 - Flight 2 - Maneuver 6

Figure 8.23: Validation – 9/5/2012 - Flight 2 - Maneuver 6
Validation – Time Domain Results

Roll Angle (rad)

Time (s)

Ay (g)

Figure 8.24: Validation – 9/5/2012 - Flight 2 - Maneuver 6
8.2.3 – Off-line Time Domain & Frequency Domain PID Comparison

8.2.3.1 – Longitudinal

Four methods were used to compare the time domain and frequency domain results. First, the number of maneuvers that were badly scaled and caused the MLE to fail was compared. The results of this analysis are shown in Table 8.15.

Table 8.15: Comparison of Number of Longitudinal Maneuvers Available for Estimation

<table>
<thead>
<tr>
<th></th>
<th>Time Domain</th>
<th>Frequency Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Available Maneuvers</td>
<td>40</td>
<td>40</td>
</tr>
<tr>
<td>Maneuvers Not Used</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MLE Failed</td>
<td>12</td>
<td>6</td>
</tr>
<tr>
<td>Removed – Outlier</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td>Maneuvers Used</td>
<td>18</td>
<td>23</td>
</tr>
<tr>
<td>Estimation</td>
<td>13</td>
<td>12</td>
</tr>
<tr>
<td>Validation</td>
<td>5</td>
<td>11</td>
</tr>
</tbody>
</table>

Next the estimates of the non-dimensional derivatives and the standard error of the estimates were compared. Table 8.12 compares the non-dimensional derivative estimates and standard errors. Estimates that show greater than a 50% change between the time domain and the frequency domain are highlighted in the table. The table also shows that the time domain estimates have a smaller standard error and consequently smaller 95% confidence interval.

Table 8.16: Comparison of Non-Dimensional Derivative Estimates

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Time Domain Estimate</th>
<th>Frequency Domain Estimate</th>
<th>Time Domain Standard Error</th>
<th>Frequency Domain Standard Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{L_\alpha}$</td>
<td>3.309</td>
<td>4.582</td>
<td>0.115</td>
<td>0.177</td>
</tr>
<tr>
<td>$C_{L_q}$ *</td>
<td>41.937</td>
<td>-14.206</td>
<td>4.434</td>
<td>9.077</td>
</tr>
<tr>
<td>$C_{\alpha E}$ *</td>
<td>1.787</td>
<td>0.383</td>
<td>0.091</td>
<td>0.180</td>
</tr>
<tr>
<td>$C_{m_\alpha}$</td>
<td>-0.446</td>
<td>-0.313</td>
<td>0.014</td>
<td>0.023</td>
</tr>
<tr>
<td>$C_{m_q}$ *</td>
<td>-4.096</td>
<td>-13.184</td>
<td>0.763</td>
<td>1.385</td>
</tr>
<tr>
<td>$C_{m_E}$</td>
<td>-0.560</td>
<td>-0.775</td>
<td>0.015</td>
<td>0.028</td>
</tr>
</tbody>
</table>

* Indicates Estimates that vary significantly (greater than a 50% change) between time and frequency domain.
While, the state-space models are derived using the estimates of the non-dimensional derivatives, they are also compared because they are used as truth-values for the real-time PID which estimates the dimensional derivatives. The state space models from the time domain and frequency domain off-line estimation are shown in Equation (8.20) and (8.21), respectively. The parameters that vary as a result of the differences in the non-dimensional coefficient estimates are highlighted in red.

\[
\begin{bmatrix}
\dot{V} \\
\dot{\alpha} \\
\dot{q} \\
\dot{\theta}
\end{bmatrix} =
\begin{bmatrix}
0.0275 & 51.7097 & 0 & -31.5757 \\
0 & -3.2482 & 0.7628 & -0.0650 \\
0 & -16.7744 & -0.8751 & 0 \\
0 & 0 & 1 & 0
\end{bmatrix}
\begin{bmatrix}
V \\
\alpha \\
q \\
\theta
\end{bmatrix}
+ 
\begin{bmatrix}
15.9835 \\
-1.7784 \\
-21.0543 \\
0
\end{bmatrix} \{\Delta \delta_E\} 
\tag{8.20}
\]

\[
\begin{bmatrix}
\dot{V} \\
\dot{\alpha} \\
\dot{q} \\
\dot{\theta}
\end{bmatrix} =
\begin{bmatrix}
0.3881 & 23.3029 & 0 & -31.5757 \\
0 & -4.5157 & 1.0804 & -0.0650 \\
0 & -11.7676 & -2.8168 & 0 \\
0 & 0 & 1 & 0
\end{bmatrix}
\begin{bmatrix}
V \\
\alpha \\
q \\
\theta
\end{bmatrix}
+ 
\begin{bmatrix}
19.2229 \\
-0.3807 \\
-29.1149 \\
0
\end{bmatrix} \{\Delta \delta_E\} 
\tag{8.21}
\]

Finally, the mean residual and standard deviation of the residuals from the validations maneuvers are compared in Table 8.17. The lower value for the mean residual is highlighted for comparison.

\textit{Table 8.17: Comparison of Residuals}

<table>
<thead>
<tr>
<th>Parameter</th>
<th>TD - Mean Residual</th>
<th>FD - Mean Residual</th>
<th>TD - Residual Standard Deviation</th>
<th>FD - Residual Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>V</td>
<td>-4.62E-02</td>
<td>-3.70E+00</td>
<td>2.08E-01</td>
<td>1.85E-03</td>
</tr>
<tr>
<td>\alpha</td>
<td>-2.65E-03</td>
<td>-4.54E-04</td>
<td>3.09E-03</td>
<td>6.78E-03</td>
</tr>
<tr>
<td>q</td>
<td>7.87E-03</td>
<td>8.17E-03</td>
<td>7.69E-03</td>
<td>3.15E-03</td>
</tr>
<tr>
<td>\theta</td>
<td>6.67E-02</td>
<td>6.44E-02</td>
<td>8.25E-02</td>
<td>3.43E-02</td>
</tr>
<tr>
<td>a_z</td>
<td>-1.57E-16</td>
<td>1.06E-16</td>
<td>2.90E-16</td>
<td>1.67E-17</td>
</tr>
</tbody>
</table>

70
8.2.3.2 – Lateral-Directional

Four methods were used to compare the time domain and frequency domain results. First, the number of maneuvers that were badly scaled and caused the MLE to fail were compared. The results of this analysis are shown in Table 8.18.

Table 8.18: Comparison of Number of Lateral Maneuvers Available for Estimation

<table>
<thead>
<tr>
<th></th>
<th>Time Domain</th>
<th>Frequency Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Available Maneuvers</td>
<td>40</td>
<td>40</td>
</tr>
<tr>
<td>Maneuvers Not Used</td>
<td>23</td>
<td>12</td>
</tr>
<tr>
<td>MLE Failed</td>
<td>17</td>
<td>4</td>
</tr>
<tr>
<td>Removed – Outlier</td>
<td>6</td>
<td>10</td>
</tr>
<tr>
<td>Maneuvers Used</td>
<td>17</td>
<td>26</td>
</tr>
<tr>
<td>Estimation</td>
<td>13</td>
<td>20</td>
</tr>
<tr>
<td>Validation</td>
<td>4</td>
<td>6</td>
</tr>
</tbody>
</table>

Next the estimates of the non-dimensional derivatives and the standard error of the estimates were compared. Table 8.19 compares the non-dimensional derivative estimates and standard errors. Estimates that show greater than a 50% change between the time domain and the frequency domain (and) and are highlighted in the table. The table also shows that the frequency domain estimates have a smaller standard error and consequently smaller 95% confidence interval.

Table 8.19: Comparison of Non-Dimensional Derivative Estimates

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Time Domain Estimate</th>
<th>Frequency Domain Estimate</th>
<th>Time Domain Standard Error</th>
<th>Frequency Domain Standard Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{Yg}$</td>
<td>-0.271</td>
<td>-0.276</td>
<td>0.037</td>
<td>0.011</td>
</tr>
<tr>
<td>$C_{Yr}$</td>
<td>0.217</td>
<td>0.058</td>
<td>0.286</td>
<td>0.077</td>
</tr>
<tr>
<td>$C_{Yp}$</td>
<td>0.078</td>
<td>0.045</td>
<td>0.039</td>
<td>0.013</td>
</tr>
<tr>
<td>$C_{lp}$</td>
<td>-0.028</td>
<td>-0.024</td>
<td>0.005</td>
<td>0.001</td>
</tr>
<tr>
<td>$C_{lp}$</td>
<td>-0.157</td>
<td>-0.173</td>
<td>0.031</td>
<td>0.005</td>
</tr>
<tr>
<td>$C_{n}$</td>
<td>0.100</td>
<td>0.109</td>
<td>0.045</td>
<td>0.007</td>
</tr>
<tr>
<td>$C_{n}$</td>
<td>-0.066</td>
<td>-0.072</td>
<td>0.010</td>
<td>0.002</td>
</tr>
<tr>
<td>$C_{ng}$</td>
<td>0.013</td>
<td>0.012</td>
<td>0.005</td>
<td>0.001</td>
</tr>
<tr>
<td>$C_{ng}$</td>
<td>0.047</td>
<td>0.050</td>
<td>0.006</td>
<td>0.001</td>
</tr>
<tr>
<td>$C_{np}$</td>
<td>-0.083</td>
<td>-0.088</td>
<td>0.034</td>
<td>0.005</td>
</tr>
</tbody>
</table>
**Parameter** | **Time Domain Estimate** | **Frequency Domain Estimate** | **Time Domain Standard Error** | **Frequency Domain Standard Error**
--- | --- | --- | --- | ---
$C_{n_s}$ | -0.226 | -0.145 | 0.050 | 0.006
$C_{n_{sb}}$ | -0.009 | -0.018 | 0.011 | 0.002
$C_{n_{sr}}$ | -0.020 | -0.018 | 0.007 | 0.001

* Indicates Estimates that vary significantly (greater than a 50% change) between time and frequency domain.

While, the state-space models are derived using the estimates of the non-dimensional derivatives, they are also compared because they are used as truth-values for the real-time PID which estimates the dimensional derivatives. The state space models from the time domain and frequency domain off-line estimation are shown in Equation (5.22) and (5.23), respectively. The parameters that vary as a result of the differences in the non-dimensional coefficient estimates are highlighted in red.

\[
\begin{bmatrix}
\beta \\
p \\
r \\
\phi
\end{bmatrix} =
\begin{bmatrix}
-0.2692 & 0.0511 & -0.9903 & 0.3382 \\
-25.9967 & -6.2392 & 3.5113 & 0 \\
8.7436 & -0.9330 & -1.6817 & 0 \\
0 & 1 & 0.0527 & 0
\end{bmatrix}
\begin{bmatrix}
\beta \\
p \\
r \\
\phi
\end{bmatrix} +
\begin{bmatrix}
0 & 0 & 0.0446 \\
-22.0408 & -7.3250 & 4.2387 & 0 \\
9.5833 & -1.0739 & -1.0711 & 0
\end{bmatrix}
\begin{bmatrix}
\delta_a \\
\delta_r
\end{bmatrix}
\]

(8.22)

\[
\begin{bmatrix}
\beta \\
p \\
r \\
\phi
\end{bmatrix} =
\begin{bmatrix}
-0.2935 & 0.0511 & -0.992 & 0.3608 \\
-22.0408 & -7.3250 & 4.2387 & 0 \\
9.5833 & -1.0739 & -1.0711 & 0 \\
0 & 1 & 0.0527 & 0
\end{bmatrix}
\begin{bmatrix}
\beta \\
p \\
r \\
\phi
\end{bmatrix} +
\begin{bmatrix}
0 & 0 & 0.0476 \\
-22.0408 & -7.3250 & 4.2387 & 0 \\
9.5833 & -1.0739 & -1.0711 & 0
\end{bmatrix}
\begin{bmatrix}
\delta_a \\
\delta_r
\end{bmatrix}
\]

(8.23)

Finally, the mean residual and standard deviation of the residuals from the validations maneuvers are compared in Table 8.20. The lower value for the mean residual is highlighted for comparison.

**Table 8.20: Comparison of Residuals**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>TD - Mean Residual</th>
<th>FD - Mean Residual</th>
<th>TD - Residual Standard Deviation</th>
<th>FD - Residual Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta$</td>
<td>-1.73E-03</td>
<td>-2.53E-03</td>
<td>1.79E-03</td>
<td>1.85E-03</td>
</tr>
<tr>
<td>$p$</td>
<td>-2.33E-03</td>
<td>-1.18E-03</td>
<td>1.03E-02</td>
<td>6.78E-03</td>
</tr>
<tr>
<td>$r$</td>
<td>1.08E-04</td>
<td>1.22E-03</td>
<td>6.07E-03</td>
<td>3.15E-03</td>
</tr>
<tr>
<td>$\phi$</td>
<td>-2.61E-02</td>
<td>-2.47E-02</td>
<td>2.68E-02</td>
<td>3.43E-02</td>
</tr>
<tr>
<td>$ay$</td>
<td>-1.40E-17</td>
<td>5.96E-18</td>
<td>3.25E-17</td>
<td>1.67E-17</td>
</tr>
</tbody>
</table>
8.3 - **Real-Time Parameter Identification**

As described in Chapter 1, the real-time PID can be implemented onboard the aircraft or in a simulation on a desktop PC. This section will first present the graphical results from the real-time parameter identification from onboard the Phastball and then a quantitative analysis of the real-time estimates including average error between the real-time and off-line estimates, the average settling time, and the average standard error of the real-time estimates (the average standard error had to be done using desktop computer simulations).

8.3.1 - **Longitudinal Results**

The original implementation of the onboard program used the short period approximation and did not estimate the confidence interval. The desktop PC version of the real-time PID program was updated to use the linearized longitudinal equations but not the short period approximation and to estimate the confidence interval. Since the real-time program was updated, flight testing has not taken place, so the error analysis and analysis of the confidence interval was done using the desktop computer real-time PID program.

Figures 8.25 and 8.26 show the onboard estimates from October 27, 2012, Flight 1, Activation1. The red band represents the off-line time domain parameter estimate, and the black band represents the off-line frequency domain estimate. Figures 8.27 and 8.28 are the results of the desktop PC real-time parameter estimation for October 3, 2012, Flight 2 Activation 1. Again, the red band represents the off-line time domain parameter estimate, and the black band represents the off-line time frequency domain estimate. The blue error bars represent the 95% confidence interval of the real-time estimate.
Figure 8.25: Onboard Z Parameter Estimates - October 27, 2012 - Flight 1 - Activation 1

Figure 8.26: Onboard M Parameter Estimates - October 27, 2012 - Flight 1 - Activation 1
Figure 8.27 - Onboard Z Parameter Estimates - October 3, 2012 - Flight 2 - Activation 1

Figure 8.28 - Onboard M Parameter Estimates - October 3, 2012 - Flight 2 - Activation 1
To analyze the error value for the real-time estimates was evaluated using the parameter estimate at \( t=8 \) sec. The value at this time was used for the error analysis, because at eight seconds, the multisine was complete but the Fourier transform (which acts as a memory) did not yet include data without an input excitation. Table 8.21 shows the difference between the off-line time domain parameter estimate and the real-time estimate and the off-line frequency domain parameter estimate and the real-time estimate (the lower error for each parameter is highlighted). Table 8.22 shows the number of times the real-time estimate converged to within the confidence interval of either the off-line time domain or off-line frequency domain estimate, the average settling time, and the average standard error of the real-time estimates.

**Table 8.21 - Average Error between Real-Time Parameter Estimate and Off-line Estimates**

<table>
<thead>
<tr>
<th></th>
<th>( Z_{\alpha} )</th>
<th>( Z_q )</th>
<th>( Z_{de} )</th>
<th>( M_{\alpha} )</th>
<th>( M_q )</th>
<th>( M_{de} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>TD</td>
<td>1.954</td>
<td>-0.216</td>
<td>0.101</td>
<td>1.981</td>
<td>10.916</td>
<td>10.730</td>
</tr>
<tr>
<td>FD</td>
<td>0.686</td>
<td>0.583</td>
<td>5.909</td>
<td>-0.669</td>
<td>-2.611</td>
<td>2.669</td>
</tr>
<tr>
<td>Average Error</td>
<td>1.954</td>
<td>0.686</td>
<td>-0.216</td>
<td>1.981</td>
<td>10.916</td>
<td>10.730</td>
</tr>
</tbody>
</table>

**Table 8.22 - Average Standard Error, Average Settling Time, and Convergence Statistics**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Average Standard Error</th>
<th>Average Settling Time (sec)</th>
<th># of Times Estimate Converged to Off-line Estimate</th>
</tr>
</thead>
<tbody>
<tr>
<td>( Z_{\alpha} )</td>
<td>1.18</td>
<td>6</td>
<td>13</td>
</tr>
<tr>
<td>( Z_q )</td>
<td>0.27</td>
<td>5</td>
<td>19</td>
</tr>
<tr>
<td>( Z_{de} )</td>
<td>0.96</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>( M_{\alpha} )</td>
<td>5.05</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>( M_q )</td>
<td>1.05</td>
<td>6</td>
<td>14</td>
</tr>
<tr>
<td>( M_{de} )</td>
<td>3.87</td>
<td>4</td>
<td>18</td>
</tr>
</tbody>
</table>
8.3.2 - Lateral-Directional Results

The original implementation of the onboard program did not estimate the confidence interval, but the desktop PC version of the real-time PID program was updated to use the linearized to estimate the confidence interval. Since the real-time program was updated, flight testing has not taken place, so the error analysis and analysis of the confidence interval was done using the desktop computer real-time PID program.

Figures 8.29 - 8.33 show the onboard estimates from October 27, 2012, Flight 1, Activation 1. The red band represents the off-line time domain parameter estimate, and the black band represents the off-line frequency domain estimate. Figures 8.34-8.38 are the results of the desktop PC real-time parameter estimation for October 3, 2012, Flight 2 Activation 1. Again, the red band represents the off-line time domain parameter estimate, and the black band represents the off-line time frequency domain estimate. The blue error bars represent the 95% confidence interval of the real-time estimate.

Figure 8.29 - $Y$ Parameter Estimates - October 27, 2012 - Flight 1 - Activation 1
Figure 8.30 - L Parameter Estimates (stability) - October 27, 2012 - Flight 1 - Activation 1

Figure 8.31 - L Parameter Estimates (control) - October 27, 2012 - Flight 1 - Activation 1
Figure 8.32 - N Parameter Estimates (stability) - October 27, 2012 - Flight 1 - Activation 1

Figure 8.33 - N Parameter Estimates (control) - October 27, 2012 - Flight 1 - Activation 1
Figure 8.34 - Y Parameter Estimates - October 3 2012 - Flight 2 - Activation 1

Figure 8.35 - L Parameter Estimates (stability) - October 3 2012 - Flight 2 - Activation 1
Figure 8.36 - L Parameter Estimates (control) - October 3 2012 - Flight 2 - Activation 1

Figure 8.37 - N Parameter Estimates (stability) - October 3 2012 - Flight 2 - Activation 1
To analyze the error value for the real-time estimates was evaluated using the parameter estimate at \( t=8 \) sec. The value at this time was used for the error analysis, because at eight seconds, the multisine was complete but the Fourier transform (which acts as a memory) does not yet include data without an input excitation. Tables 8.23-8.25 shows the difference between the off-line time domain parameter estimate and the real-time estimate and the off-line frequency domain parameter estimate and the real-time estimate (the lower error for each parameter is highlighted). Table 8.26 shows the the number of times the real-time estimate converged to within the confidence interval of either the off-line time domain or off-line frequency domain estimate, the average settling time, and the average standard error of the real-time estimates.

**Table 8.23 - Average Error between Real-Time Parameter Estimate and Off-line Estimates - Y Parameters**

<table>
<thead>
<tr>
<th></th>
<th>Y(_{\beta})</th>
<th>Y(_{r})</th>
<th>Y(_{dr})</th>
</tr>
</thead>
<tbody>
<tr>
<td>TD</td>
<td>-0.089</td>
<td>0.253</td>
<td>0.336</td>
</tr>
<tr>
<td>FD</td>
<td>-0.095</td>
<td>0.518</td>
<td>-0.828</td>
</tr>
</tbody>
</table>

**Figure 8.38 - N Parameter Estimates (control) - October 3 2012 - Flight 2 - Activation 1**

To analyze the error value for the real-time estimates was evaluated using the parameter estimate at \( t=8 \) sec. The value at this time was used for the error analysis, because at eight seconds, the multisine was complete but the Fourier transform (which acts as a memory) does not yet include data without an input excitation. Tables 8.23-8.25 shows the difference between the off-line time domain parameter estimate and the real-time estimate and the off-line frequency domain parameter estimate and the real-time estimate (the lower error for each parameter is highlighted). Table 8.26 shows the the number of times the real-time estimate converged to within the confidence interval of either the off-line time domain or off-line frequency domain estimate, the average settling time, and the average standard error of the real-time estimates.
Table 8.24 - Average Error between Real-Time Parameter Estimate and Off-line Estimates - L Parameters

<table>
<thead>
<tr>
<th></th>
<th>Lbeta</th>
<th>Lp</th>
<th>Lr</th>
<th>Lda</th>
<th>Ldr</th>
</tr>
</thead>
<tbody>
<tr>
<td>TD</td>
<td>-13.880</td>
<td>-1.300</td>
<td>-2.910</td>
<td>-4.120</td>
<td>-10.586</td>
</tr>
</tbody>
</table>

Table 8.25 - Average Error between Real-Time Parameter Estimate and Off-line Estimates - N Parameters

<table>
<thead>
<tr>
<th></th>
<th>Nbeta</th>
<th>Np</th>
<th>Nr</th>
<th>Nda</th>
<th>Ndr</th>
</tr>
</thead>
<tbody>
<tr>
<td>TD</td>
<td>0.670</td>
<td>-0.749</td>
<td>-1.364</td>
<td>-5.197</td>
<td>-7.230</td>
</tr>
<tr>
<td>FD</td>
<td>0.171</td>
<td>-0.823</td>
<td>-0.687</td>
<td>1.510</td>
<td>1.763</td>
</tr>
</tbody>
</table>

Table 8.26 - Average Standard Error, Average Settling Time, and Convergence Statistics

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Average Standard Error</th>
<th>Average Settling Time (sec)</th>
<th># of Times Estimate Converged to Off-line Estimate</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Y_\beta$</td>
<td>0.54</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>$Y_r$</td>
<td>0.21</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>$Y_{\delta_k}$</td>
<td>0.48</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>$L_\beta$</td>
<td>7.84</td>
<td>4</td>
<td>17</td>
</tr>
<tr>
<td>$L_p$</td>
<td>1.09</td>
<td>5</td>
<td>28</td>
</tr>
<tr>
<td>$Y_r$</td>
<td>2.74</td>
<td>5</td>
<td>27</td>
</tr>
<tr>
<td>$L_{\delta_\alpha}$</td>
<td>9.46</td>
<td>7</td>
<td>28</td>
</tr>
<tr>
<td>$L_{\delta_k}$</td>
<td>7.02</td>
<td>7</td>
<td>27</td>
</tr>
<tr>
<td>$N_\beta$</td>
<td>1.99</td>
<td>7</td>
<td>22</td>
</tr>
<tr>
<td>$N_p$</td>
<td>0.29</td>
<td>4</td>
<td>22</td>
</tr>
<tr>
<td>$N_r$</td>
<td>0.58</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>$N_{\delta_k}$</td>
<td>2.47</td>
<td>7</td>
<td>24</td>
</tr>
<tr>
<td>$N_{\delta_\alpha}$</td>
<td>1.91</td>
<td>6</td>
<td>28</td>
</tr>
</tbody>
</table>
Chapter 9: Conclusions and Future Work

9.1 - Conclusions

The ultimate goal of this study project was to create a real-time parameter identification (RTPID) algorithm to run onboard the aircraft during flight. Before implementing the onboard PID program, off-line maximum likelihood estimation programs were implemented in both the time and frequency domains. Some of the longitudinal estimates show some disagreement between the time and frequency domains, most notably $C_{L_{q}}$, $C_{L_{\delta_{E}}}$, and $C_{m_{q}}$. In general the off-line frequency domain PID had more flight data segments that resulted in converged parameter estimates than the time domain process and took less time to run. However, the time domain parameters had lower standard errors. The results of the lateral off-line PID show agreement between the time domain and frequency domain estimates with the exceptions being $C_{Y_{r}}$ and $C_{N_{\delta_{A}}}$. Again more flight segments resulted in converged estimates of the parameters in the frequency domain than the time domain and the frequency domain PID produced parameters estimates in less time. For the lateral-directional dynamics, however, the frequency domain results had the small standard errors.

The onboard parameter estimation used a frequency domain implementation of the least-squares estimation. This technique was chosen for its resistance to measurement noise and data dropouts as well as its low computational cost. The states and control surface deflections were transformed into the frequency domain using the recursive Fourier transform which was update with a rate of 25 Hz. The parameter estimates were updated with at a rate of 1 Hz. For the longitudinal dynamics, the real-time estimates converged to the off-line time domain or frequency domain estimate the majority of time with the exception of $Z_{\delta_{E}}$ and $M_{\alpha}$ which showed some bias. The settling time for the longitudinal parameters was four to six seconds depending on the parameter. For the lateral-directional dynamics, the real-time estimates converged to the off-line time domain or frequency domain estimate the majority of time with the exception of the side force parameters and $N_{r}$ which showed some bias. The settling time for the longitudinal parameters was four to eight seconds depending on the parameter.
9.2 - Future Work

There are several areas of future work based on this project. First, computing the confidence interval of the real-time estimates onboard will be a priority for future use in a fault tolerant control system. The parameter estimation program, like all systems, is subject to error and a measure for when to trust the estimates will be necessary for application within a fault tolerant control system. The on-line parameter estimation program currently is formulated to estimate the dimensional stability and control derivatives; however, the estimation of the non-dimensional stability and control derivatives may prove useful for fault detection and identification. Finally, it will be necessary to address the issue of lack of system excitation if the real-time algorithm is to be implemented in a fault tolerant control system. A deeper investigation into the divergence of the estimates in the absence of an input will be necessary.
### Appendix A: WVU Phastball Geometric Properties

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Wing Properties</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Airfoil</td>
<td>NACA 2410</td>
<td>---</td>
</tr>
<tr>
<td><strong>Inboard Section</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Span</td>
<td>3.33</td>
<td>feet</td>
</tr>
<tr>
<td>Root Chord</td>
<td>1.167</td>
<td>feet</td>
</tr>
<tr>
<td>Tip Chord</td>
<td>1.167</td>
<td>feet</td>
</tr>
<tr>
<td>Leading Edge Angle</td>
<td>0</td>
<td>degrees</td>
</tr>
<tr>
<td><strong>Outboard Section (x2)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Span</td>
<td>2</td>
<td>feet</td>
</tr>
<tr>
<td>Root Chord</td>
<td>1.166</td>
<td>feet</td>
</tr>
<tr>
<td>Tip Chord</td>
<td>0.8125</td>
<td>feet</td>
</tr>
<tr>
<td>Leading Edge Angle</td>
<td>0</td>
<td>degrees</td>
</tr>
<tr>
<td><strong>Overall Dimensions</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wingspan</td>
<td>7.33</td>
<td>feet</td>
</tr>
<tr>
<td>Mean Aerodynamic Chord</td>
<td>1.08</td>
<td>feet</td>
</tr>
<tr>
<td>Planform Area</td>
<td>7.847</td>
<td>feet²</td>
</tr>
<tr>
<td>Dimension</td>
<td>Value</td>
<td>Unit</td>
</tr>
<tr>
<td>------------------</td>
<td>----------</td>
<td>----------</td>
</tr>
<tr>
<td><strong>Horizontal Stabilizer Properties</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Airfoil</td>
<td>NACA 0009</td>
<td>---</td>
</tr>
<tr>
<td>Span</td>
<td>2.83</td>
<td>feet</td>
</tr>
<tr>
<td>Root Chord</td>
<td>0.80</td>
<td>feet</td>
</tr>
<tr>
<td>Tip Chord</td>
<td>0.59</td>
<td>feet</td>
</tr>
<tr>
<td>Leading Edge Angle</td>
<td>5.04</td>
<td>degrees</td>
</tr>
<tr>
<td>Taper Ratio</td>
<td>0.74</td>
<td>degrees</td>
</tr>
<tr>
<td>Aspect Ratio</td>
<td>4.00</td>
<td>---</td>
</tr>
<tr>
<td>Planform Area</td>
<td>2.01</td>
<td>feet²</td>
</tr>
<tr>
<td><strong>Vertical Stabilizer Properties</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Airfoil</td>
<td>NACA 0009</td>
<td>---</td>
</tr>
<tr>
<td>Span</td>
<td>0.92</td>
<td>feet</td>
</tr>
<tr>
<td>Root Chord</td>
<td>1.11</td>
<td>feet</td>
</tr>
<tr>
<td>Tip Chord</td>
<td>0.79</td>
<td>feet</td>
</tr>
<tr>
<td>Leading Edge Angle</td>
<td>21.09</td>
<td>degrees</td>
</tr>
<tr>
<td>Taper Ratio</td>
<td>0.71</td>
<td>degrees</td>
</tr>
<tr>
<td>Aspect Ratio</td>
<td>1.92</td>
<td>---</td>
</tr>
<tr>
<td>Planform Area</td>
<td>1.75</td>
<td>feet²</td>
</tr>
</tbody>
</table>
Bibliography


