



UNIVERSITEIT•STELLENBOSCH•UNIVERSITY  
jou kennisvennoot • your knowledge partner

# Aggressive Flight Control Techniques for a Fixed-Wing Unmanned Aerial Vehicle

by

Dunross Rudi Gaum



*Thesis presented at Stellenbosch University in partial  
fulfilment of the requirements for the degree of*

Master of Science in Engineering

Department of Electrical and Electronic Engineering  
University of Stellenbosch  
Private Bag X1, 7602, Matieland, South Africa

Supervisor: Dr I.K. Peddle

March 2009

# Declaration

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the owner of the copyright thereof (unless to the extent explicitly otherwise stated) and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: 25 November 2008

Copyright © 2009 Stellenbosch University  
All rights reserved.

# Abstract

This thesis investigates aggressive all-attitude flight control systems. These are flight controllers capable of controlling an aircraft at any attitude and will enable the autonomous execution of manoeuvres such as high bank angle turns, steep climbs and aerobatic flight manoeuvres. This class of autopilot could be applied to carry out evasive combat manoeuvres or to create more efficient and realistic target drones.

A model for the aircraft's dynamics is developed in such a way that its high bandwidth specific force and moment model is split from its lower bandwidth kinematic model. This split is done at the aircraft's specific acceleration and roll rate, which enables the design of simple, decoupled, linear attitude independent inner loop controllers to regulate these states. Two outer loop kinematic controllers are then designed to interface with these inner loop controllers to guide the aircraft through predefined reference trajectories. The first method involves the design of a linear quadratic regulator (LQR) based on the successively linearised kinematics, to optimally control the system. The second method involves specific acceleration matching (SAM) and results in a linear guidance controller that makes use of position based trajectories. These position based trajectories allow the aircraft's velocity magnitude to be regulated independently of the trajectory tracking. To this end, two velocity regulation algorithms were developed. These involved methods of optimal control, implemented using dynamic programming, and energy analysis to regulate the aircraft's velocity in a predictive manner and thereby providing significantly improved velocity regulation during aggressive aerobatic type manoeuvres.

Hardware in the loop simulations and practical flight test data verify the theoretical results of all controllers presented.

# Opsomming

In hierdie tesis word aggressiewe vlugbeheertegniese ondersoek. Dit is beheerders wat in staat is om 'n vliegtuig by enige oriëntasie te kan beheer en die vliegtuig in staat stel om maneuvres soos aggressiewe draaie, steil stygvlugte en akrobatiese vlugmaneuvres outonoom uit te voer. Hierdie tipe outoloods kan gebruik word om gevegsontwikkelingsmaneuvres uit te voer of meer effektiewe en realistiese teikenvliegtuie te ontwerp.

'n Model vir die vliegtuig se dinamika word op só 'n manier ontwikkel dat sy hoë bandwydte spesifieke-krag-en-moment-model van sy laer bandwydte kinematiese model geskei word. Hierdie skeiding word by die vliegtuig se spesifieke versnelling, sowel as sy roltempo gedoen. Dit stel die navorser in staat om vereenvoudigde en ontkoppelde lineêre oriëntasie-onafhanklike binnelusbeheerders vir die regulering van hierdie toestande te ontwerp. Twee kinematiese buitelusbeheerders word dan ontwerp om met hierdie binnelusbeheerders te koppel en die vliegtuig deur voorafbepaalde verwysingstrajekte te stuur. Die eerste metode behels die ontwerp van 'n lineêre kwadratiese reguleerder, gebaseer op die opeenvolgend-hergelineariseerde kinematika, om die stelsel optimaal te beheer. Die tweede metode behels spesifieke versnellingsgelykstelling en het 'n lineêre stuurbeheerder wat van posisie-gebaseerde verwysingstrajekte gebruik maak tot gevolg. Hierdie posisie-gebaseerde verwysingstrajekte maak die regulering van die vliegtuig se snelheid onafhanklik van sy trajekvolging. Om dit te vermag, word twee snelheidsreguleringsalgoritmes ontwikkel. Dit behels metodes van optimale beheer, geïmplementeer deur dinamiese programmering, en energie-analise wat gebruik word om die vliegtuig se snelheid op 'n vooruitskouende manier te reguleer. Daardeur word die vliegtuig se snelheidsregulasie beduidend verbeter wanneer aggressiewe akrobatiese tipe maneuvres uitgevoer word.

Hardeware-in-die-lus-simulasies, sowel as praktiese toetsvlugdata, bevestig die teoretiese resultate van al die beheerders wat in hierdie tesis ondersoek is.

# Acknowledgements

The author would like to thank the following people for their contribution towards this project.

- Dr. I.K. Peddle, for all your guidance and providing me with insight and understanding on many topics.
- The National Aerospace Centre of Excellence (NACoE), for helping to fund this project during its first year.
- The National Research Foundation (NRF), for providing funding for the project during its second year.
- Willem Hough, for your contribution to building the CAP232 aerobatic aircraft and providing a platform to test aggressive flight control systems.
- My father, for inspiring me to study engineering and supporting me throughout this project.
- Philip Smit, for all your assistance at the flight tests or whenever I needed an extra hand.
- Bernard Visser, for all the helpful advice on RC aircraft related topics.
- All my other friends in the ESL, especially Deon Blaauw and Ruan de Hart, for creating a fun and enjoyable working atmosphere in the lab.
- Micheal Basson and Dr. Kas Hamman, for both doing an excellent job at flying the aircraft.
- Wessel Kroukamp, Johan Arendse and Willie van Rooyen, for providing technical assistance with various aspects of the project.
- And to my girlfriend, Carla-Marié Spies, thank you for your unconditional love, support and understanding during this project.

# Contents

<b>Declaration</b>	<b>i</b>
<b>Abstract</b>	<b>ii</b>
<b>Opsomming</b>	<b>iii</b>
<b>Acknowledgements</b>	<b>iv</b>
<b>Contents</b>	<b>v</b>
<b>List of Figures</b>	<b>viii</b>
<b>List of Tables</b>	<b>xii</b>
<b>Nomenclature</b>	<b>xiii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Background . . . . .	1
1.2 The All-attitude Aggressive Flight Control Problem . . . . .	1
1.3 Thesis Outline . . . . .	6
<b>2 Aircraft Modelling and Simulation</b>	<b>7</b>
2.1 Definitions . . . . .	7
2.2 Aircraft Model Overview . . . . .	10
2.3 Outer Loop Model – Point Mass Kinematics . . . . .	11
2.4 Inner Loop Model – Specific Forces and Moments . . . . .	14
2.5 Non-Linear Simulation . . . . .	20
<b>3 Inner Loop Controllers</b>	<b>22</b>
3.1 Simplifying and Decoupling the Model . . . . .	22
3.2 Axial Specific Acceleration (ASA) Controller . . . . .	24
3.3 Normal Specific Acceleration (NSA) Controller . . . . .	32
3.4 Decoupling the Lateral Dynamics . . . . .	40

3.5	Roll Rate Controller . . . . .	42
3.6	Lateral Specific Acceleration (LSA) Controller . . . . .	45
3.7	Obtaining the Wind Axes Measurements . . . . .	51
3.8	Review of the Inner Loop Controllers . . . . .	53
3.9	Practical Results and Testing . . . . .	53
<b>4</b>	<b>Continuous Re-Linearisation LQR Kinematic Controller</b>	<b>60</b>
4.1	Design Architecture . . . . .	60
4.2	Modelling . . . . .	64
4.3	Implementation . . . . .	68
4.4	Time Based Trajectory Generation . . . . .	79
4.5	Simulated Trajectory Flight Results . . . . .	81
4.6	Practical Flight Test Data . . . . .	83
4.7	Summary . . . . .	87
<b>5</b>	<b>Specific Acceleration Matching Kinematic Controller</b>	<b>88</b>
5.1	Overview . . . . .	88
5.2	Control Architecture . . . . .	90
5.3	Implementation . . . . .	96
5.4	Position Based Trajectory Generation . . . . .	100
5.5	Simulated Trajectory Flight Results . . . . .	103
5.6	Practical Flight Test Results . . . . .	105
5.7	Summary . . . . .	108
<b>6</b>	<b>Predictive Velocity Regulation</b>	<b>109</b>
6.1	The Velocity Regulation Problem . . . . .	110
6.2	Modelling . . . . .	111
6.3	Optimal Control . . . . .	112
6.4	Kinetic Energy Analysis . . . . .	119
6.5	Summary . . . . .	122
<b>7</b>	<b>Comparison and Conclusion</b>	<b>123</b>
7.1	Control System Summaries . . . . .	123
7.2	Control System Comparison and Analysis . . . . .	125
7.3	Conclusion . . . . .	131
7.4	Recommendations . . . . .	132
	<b>Appendices</b>	<b>134</b>
<b>A</b>	<b>Attitude Representation Methods</b>	<b>135</b>
A.1	Attitude Representation Methods . . . . .	135

A.2	Attitude Conversions . . . . .	137
<b>B</b>	<b>Mathematical Principles and Equations</b>	<b>139</b>
B.1	Dot and Cross Products . . . . .	139
B.2	Cross Product Transformation Matrix . . . . .	139
B.3	Derivative of a Vector in a Rotating Reference Frame . . . . .	140
B.4	Moment of Inertia . . . . .	140
B.5	Rotation Matrices . . . . .	141
B.6	Special Rotation Matrices . . . . .	142
B.7	Small Angle Assumptions . . . . .	143
<b>C</b>	<b>Additional Control System Design Details and Results</b>	<b>144</b>
C.1	Pitch Rate Damping NSA Controller . . . . .	144
C.2	Feedback Linearisation of the Directional Coupling . . . . .	148
C.3	Linearisation of the LQR State and Input Matrices . . . . .	149
C.4	Additional LQR Control System Test Results . . . . .	152
C.5	Additional SAM Control System Test Results . . . . .	152
<b>D</b>	<b>Position Based Trajectory Design</b>	<b>155</b>
D.1	Straight Line . . . . .	155
D.2	Aileron Roll . . . . .	158
D.3	Vertical Spiral Arc . . . . .	159
D.4	Horizontal Spiral Turn . . . . .	163
<b>E</b>	<b>Avionics and Ground Station</b>	<b>167</b>
E.1	Avionics Overview . . . . .	168
E.2	Ground Station . . . . .	170
<b>F</b>	<b>Aircraft Specifications and Modelling</b>	<b>174</b>
F.1	Modelling Constants . . . . .	174
F.2	Physical Specifications . . . . .	175
F.3	Aerodynamic Model . . . . .	176
F.4	Dimensional Stability and Control Derivative Notation . . . . .	178
	<b>Bibliography</b>	<b>180</b>



# List of Figures

1.1	Boeing X-45 Unmanned Combat Air Vehicle . . . . .	2
1.2	CAP232 Model Aerobatic Aircraft Used by [15] and in this Project . . . . .	3
1.3	Aileron Roll Trajectory . . . . .	4
1.4	High Bank Angle Turn Trajectory . . . . .	4
1.5	Aggressive Climb Trajectory . . . . .	5
1.6	Vertical Loop Trajectory . . . . .	5
1.7	Immelmann Trajectory . . . . .	5
2.1	Inertial Axis System Definition [26] . . . . .	8
2.2	Body Axis System Definition . . . . .	9
2.3	Wind Axis System Definition . . . . .	9
2.4	Aircraft Model Overview [1] . . . . .	10
2.5	Dynamic Response of the Aircraft's Throttle . . . . .	19
2.6	Phase Plot of the Throttle Models . . . . .	20
2.7	Visual Simulation Environment . . . . .	21
3.1	Step Response and Actuator Commands of Final Controller Design . . . . .	27
3.2	Consecutive Root Locus Design of the ASA Controller . . . . .	27
3.3	Robustness Analysis of the ASA Controller on a Throttle Model with a 1.2 s Time Lag . . . . .	28
3.4	Bode Magnitude Plot of Return Disturbance Transfer Function and its Com- ponents . . . . .	29
3.5	Moment Arm Lengths for Aircraft Forces . . . . .	33
3.6	Open Loop Poles and Zeros of the NSA Dynamics with Step Response . . . . .	34
3.7	Pole Placement Results due to Control System Delay . . . . .	37
3.8	Allowable Region for the Closed Loop NSA Poles at the Aircraft's Trim Airspeed	38
3.9	Open Loop Poles and Zeros of the NSA Dynamics Over a Range of Velocities	39
3.10	Analysis of CL Pole Deviation due to Controller Delay for the NSA Controller	40
3.11	Step Response of the NSA Controller . . . . .	40
3.12	Analysis of CL Pole Deviation due to Controller Delay for the Roll Rate Con- troller . . . . .	44

3.13	Roll Rate Controller Step Response . . . . .	44
3.14	Coupling from the Lateral Directional Dynamics into the Roll Rate . . . . .	45
3.15	Overview of the LSA Control Architecture . . . . .	46
3.16	Dutch Roll Damper Root Locus Design . . . . .	47
3.17	Damping of the Dutch Roll Mode . . . . .	48
3.18	Analysis of the Lateral Specific Acceleration Dynamics . . . . .	49
3.19	LSA Regulator Disturbance Rejection Analysis . . . . .	51
3.20	Overview of the Full Control Architecture . . . . .	53
3.21	Overview of the Hardware in the Loop Simulation Setup . . . . .	54
3.22	Practical Response of the ASA Controller . . . . .	55
3.23	Results of Safety Pilot Commanding Inner Loop Controllers . . . . .	55
3.24	Practical Response of the Normal Specific Acceleration Controller . . . . .	56
3.25	Practical Response of the Roll Rate Controller . . . . .	57
3.26	Open Loop Practical Response of the Dutch Roll Mode . . . . .	58
3.27	Practical Response of the Dutch Roll Damper . . . . .	58
3.28	Practical Response of the Lateral Specific Acceleration Regulator . . . . .	59
4.1	Continuous Re-linearisation of the Reference Trajectory . . . . .	61
4.2	Resulting $q_0$ with Variation of Different Euler 3-2-1 Attitude Angles . . . . .	65
4.3	Inertial Axes Location of the Singularity for Each of the Euler Angle Sequences . . . . .	66
4.4	Overview of the Kinematic State Estimator [15] . . . . .	69
4.5	LQR Gain Settling for Different Sampling Frequencies . . . . .	72
4.6	Gain Conversion of the LQR Feedback Gains . . . . .	73
4.7	LQR Pole Placement for Two Different Points on a Reference Trajectory . . . . .	75
4.8	Velocity Step Response of the LQR Controller . . . . .	76
4.9	Pitch Angle Step Response of the LQR Controller . . . . .	76
4.10	Yaw Angle Step Response of the LQR Controller . . . . .	77
4.11	Lateral Position Step Response of the LQR Controller . . . . .	77
4.12	Altitude Step Response of the LQR Controller . . . . .	78
4.13	Architecture of the Reference Trajectory Generator . . . . .	79
4.14	Aileron Roll Trajectory Simulation for the LQR Controller . . . . .	81
4.15	High Angle Turn Trajectory Simulation for the LQR Controller . . . . .	82
4.16	Vertical Loop Trajectory Simulation for the LQR Controller . . . . .	82
4.17	Immelmann Trajectory Simulation for the LQR Controller . . . . .	82
4.18	Practical Flight Results of the Level Flight by the LQR Controller . . . . .	84
4.19	Practical Flight Results of the Aileron Roll by the LQR Controller . . . . .	84
4.20	Practical Flight Results of the High Angle Turn by the LQR Controller . . . . .	85
4.21	Practical Flight Results of the Aggressive Climb by the LQR Controller . . . . .	85
4.22	Practical Flight Results of the Vertical Loop by the LQR Controller . . . . .	86

5.1	Specific Acceleration Matching in the Aircraft's YZ-plane . . . . .	89
5.2	Position Based Trajectories . . . . .	90
5.3	Consecutive Root Locus Plots for the Design of the Velocity Controller . . . . .	96
5.4	Large Bank Angle Command Generated for a Small NSA Vector . . . . .	97
5.5	Lateral Position Step Response of the SAM Controller . . . . .	98
5.6	Altitude Step Response of the SAM Controller . . . . .	99
5.7	Step Response of the Velocity Controller . . . . .	99
5.8	Straight Line Position Based Trajectory Building Block . . . . .	101
5.9	Aileron Roll Position Based Trajectory Building Block . . . . .	101
5.10	Vertical Spiral Arc Position Based Trajectory Building Block . . . . .	102
5.11	Horizontal Spiral Turn Position Based Trajectory Building Block . . . . .	102
5.12	Assembly of a Complex Trajectory Using Basic Building Blocks . . . . .	103
5.13	Simulation of the Aileron Roll Trajectory Flown by the SAM Controller . . . . .	104
5.14	Simulation of the Horizontal Spiral Turn Trajectory Flown by the SAM Controller . . . . .	104
5.15	Simulation of the Vertical Loop Trajectory Flown by the SAM Controller . . . . .	104
5.16	Simulation of the Immelmann Trajectory Flown by the SAM Controller . . . . .	105
5.17	Practical Flight Results for the Aileron Roll by the SAM Controller . . . . .	106
5.18	Practical Flight Results for the Horizontal Spiral Turn by the SAM Controller . . . . .	106
5.19	Practical Flight Results for the Vertical Loop by the SAM Controller . . . . .	107
5.20	Practical Flight Results for the Immelmann by the SAM Controller . . . . .	108
6.1	Initial Attempt at the Vertical Loop Trajectory without Predictive Velocity Algorithms . . . . .	110
6.2	Optimal Path in a Multistage Decision Process . . . . .	114
6.3	Velocity Regulation Control Law Obtained From the Optimal Control Algorithm . . . . .	117
6.4	Error Rejection of the Optimal Velocity Regulation Control Algorithm . . . . .	117
6.5	Simulation of the Loop Trajectory using the Optimal Velocity Regulation Algorithm . . . . .	118
6.6	Trajectory Discretisation Intervals . . . . .	119
6.7	Logic Switching used to Command the Velocity Controller . . . . .	121
6.8	Simulation of the Loop Trajectory using the Kinetic Energy Analysis Algorithm . . . . .	122
7.1	Continuous Re-Linearisation LQR Control System Architecture . . . . .	124
7.2	SAM Control System Architecture . . . . .	125
A.1	Quaternion Definition . . . . .	137
B.1	Single Rotation Example . . . . .	141
C.1	Effect of Control System Delay on Closed Loop Pole Placement . . . . .	146

C.2	Consecutive Root Locus Design of the Control System . . . . .	147
C.3	Step Response of the Pitch Rate Damping NSA Controller . . . . .	147
C.4	Straight and Level Flight Trajectory Simulation for the LQR Controller . . . . .	152
C.5	Aggressive Climb Trajectory Simulation for the LQR Controller . . . . .	152
C.6	Straight and Level Flight Trajectory Simulation for the SAM Controller . . . . .	153
C.7	Aggressive Climb Trajectory Simulation for the SAM Controller . . . . .	153
C.8	Barrel Roll Trajectory Simulation for the SAM Controller . . . . .	153
C.9	Practical Flight Results of the Level Flight by the SAM Controller . . . . .	154
C.10	Practical Flight Results of the Aggressive Climb by the SAM Controller . . . . .	154
D.1	Straight Line Position Based Trajectory Design . . . . .	156
D.2	Aileron Roll Position Based Trajectory Design . . . . .	158
D.3	Vertical Spiral Arc Position Based Trajectory Design . . . . .	159
D.4	Vertical Spiral Arc Coordinated Into the Loop's Plane . . . . .	160
D.5	Horizontal Spiral Turn Position Based Trajectory Design . . . . .	163
D.6	Horizontal Spiral Turn Climb Angle . . . . .	164
D.7	Horizontal Spiral Turn NSA Vector Analysis . . . . .	165
E.1	Avionics System and Ground Station Overview . . . . .	167
E.2	Microstrip Line Illustration . . . . .	169
E.3	Main Ground Station Tab in the Ground Station Interface . . . . .	172
E.4	Controller Test Tab in the Ground Station Interface . . . . .	172
E.5	LQR Control System Command Tab in the Ground Station Interface . . . . .	173
E.6	SAM Control System Command Tab in the Ground Station Interface . . . . .	173
F.1	Airfoil Illustration . . . . .	176
F.2	AVL Geometry Plot of the Aircraft . . . . .	177

# List of Tables

2.1	Aircraft Control Surfaces . . . . .	10
2.2	Simulation Sensor Noise Specifications . . . . .	21
4.1	Kinematic Estimator RMS State Errors . . . . .	69
4.2	Maximum Desired State Deviations . . . . .	78
4.3	Maximum Desired Control Deviations . . . . .	79
7.1	Comparison Between Various Aggressive Flight Control Architectures . . . . .	131
F.1	Moment of Inertia Measurements [15] . . . . .	175
F.2	Airfoil Properties [15] . . . . .	176
F.3	Longitudinal Stability Derivatives [15] . . . . .	178
F.4	Lateral Stability Derivatives [15] . . . . .	178
F.5	Control Derivatives [15] . . . . .	178
F.6	Dimensional Stability and Control Derivatives (Forces) . . . . .	179
F.7	Dimensional Stability and Control Derivatives (Moments) . . . . .	179

# Nomenclature

## Subscripts and Superscripts:

$W, I, B, R$  Wind, inertial, body and reference axis systems

## Recurring Notations:

$\mathbf{P}_C^{AB}$  Denotes vector  $\mathbf{P}$  with data about axis system  $A$  relative to axis system  $B$ , coordinated in axis system  $C$

$e_{mn}^{AB}$  The element from row  $m$  and column  $n$  of the direction cosine matrix relating the rotation of axis system  $A$  relative to  $B$

## Vectors and Tensors:

$\mathbf{P}, \mathbf{V}, \mathbf{A}$  Position, Velocity and Acceleration

$\mathbf{F}, \mathbf{M}$  Force and Moment vectors

$\mathbf{G}$  Gravitational acceleration vector

$\mathbf{H}$  Angular momentum vector

$\boldsymbol{\omega}$  Angular rate vector

$\mathbf{I}$  Moment of inertia matrix

$\mathbf{q}$  Quaternion vector

$\mathbf{S}$  Cross product matrix

$\mathbf{DCM}$  Direction cosine matrix

$\mathbf{T}^i, \mathbf{T}^j, \mathbf{T}^k$  Single rotation matrices about the  $X, Y$  and  $Z$  axes

$\mathbf{i}, \mathbf{j}, \mathbf{k}$  Unit vectors in the  $X, Y$  and  $Z$  directions respectively

## State Space:

$\mathbf{F}, \boldsymbol{\Phi}$  Continuous and discrete state matrices respectively

$\mathbf{G}, \boldsymbol{\Gamma}$  Continuous and discrete input matrices respectively

$\mathbf{H}$  Output matrix

$\underline{\mathbf{x}}$  State vector

$\underline{u}$  Control vector

### Optimal Control:

$J$  Cost function  
 $\mathbf{Q}_1, \mathbf{Q}_2$  State and Control weighting matrices  
 $\mathbf{K}$  Feedback gain matrix  
 $N$  Time steps  
 $J_{a,b}^*$  Optimal cost to proceed from a to b  
 $\underline{u}^*$  Optimal control vector

### Coordinate Vectors:

$P_x, P_y, P_z$  Position vector components in the North, East and Down directions  
 $\bar{V}, \alpha, \beta$  Velocity magnitude, angle of attack and angle of sideslip  
 $\phi, \theta, \psi$  Third, second and first Euler rotation angles  
 $I_x, I_y, I_z$  Principal moments of inertial about the X, Y and Z axes  
 $P, Q, R$  Roll, pitch and yaw rates  
 $L, M, N$  Rolling, pitching and yawing moments  
 $X, Y, Z$  X, Y and Z-axis forces  
 $A, B, C$  Specific accelerations in the X, Y and Z-axis directions  
 $q_{0-3}$  Components of the Quaternion vector

### Aerodynamic Coefficients:

$C_L$  Coefficient of aerodynamic lift  
 $C_D$  Coefficient of aerodynamic drag  
 $C_y$  Coefficient of aerodynamic side force  
 $C_l$  Aerodynamic roll moment coefficient  
 $C_m$  Aerodynamic pitch moment coefficient  
 $C_n$  Aerodynamic yaw moment coefficient

### Aerodynamic Modelling Symbols:

$\bar{V}_a$  Airspeed  
 $q$  Dynamic pressure  
 $\rho$  Density of air  
 $S$  Wing reference area  
 $\bar{c}$  Mean aerodynamic chord  
 $b$  Wing span

$A$	Aspect ratio
$e$	Oswald efficiency factor

**Additional Modelling Symbols:**

$\delta_E, \delta_A, \delta_R$	Elevator, Aileron and Rudder deflections
$T, T_C$	Actual and commanded thrust respectively
$g$	Gravitational acceleration
$m$	Aircraft mass
$\tau$	Engine time constant
$C^{DS}$	Static to dynamic thrust conversion constant
$R^{LD}$	Lift to drag ratio

**Time and Frequency:**

$t$	Time
$T_S$	Sample period
$f_S$	Sample frequency

**System Dynamics:**

$\omega_n, \zeta$	Natural frequency and damping of a second order system
$p$	Pole location
$z$	Zero location
$\alpha_x$	Term $x$ of a system's characteristic equation

**Abbreviations:**

AC	Aerodynamic Centre
ASA	Axial Specific Acceleration
AVL	Athena Vortex Lattice
CAN	Controller Area Network
CFD	Computational Fluid Dynamics
CG	Centre of Gravity
CL	Closed Loop
CRL	Continuous Re-Linearisation
DCM	Direction Cosine Matrix
DOF	Degree of Freedom
EM	Electromagnetic
GPS	Global Positioning System



HIL	Hardware in the Loop
IMU	Inertial Measurement Unit
ISA	Industry Standard Architecture
LAN	Local Area Network
LQR	Linear Quadratic Regulator
LSA	Lateral Specific Acceleration
MEMS	Microelectromechanical Systems
NMP	Non-minimum Phase
NP	Neutral Point
NSA	Normal Specific Acceleration
NSAVDC	Normal Specific Acceleration Vector Direction Controller
MP	Minimum Phase
OBC	Onboard Computer
PCB	Printed Circuit Board
RC	Radio Control
RF	Radio Frequency
RMS	Root Mean Square
SAM	Specific Acceleration Matching
TSS	Time Scale Separation
UART	Universal Asynchronous Receiver/Transmitter
UAV	Unmanned Aerial Vehicle
UCAV	Unmanned Combat Air Vehicle

# Chapter 1

## Introduction

### 1.1 Background

An Unmanned Aerial Vehicle (UAV) is defined by [23] as an aerial vehicle using both aerodynamic and propulsion forces to navigate along a commanded flight path without assistance from an on-board human pilot. There are many civil as well as military applications for UAVs including reconnaissance, electronic warfare, radio and data relay, search and rescue, fire suppression and law enforcement [24].

Research in autonomous navigation, guidance and flight control of UAVs is growing all the more popular. A recent survey done by the Teal Group, a market analysis firm based in the US, shows that the UAV sector is the most dynamic growth sector in the aerospace industry and projects the worldwide UAV expenditures to more than double within the next decade [27]. In order to contribute to this dynamic field of study, the *Centre of Expertise (CoX) in Autonomous Systems* within the Department of Electrical & Electronic Engineering at the University of Stellenbosch performs active research on this topic and currently consists of more than 50 academic, research and technical staff members. Current research topics include autonomous take-off [21] and landing, control of a variable stability UAV [20] and autonomous helicopter flight control [17].

### 1.2 The All-attitude Aggressive Flight Control Problem

For this project the design of an aggressive flight control system for a fixed-wing UAV is investigated. This is defined as a controller capable of performing aggressive flight manoeuvres such as an Immelmann or a high angle turn. It should be capable of controlling the aircraft at any attitude and over a wide airspeed range. A controller such as this has various practical applications such as:

- Allowing an unmanned combat air vehicle (UCAV), such as the Boeing X-45 shown in figure 1.1, to carry out evasive combat manoeuvres.

- Creating more efficient and realistic target drones.
- Enabling a UAV to carry out terrain following manoeuvres and thereby fly very close to the earth's surface to avoid radar detection.
- Navigating through an urban environment with a UAV by avoiding structures and other obstacles.
- Adding safety and robustness to a UAV's flight control system by enabling successful recovery from larger disturbances.



**Figure 1.1:** Boeing X-45 Unmanned Combat Air Vehicle

Conventional aircraft control systems are not capable of such aggressive flight control, since the non-linear aircraft model is traditionally linearised about a trim flight condition and small perturbation theory is used to design the control system. This allows flight control over a very limited attitude and airspeed range as large perturbations from the linearised flight condition cause the linearisation assumptions to become less and less accurate. This will at some point lead to control system instability. Typically, a linearised controller strategy such as this will be able to achieve 20 degree bank angles and velocity control of 25 per cent from trim. Therefore, in order to design an all-attitude, aggressive flight control system, other strategies will have to be investigated.

### 1.2.1 Previous Research on this Topic

In a previous project, at the University of Stellenbosch, aggressive flight control was already investigated by [15]. The technique used in this project was to re-linearise the aircraft's entire 14 state model at every discrete sample instance of the controller and to use a linear quadratic regulator (LQR) algorithm to minimise a cost function and thereby place stable closed loop poles. This controller was successfully tested on a CAP232 model aircraft, shown in figure 1.2, using the OBC avionics package presented in appendix E.



**Figure 1.2:** CAP232 Model Aerobatic Aircraft Used by [15] and in this Project

Despite its success, the design architecture of this controller has certain shortcomings. Firstly, using an LQR algorithm on a 14 state system becomes difficult, since one has to select 14 different state weightings and 4 actuation weightings. Secondly, insight into the closed loop dynamics is greatly reduced with such a large system and no direct control is available over closed loop pole placement. Lastly, the iterative matrix inversions required by the LQR algorithm is very computationally demanding and therefore the controller can only be implemented on a platform with significant processing power available to it.

Therefore, a project was defined to investigate more elegant methods to solve the aggressive flight control problem and thereby develop a state of the art autopilot that is both intuitive in nature and relatively computationally efficient.

### 1.2.2 Thesis Approach and Goals

An innovative approach to aircraft control, suitable for all-attitude flight and aggressive manoeuvres, has been developed by [1]. The strategy involves splitting the aircraft's model into its faster and slower dynamics. The ability to design various smaller controllers for the aircraft's faster inner loop dynamics using direct pole placement then becomes possible. The principle of time scale separation is then used between the aircraft's faster inner loop and slower outer loop dynamics, where the inner loop control commands are seen as immediately attainable from an outer loop perspective. A simpler kinematics controller can then be designed to control the aircraft's guidance dynamics.

This split in the aircraft's model creates various options for simpler and more computationally efficient controller design. The goals of this project are to investigate these options and can be listed as follows:

- The development of an *LQR kinematic controller* as well as the implementation of the *Specific Acceleration Matching kinematic controller* developed by [1].
- The implementation of different configurations and variations to the inner loop controllers proposed by [1] that are to be used with the above kinematic controllers.

- The practical testing of these algorithms on the CAP232 model aircraft, shown in figure 1.2, and the investigation and correction of any implementation problems.
- The development of any additional algorithms, such as *predictive velocity regulation* discussed in Chapter 6, and supporting tools, such as the *position based trajectories* discussed in Appendix D, to be used with these controllers.
- To investigate the performance of these controllers and compare them with each other as well as with a previous aggressive flight controller developed by [15].

### 1.2.3 Demonstration Trajectories

In order to demonstrate the functionality of these controllers practically, various test trajectories were devised. The first is the *aileron roll*, shown in figure 1.3, which consists of the aircraft performing a 360 degree roll. This is a relatively simple manoeuvre and is the first step to more challenging trajectories.

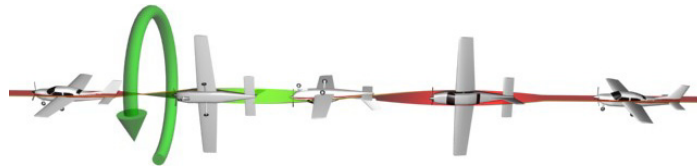


Figure 1.3: Aileron Roll Trajectory

The next trajectory, shown in figure 1.4, is the *high bank angle turn* where the aircraft has to change its heading rapidly. This trajectory shows the ability of the controller to greatly improve on the evasion and turn capabilities of a conventional autopilot.

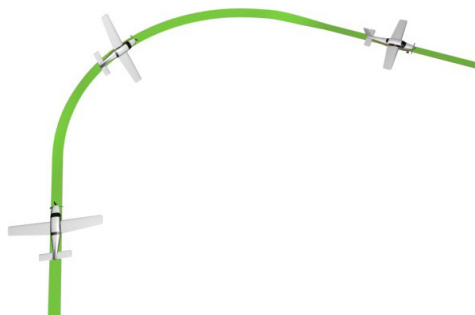


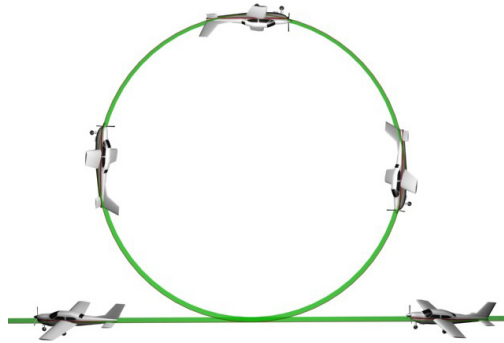
Figure 1.4: High Bank Angle Turn Trajectory

An *aggressive climb* manoeuvre has been defined as a first step to test the control system's vertical climb capabilities. For this trajectory the aircraft pitches up more than 60 degrees and climbs about 60 meters before returning to level flight, as shown in figure 1.5.



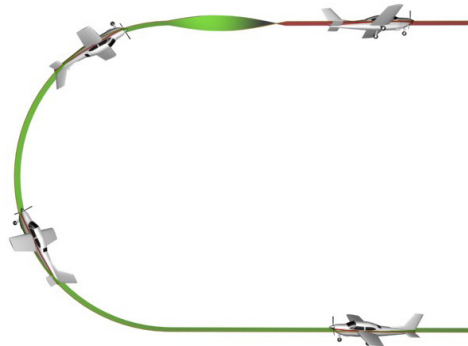
**Figure 1.5:** Aggressive Climb Trajectory

The *vertical loop* trajectory will cause the aircraft to fly a constant radius vertical circle, as illustrated in figure 1.6. This is used as an extreme test for both the aggressive flight and position tracking capabilities of the control system.



**Figure 1.6:** Vertical Loop Trajectory

The final trajectory that will be flight tested is an *Immelmann* manoeuvre. This is where the aircraft does a half loop and then rolls through 180 degrees in order to end the trajectory flying in the opposite direction, as shown by figure 1.7.



**Figure 1.7:** Immelmann Trajectory

### 1.3 Thesis Outline

This thesis covers the theoretical design of various flight controllers as well as their practical implementation and testing. Chapter 2 develops the aircraft model in such a way that it can be split into a fast *inner loop* and slower *outer loop* model. Chapter 3 then shows the design of the inner loop specific acceleration and roll rate controllers. With the help of these controllers the aircraft is reduced to a point mass with commandable specific accelerations. This simplifies the control problem to simple kinematics.

Two kinematic control strategies are then investigated. Chapter 4 develops an *LQR control strategy* for an outer loop kinematic controller. Chapter 5 then investigates *Specific Acceleration Matching*, a simplified alternative strategy for kinematic control proposed by [1]. Certain changes are made to this strategy in order to implement it practically. Chapter 6 shows the design of algorithms for *predictive velocity regulation*. These algorithms are required by the trajectories designed for the *Specific Acceleration Matching* controller of the previous chapter. Chapter 7 then provides a summary and comparison of all the control strategies and states the conclusions derived from this research.

## Chapter 2

# Aircraft Modelling and Simulation

This chapter outlines the aircraft dynamic model as presented by [1]. It will be shown how the aircraft's dynamics can be broken up into a fast set of aircraft-specific dynamics, called the *inner loop model* and a slower set of aircraft independent point mass dynamics, called the *outer loop model*. These models will then be used to design independent controllers in the later chapters.

The first section covers various definitions required for the modelling process. Using these definitions the point mass dynamics of the outer loop system can be derived. Thereafter, the inner loop model consisting of the specific forces and moments acting on the aircraft, will be investigated and modelled. The faster and slower dynamics of the aircraft will then be encapsulated by the inner and outer loop models respectively. This unique way of splitting the aircraft dynamic model will lead to an elegant approach to the control systems design in the chapters to follow.

### 2.1 Definitions

The definitions outlined in this section will form the groundwork for the development of a mathematical model for the aircraft.

#### 2.1.1 Axis System Definitions

The definition of various axis systems are required, as they will each be used in different areas of the aircraft model. An inertial reference frame as well as two aircraft fixed axis systems, namely the wind and body axes, will be defined. By describing the relative motion between these axis systems, the aircraft's inner and outer loop models can be developed.



### 2.1.1.1 Inertial Axis System

An inertial axis system is one where Newton's laws of motion will apply. It provides a fixed reference frame from which an object's absolute position, velocity and attitude can be described. In order to use the earth as an inertial axis system, its curvature and rotation has to be ignored. The inertial axis system used in this project is therefore defined as being right hand orthogonal and fixed to the earth's surface with its origin chosen at some convenient point – usually a runway or structure. The directions in which its axes are defined can be referenced from figure 2.1.

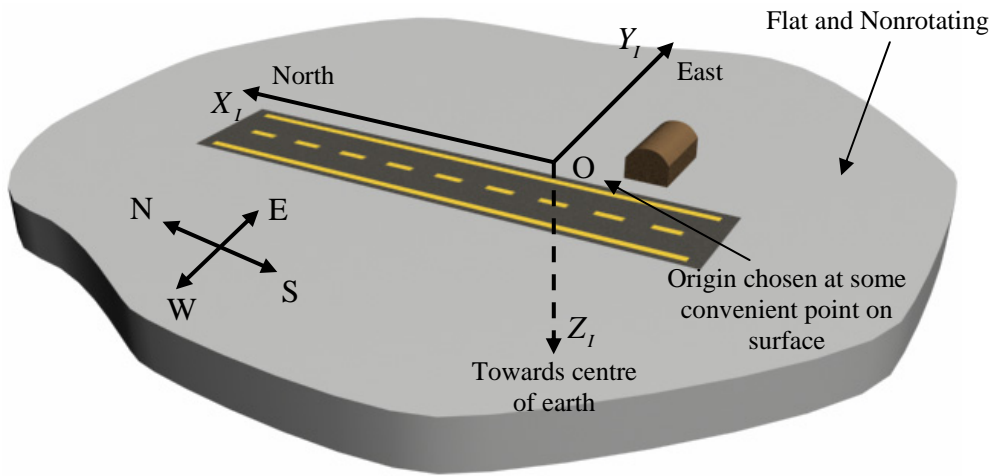


Figure 2.1: Inertial Axis System Definition [26]

### 2.1.1.2 Body Axis System

The body axis system is fixed to the aircraft and therefore both rotates and translates along with it. It is also a right hand orthogonal axis system with its origin located at the aircraft's centre of gravity. The  $X_B$ -axis points directly forward in the aircraft's plane of symmetry and runs along its longitudinal reference line. This reference line would usually run parallel to the chord of the aircraft's wing. The other two axes are perpendicular to  $X_B$  with the  $Y_B$ -axis pointing to the right along the aircraft's main wing and  $Z_B$  pointing directly down in the aircraft's plane of symmetry.

Due to the rotational freedom of this axis system, Newton's laws of motion can not be applied directly. If it is required to use these laws they have to be applied in inertial axes and then the result can be transformed into body axes. The equation of Coriolis (see section B.3) can be used to transform time derivatives of vectors between axis systems with relative rotational motion between them. Figure 2.2 depicts the definition of the body axes system and provides the notations used for forces, moments and angular rates for each of its three axes respectively.

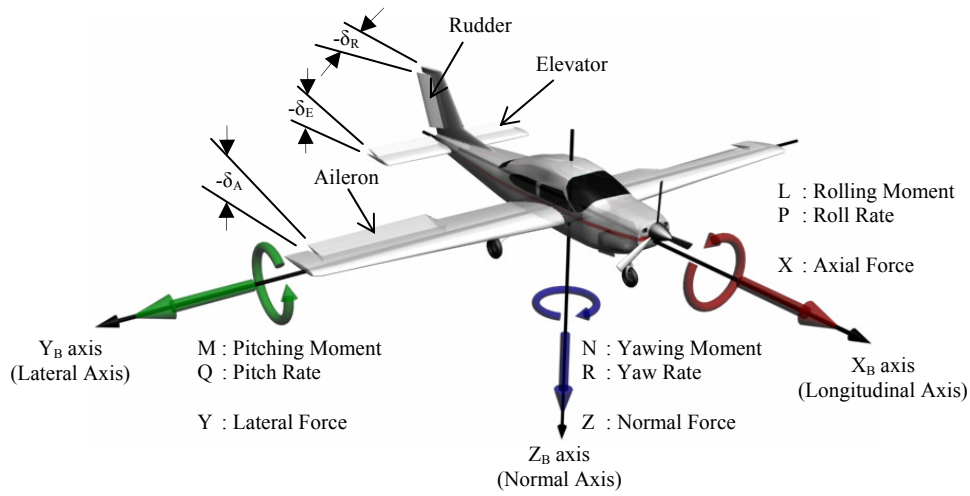


Figure 2.2: Body Axis System Definition

### 2.1.1.3 Wind Axis System

The wind axis system of an aircraft shares its origin with the body axes but its orientation is defined so that its  $X_W$ -axis always points in the direction of the oncoming free-stream velocity vector and its  $Z_W$ -axis always lies in the aircraft's plane of symmetry [26]. Two angles are defined that relate the orientation of the wind axes relative to the aircraft's body axes. They are the angles of attack ( $\alpha$ ) and sideslip ( $\beta$ ). If the body axis system is negatively pitched about the  $Y_B$ -axis through the angle of attack ( $\alpha$ ) and then positively yawed about the  $Z_W$ -axis through the angle of sideslip ( $\beta$ ), the wind axis system is obtained. This transformation is illustrated in figure 2.3. The wind axis system is ideally suited for describing the aerodynamic forces acting on an aircraft, since the forces of lift, sideslip and drag are modelled in the  $Z_W$ ,  $Y_W$  and  $X_W$  directions respectively.

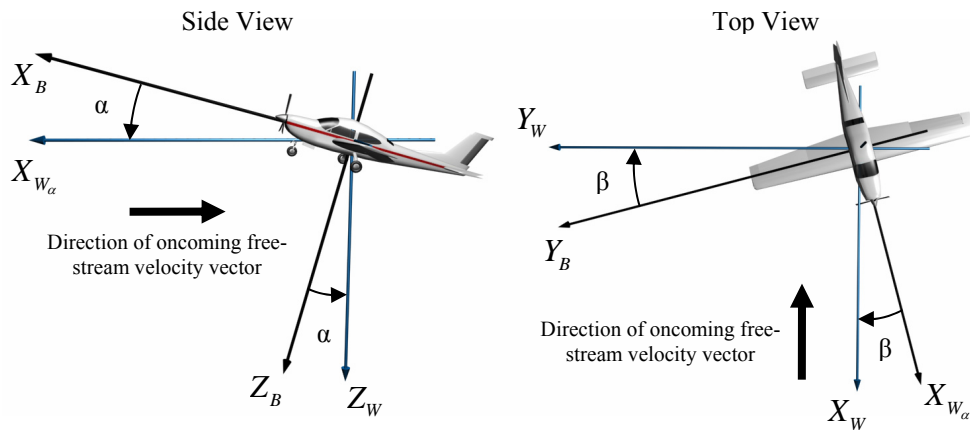


Figure 2.3: Wind Axis System Definition

### 2.1.2 Aircraft Control Surfaces

Most conventional aircraft, such as the one used in this project, are equipped with three aerodynamic actuators as well as the ability to command the magnitude of their thrust vector. These actuators are listed in table 2.1 along with their resulting effect. Refer to figure 2.2 for their locations on the aircraft and their positive deflection directions.

Actuator	Symbol	Induced Effect
Thrust Command	$T_C$	Positive $X_B$ -axis Force
Elevator Deflection	$\delta_E$	Negative Pitching Moment
Aileron Deflection	$\delta_A$	Negative Rolling Moment
Rudder Deflection	$\delta_R$	Negative Yawing Moment

Table 2.1: Aircraft Control Surfaces

## 2.2 Aircraft Model Overview

The dynamics that describe the relative angular motion between the aircraft's body and wind axes operate at a much higher frequency than the attitude dynamics of the aircraft's wind axes relative to inertial space. It is shown by [1] how these dynamics can be split through the principle of time scale separation at the aircraft's specific accelerations<sup>1</sup> to produce models for the aircraft's slower point mass dynamics, and for its faster specific force and moment dynamics. These will be referred to as the outer and inner loop models respectively. Figure 2.4 conceptually illustrates this<sup>2</sup>, with  $\mathbf{G}_I$  and  $\mathbf{G}_W$  corresponding to the gravitational acceleration vectors, coordinated in inertial and wind axes respectively.

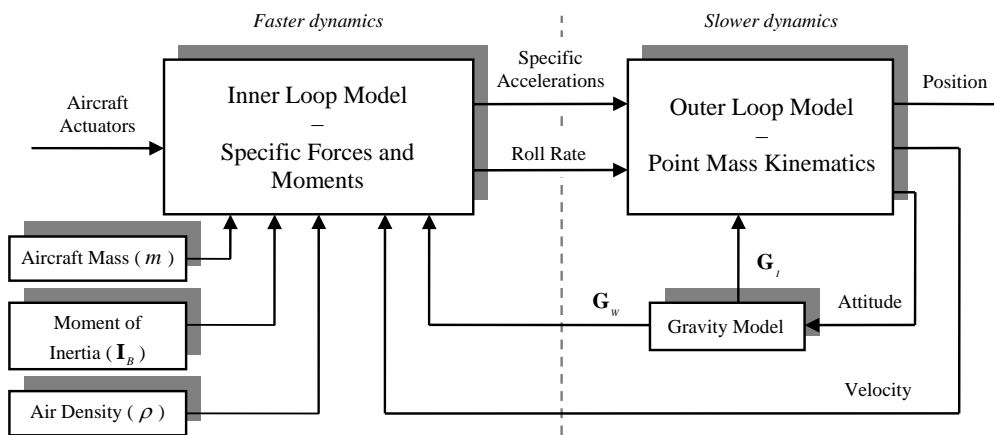


Figure 2.4: Aircraft Model Overview [1]

<sup>1</sup>Specific accelerations are all the accelerations experienced by the aircraft, except for gravity.

<sup>2</sup>In this model the density of air  $\rho$  is considered a constant, as the UAV will not be flown at high altitudes.

## 2.3 Outer Loop Model – Point Mass Kinematics

This model will describe the attitude, velocity and position dynamics of a point mass able to rotate and translate in free space. The motion of the aircraft's wind axes system will be described relative to a fixed inertial reference. For this model it is assumed that the specific accelerations ( $A_W$ ,  $B_W$  and  $C_W$ ) acting on the aircraft as well as its roll rate ( $P_W$ ) are inputs to the system. The origin of these accelerations will be modelled by the inner loop dynamics. This model will therefore encompass all of the aircraft's slower guidance dynamics and is completely aircraft independent.

### 2.3.1 Velocity Dynamics

The dynamic equations governing an object's velocity in wind axes are given by taking the time derivative of its velocity vector.

$$\frac{d}{dt} \mathbf{V}^{WI} \Big|_I = \mathbf{A}^{WI} \quad (2.3.1)$$

The total acceleration vector can be written as the sum of the specific acceleration ( $\mathbf{A}^{WIS}$ ) and gravity vectors ( $\mathbf{G}$ ).

$$\frac{d}{dt} \mathbf{V}^{WI} \Big|_I = \mathbf{A}^{WIS} + \mathbf{G} \quad (2.3.2)$$

The specific accelerations acting on the aircraft will be modelled in wind axes and therefore it is easier to work with a velocity magnitude and the attitude of the wind axes system instead of an inertially coordinated velocity vector. The time derivative should therefore be transformed to the object's wind axes. The equation of Coriolis (see section B.3) can be used to transform the time derivative of a vector with respect to an inertially fixed axis system to that of a rotating axis system.

$$\frac{d}{dt} \mathbf{V}^{WI} \Big|_W = -\boldsymbol{\omega}^{WI} \times \mathbf{V}^{WI} + \mathbf{A}^{WIS} + \mathbf{G} \quad (2.3.3)$$

The above equation can now be coordinated into wind axes and equation (B.2.1) used to simplify the cross product. The direction cosine matrix (DCM), discussed in Appendix A, is used to coordinate the gravity vector from inertial to wind axes.

$$\begin{bmatrix} \dot{\tilde{V}} \\ 0 \\ 0 \end{bmatrix} = - \begin{bmatrix} 0 & -R_W & Q_W \\ R_W & 0 & -P_W \\ -Q_W & P_W & 0 \end{bmatrix} \begin{bmatrix} \tilde{V} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} A_W \\ B_W \\ C_W \end{bmatrix} + \mathbf{DCM}^{WI} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \quad (2.3.4)$$

Here  $\tilde{V}$  is the magnitude of the aircraft's velocity vector,  $P_W$ ,  $Q_W$  and  $R_W$  are its wind axes angular rates and  $A_W$ ,  $B_W$  and  $C_W$  are the components of the specific acceleration vector coordinated in wind axes. The magnitude of the gravity vector in the inertial

down direction is given by  $g$ . Equation (2.3.4) can be split into three equations. The first is the dynamic equation for the velocity magnitude in wind axes and the other two are algebraic constraint equations.

$$\dot{\bar{V}} = A_W + g e_{13}^{WI} \quad (2.3.5)$$

$$\begin{bmatrix} R_W \\ Q_W \end{bmatrix} = \frac{1}{\bar{V}} \begin{bmatrix} g e_{23}^{WI} \\ -g e_{33}^{WI} \end{bmatrix} + \frac{1}{\bar{V}} \begin{bmatrix} B_W \\ -C_W \end{bmatrix} \quad (2.3.6)$$

Here  $e_{xy}^{WI}$  corresponds to row  $x$  and column  $y$  of the  $\mathbf{DCM}^{WI}$  matrix.

The total acceleration vector of the aircraft can also be written in terms of the total force vector ( $\mathbf{F}^{WI}$ ) and the mass of the aircraft ( $m$ ) through Newton's second law as shown below,

$$\mathbf{A}^{WI} = \frac{1}{m} \mathbf{F}^{WI} \quad (2.3.7)$$

Substituting this into equation (2.3.1) and going through the same derivation above, will lead to a different form of the algebraic constraint equations that will be used later for the inner loop model and is given by,

$$R_W = \frac{1}{m \bar{V}} Y_W \quad (2.3.8)$$

$$Q_W = -\frac{1}{m \bar{V}} Z_W \quad (2.3.9)$$

where  $Y_W$  and  $Z_W$  correspond to second and third components of the total force vector coordinated in wind axes.

### 2.3.2 Position Dynamics

In order to obtain the position dynamics, the time derivative of the position vector is taken with respect to inertial space.

$$\left. \frac{d}{dt} \mathbf{P}^{WI} \right|_I = \mathbf{V}^{WI} \quad (2.3.10)$$

As stated in the previous section, the velocity vector, coordinated in the aircraft's wind axes, is used for this model. The DCM is therefore required to convert the vector back to inertial space.

$$\dot{\mathbf{P}}_I^{WI} = \left[ \mathbf{DCM}^{WI} \right]^{-1} \mathbf{V}_W^{WI} \quad (2.3.11)$$

Using the orthogonal property of the DCM given by equation (A.1.2), the previous equation becomes.

$$\dot{\mathbf{P}}_I^{WI} = \left[ \mathbf{DCM}^{WI} \right]^T \mathbf{V}_W^{WI} \quad (2.3.12)$$

Simplifying this yields the position dynamics,

$$\dot{\mathbf{P}}_I^{WI} = \begin{bmatrix} e_{11}^{WI} \\ e_{12}^{WI} \\ e_{13}^{WI} \end{bmatrix} \bar{V} \quad (2.3.13)$$

### 2.3.3 Attitude Dynamics

The attitude dynamics of the wind axes allow for the dynamic calculation of the attitude states, when the wind axes angular rates are known. Various methods of attitude description exist as shown in section A.1. The derivation of the attitude dynamics are discussed below for the Euler 3-2-1 angle sequence, however similar derivations exist for other Euler angle sequences as well as Quaternions.

The angular rate vector can be written as the sum of the Euler angle velocities multiplied by the defined unit vector that each rotates about. For Euler 3-2-1 this is given by,

$$\boldsymbol{\omega}^{WI} = \dot{\phi} \mathbf{u}_1^\phi + \dot{\theta} \mathbf{u}_2^\theta + \dot{\psi} \mathbf{u}_3^\psi \quad (2.3.14)$$

where  $\mathbf{u}_i^x$  corresponds to the  $i^{\text{th}}$  (first, second or third) unit vector of the axis system about which the Euler angle rotation, denoted in the superscript ( $x$ ), is carried out. Refer to section A.1.2 for more information on these rotation unit vectors. It is shown by [14] how these unit vectors can be transformed to the wind axes using single rotation matrices such as the ones defined in section B.6. The result is taken from [8] and stated below,

$$\begin{bmatrix} P_W \\ Q_W \\ R_W \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \phi & \cos \theta \sin \phi \\ 0 & -\sin \phi & \cos \theta \cos \phi \end{bmatrix}_{(321)} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}_{(321)} \quad (2.3.15)$$

with the columns of the above  $3 \times 3$  transformation matrix consisting of the three unit vectors of equation (2.3.14) resolved into the wind axis frame. By making the time derivatives of the attitude states the subject of the above equation and thereby inverting the transformation matrix, the attitude dynamics are obtained. The result is shown below for the Euler 3-2-1 and 2-3-1 angle sequences as well as for Quaternions [8].

Euler 3-2-1

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}_{(321)} = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi \sec \theta & \cos \phi \sec \theta \end{bmatrix}_{(321)} \begin{bmatrix} P_W \\ Q_W \\ R_W \end{bmatrix} \quad (2.3.16)$$

Euler 2-3-1

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}_{(231)} = \begin{bmatrix} 1 & -\cos \phi \tan \theta & \sin \phi \tan \theta \\ 0 & \sin \phi & \cos \phi \\ 0 & \cos \phi \sec \theta & -\sin \phi \sec \theta \end{bmatrix}_{(231)} \begin{bmatrix} P_W \\ Q_W \\ R_W \end{bmatrix} \quad (2.3.17)$$

Quaternions

$$\begin{bmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & -P_W & -Q_W & -R_W \\ P_W & 0 & R_W & -Q_W \\ Q_W & -R_W & 0 & P_W \\ R_W & Q_W & -P_W & 0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \quad (2.3.18)$$

## 2.4 Inner Loop Model – Specific Forces and Moments

For the outer loop model derived in section 2.3 it was assumed that the specific accelerations ( $A_W$ ,  $B_W$  and  $C_W$ ) and the roll rate ( $P_W$ ) of the aircraft are inputs to the system. The inner loop model developed by [1] investigates the origin of the specific forces and moments acting on the aircraft in order to provide the dynamic equations for these specific accelerations and rates. The inner loop model therefore encompasses all of the aircraft-specific dynamics and relates the angular motion of the aircraft's body axes relative to its wind axes.

### 2.4.1 Rigid Body Rotational Dynamics

The rigid body rotational dynamics model the relative angular motion between the aircraft's wind and body axes, given the applied moments. From the definition of the wind axes, its orientation relative to the body axes is defined by two rotations. Firstly, a negative rotation through the angle of attack ( $\alpha$ ) around the  $Y_B$ -axis and then a positive rotation through the angle of sideslip ( $\beta$ ) around the new  $Z_W$ -axis (see section 2.1.1.3). Therefore the angular rate vector of the wind axes with respect to the body axes is given by the angular rates ( $\dot{\alpha}$  and  $\dot{\beta}$ ) about their respective unit vectors.

$$\omega^{WB} = -\dot{\alpha}\mathbf{j}^B + \dot{\beta}\mathbf{k}^W \quad (2.4.1)$$

The angular velocity of the wind axes with respect to inertial space can be written as the sum of the angular velocity of the body axes relative to inertial space and the wind axes relative to the body axes.

$$\omega^{WI} = \omega^{WB} + \omega^{BI} \quad (2.4.2)$$

Substituting equation (2.4.1) into equation (2.4.2) gives,

$$\omega_B^{BI} = \dot{\alpha} \mathbf{j}_B^B - \dot{\beta} \mathbf{k}_B^W + \omega_B^{WI} \quad (2.4.3)$$

with the added subscripts  $B$  indicating that the vectors are now coordinated into the body axes. This equation can be rewritten using the wind to body axes rotation matrix (see section B.6) from equation (B.6.5), to convert the vectors from their native axis systems.

$$\omega_B^{BI} = \dot{\alpha} \mathbf{j}_B^B - \dot{\beta} \mathbf{DCM}^{BW} \mathbf{k}_W^W + \mathbf{DCM}^{BW} \omega_W^{WI} \quad (2.4.4)$$

Expanding the above equation yields,

$$\begin{bmatrix} P \\ Q \\ R \end{bmatrix} = \begin{bmatrix} 0 & \sin \alpha \\ 1 & 0 \\ 0 & -\cos \alpha \end{bmatrix} \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \end{bmatrix} + \begin{bmatrix} \cos \alpha \cos \beta & -\cos \alpha \sin \beta & -\sin \alpha \\ \sin \beta & \cos \beta & 0 \\ \sin \alpha \cos \beta & -\sin \alpha \sin \beta & \cos \alpha \end{bmatrix} \begin{bmatrix} P_W \\ Q_W \\ R_W \end{bmatrix} \quad (2.4.5)$$

By making  $\dot{\alpha}$ ,  $\dot{\beta}$  and  $P_W$  the subject of the equations and substituting the two algebraic constraints from equations (2.3.8) and (2.3.9), three equations are obtained. The first two provide the attitude dynamics of the wind axes relative to the body axes, given the body axes angular rates and wind axes forces. The third equation provides a constraint that keeps the wind axes normal vector in the aircraft's plane of symmetry.

$$\begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ P_W \end{bmatrix} = \begin{bmatrix} -\cos \alpha \tan \beta & 1 & -\sin \alpha \tan \beta \\ \sin \alpha & 0 & -\cos \alpha \\ \cos \alpha \sec \beta & 0 & \sin \alpha \sec \beta \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix} + \frac{1}{m\bar{V}} \begin{bmatrix} \sec \beta & 0 \\ 0 & 1 \\ -\tan \beta & 0 \end{bmatrix} \begin{bmatrix} Z_W \\ Y_W \end{bmatrix} \quad (2.4.6)$$

In order to write the attitude dynamics of the wind axes in terms of the applied forces and moments, the dynamics of the angular rates in the above equation have to be obtained. Euler's law for rigid bodies states that the time derivative relative to the inertial reference frame of an object's angular momentum ( $\mathbf{H}$ ), referenced to its centre of mass, is equal to the externally applied moment ( $\mathbf{M}$ ) [2].

$$\mathbf{M} = \left. \frac{d}{dt} \mathbf{H} \right|_I \quad (2.4.7)$$

The time derivative in the above equation can be transformed to the aircraft's body axes using equation (B.3.1) for the conversion,

$$\mathbf{M} = \left. \frac{d}{dt} \mathbf{H} \right|_B + \omega^{BI} \times \mathbf{H} \quad (2.4.8)$$

As stated by [1], the angular momentum vector about the centre of mass ( $\mathbf{H}$ ) takes on its simplest form when coordinated into body axes, since in this axis system the mass distri-



bution remains constant and is independent of the aircraft's translational and rotational motion. Coordinated in body axes this vector is given by [2] as,

$$\mathbf{H}_B = \mathbf{I}_B \boldsymbol{\omega}_B^{BI} \quad (2.4.9)$$

with  $\mathbf{I}_B$  being the moment of inertia tensor<sup>3</sup> referenced to the body axis system. Due to the aircraft's symmetry equation (B.4.3) can be used for  $\mathbf{I}_B$ . Equation (2.4.8) can now be coordinated into body axes and equation (2.4.9) substituted for the angular momentum vector to yield,

$$\dot{\boldsymbol{\omega}}_B^{BI} = \mathbf{I}_B^{-1} \left( \mathbf{M}_B - \mathbf{S}_{\boldsymbol{\omega}_B^{BI}} \mathbf{I}_B \boldsymbol{\omega}_B^{BI} \right) \quad (2.4.10)$$

In the above equation  $\mathbf{S}_{\boldsymbol{\omega}_B^{BI}}$  is given by equation (B.2.2) and is a matrix used to represent the cross product. Expanding the above equation and combining it with equation (2.4.6) provides the full rigid body rotational dynamics.

$$\begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \end{bmatrix} = \begin{bmatrix} -\cos \alpha \tan \beta & 1 & -\sin \alpha \tan \beta \\ \sin \alpha & 0 & -\cos \alpha \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix} + \frac{1}{m\bar{V}} \begin{bmatrix} \sec \beta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} Z_W \\ Y_W \end{bmatrix} \quad (2.4.11)$$

$$\begin{bmatrix} \dot{P} \\ \dot{Q} \\ \dot{R} \end{bmatrix} = \mathbf{I}_B^{-1} \left( \begin{bmatrix} L \\ M \\ N \end{bmatrix} - \begin{bmatrix} 0 & -R & Q \\ R & 0 & -P \\ -Q & P & 0 \end{bmatrix} \mathbf{I}_B \begin{bmatrix} P \\ Q \\ R \end{bmatrix} \right) \quad (2.4.12)$$

with constraint,

$$P_W = \begin{bmatrix} \cos \alpha \sec \beta & 0 & \sin \alpha \sec \beta \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix} + \frac{1}{m\bar{V}} \begin{bmatrix} -\tan \beta & 0 \end{bmatrix} \begin{bmatrix} Z_W \\ Y_W \end{bmatrix} \quad (2.4.13)$$

## 2.4.2 Specific Forces and Moments

This section investigates the specific forces<sup>4</sup> and moments acting on the aircraft. These forces can be divided into aerodynamic forces and the force caused by the aircraft's thrust vector. For this model it is assumed that the direction of the thrust vector coincides with the aircraft's  $X_B$ -axis and that any moment caused by the thrust is negligibly small. The aerodynamic forces of lift, drag and sideslip are all defined as being either parallel or perpendicular to the aircraft's velocity vector and therefore the wind axes is the logical choice for this model. The specific forces ( $X_W$ ,  $Y_W$ ,  $Z_W$ ) and moments ( $L_W$ ,  $M_W$ ,  $N_W$ ) acting on the aircraft, as given by the small incidence angle aerodynamic model [4], and

<sup>3</sup>See section B.4 for more information on the moment of inertia tensor

<sup>4</sup>Specific forces are all the forces acting on the aircraft, except for gravity.

coordinated in the wind axes are given by,

$$\begin{bmatrix} X_W \\ Y_W \\ Z_W \end{bmatrix} = qS \begin{bmatrix} -C_D \\ C_y \\ -C_L \end{bmatrix} + \begin{bmatrix} \cos \alpha \cos \beta \\ -\cos \alpha \sin \beta \\ -\sin \alpha \end{bmatrix} T \quad (2.4.14)$$

$$\begin{bmatrix} L_W \\ M_W \\ N_W \end{bmatrix} = qS \begin{bmatrix} b & 0 & 0 \\ 0 & \bar{c} & 0 \\ 0 & 0 & b \end{bmatrix} \begin{bmatrix} C_l \\ C_m \\ C_n \end{bmatrix} \quad (2.4.15)$$

with the dynamic pressure ( $q$ ) given by,

$$q = \frac{1}{2} \rho \bar{V}_a^2 \quad (2.4.16)$$

Here  $T$  is the magnitude of the aircraft's thrust vector,  $\bar{V}_a$  is the airspeed,  $\rho$  is the air density,  $S$  is the wing reference area,  $b$  is the wing span and  $\bar{c}$  is the mean aerodynamic chord. The aerodynamic force coefficients of *lift*, *side force* and *drag* are denoted by  $C_L$ ,  $C_y$  and  $C_D$  respectively and the *roll*, *pitch* and *yaw* moment coefficients are given by  $C_l$ ,  $C_m$  and  $C_n$  respectively. These coefficients can be expanded using dimensionless stability and control derivatives. Using these derivatives, the aerodynamic force and moment coefficients can be defined as the sum of various aircraft states and actuator inputs as shown by [4],

$$C_D = C_{D_0} + \frac{C_L^2}{\pi A e} \quad (2.4.17)$$

$$\begin{aligned} \begin{bmatrix} C_y \\ C_L \end{bmatrix} &= \begin{bmatrix} 0 \\ C_{L_0} \end{bmatrix} + \begin{bmatrix} 0 & C_{y\beta} & \frac{b}{2\bar{V}_a} C_{y_P} & 0 & \frac{b}{2\bar{V}_a} C_{y_R} \\ C_{L\alpha} & 0 & 0 & \frac{\bar{c}}{2\bar{V}_a} C_{L_Q} & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \\ P \\ Q \\ R \end{bmatrix} \\ &+ \begin{bmatrix} C_{y\delta_A} & 0 & C_{y\delta_R} \\ 0 & C_{L\delta_E} & 0 \end{bmatrix} \begin{bmatrix} \delta_A \\ \delta_E \\ \delta_R \end{bmatrix} \end{aligned} \quad (2.4.18)$$

$$\begin{aligned} \begin{bmatrix} C_l \\ C_m \\ C_n \end{bmatrix} &= \begin{bmatrix} 0 \\ C_{m_0} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & C_{l\beta} & \frac{b}{2\bar{V}_a} C_{l_P} & 0 & \frac{b}{2\bar{V}_a} C_{l_R} \\ C_{m\alpha} & 0 & 0 & \frac{\bar{c}}{2\bar{V}_a} C_{m_Q} & 0 \\ 0 & C_{n\beta} & \frac{b}{2\bar{V}_a} C_{n_P} & 0 & \frac{b}{2\bar{V}_a} C_{n_R} \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \\ P \\ Q \\ R \end{bmatrix} \\ &+ \begin{bmatrix} C_{l\delta_A} & 0 & C_{l\delta_R} \\ 0 & C_{m\delta_E} & 0 \\ C_{n\delta_A} & 0 & C_{n\delta_R} \end{bmatrix} \begin{bmatrix} \delta_A \\ \delta_E \\ \delta_R \end{bmatrix} \end{aligned} \quad (2.4.19)$$

with  $e$  defined as the Oswald efficiency factor and  $A$  as the aspect ratio of the wing. The static lift and pitching moment coefficients are given by  $C_{L_0}$  and  $C_{m_0}$  respectively. The non-dimensional stability and control derivatives are terms of the form,

$$C_{A_x} = \frac{\partial C_A}{\partial x'} \quad (2.4.20)$$

where,

$$x' = n_c x \quad (2.4.21)$$

normalises  $x$  through a normalising coefficient,  $n_c$ . For derivatives with respect to pitch rate this coefficient is given by  $\bar{c}/2\bar{V}_a$  and for derivatives with respect to roll and yaw rate this coefficient is  $b/2\bar{V}_a$ . Angles of incidence as well as control perturbation angles have a unity normalising coefficient.

Methods such as computational fluid dynamics (CFD) or wind tunnel tests can be used to obtain these stability and control derivatives, however for this project a vortex-lattice program developed at MIT called *Athena Vortex Lattice* (AVL), is used to model the aircraft and calculate these derivatives. Refer to Appendix F for further details.

The following assumptions are required to reduce the aerodynamic model to the form shown above:

1. Ignore the effects of main wing downwash lag [3] on the horizontal tail as well as added mass [4] effects as they are typically negligibly small. Therefore  $C_{m\dot{\alpha}} = 0$  and  $C_{L\dot{\alpha}} = 0$ .
2. Assume the aircraft is operating in pre-stall flight conditions and therefore the incidence angles are small.

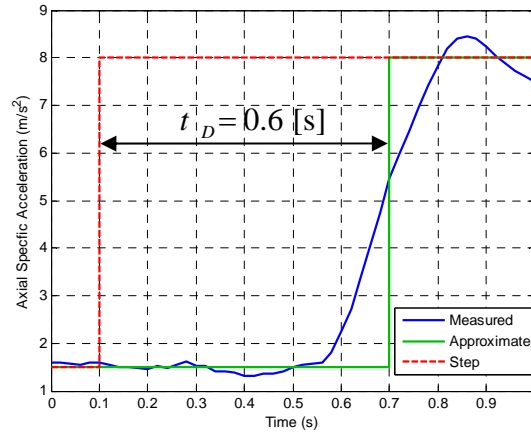
The aerodynamic model presented in this section provides the forces and moments acting on the aircraft in wind axes. The rigid body rotational dynamics model, derived in section 2.4.1, requires the aircraft's moments in body axes. The  $\mathbf{DCM}^{BW}$  transformation matrix from equation (B.6.5) can be used to coordinate the angular rates from wind to body axes as follows,

$$\mathbf{M}_B = \mathbf{DCM}^{BW} \mathbf{M}_W \quad (2.4.22)$$

### 2.4.3 Throttle Dynamics

For most aircraft there exists some form of lag dynamics between when a thrust setting is commanded and when it is achieved by the aircraft's engine. In order to model this effect for the GMS 1.20 cubic inch model aircraft engine used in this project, a step is commanded on the aircraft's throttle and the resulting axial acceleration in body axes is analysed.

Figure 2.5 shows this step response with the throttle step commanded at  $t = 0.1$  s. From the figure it is clear that the response of the system consists of a time lag of approx-



**Figure 2.5:** Dynamic Response of the Aircraft's Throttle

imately 0.5 s and then a fast response which can be approximated as first order thereafter. This model will be used in the non-linear simulator discussed in section 2.5.

In order to model the throttle dynamics for control systems design purposes, only the predominant response will be used and can be approximated with a time delay of 0.6 s (see figure 2.5). Two methods of approximating this time lag are used in this project.

A Padé approximation can be used to model the phase lost due to the delay. The first order Padé approximation will model this with minimal error up to a frequency of about 3 rad/s as shown in figure 2.6. This model consists of a single pole and right half plane zero as given by the first two numerator and denominator terms of the Taylor series expansion of  $e^{-t_D s}$ ,

$$\frac{T}{T_C} = \frac{-s + \frac{1}{\tau_p}}{s + \frac{1}{\tau_p}} \quad (2.4.23)$$

with  $\tau_p$  given by,

$$\tau_p = t_D / 2 = 0.3 \quad (2.4.24)$$

where  $t_D$  corresponds to the amount of time delay being modelled. Written in state space form this model becomes,

$$\begin{aligned} \dot{T}_s &= \left[ -\frac{1}{\tau_p} \right] T_s + \left[ \frac{2}{\tau_p} \right] T_c \\ T &= T_s - T_c \end{aligned} \quad (2.4.25)$$

where  $T$  is the actual thrust and  $T_s$  is a state used to model the dynamics.

A second model that is also used, is a simple first order lag. It is less accurate than the Padé approximation model, but is also less complex, since it only consists of one pole and does not have any added zeros. Figure 2.6 shows that this model predicts the phase loss of the delay with minimal error up to about 2 rad/s. Mathematically this model is

expressed as follows,

$$\dot{T} = \left[ -\frac{1}{\tau} \right] T + \left[ \frac{1}{\tau} \right] T_c \quad (2.4.26)$$

with,

$$\tau = 0.75 \quad (2.4.27)$$

It will be shown with the design of the axial specific acceleration controller in section 3.2 that the dominant frequency of interest lies just above 1 rad/s and therefore both of the above models will be sufficient for control systems design purposes. A Padé approximation model does however provide better insight into the exact dynamics and will be required for a more in depth analysis of any throttle pole placements – see section 3.2.3.

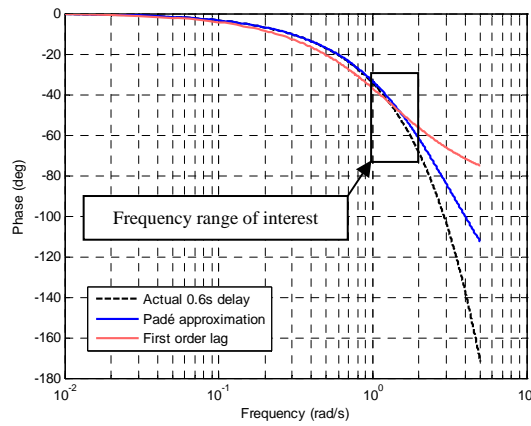


Figure 2.6: Phase Plot of the Throttle Models

## 2.5 Non-Linear Simulation

By combining the inner and outer loop models, a full non-linear aircraft model is obtained. This model can be used to create a virtual aircraft simulation environment in *Simulink* and can provide a relatively accurate testbed for the controllers that will be derived in the chapters to follow. This will help verify the validity of any assumptions and simplifications that were made to the model in the controller design process, as well as evaluate the performance of the controllers.

Quaternion attitude dynamics are used for this model, as these dynamics have no inherent singularities. The throttle is modelled by the full time lag and first order model discussed in section 2.4.3. The final result is a virtual simulation model able to describe the aircraft's attitude, position and velocity relative to inertial space with the actuator commands provided as inputs.

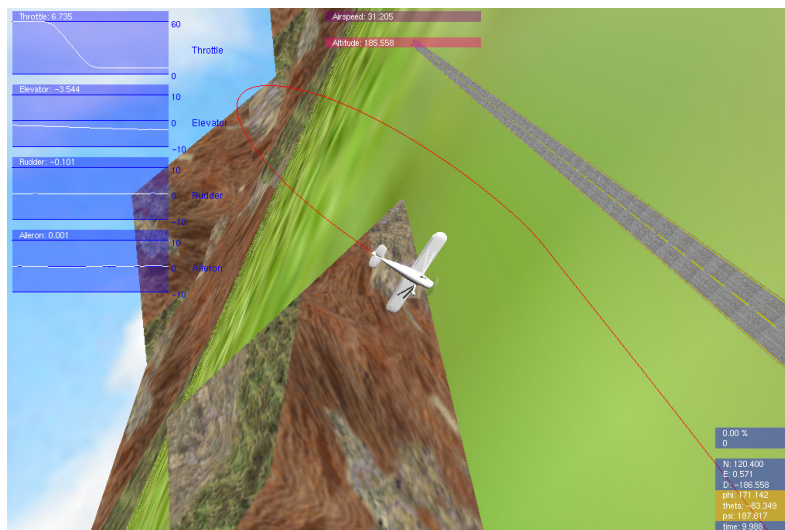
Sensor noise is also added to the simulation in order to closely simulate the real-world sensors used in this project. Table 2.2 shows the RMS values of the simulated noise for each sensor. The noise values for the rate gyroscopes and accelerometers shown in this table does not correlate with the values provided by the respective datasheets. These values additionally take the noise on these sensors due to the vibrations of the aircraft's engine into account.

Sensor	RMS Noise Values	Units
Accelerometers	0.4	$\text{m/s}^2$
Rate Gyroscopes	0.14	$\text{rad/s}$
Magnetometer	0.02	$\text{rad/s}$
Airspeed	0.5	$\text{m/s}$
Pressure Altitude	0.5	m
GPS 2D Position	3	m
GPS Altitude	10	m
GPS Velocity	0.5	$\text{m/s}$

**Table 2.2:** Simulation Sensor Noise Specifications

Additionally, the simulation also provides the ability to generate random wind gusts as well as constant wind shear which acts on the aircraft. This will test the robustness and position tracking accuracy of any controllers that are designed in this project.

A graphical output from this simulation environment has been developed by [25] and can be used to visualise any autonomous flight trajectories that are flown, as shown in figure 2.7.



**Figure 2.7:** Visual Simulation Environment

## Chapter 3

# Inner Loop Controllers

Chapter 2 showed the derivation of the inner loop acceleration and angular rate dynamics model. This chapter will use the ideas presented by [1], which show how the inner loop model can be further simplified and decoupled into axial, normal and lateral models. These models will then be individually used to create four separate controllers – three *specific acceleration controllers*, and a *roll rate controller*. These controllers will be designed to be *attitude independent*, thereby greatly simplifying the ability for all-attitude flight control. In order to account for any disturbances and model uncertainties, high bandwidth feedback control systems will be developed, with augmented integrators for steady state disturbance rejection. All aircraft model uncertainties will then be encapsulated behind fast integrators which ensure that controller commands are quickly achieved.

The theoretical design process will first be outlined for each of these controllers. Thereafter the practical considerations for the implementation of each will be discussed. Simulation as well as practical results will then be shown.

### 3.1 Simplifying and Decoupling the Model

As stated by [1] the inertial cross coupling terms of equation (2.4.12) can be ignored for most autopilot applications. The cross coupling only presents itself when the aircraft is experiencing large angular velocities around two of its axes simultaneously. For normal or even aggressive flight this will rarely be the case and therefore the rigid body rotational dynamics of equations (2.4.12) and (2.4.11) become,

$$\begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \end{bmatrix} = \begin{bmatrix} -\cos \alpha \tan \beta & 1 & -\sin \alpha \tan \beta \\ \sin \alpha & 0 & -\cos \alpha \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix} + \frac{1}{m\bar{V}} \begin{bmatrix} \sec \beta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} Z_W \\ Y_W \end{bmatrix} \quad (3.1.1)$$

$$\begin{bmatrix} \dot{P} \\ \dot{Q} \\ \dot{R} \end{bmatrix} = \begin{bmatrix} \frac{1}{I_x} & 0 & 0 \\ 0 & \frac{1}{I_y} & 0 \\ 0 & 0 & \frac{1}{I_z} \end{bmatrix} \begin{bmatrix} L \\ M \\ N \end{bmatrix} \quad (3.1.2)$$

The specific forces and moments derived in section 2.4.2 are now substituted into these equations and the assumption is made that the wind axes moments can be used without conversion in the body axes. This is shown by [1] to be a valid assumption for small incidence angles since the intrinsic uncertainty in the aerodynamic model is far greater than the added inaccuracy. The full inner loop dynamic model with the throttle model of equation (2.4.26) being used, is given by,

$$\begin{bmatrix} \dot{\alpha} \\ \dot{Q} \\ \dot{\beta} \\ \dot{P} \\ \dot{R} \end{bmatrix} = \begin{bmatrix} -\frac{\bar{L}_\alpha}{m\bar{V}} \sec \beta & 1 - \frac{\bar{L}_Q}{m\bar{V}} \sec \beta & 0 & -\cos \alpha \tan \beta & -\sin \alpha \tan \beta \\ \frac{\bar{M}_\alpha}{I_y} & \frac{\bar{M}_Q}{I_y} & 0 & 0 & 0 \\ 0 & 0 & \frac{\bar{Y}_\beta}{m\bar{V}} & \sin \alpha + \frac{\bar{Y}_P}{m\bar{V}} & \frac{\bar{Y}_R}{m\bar{V}} - \cos \alpha \\ 0 & 0 & \frac{\bar{L}_\beta}{I_x} & \frac{\bar{L}_P}{I_x} & \frac{\bar{L}_R}{I_x} \\ 0 & 0 & \frac{\bar{N}_\beta}{I_z} & \frac{\bar{N}_P}{I_z} & \frac{\bar{N}_R}{I_z} \end{bmatrix} \begin{bmatrix} \alpha \\ Q \\ \beta \\ P \\ R \end{bmatrix} + \begin{bmatrix} -\frac{\sin \alpha}{m\bar{V}} \sec \beta & -\frac{\bar{L}_{\delta_E}}{m\bar{V}} \sec \beta & 0 & 0 \\ 0 & \frac{\bar{M}_{\delta_E}}{I_y} & 0 & 0 \\ -\frac{\cos \alpha}{m\bar{V}} \sin \beta & 0 & \frac{\bar{Y}_{\delta_A}}{m\bar{V}} & \frac{\bar{Y}_{\delta_R}}{m\bar{V}} \\ 0 & 0 & \frac{\bar{L}_{\delta_A}}{I_x} & \frac{\bar{L}_{\delta_R}}{I_x} \\ 0 & 0 & \frac{\bar{N}_{\delta_A}}{I_z} & \frac{\bar{N}_{\delta_R}}{I_z} \end{bmatrix} \begin{bmatrix} T \\ \delta_E \\ \delta_A \\ \delta_R \end{bmatrix} + \begin{bmatrix} \frac{g}{\bar{V}} e_{33}^{WI} \sec \beta \\ 0 \\ \frac{g}{\bar{V}} e_{23}^{WI} \\ 0 \\ 0 \end{bmatrix} \quad (3.1.3)$$

$$\dot{T} = \left[ -\frac{1}{\tau} \right] T + \left[ \frac{1}{\tau} \right] T_c \quad (3.1.4)$$

The dimensional derivative notation used to simplify the representation of the above equations can be referenced in section F.4. Due to the symmetrical camber of the CAP232 model aircraft's wing, the zero angle of attack lift ( $C_{L_0}$ ) and pitching moment ( $C_{m_0}$ ) coefficients are well approximated as zero and have been removed from the above model. It will be additionally shown in section 3.3 that they are not important to the control system design. A few final assumptions are now made in order to decouple the model:

1. Standard small angle approximations (see section B.7) are now made with regard to the angles of attack ( $\alpha$ ) and sideslip ( $\beta$ ). In the dynamic equation for the angle of attack however, it is assumed that  $\beta$  is zero altogether. This assumption is made possible by the coordinated nature of normal flight. It is almost always undesirable to fly a manoeuvre while crabbing to one side. Therefore if the lateral acceleration is always coordinated to zero,  $\beta$  will always be a very small angle.
2. The effect of the thrust vector on the normal and lateral dynamics is ignored. This is due to the very low bandwidth of the throttle in comparison to these dynamics. The small coupling of the thrust will be treated as a disturbance by these controllers and the high bandwidth integrators will quickly remove the error.

After these simplifications the three decoupled inner loop models are given by,



Axial

$$\dot{T} = \left[ -\frac{1}{\tau} \right] T + \left[ \frac{1}{\tau} \right] T_C \quad (3.1.5)$$

$$A_W = \left[ \frac{1}{m} \right] T + \left[ -\frac{qS}{m} C_D \right] \quad (3.1.6)$$

Normal

$$\begin{bmatrix} \dot{\alpha} \\ \dot{Q} \end{bmatrix} = \begin{bmatrix} -\frac{\bar{L}_\alpha}{m\bar{V}} & 1 - \frac{\bar{L}_Q}{m\bar{V}} \\ \frac{M_\alpha}{I_y} & \frac{M_Q}{I_y} \end{bmatrix} \begin{bmatrix} \alpha \\ Q \end{bmatrix} + \begin{bmatrix} -\frac{L_{\delta_E}}{m\bar{V}} \\ \frac{M_{\delta_E}}{I_y} \end{bmatrix} \delta_E + \begin{bmatrix} \frac{g}{\bar{V}} e_{33}^{WI} \\ 0 \end{bmatrix} \quad (3.1.7)$$

$$\begin{bmatrix} C_W \end{bmatrix} = \begin{bmatrix} -\frac{\bar{L}_\alpha}{m} & -\frac{\bar{L}_Q}{m} \end{bmatrix} \begin{bmatrix} \alpha \\ Q \end{bmatrix} + \begin{bmatrix} -\frac{\bar{L}_{\delta_E}}{m} \end{bmatrix} \delta_E \quad (3.1.8)$$

Lateral

$$\begin{bmatrix} \dot{\beta} \\ \dot{P} \\ \dot{R} \end{bmatrix} = \begin{bmatrix} \frac{\bar{Y}_\beta}{m\bar{V}} & \frac{\bar{Y}_P}{m\bar{V}} & \frac{\bar{Y}_R}{m\bar{V}} - 1 \\ \frac{L_\beta}{I_x} & \frac{L_P}{I_x} & \frac{L_R}{I_x} \\ \frac{N_\beta}{I_z} & \frac{N_P}{I_z} & \frac{N_R}{I_z} \end{bmatrix} \begin{bmatrix} \beta \\ P \\ R \end{bmatrix} + \begin{bmatrix} \frac{\bar{Y}_{\delta_A}}{m\bar{V}} & \frac{\bar{Y}_{\delta_R}}{m\bar{V}} \\ \frac{L_{\delta_A}}{I_x} & \frac{L_{\delta_R}}{I_x} \\ \frac{N_{\delta_A}}{I_z} & \frac{N_{\delta_R}}{I_z} \end{bmatrix} \begin{bmatrix} \delta_A \\ \delta_R \end{bmatrix} + \begin{bmatrix} \frac{g}{\bar{V}} e_{23}^{WI} \\ 0 \\ 0 \end{bmatrix} \quad (3.1.9)$$

$$\begin{bmatrix} B_W \\ P_W \end{bmatrix} = \begin{bmatrix} \frac{\bar{Y}_\beta}{m} & \frac{\bar{Y}_P}{m} & \frac{\bar{Y}_R}{m} \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \beta \\ P \\ R \end{bmatrix} + \begin{bmatrix} \frac{\bar{Y}_{\delta_A}}{m} & \frac{\bar{Y}_{\delta_R}}{m} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_A \\ \delta_R \end{bmatrix} \quad (3.1.10)$$

In the normal and lateral models above the velocity of the aircraft is considered a constant, where in fact it is a dynamic variable in the outer loop point mass kinematic model. However, if the dynamics of the outer loop system is kept much slower than these inner loop dynamics, the velocity of the aircraft can be considered a constant through the principle of time scale separation (TSS). The inner loop models will therefore change depending on the current velocity. Typically the TSS assumption can be considered valid when the outer loop dynamics are a magnitude of five or more times slower than the inner loop dynamics. Therefore, it is desired to create high bandwidth inner loop controllers so that fast outer loop guidance poles can be placed for aggressive flight control without violating the TSS assumption.

### 3.2 Axial Specific Acceleration (ASA) Controller

The axial specific acceleration (ASA) controller is responsible for controlling the aircraft's throttle in order to achieve a commanded wind axis ASA. Due to the low bandwidth of the throttle actuator, this controller will not be as fast as the other inner loop controllers.

### 3.2.1 Design

Restated below is the model for the aircraft's ASA, given by equations (3.1.5) and (3.1.6),

$$\dot{T} = \left[ -\frac{1}{\tau} \right] T + \left[ \frac{1}{\tau} \right] T_C \quad (3.2.1)$$

$$A_W = \left[ \frac{1}{m} \right] T + \left[ -\frac{qS}{m} C_D \right] \quad (3.2.2)$$

Using measurements from a wind axis axial accelerometer, an ASA controller can be designed. The drag term in the  $A_W$  output equation will be considered a disturbance in the design of this control system. The effect of this assumption will be further investigated in section 3.2.4. In order to reject this as well as any other disturbances to the ASA in the steady state, an integrator will be augmented to the system and is defined as follows,

$$\dot{E}_A = A_W - A_{W_R} \quad (3.2.3)$$

where  $A_{W_R}$  is the reference ASA in wind axes. The augmented system now becomes,

$$\begin{bmatrix} \dot{T} \\ \dot{E}_A \end{bmatrix} = \begin{bmatrix} -\frac{1}{\tau} & 0 \\ \frac{1}{m} & 0 \end{bmatrix} \begin{bmatrix} T \\ E_A \end{bmatrix} + \begin{bmatrix} \frac{1}{\tau} \\ 0 \end{bmatrix} T_C + \begin{bmatrix} 0 \\ -1 \end{bmatrix} A_{W_R} \quad (3.2.4)$$

A throttle control law for full pole placement is defined as follows,

$$T_C = -K_A A_W - K_E E_A + \bar{N}_A A_{W_R} \quad (3.2.5)$$

Substituting the control law into the dynamics, yields the closed loop system.

$$\begin{bmatrix} \dot{T} \\ \dot{E}_A \end{bmatrix} = \begin{bmatrix} -\frac{1}{\tau} - K_A \frac{1}{m\tau} & -K_E \frac{1}{\tau} \\ \frac{1}{m} & 0 \end{bmatrix} \begin{bmatrix} T \\ E_A \end{bmatrix} + \begin{bmatrix} \frac{\bar{N}_A}{\tau} \\ -1 \end{bmatrix} A_{W_R} \quad (3.2.6)$$

The characteristic equation of this closed loop system is given by,

$$\alpha_c(s) = s^2 + s \left( \frac{1}{\tau} + K_A \frac{1}{m\tau} \right) + \frac{K_E}{m\tau} \quad (3.2.7)$$

With a desired axial characteristic equation defined as follows,

$$\alpha_c(s) = s^2 + \alpha_1 s + \alpha_0 \quad (3.2.8)$$

the control law gains can now be solved and are given by [1] as,

$$K_A = m(\tau\alpha_1 - 1) \quad (3.2.9)$$

$$K_E = m\tau\alpha_0 \quad (3.2.10)$$

The  $\bar{N}_A$  feed-forward term in the control law can be used to place a zero in the closed loop system. This can be used to cancel out the integrator pole so that the dynamic response of the integrator can not be seen from the reference input. The equation used to place this zero is given by [10] as,

$$\bar{N}_A = -\frac{K_E}{z_f} \quad (3.2.11)$$

where  $z_f$  is the location of the desired feed-forward zero.

### 3.2.2 Pole Placement and Simulation

As shown by the throttle control law of equation (3.2.5), feedback from both the states in the augmented axial model is used. This allows for full pole placement of the closed loop system. In order to better understand where to place these closed loop poles, the open loop pole location of the throttle dynamics is investigated. The single real pole, obtained from the dynamics of equation (2.4.26) with  $\tau$  given by equation (2.4.27), is located at,

$$p_a = -1.33 \text{ [rad/s]} \quad (3.2.12)$$

Because of the decision to model drag as a disturbance to the system, the entire ASA controller is required to respond as fast as the throttle actuator's bandwidth will allow in order to reject any disturbances as quickly as possible. In order to obtain the best performance it is decided to place both the system's poles at the same frequency. The desired characteristic equation is therefore defined as follows,

$$\alpha_c(s) = s^2 + 2\zeta\omega_n s + \omega_n^2 \quad (3.2.13)$$

The closed loop frequency is chosen to be slightly slower than the open loop frequency with a somewhat conservative damping ratio. This is given by,

$$\omega_n = 1.05 \text{ [rad/s]} \quad (3.2.14)$$

$$\zeta = 0.8 \quad (3.2.15)$$

As stated earlier, the closed loop zero which can be placed by the  $\bar{N}_A$  feed-forward term can be used to remove the integrator's dynamics from the reference input. If the zero is placed at the same frequency as the integrator, it provides a somewhat excessive amount of feed-forward. This is due to the time lag behaviour of the actual system which causes the integrator to wind up and create significant overshoot before the commanded value is reached. By placing the zero at a slightly higher frequency and thereby providing less feed-forward, an acceptable response is achieved as shown in figure 3.1. The closed loop zero is therefore placed at,

$$z_f = -1.65 \text{ [rad/s]} \quad (3.2.16)$$

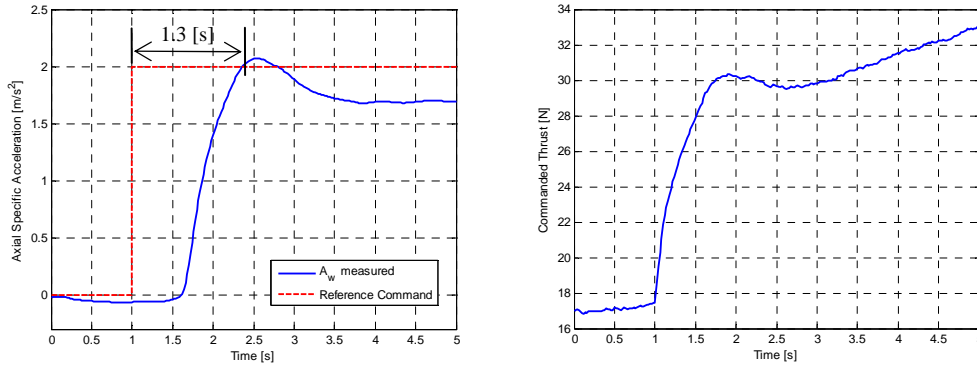


Figure 3.1: Step Response and Actuator Commands of Final Controller Design

### 3.2.3 Further Design Insight

The system's closed loop poles can unfortunately not be placed any faster than those specified in section 3.2.2. This is due to the time lag behaviour of the aircraft's throttle dynamics. As shown in section 2.4.3, these dynamics are highly non-linear and the first order lag model used to approximate these dynamics in this control architecture is not very accurate. In order to gain further insight into the effect of the feedback gains on the closed loop dynamics, a more accurate second order Padé approximation model of the ASA and throttle dynamics can be used in a consecutive root locus design technique with the same feedback gains.

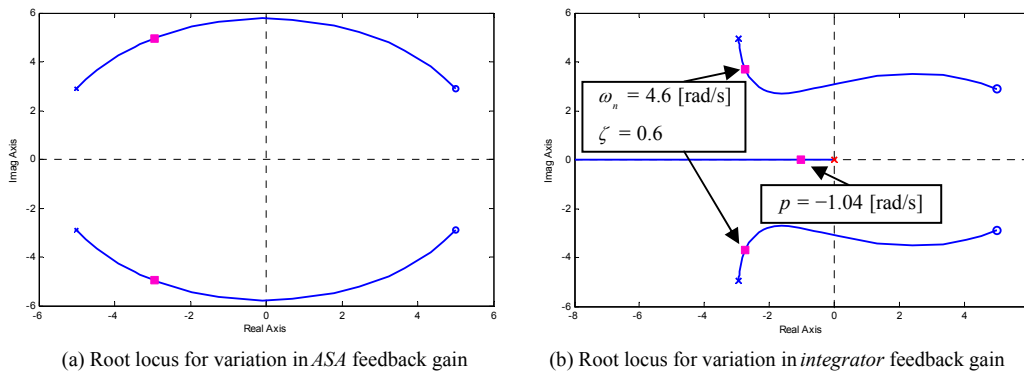
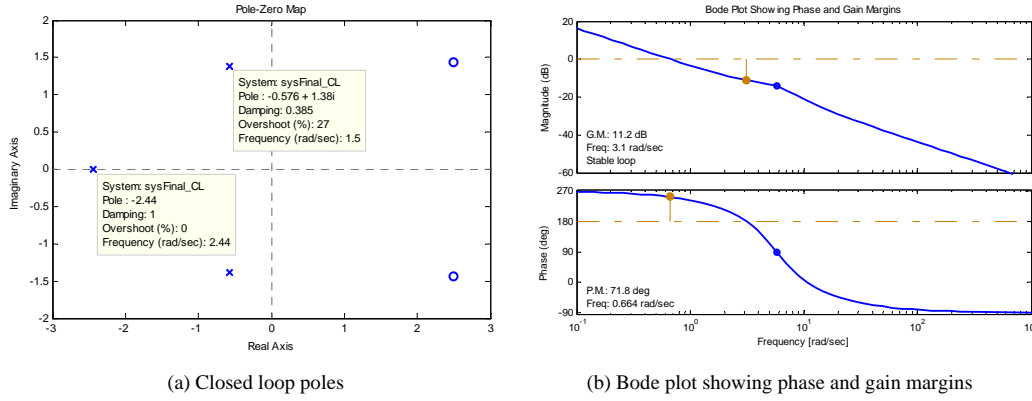


Figure 3.2: Consecutive Root Locus Design of the ASA Controller

Figure 3.2 shows the root loci for the ASA as well as integrator feedback gains. It can be seen that if either of these gains are chosen much higher, the system will become unstable – a fact not illustrated by the simple first order lag model used in the controller design. The closed loop poles are somewhat different to those desired by the design in section 3.2.2. The system's response is dominated by a single real pole with a well

damped complex pole pair located at a somewhat higher frequency. It can be seen however that for the chosen pole locations the system is still highly stable and some robustness exists in the design. This can be better illustrated by figure 3.3 where the model's throttle lag is doubled to 1.2 s and the effective pole placement as well as phase and gain margins are investigated. Even though the high frequency pole pair in the system has shifted from its well damped location to a damping of 0.386, the system is still stable and has a phase margin of 71.8 degrees and gain margin of 11.2 dB.



**Figure 3.3:** Robustness Analysis of the ASA Controller on a Throttle Model with a 1.2 s Time Lag

### 3.2.4 Analysing the Effect of Drag

The aerodynamic drag force has been modelled as a disturbance to the ASA dynamics and the effects of this assumption have to be investigated. The drag force is in fact given by the following equation taken from section 2.4.2,

$$X_W^D = -qS \left( C_{D_0} + \frac{C_L^2}{\pi A e} \right) \quad (3.2.17)$$

with,

$$q = \frac{1}{2} \rho \bar{V}_a^2 \quad (3.2.18)$$

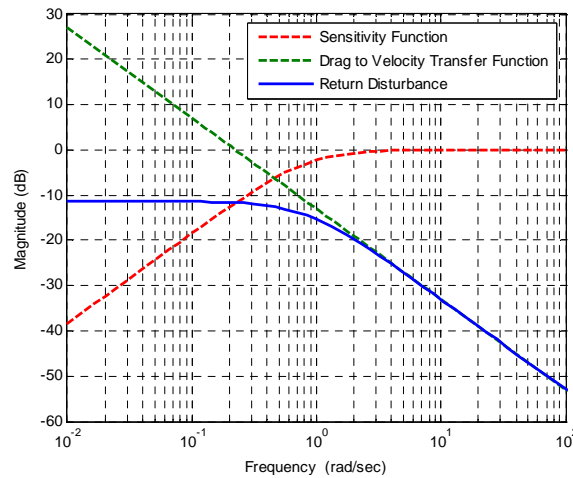
The first and second terms in equation (3.2.17) correspond to the parasitic and dynamic drag respectively. The dynamic drag is the drag induced due to the aircraft's lift and is responsible for the higher frequency components in the drag force. From the point mass kinematics derived in section 2.3 it can be seen that the ASA only effects the aircraft's velocity magnitude dynamics. A simplified sensitivity function, derived by [1], shows how effective the ASA controller will be at rejecting disturbances at different frequencies. This function is given by,

$$S_D(s) = -\frac{s}{s + \frac{1}{\tau \alpha_0}} \quad (3.2.19)$$

with  $\tau$  and  $\alpha_0$  corresponding to the throttle's time constant and the final term in the closed loop characteristic equation, both shown in section 3.2.1. As shown by [1] this can be combined with the natural integration effect from ASA to velocity magnitude to yield the *return disturbance transfer function*. This function analyses the effect of normal specific acceleration, which is generated by the aircraft's lift force, on velocity magnitude due to its coupling into the drag. The return disturbance transfer function is given by,

$$\Delta C_W(s) = -2 \frac{C_W}{\bar{V} R_{LD}} \frac{S_D(s)}{s} \quad (3.2.20)$$

where  $C_W$  corresponds to the aircraft's normal specific acceleration and  $R_{LD}$  to its lift to drag ratio. By selecting a maximum NSA magnitude of 2.5 g's, a minimum velocity of 22 m/s and a conservative estimate of 10 for the minimum lift to drag ratio, an analysis of the maximum coupling into velocity can be done. Figure 3.4 shows the Bode magnitude plot of this transfer function, where it can be seen that a rejection of more than 11 db is obtained across the entire frequency range. For normal specific accelerations larger than 2.5 g's it is assumed that the aircraft is executing a vertical manoeuvre and will cause the ASA controller to completely exit the throttle's dynamic range. This scenario will be dealt with by the algorithms developed in Chapter 6.



**Figure 3.4:** Bode Magnitude Plot of Return Disturbance Transfer Function and its Components

It can therefore be concluded that NSA, acting through the effect of the dynamic drag, has a negligibly small influence on the velocity magnitude. This proves that the dynamic drag can safely be ignored in the design of the ASA controller. Another effect of the drag can be observed in figure 3.1, where a small steady state following error is present on the ASA step response. This can be explained as follows. During a step a constant positive wind axes ASA is applied which leads to a linear increase in velocity. It can then be seen from equations (3.2.17) and (3.2.18) that this will cause a quadratic increase in the drag

force which the first order system will not be able to track without a steady state error. The following two methods can be used to counter this:

1. Augment the system with two more integrators in order to make it a third order system which will be able to reject quadratic disturbances with no steady state error. This will however add too much complexity to the system.
2. The throttle actuator can be used to counter the drag disturbance through open loop dynamic inversion. But due to the open loop nature of this control strategy and the fact that the throttle and drag models of the aircraft are not very accurate, it is very likely that a (usually smaller) steady state error will still exist. The more accurate Padé approximation model for the throttle dynamics also highlights the non-minimum phase nature of these dynamics which implies that dynamic inversion can not be employed successfully.

It would be preferable not to use either of these methods since they will introduce more complexity and uncertainties to the system. Therefore the nature of the steady state error is investigated further. Equations (3.2.17) and (3.2.18) show that the parasitic drag will remain constant while the airspeed is not increasing or decreasing. This constant error can be removed completely by the single integrator in the control system. The steady state error will therefore become smaller as the change in velocity ( $\dot{V}_a$ ) approaches zero. Because the ASA will mostly be used to regulate the aircraft's velocity, this error will merely result in a velocity command being carried out slightly slower than expected. This can be considered an acceptable trade-off for creating a simple and easily implemented ASA controller.

### 3.2.5 Integrator Anti-windup

Integrator windup occurs when the control input is saturated and an error still exists between the reference command and what is achieved by the controller – usually due to an excessively large reference command. This error causes the integrator's state to keep winding up. As soon as the error's sign changes, all of this windup first has to be undone before the control input can exit saturation. This problem can be overcome by simply disabling the propagation of the integrator state in the following two cases:

- When the control has reached its maximum and an error exists that will cause the integrator to command an even larger control signal.
- When the control has reached its minimum and an error exists that will cause the integrator to command an even smaller control signal.

For the ASA controller, integrator windup will most commonly occur when the aircraft is flying at a very large positive or negative pitch angle and the controller is attempting to counter the effect of gravity. The maximum thrust is somewhat smaller than

the force of gravity and therefore the throttle will be driven into saturation. Integrator windup is most common on the ASA controller, but can also occur on any of the other controllers that will be developed in this chapter. Therefore anti-windup will be implemented on all the inner loop controllers.

### 3.2.6 Practical Considerations

Before this controller is practically implemented a few aspects have to be considered first.

**Static to Dynamic Thrust Conversion** As the airspeed increases during flight, the angle of attack on the aircraft's propeller with respect to the incoming wind becomes less and therefore the amount of thrust being generated also decreases. A maximum static thrust measurement, as performed in Appendix F, will therefore not be an accurate figure in predicting the maximum achievable thrust during flight. Through practical measurements a conversion ratio from static to dynamic thrust has been determined at the aircraft's trim velocity. Because this ratio does not greatly change over the aircraft's entire velocity range, the simple conversion constant is used instead of a linear model. This constant has been determined as,

$$C^{DS} = 0.62 \quad (3.2.21)$$

The ASA controller can now be implemented by using the maximum commandable thrust as given by,

$$T_{\max} = C^{DS} T_{\max}^S \quad (3.2.22)$$

where  $T_{\max}^S$  and  $T_{\max}$  is the maximum static and dynamic thrust respectively.

**Throttle Output Filter** Direct feedback of ASA is used in the design architecture of this controller. This measurement can contain a relatively large amount of noise in practice, especially due to this sensor being sensitive to any vibrations on the airframe, such as those caused by the aircraft's engine. This noise will translate directly to the throttle command. It is unwanted for the engine's throttle servo to keep opening and closing the engine's butterfly valve at frequencies higher than the bandwidth of the throttle dynamics. The throttle command should therefore pass through a low pass filter to remove most of this sensor noise. If the cut off frequency of the filter is chosen high enough, it will have no effect on the control system dynamics. The transfer function for the first order low pass filter is given by,

$$G(s) = \frac{1}{\frac{s}{\omega_f} + 1} \quad (3.2.23)$$

with a cut-off frequency selected at,

$$\omega_f = 7.5 \text{ [rad/s]} \quad (3.2.24)$$



### 3.3 Normal Specific Acceleration (NSA) Controller

This controller uses the aircraft's elevator in order to create a commanded wind axes normal specific acceleration (NSA). The control architecture will provide full pole placement freedom of the NSA dynamics.

#### 3.3.1 Design

The decoupled model of the aircraft's NSA dynamics is taken from equations (3.1.7) and (3.1.8) and restated below using the notations outlined in section F.4.

$$\begin{bmatrix} \dot{\alpha} \\ \dot{Q} \end{bmatrix} = \begin{bmatrix} -\frac{\bar{L}_\alpha}{m\bar{V}} & 1 - \frac{\bar{L}_Q}{m\bar{V}} \\ \frac{M_\alpha}{I_y} & \frac{M_Q}{I_y} \end{bmatrix} \begin{bmatrix} \alpha \\ Q \end{bmatrix} + \begin{bmatrix} -\frac{\bar{L}_{\delta_E}}{m\bar{V}} \\ \frac{M_{\delta_E}}{I_y} \end{bmatrix} \delta_E + \begin{bmatrix} \frac{g}{\bar{V}} e_{33}^{WI} \\ 0 \end{bmatrix} \quad (3.3.1)$$

$$\begin{bmatrix} C_W \end{bmatrix} = \begin{bmatrix} -\frac{\bar{L}_\alpha}{m} & -\frac{\bar{L}_Q}{m} \end{bmatrix} \begin{bmatrix} \alpha \\ Q \end{bmatrix} + \begin{bmatrix} -\frac{\bar{L}_{\delta_E}}{m} \end{bmatrix} \delta_E \quad (3.3.2)$$

Gravitational acceleration acts as an unwanted input to the system shown above. In order to create an *attitude independent* NSA controller this coupling has to be removed. Because the model accurately describes how this term enters into the dynamics, direct feedback linearisation can be used to cancel its effect, as shown by [1].

In order to use feedback linearisation, the non-minimum phase nature of the NSA dynamics has to be ignored. The NMP is caused by the small lift force due to the induced angle of incidence on the tail when the elevator is perturbed and places a right half plane zero in the system. This lift force can be ignored by assuming that,

$$C_{L_{\delta_E}} = 0 \quad (3.3.3)$$

Referring to figure 3.6 it can be seen that a left half plane zero is also present in the system's dynamics. This zero typically has no real impact on the dynamics as it is located at such a high frequency that its effect will be negated by unmodelled higher frequency poles from effects such as the servo lag or control system delay. In order to also completely remove this zero from the dynamics, an additional assumption is made. The lift force generated by the induced angle of incidence on the tail (and to a much smaller extent the main wing) when the aircraft is experiencing a pitch rate, is ignored. Therefore,

$$C_{L_Q} = 0 \quad (3.3.4)$$

Only the moments generated by both these forces will be considered by the model. This assumption is valid for any airframe where these forces will have long moment arms to the centre of gravity (CG), such as one with a tailplane located far behind the CG. The NMP effect places a right half plane zero in the system with the length of the moment

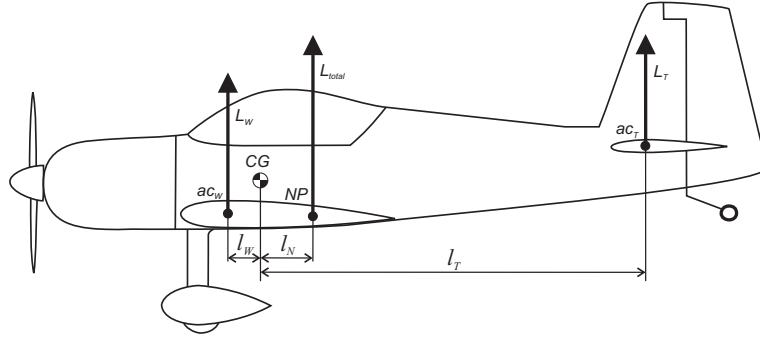
arm from the CG to the aerodynamic centre<sup>1</sup> of the tailplane mainly determining its frequency. An approximate equation for the frequency of the NMP zero has been derived by [1] and is given by,

$$z_{NMP} \approx \sqrt{\frac{\bar{L}_\alpha}{I_y} (l_T - l_N)} \quad (3.3.5)$$

where  $l_N$  and  $l_T$  are approximated as the respective lengths to the aircraft's CG from the neutral point<sup>2</sup> and the aerodynamic centre of the tailplane, as shown in figure 3.5. These lengths can be calculated from the aircraft's derivatives as follows,

$$l_N = -\frac{M_\alpha}{\bar{L}_\alpha} \quad \text{and} \quad l_T = -\frac{M_{\delta_E}}{\bar{L}_{\delta_E}} \quad (3.3.6)$$

As the CG is usually located much closer to the neutral point than the tail plane, the length  $l_T$  will dominate equation (3.3.5).



**Figure 3.5:** Moment Arm Lengths for Aircraft Forces

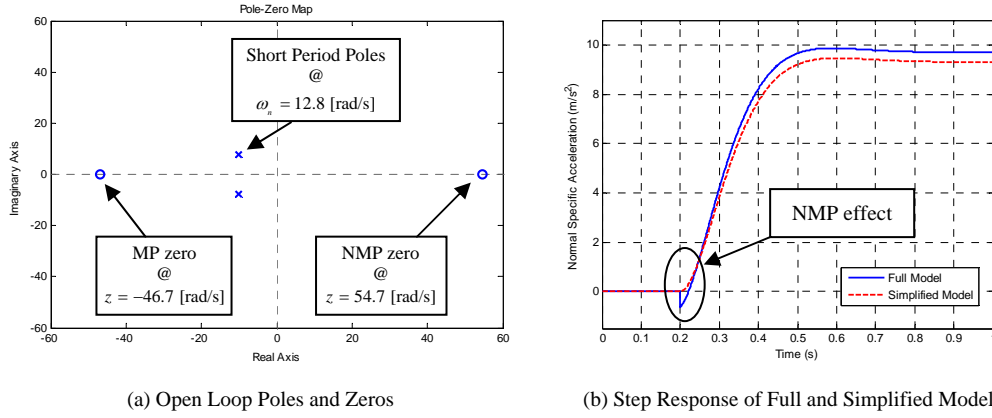
According to [1], if the dominant open as well as closed loop poles of the system are at least three times slower in frequency than the NMP zero, its effect on the system can be considered negligibly small. For the airframe used in this project, equation (3.3.5) yields an approximate NMP zero location of,

$$z_{NMP} \approx 50 \text{ [rad/s]} \quad (3.3.7)$$

at the aircraft's trim velocity, which is about four times faster than the open loop poles and can therefore safely be ignored. Figure 3.6 shows the location of the poles and zeros for the CAP232 model aircraft at its trim velocity as well as the small resulting change in the open loop step response after both zeros have been ignored.

<sup>1</sup>The aerodynamic centre of an aerofoil is the point where the pitching moment generated by this aerofoil does not vary with lift coefficient, i.e. angle of incidence.

<sup>2</sup>The neutral point is the longitudinal aerodynamic centre of the aircraft and is defined as the CG location where the pitching moment coefficient would be independent of the angle of incidence ( $C_{m_\alpha} = 0$ ).



**Figure 3.6:** Open Loop Poles and Zeros of the NSA Dynamics with Step Response

After simplification, the NSA dynamic equations are given by,

$$\begin{bmatrix} \dot{\alpha} \\ \dot{Q} \end{bmatrix} = \begin{bmatrix} -\frac{\bar{L}_\alpha}{m\bar{V}} & 1 \\ \frac{M_\alpha}{I_y} & \frac{M_Q}{I_y} \end{bmatrix} \begin{bmatrix} \alpha \\ Q \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{M_{\delta_E}}{I_y} \end{bmatrix} \delta_E + \begin{bmatrix} \frac{g}{\bar{V}} e_{33}^{WI} \\ 0 \end{bmatrix} \quad (3.3.8)$$

$$C_W = \begin{bmatrix} -\frac{\bar{L}_\alpha}{m} \end{bmatrix} \begin{bmatrix} \alpha \end{bmatrix} \quad (3.3.9)$$

To remove any steady state errors on NSA, such as those caused when the zero angle of attack lift ( $C_{L_0}$ ) and pitching moment ( $C_{m_0}$ ) coefficients are not exactly zero as assumed in section 3.1, the system is augmented with an integrator.

$$\dot{E}_C = C_W - C_{W_R} \quad (3.3.10)$$

By substituting the simplified equation for  $\dot{\alpha}$  into the derivative of equation (3.3.9), the dynamics can be written in terms of NSA, pitch rate and the augmented integrator.

$$\begin{bmatrix} \dot{C}_W \\ \dot{Q} \\ \dot{E}_C \end{bmatrix} = \begin{bmatrix} -\frac{\bar{L}_\alpha}{m\bar{V}} & -\frac{\bar{L}_\alpha}{m} & 0 \\ -\frac{mM_\alpha}{I_y\bar{L}_\alpha} & \frac{M_Q}{I_y} & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} C_W \\ Q \\ E_C \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{M_{\delta_E}}{I_y} \\ 0 \end{bmatrix} \delta_E + \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} C_{W_R} + \begin{bmatrix} -\frac{\bar{L}_\alpha}{m\bar{V}} g e_{33}^{WI} \\ 0 \\ 0 \end{bmatrix} \quad (3.3.11)$$

The elevator control law shown below is defined with feedback from all three states to allow for full pole placement as well as feed-forward from the reference input to place a closed loop zero.

$$\delta_E = -K_Q Q - K_C C_W - K_E E_C + \bar{N}_C C_{W_R} + \delta_{E_G} \quad (3.3.12)$$

The feedback linearisation term ( $\delta_{E_G}$ ) is used to remove the effect of gravity on the system. Feedback linearisation is an open loop control method through which a well-modelled unwanted input (in this case the effect of gravity on the dynamics) can be removed from a system's dynamics. The system's output equation is differentiated until the control input

as well as the unwanted coupling is located in the same equation. The feedback linearisation term ( $\delta_{E_G}$ ) in the control input is then selected in such a way that it cancels the unwanted coupling. The system's control input will therefore create the exact opposite effect as that of the unwanted coupling on the output of interest, thereby removing its effect. The feedback linearisation control law derived by [1] to cancel the effect of gravity on the NSA dynamics is given by,

$$\delta_{E_G} = \frac{gI_y}{\bar{V}M_{\delta_E}} \left[ \left( \frac{M_Q}{I_y} - \frac{M_{\delta_E}}{I_y} K_Q \right) e_{33}^{WI} + \left( \frac{C_W + e_{33}^{WI} g}{\bar{V}} \right) e_{13}^{WI} + P_W e_{23}^{WI} \right] \quad (3.3.13)$$

By now substituting the elevator control law into the NSA dynamics and removing the  $\delta_{E_G}$  and gravity terms, as they will cancel each other out, the closed loop system becomes,

$$\begin{bmatrix} \dot{C}_W \\ \dot{Q} \\ \dot{E}_C \end{bmatrix} = \begin{bmatrix} -\frac{\bar{L}_\alpha}{m\bar{V}} & -\frac{\bar{L}_\alpha}{m} & 0 \\ -\frac{mM_\alpha}{I_y\bar{L}_\alpha} - \frac{M_{\delta_E}}{I_y} K_C & \frac{M_Q}{I_y} - \frac{M_{\delta_E}}{I_y} K_Q & -\frac{M_{\delta_E}}{I_y} K_E \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} C_W \\ Q \\ E_C \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{M_{\delta_E}}{I_y} \bar{N}_C \\ -1 \end{bmatrix} C_{W_R} \quad (3.3.14)$$

Calculating the closed loop characteristic equation yields,

$$\begin{aligned} \alpha_c(s) = s^3 + s^2 \left( \frac{M_{\delta_E}}{I_y} K_Q + \frac{\bar{L}_\alpha}{m\bar{V}} - \frac{M_Q}{I_y} \right) + s \left( \frac{\bar{L}_\alpha M_{\delta_E}}{m\bar{V}I_y} (K_Q - \bar{V}K_C) \right. \\ \left. - \frac{\bar{L}_\alpha M_Q}{m\bar{V}I_y} - \frac{M_\alpha}{I_y} \right) - \frac{\bar{L}_\alpha M_{\delta_E}}{mI_y} K_E \end{aligned} \quad (3.3.15)$$

With the desired characteristic equation for the NSA dynamics defined as follows,

$$\alpha_c(s) = s^3 + \alpha_2 s^2 + \alpha_1 s + \alpha_0 \quad (3.3.16)$$

the control law gains can be solved and are given by [1] as,

$$K_Q = \frac{I_y}{M_{\delta_E}} \left( \alpha_2 - \frac{\bar{L}_\alpha}{m\bar{V}} + \frac{M_Q}{I_y} \right) \quad (3.3.17)$$

$$K_C = -\frac{mI_y}{\bar{L}_\alpha M_{\delta_E}} \left[ \alpha_1 + \frac{M_\alpha}{I_y} - \frac{\bar{L}_\alpha}{m\bar{V}} \alpha_2 + \left( \frac{\bar{L}_\alpha}{m\bar{V}} \right)^2 \right] \quad (3.3.18)$$

$$K_E = -\frac{mI_y}{\bar{L}_\alpha M_{\delta_E}} \alpha_0 \quad (3.3.19)$$

The equation for the closed loop feed-forward zero placed by  $\bar{N}_C$  is given by,

$$\bar{N}_C = -\frac{K_E}{z_f} \quad (3.3.20)$$

where  $z_f$  is the location of the desired zero.

### 3.3.2 Control System Delays

For the NSA and most of the other inner loop controllers presented in this thesis, full pole placement freedom is provided through their control architectures. This means that in theory the response of the system can be made as fast or as slow as is needed. However, there are a few practical and theoretical limitations to this design freedom. These include actuator slew rate, actuator limits, control system delay, model simplification assumptions and measurement noise.

The main practical limiting factor for pole placement in this project, namely control system delay, will now be discussed. The OBC avionics package (see Appendix E) used in this project is designed to use a full sample period to calculate its control signals and then output these signals at the start of the next cycle. For this 50 Hz system, this implies a one sample period delay of 0.02 s from when the measurements are taken to when the control is applied.

The open loop Short Period mode poles of the aircraft's NSA dynamics at its trim velocity of 30 m/s are located at a natural frequency and damping of,

$$\omega_n = 12.8 \text{ [rad/s]}, \quad \zeta = 0.794 \quad (3.3.21)$$

Consider the following two scenarios:

**Attempted Increase in Natural Frequency** The NSA controller now attempts to place these poles at a somewhat higher frequency as given by,

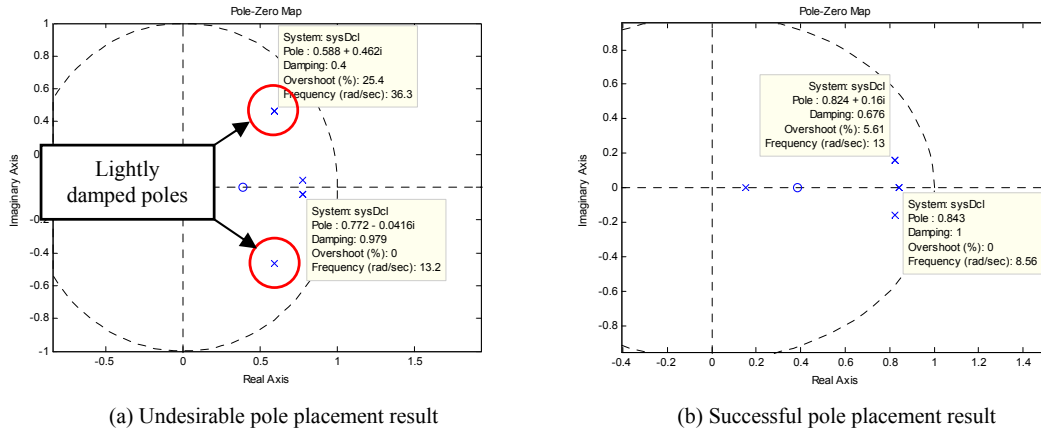
$$\omega_n = 18 \text{ [rad/s]}, \quad \zeta = 0.8, \quad p_i = -10 \text{ [rad/s]} \quad (3.3.22)$$

where  $p_i$  corresponds to the location of the integrator pole. The resulting pole placement can be investigated in the Z-plane, where a single sample period delay is added to the model before the calculated feedback gains are applied. Figure 3.7(a) shows this result, where it can be seen that even though this desired placement falls within the upper limit imposed by the NMP zero, the Short Period mode poles have ended up at double the desired frequency and at a damping ratio of  $\zeta = 0.4$ . The extra pole caused by the delay has also moved into the frequency range of the system and formed a complex pole pair with the integrator pole. This result greatly differs for the desired pole placement and will adversely affect the performance of the system. Should any further increase in natural frequency be attempted, the closed loop system will start to become unstable.

**No Change in Natural Frequency** If the closed loop poles are selected at or close to their open loop frequency and the integrator is placed slightly slower than this as follows,

$$\omega_n = 13 \text{ [rad/s]}, \quad \zeta = 0.707, \quad p_i = -7 \text{ [rad/s]} \quad (3.3.23)$$

it results in a stable system with all the poles located relatively close to where they were placed. The delay pole is also located at a much higher frequency where it will not interfere with the system's dynamics. This result is shown in figure 3.7(b).



**Figure 3.7:** Pole Placement Results due to Control System Delay

The conclusion that can be drawn from this is that if it is not attempted to place the closed loop poles far from their open loop frequency or the integrator pole at too high a bandwidth, and thereby requiring a significant amount of control effort in the process, the controller delay will not greatly affect the resulting pole placement. This limiting factor for the placement of the closed loop poles should be considered for all the inner loop controllers being designed.

**Pitch Rate Damping NSA Controller** An alternative design strategy for the NSA controller which does not use NSA feedback to alter the system's natural dynamics is developed in section C.1. This controller only uses feedback from pitch rate to damp the Short Period mode response and feedback from the NSA integrator in order to achieve the controller command without a steady state error. It is ideally suited for an aircraft where the natural NSA dynamics is already relatively acceptable and will be much more robust against the effects of control system delay and model inaccuracies. This controller has not been practically tested in this project, as it has been developed in response to an analysis of the flight test results obtained from the NSA controller proposed by [1].

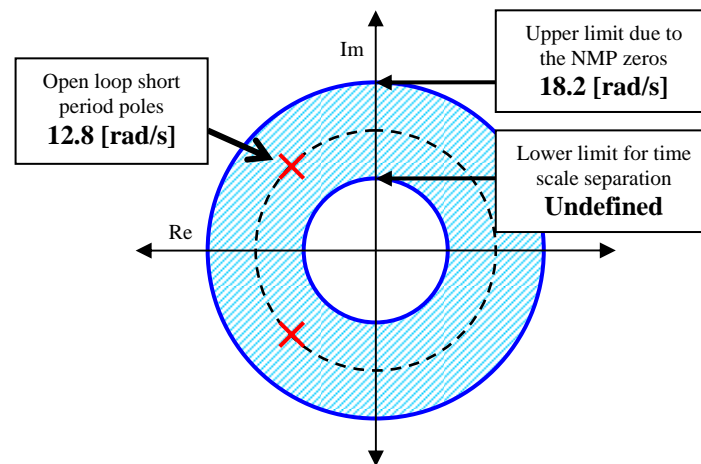
### 3.3.3 Pole Placement and Simulation

Limiting factors for the pole placement freedom of the NSA dynamics are listed below:

1. The lower bound for the placement of these poles is the time scale separation that needs to exist between the inner and outer loop controllers. This bound is placed at five times the outer loop frequency.

2. The upper limit is determined by the NMP zeros that have been ignored from the NSA dynamics. If the poles are placed any faster than one third [1] of the frequency of these zeros they will start to affect the dynamics.
3. The third limit is on the movement of the poles from their open loop location which is caused by the delay in the system as discussed in section 3.3.2.

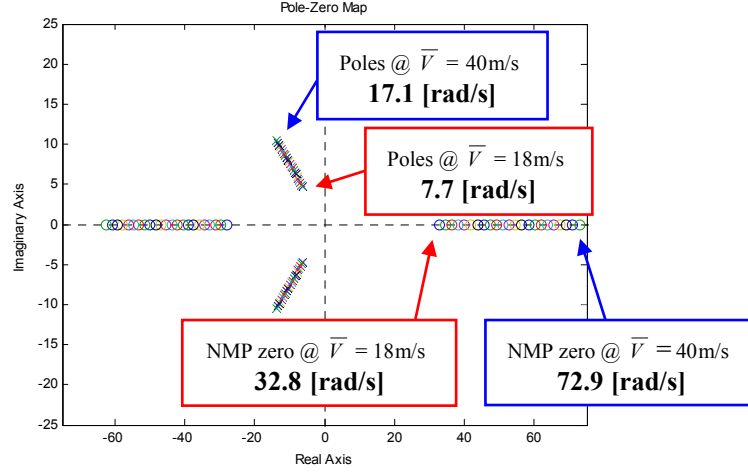
These constraints can be analysed for the CAP232 model aircraft used in this project at its trim velocity. The NMP zero is located at 54.7 rad/s and therefore the upper bound for the poles is set at 18.2 rad/s. The lower bound is as yet undefined as the performance specification of the outer loop controller still has to be selected. The closed loop locations of the inner loop poles can be used to define this specification after all the controllers have been designed, which will then create an upper limit for the outer loop poles. Figure 3.8 shows these constraints as well as the location of the open loop poles, which is safely within the allowable region for closed loop pole placement.



**Figure 3.8:** Allowable Region for the Closed Loop NSA Poles at the Aircraft's Trim Airspeed

This controller should operate over the full range of normal flyable velocities of the aircraft and therefore the above analysis should be done over this entire range. For the aircraft used in this project this range is defined from 18 m/s to 40 m/s. Figure 3.9 shows the open loop poles and zeros of the system over this velocity range. It can be seen how velocity acts as a global scaling factor for these poles and zeros while their relative scaling to each other remains the same. Therefore the allowable region for pole placement will also be similarly scaled with velocity. This implies that any selected closed loop poles will have to be scaled accordingly in order to keep them within this allowable region.

The closed loop poles are therefore selected at the frequency of their open loop counterparts but with a damping ratio of  $\zeta = 0.707$  for all velocities. This placement is chosen due to the limited allowable movement of these poles and the fact that this will cause



**Figure 3.9:** Open Loop Poles and Zeros of the NSA Dynamics Over a Range of Velocities

them to remain inside the region of constraint over the entire velocity range. The open loop frequency can be obtained from the open loop characteristic equation.

$$\alpha_c(s) = s^2 + s \left( \frac{\bar{L}_\alpha}{m\bar{V}} - \frac{M_Q}{I_y} \right) + \left( -\frac{\bar{L}_\alpha}{m\bar{V}} \frac{M_Q}{I_y} - \frac{M_\alpha}{I_y} \right) \quad (3.3.24)$$

Therefore the frequency of the open loop Short Period mode poles is given by,

$$\omega_n = \sqrt{-\frac{\bar{L}_\alpha}{m\bar{V}} \frac{M_Q}{I_y} - \frac{M_\alpha}{I_y}} \quad (3.3.25)$$

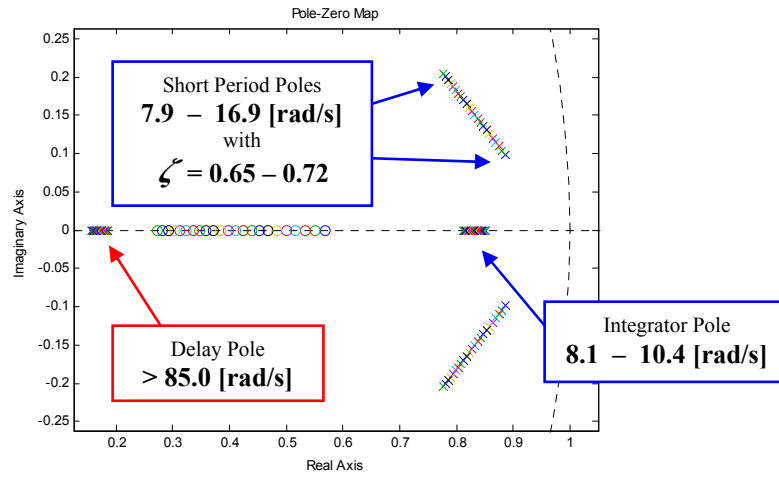
The integrator pole location has not yet been selected. It is chosen at a frequency somewhat lower than the NSA dynamics, yet high enough to create a time scale decoupled inner loop controller. It is also scheduled over velocity as follows,

$$p_i = -6.5 \text{ @ } \bar{V} = 18 \text{ [m/s]} \quad \text{to} \quad p_i = -8.1 \text{ @ } \bar{V} = 40 \text{ [m/s]} \quad (3.3.26)$$

The closed loop feed-forward zero is placed at a frequency 60% higher than the integrator's pole. This will remove some of the integrator dynamics from the reference input, but avoid providing too much feed-forward due to possible modelling inaccuracies which could create an undesirable amount of overshoot in the practical system.

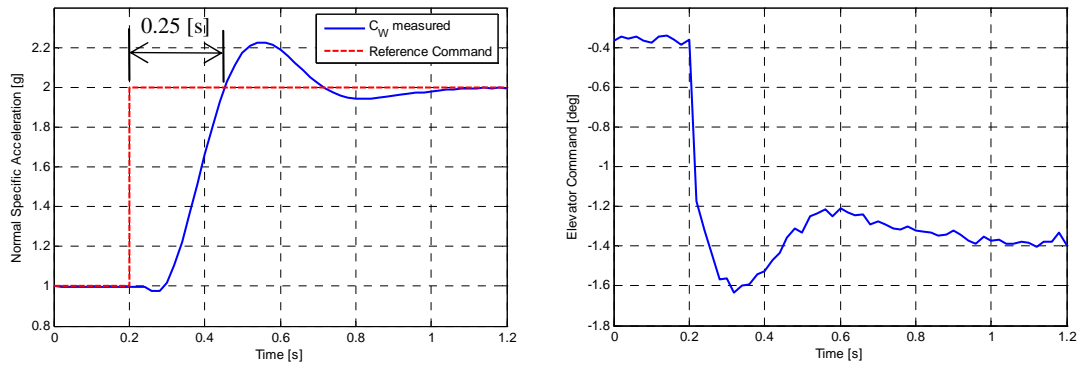
A Z-plane analysis of the resulting pole placement due to the NMP zeros and the system delay can be done in order to verify its success. Figure 3.10 shows this result over the entire velocity range. The closed loop poles are located relatively near their desired locations with the only notable difference being a slight upward shift in the integrator's frequency due to the delay.





**Figure 3.10:** Analysis of CL Pole Deviation due to Controller Delay for the NSA Controller

With the closed loop poles selected, the controller's step response can be analysed. A one sample period actuation delay is added to the non-linear simulation model to incorporate the effect that control system delay has on the system. Figure 3.11 shows the simulated step response of the NSA controller. A small amount of non-minimum phase effect can be seen just after the step, with the Short Period mode poles dominating the response thereafter. The controller's response is fast and well damped and therefore the design can be considered acceptable.



**Figure 3.11:** Step Response of the NSA Controller

### 3.4 Decoupling the Lateral Dynamics

The aircraft's lateral model still consists of both the directional and roll mode dynamics. As stated by [1] it might be possible to decouple these dynamics and design two inde-

pendent controllers. Each of the controllers will have an integrator on its output that will assist in the removal of errors, including those caused by the decoupling simplifications.

It is shown by [4] how the lateral force produced due to aileron deflection and roll rate can be considered negligible. These are commonly used assumptions with regard to the lateral dynamics. These assumptions can be verified numerically by assuming a maximum roll rate of 270 deg/s and aileron deflection of 15 deg. The resulting lateral acceleration produced in these cases are given by 0.075 m/s<sup>2</sup> and 0.11 m/s<sup>2</sup> respectively. Both of these values are negligibly small and therefore verifies the validity of the assumptions. This allows for the lateral dynamics to be written as follows,

$$\begin{bmatrix} \dot{\beta} \\ \dot{P} \\ \dot{R} \end{bmatrix} = \begin{bmatrix} \frac{\bar{Y}_\beta}{mV} & 0 & \frac{\bar{Y}_R}{mV} - 1 \\ \frac{L_\beta}{I_x} & \frac{L_P}{I_x} & \frac{L_R}{I_x} \\ \frac{N_\beta}{I_z} & \frac{N_P}{I_z} & \frac{N_R}{I_z} \end{bmatrix} \begin{bmatrix} \beta \\ P \\ R \end{bmatrix} + \begin{bmatrix} 0 & \frac{\bar{Y}_{\delta_R}}{mV} \\ \frac{L_{\delta_A}}{I_x} & \frac{L_{\delta_R}}{I_x} \\ \frac{N_{\delta_A}}{I_z} & \frac{N_{\delta_R}}{I_z} \end{bmatrix} \begin{bmatrix} \delta_A \\ \delta_R \end{bmatrix} + \begin{bmatrix} \frac{g}{V} e_{23}^{WI} \\ 0 \\ 0 \end{bmatrix} \quad (3.4.1)$$

$$\begin{bmatrix} B_W \\ P_W \end{bmatrix} = \begin{bmatrix} \frac{\bar{Y}_\beta}{m} & 0 & \frac{\bar{Y}_R}{m} \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \beta \\ P \\ R \end{bmatrix} + \begin{bmatrix} \frac{\bar{Y}_{\delta_R}}{m} \\ 0 \end{bmatrix} \begin{bmatrix} \delta_R \end{bmatrix} \quad (3.4.2)$$

For most conventional aircraft the coupling between the directional and the roll rate dynamics in the above system are quite weak. It is shown by [1] that if the conditions analysed below hold, lateral decoupling assumptions can be made.

$$\left| \frac{C_{n_P}}{C_{l_P}} \right| \ll \left| \frac{C_{n_R}}{C_{l_R}} \right| \quad (47 \text{ times smaller}) \quad (3.4.3)$$

$$\left| \frac{C_{n_P}}{C_{l_P}} \right| \ll \left| \frac{C_{n_\beta}}{C_{l_\beta}} \right| \quad (44 \text{ times smaller}) \quad (3.4.4)$$

$$\left| \frac{C_{n_P}}{C_{l_P}} \right| \ll \left| \frac{C_{n_{\delta_R}}}{C_{l_{\delta_R}}} \right| \quad (239 \text{ times smaller}) \quad (3.4.5)$$

$$\left| \frac{C_{n_{\delta_A}}}{C_{l_{\delta_A}}} \right| \ll \left| \frac{C_{n_R}}{C_{l_R}} \right| \quad (159 \text{ times smaller}) \quad (3.4.6)$$

$$\left| \frac{C_{n_{\delta_A}}}{C_{l_{\delta_A}}} \right| \ll \left| \frac{C_{n_\beta}}{C_{l_\beta}} \right| \quad (149 \text{ times smaller}) \quad (3.4.7)$$

From the above analysis it is clear that all the conditions hold by a rather large margin – a factor of 10 or more would already have been considered acceptable. Therefore the lateral dynamics can be safely decoupled into roll rate and lateral specific acceleration models and separate controllers can be designed for each.

It can now be assumed that feedback from the roll rate to the ailerons will only affect the roll rate dynamics and feedback from the yaw rate and lateral specific acceleration to the rudder will only affect the directional dynamics. An analysis of this assumption is done in section 3.5.2 where the coupling effect (if any) of the directional dynamics on the

roll rate is shown. The decoupled models are now given by,

Roll Rate

$$\dot{P} = \left[ \frac{L_P}{I_x} \right] P + \left[ \frac{L_{\delta_A}}{I_x} \right] \delta_A \quad (3.4.8)$$

$$P_W = P \quad (3.4.9)$$

Lateral Specific Acceleration

$$\begin{bmatrix} \dot{\beta} \\ \dot{R} \end{bmatrix} = \begin{bmatrix} \frac{\bar{Y}_\beta}{m\bar{V}} & \frac{\bar{Y}_R}{m\bar{V}} - 1 \\ \frac{\bar{N}_\beta}{I_z} & \frac{\bar{N}_R}{I_z} \end{bmatrix} \begin{bmatrix} \beta \\ R \end{bmatrix} + \begin{bmatrix} \frac{\bar{Y}_{\delta_R}}{m\bar{V}} \\ \frac{\bar{N}_{\delta_R}}{I_z} \end{bmatrix} \delta_R + \begin{bmatrix} \frac{g}{\bar{V}} e_{23}^{WI} \\ 0 \end{bmatrix} \quad (3.4.10)$$

$$B_W = \begin{bmatrix} \frac{\bar{Y}_\beta}{m} & \frac{\bar{Y}_R}{m} \end{bmatrix} \begin{bmatrix} \beta \\ R \end{bmatrix} + \begin{bmatrix} \frac{\bar{Y}_{\delta_R}}{m} \end{bmatrix} \delta_R \quad (3.4.11)$$

### 3.5 Roll Rate Controller

The roll rate dynamics of an aircraft is given by a single, very fast, real pole. The roll rate controller uses the aircraft's ailerons to regulate the roll rate around the  $X_W$ -axis.

#### 3.5.1 Design

The decoupled equations for the roll rate dynamics are given by,

$$\dot{P} = \left[ \frac{L_P}{I_x} \right] P + \left[ \frac{L_{\delta_A}}{I_x} \right] \delta_A \quad (3.5.1)$$

$$P_W = P \quad (3.5.2)$$

An integrator is augmented to the system to remove any steady state errors and counter any cross coupling effects from the directional dynamics as follows,

$$\dot{E}_P = P_W - P_{W_R} \quad (3.5.3)$$

The aileron control law is defined for full pole placement of the roll rate dynamics,

$$\delta_A = -K_P P_W - K_E E_P + \bar{N}_P P_{W_R} + \delta_{A_F} \quad (3.5.4)$$

where  $\delta_{A_F}$  is the feedback linearisation control law derived in section C.2 that will further assist in cancelling some of the weak coupling effects from the lateral specific acceleration (LSA) dynamics into the roll rate system. Equation (C.2.6) defines  $\delta_{A_F}$  as,

$$\delta_{A_F} = \frac{1}{L_{\delta_A}} \left( \left[ -L_\beta \frac{m}{\bar{Y}_\beta} \right] B_W + \left[ L_\beta \frac{\bar{Y}_R}{\bar{Y}_\beta} - L_R \right] R + \left[ L_\beta \frac{\bar{Y}_{\delta_R}}{\bar{Y}_\beta} - L_{\delta_R} \right] \delta_R \right) \quad (3.5.5)$$

The aileron control law of equation (3.5.4) can be substituted into the roll rate dynamics to get the dynamic equations of the closed loop system,

$$\begin{bmatrix} \dot{P}_W \\ \dot{E}_P \end{bmatrix} = \begin{bmatrix} \frac{L_P}{I_x} - \frac{L_{\delta_A}}{I_x} K_P & -\frac{L_{\delta_A}}{I_x} K_E \\ 1 & 0 \end{bmatrix} \begin{bmatrix} P_W \\ E_P \end{bmatrix} + \begin{bmatrix} \frac{L_{\delta_A}}{I_x} \bar{N}_P \\ -1 \end{bmatrix} P_{W_R} \quad (3.5.6)$$

The system's characteristic equation can be calculated from the closed loop system matrix and is given by,

$$\alpha_c(s) = s^2 + s \left( \frac{L_{\delta_A}}{I_x} K_P - \frac{L_P}{I_x} \right) + \frac{L_{\delta_A}}{I_x} K_E \quad (3.5.7)$$

With a desired characteristic equation is defined as,

$$\alpha_c(s) = s^2 + \alpha_1 s + \alpha_0 \quad (3.5.8)$$

the control gains can be solved and are given by [1] as,

$$K_P = \frac{\alpha_1 I_x + L_P}{L_{\delta_A}} \quad (3.5.9)$$

$$K_E = \frac{\alpha_0 I_x}{L_{\delta_A}} \quad (3.5.10)$$

A feed-forward zero can again be placed in the closed loop system using the following equation for  $\bar{N}_P$ ,

$$\bar{N}_P = -\frac{K_E}{z_f} \quad (3.5.11)$$

with  $z_f$  being the location of the desired zero.

### 3.5.2 Pole Placement and Simulation

By similar arguments as used for the NSA controller in section 3.3, the closed loop roll rate pole is placed at the location of its open loop counterpart which is given by,

$$p_P = \frac{L_P}{I_x} \quad (3.5.12)$$

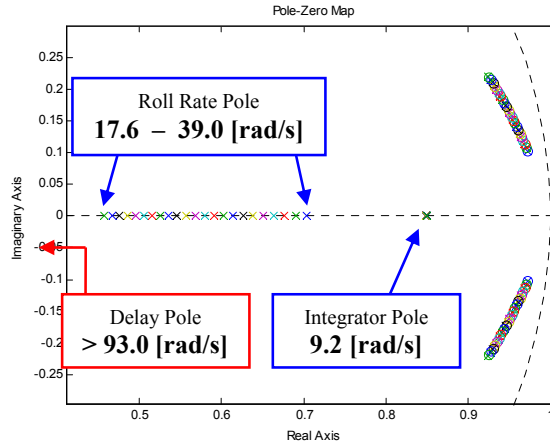
The integrator is not placed at too high a frequency due to practical concerns such as control system delay, but is still placed fast enough for time scale separation to an outer loop control system.

$$p_i = -6.5 \text{ [rad/s]} \quad (3.5.13)$$

The feed-forward zero is once again used to remove most of the integrator dynamics from the system. It is placed at a frequency that is 40% higher than the integrator pole to prevent the possibility of too much feed-forward causing excessive overshoot due to

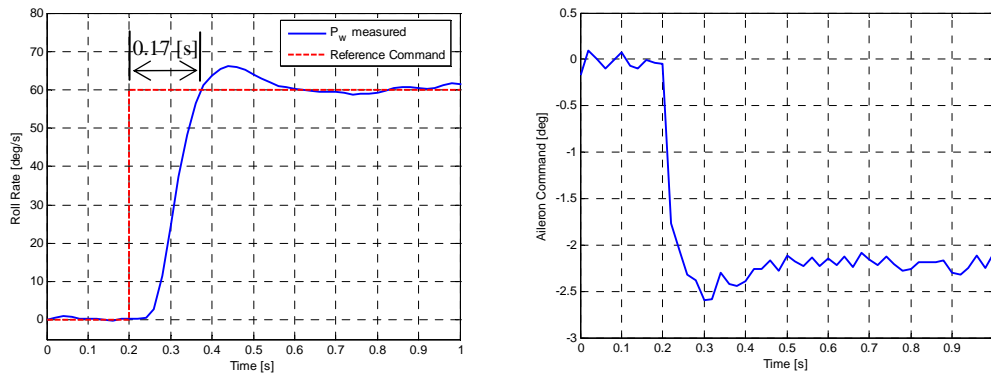
model inaccuracies, as already discussed for the previous inner loop controllers.

In order to ensure that the closed loop poles end up where they have been placed, a Z-plane analysis of the pole placement is done with the one sample period delay of the practical system, discussed in section 3.3.2, modelled in. Figure 3.12 shows the result with the only discrepancy being a slight upward shift in integrator frequency.



**Figure 3.12:** Analysis of CL Pole Deviation due to Controller Delay for the Roll Rate Controller

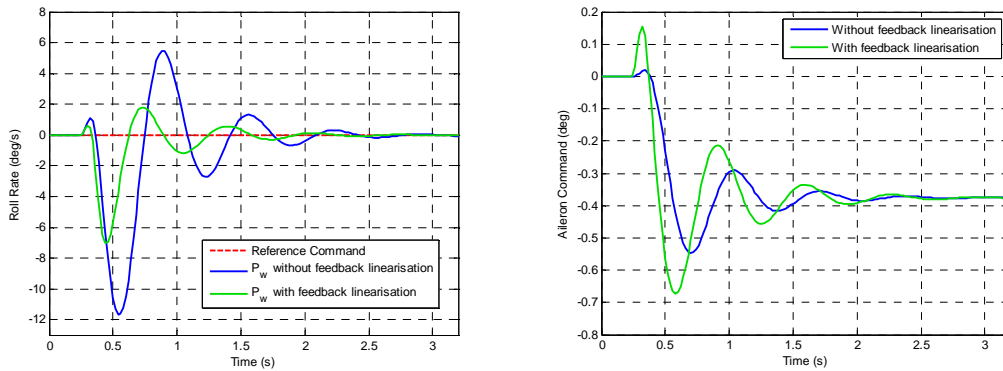
The simulated step response of the roll rate controller using the full non-linear aircraft model is shown in figure 3.13. It can be seen that the system's response is very fast and will not have any difficulty with time scale separation to an outer loop controller.



**Figure 3.13:** Roll Rate Controller Step Response

**Lateral Decoupling Analysis** The coupling of the LSA dynamics into the roll rate control system has to be investigated in order to confirm the validity of any decoupling assumptions that were made during the controller design process. Figure 3.14 shows the effect on the roll rate if a rather large step of 4 degrees is commanded on the aircraft's

rudder. The rudder will directly cause a small roll rate as well as excite the lateral directional dynamics which also have a weak coupling into the roll rate system. This will show the full effect of the coupling between the two sets of dynamics.



**Figure 3.14:** Coupling from the Lateral Directional Dynamics into the Roll Rate

The undamped Dutch Roll mode can be seen in the roll rate response. This rather small oscillation in the roll rate will not greatly affect the system from an outer loop perspective, as the total effect on the roll angle will be very close to zero. Furthermore, when the Dutch Roll mode is damped, as will be the case with the lateral specific acceleration controller, it will cause the coupling effect to die away much faster. If feedback linearisation is also used in the aileron control law, the effect of the coupling can further be reduced as shown in figure 3.14. This shows that feedback linearisation is not necessary for the decoupling of the lateral dynamics but, for a model where the parameters are quite well known, it will help to further reject any coupling effects.

### 3.6 Lateral Specific Acceleration (LSA) Controller

The aircraft's rudder can be used by a control system to attain the following two goals:

1. It can damp the Dutch Roll mode poles which primarily govern the aircraft's yaw rate and sideslip angle dynamics. For normal aircraft these poles are quite lightly damped and increasing their damping will improve the aircraft's LSA response and cause their weak coupling into the roll rate dynamics to be less consequential.
2. The rudder can also be used to coordinate the aircraft's turns by regulating the lateral specific acceleration (LSA) to zero in the steady state and thereby also aiding in the validity of certain model simplification assumptions that were made.

These goals can be attained by creating two separate controllers that use the same actuator. The Dutch Roll damper will add the high frequency portion of the rudder control

signal and the LSA regulator will operate at a much lower frequency. These two frequencies will be so far apart that a time scale separation will exist between the two and the outer acceleration regulator will be able to safely approximate the Dutch Roll dynamics as a simple steady state gain. The control signal will be given by,

$$\delta_R = \delta_{R_{DR}} + \delta_{R_L} \quad (3.6.1)$$

where  $\delta_{R_{DR}}$  is the high frequency Dutch Roll damper signal and  $\delta_{R_L}$  is the low frequency LSA regulator signal. Figure 3.15 provides an overview of the control architecture.

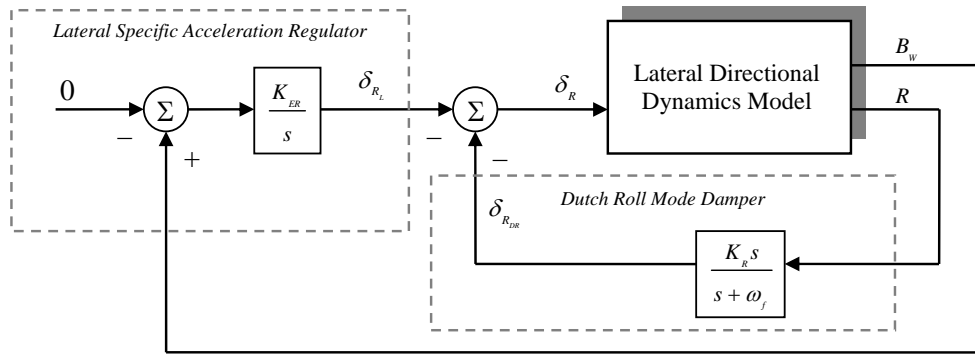


Figure 3.15: Overview of the LSA Control Architecture

### 3.6.1 Dutch Roll Damper Design

For this controller [1] proposes a control architecture where the aircraft's yaw rate is directly fed back into the rudder control signal. This creates a scenario where the frequency of the Dutch Roll mode will remain unchanged while the damping can be selected by a feedback gain. This controller will however actively counter constant yaw rates as produced when the aircraft is in a coordinated turn. Therefore this controller can not be used individually and has to be used in conjunction with the lateral specific acceleration regulator which provides the correct steady state rudder deflection for turn coordination.

A more conventional control architecture is adapted from [3] instead, where a high pass Dutch Roll damper filter for the yaw rate signal is used in the feedback loop to the rudder control signal. The transfer function of this filter is given by,

$$D(s) = \frac{K_R s}{s + \omega_f} \quad (3.6.2)$$

Here  $\omega_f$  is the filter's cut-off frequency and  $K_R$  is its feedback gain. This filter will cause the controller to counter any higher frequency Dutch Roll effects, but will filter out and ignore the low frequency effects of constant turn rates. This controller can therefore still

be used in conjunction with the lateral specific acceleration regulator for turn coordination, but can also function independently without actively countering constant turn rates. The open loop damping of the Dutch Roll mode poles for the aircraft used in this project is given by,

$$\zeta_{OL} = 0.209 \quad (3.6.3)$$

and is clearly very lightly damped. Figure 3.16 shows the root locus plot for the variation in the filter feedback gain for the Dutch Roll damper design at the aircraft's trim velocity ( $\bar{V} = 30$  m/s). The gain has been selected to place the Dutch Roll mode poles at a damping ratio of,

$$\zeta_{CL} = 0.65 \quad (3.6.4)$$

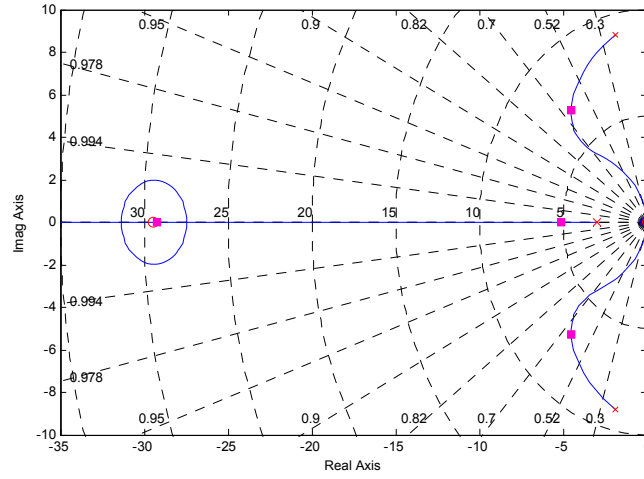


Figure 3.16: Dutch Roll Damper Root Locus Design

The cut-off frequency of the filter is chosen at one third the frequency of the open loop poles to ensure that the Dutch Roll mode response lies in the filter's passband. The filter's transfer function is therefore given by,

$$D_{Trim}(s) = \frac{-0.0706s}{s + 3} \quad (3.6.5)$$

Over the large velocity envelope that the aircraft's control system should be able to function, the poles and zeros of the lateral dynamics will scale up and down in frequency in a similar manner as the *Short Period mode* poles of the aircraft's NSA dynamics (see section 3.3). Because velocity applies a global scaling factor to all the poles and zeros of the LSA dynamics and does not affect their relative scaling with each other, the following



general equation for the Dutch Roll damper over all velocities can be adopted.

$$D(s) = \frac{K'_R}{\eta_d \omega_{DR}} \frac{s}{(s + \eta_d \omega_{DR})} \quad (3.6.6)$$

Here  $\omega_{DR}$  is the frequency of the open loop Dutch Roll mode,  $\eta_d$  is the fraction of this frequency where the filter will cut off, and  $K'_R$  is the normalised feedback gain. From equation (3.6.5) for the filter's design at the trim velocity, the above coefficients are easily attained as,

$$\eta_d = 1/3 \quad (3.6.7)$$

$$K'_R = -0.212 \quad (3.6.8)$$

An equation for the frequency of the open loop Dutch Roll mode can be obtained from the characteristic equation of the open loop decoupled LSA dynamics.

$$\alpha_c(s) = s^2 + \left[ -\frac{N_R}{I_z} - \frac{\bar{Y}_\beta}{m\bar{V}} \right] s + \frac{1}{I_z} \left[ \frac{\bar{Y}_\beta}{m\bar{V}} N_R + N_\beta - N_\beta \frac{\bar{Y}_R}{m\bar{V}} \right] \quad (3.6.9)$$

Therefore,

$$\omega_{DR} = \sqrt{\frac{1}{I_z} \left[ \frac{\bar{Y}_\beta}{m\bar{V}} N_R + N_\beta - N_\beta \frac{\bar{Y}_R}{m\bar{V}} \right]} \quad (3.6.10)$$

Using equation (3.6.6) as the transfer function for the Dutch Roll damper filter will ensure the damping of the Dutch Roll dynamics at any given velocity. Figure 3.17 shows the result of this improved damping on the aircraft's yaw rate. A step is super imposed on the rudder control signal in order to excite the Dutch Roll mode dynamics. It can be seen how the damping of these dynamics are greatly improved by the control system.

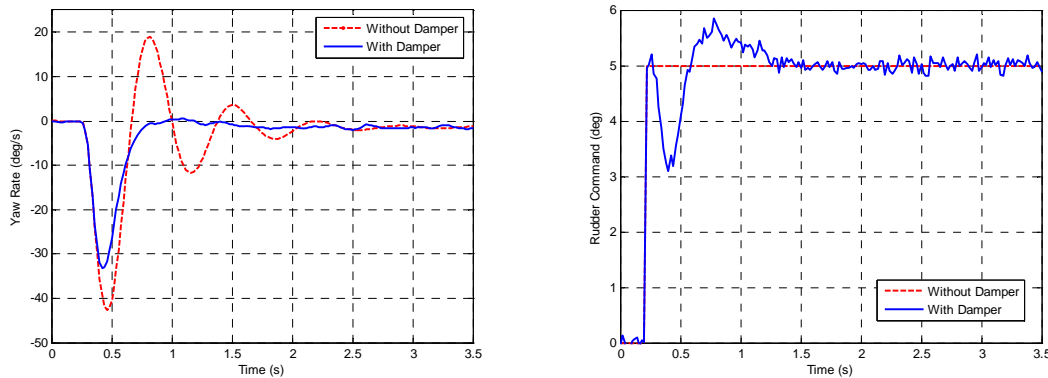
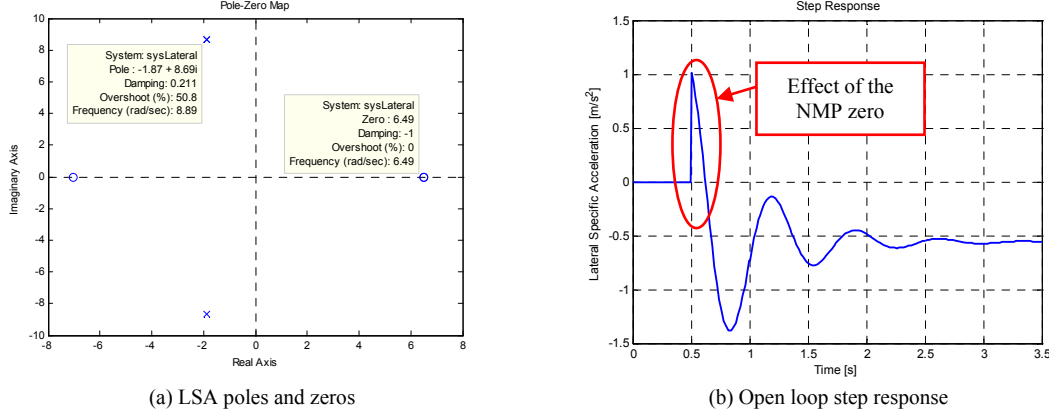


Figure 3.17: Damping of the Dutch Roll Mode

### 3.6.2 Lateral Specific Acceleration Regulator Design

The second design goal of the lateral controller is to coordinate the aircraft's turns by regulating the LSA to zero in the steady state.



**Figure 3.18:** Analysis of the Lateral Specific Acceleration Dynamics

Consider the poles and zeros of the decoupled LSA dynamics shown in figure 3.18 (a). The dynamics are similar in nature to those that govern the NSA (see section 3.3) but, due to the smaller lift surface available to the lateral system, the effect of the non-minimum phase zero is much greater. This effect, caused by the induced lateral force due to rudder deflection can be seen in the system's open loop response, shown in figure 3.18 (b).

It can be seen that the NMP zero is in fact at a lower frequency than the Dutch Roll mode poles. Therefore, the assumption to ignore the effect of this zero that was made for the NSA controller can not be made here, as its frequency is not three times faster than that of the poles. It is therefore proposed by [1] to create a much slower controller that will be time scale decoupled from these dynamics. This will greatly simplify the control architecture and will be sufficient as a fast response is not required (or possible due to the NMP zero) from this controller. If the closed loop poles are placed at a much slower frequency than these dynamics, the system can be approximated by a steady state gain as follows,

$$B_W \approx K_{SS} \delta_{R_L} \quad (3.6.11)$$

where  $\delta_{R_L}$  is the rudder deflection commanded by the LSA regulator. This steady state gain can be determined from the transfer function of the rudder to the lateral specific acceleration and is given by [1] as,

$$K_{SS} = \frac{Y_{\delta_R}}{m} \frac{Y_{\beta}}{I_Z} \left( \frac{N_{\beta}}{Y_{\beta}} - \frac{N_{\delta_R}}{Y_{\delta_R}} \right) \frac{1}{\omega_{DR}^2} \quad (3.6.12)$$

with  $\omega_{DR}$  being the natural frequency of the open loop Dutch Roll mode and is given by equation (3.6.10). Because the Dutch Roll damper from the previous section does not change the steady state gain of the system, the above equation will hold whether the damper is in use or not.

An integrator can be augmented to the system in order to regulate the LSA to zero in the steady state.

$$\dot{E}_B = B_W - B_{WR} \quad (3.6.13)$$

The  $B_{WR}$  term is the LSA reference and will always be commanded to zero. In order to place the single integrator pole, the control law is defined as,

$$\delta_{R_L} = -K_E E_B \quad (3.6.14)$$

By substituting the control law into the LSA dynamics, the closed loop system is obtained as,

$$\dot{E}_B = [-K_{SS}K_E] E_B + [-1] B_{WR} \quad (3.6.15)$$

with characteristic equation,

$$\alpha_c(s) = s + K_{SS}K_E \quad (3.6.16)$$

The single real pole can be placed by solving the control gain,

$$K_E = \frac{\alpha_0}{K_{SS}} \quad (3.6.17)$$

with  $\alpha_0$  being the final term in a desired first order characteristic equation of the closed loop system. This integrator pole can be placed as a function of the open loop Dutch Roll frequency and thereby a constant degree of separation is kept between these dynamics.

$$p_i = -\eta_i \omega_{DR} \quad (3.6.18)$$

with  $\eta_i$  chosen as,

$$\eta_i = 1/12 \quad (3.6.19)$$

The system's response can now be analysed. The same rudder disturbance is superimposed on the control signal as with the Dutch Roll test in the previous section. The response in figure 3.19 shows the Dutch Roll damper and lateral specific acceleration regulator working together to damp the induced Dutch Roll disturbance and then return the aircraft's LSA to zero. It can be noted from the yaw rate graph how the damping of the Dutch Roll mode has decreased somewhat with the addition of the second controller. This is due to the inaccuracies introduced by the time scale separation assumption that was made in the design of the LSA regulator. The damping of this mode is still much higher than it would be without the controller and is still deemed acceptable.

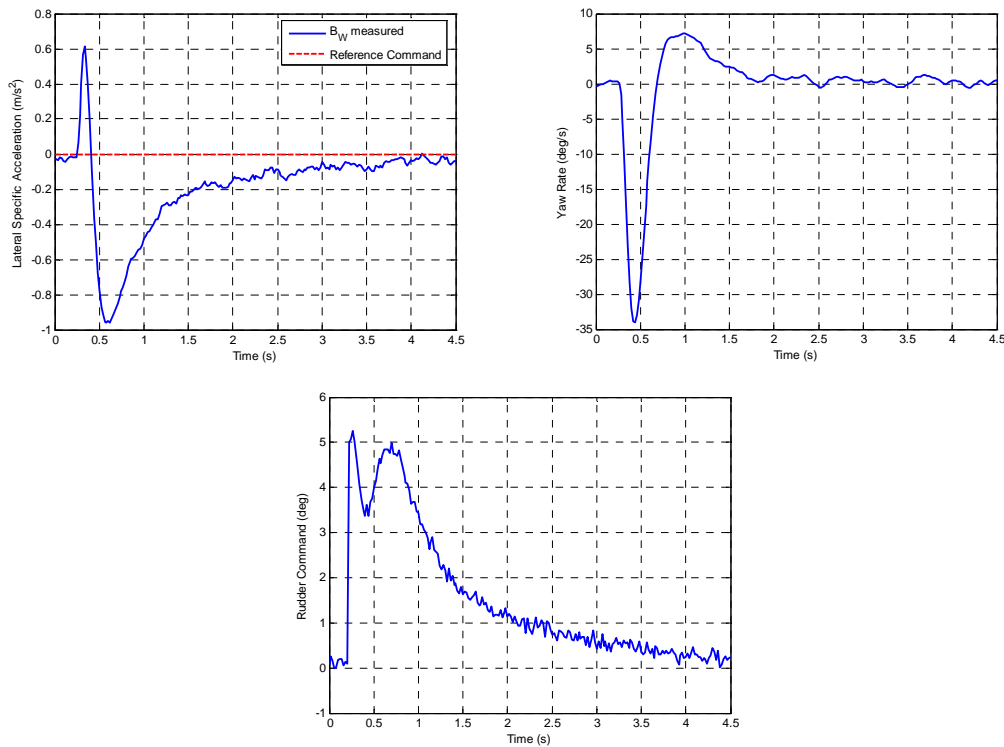


Figure 3.19: LSA Regulator Disturbance Rejection Analysis

### 3.6.3 Review

The controllers from the previous two subsections are intended to work together but can also be used individually, if required. The LSA regulator will however not function well without the Dutch Roll damper, because it removes some damping from the Dutch Roll mode and the open loop poles are already very lightly damped.

During normal flight for an aerobatic aircraft such as the one used in this project, a pilot will rarely use the aircraft's rudder. This is because such an aircraft will naturally fly turns that are very close to coordinated and the Dutch Roll dynamics, however lightly damped, are still stable. Therefore it would be possible for the other aircraft controllers to function without these lateral controllers being armed. Each of these controllers do however assist in improving the flight control of the aircraft.

## 3.7 Obtaining the Wind Axes Measurements

For all the controllers developed in this chapter, feedback from accelerations in wind axes as well as the roll rate about the velocity vector are used. A problem that therefore presents itself is obtaining the wind axes specific accelerations and roll rate from those measured in the aircraft's body axes. For the roll rate the assumption has already been made to use the body axis rate directly. But for the specific accelerations, a body to wind

axes conversion is required. The DCM can be used to relate the body and wind axes measurements as follows,

$$\mathbf{A}_W^{WI} = \mathbf{DCM}^{WB} \mathbf{A}_B^{WI} \quad (3.7.1)$$

Here  $\mathbf{DCM}^{WB}$  is the transformation matrix given by equation (B.6.7). The aerodynamic side force does not significantly effect the body axes measurements as it is always regulated to zero through the control architecture. Therefore  $\beta$  can be approximated as zero in the above transformation matrix. This simplifies the conversion to,

$$\begin{bmatrix} A_W \\ B_W \\ C_W \end{bmatrix} = \begin{bmatrix} \cos \alpha & 0 & \sin \alpha \\ 0 & 1 & 0 \\ -\sin \alpha & 0 & \cos \alpha \end{bmatrix} \begin{bmatrix} A_B \\ B_B \\ C_B \end{bmatrix} \quad (3.7.2)$$

It can be seen that a small component of the wind axes NSA will be measured by the body axes axial accelerometer on the aircraft. Should the body axes accelerations have been used without conversion by the controllers, this component would have caused a significant error on ASA as the lift force is typically much larger than the thrust force.

An equation for  $\alpha$  that is required by the conversion of equation (3.7.2) can be derived by rearranging equation (3.3.9) for the simplified output equation of the NSA dynamics when  $C_{L_0} = 0$ ,

$$\alpha = \left[ -\frac{m}{\bar{L}_\alpha} \right] C_W \quad (3.7.3)$$

The relationship between the wind axes NSA and the body axes measurements can be substituted from equation (3.7.2).

$$\alpha = \left[ -\frac{m}{\bar{L}_\alpha} \right] (-\sin \alpha A_B + \cos \alpha C_B) \quad (3.7.4)$$

By making small angle assumptions for  $\alpha$  this becomes,

$$\alpha = \left[ -\frac{m}{\bar{L}_\alpha} \right] (-\alpha A_B + C_B) \quad (3.7.5)$$

Because lift is usually the largest force acting on the aircraft and it is mainly measured on the body axes normal accelerometer, the product of the small angle,  $\alpha$  and the axial acceleration can also be ignored in the above equation.

$$\alpha = \left[ -\frac{m}{\bar{L}_\alpha} \right] C_B \quad (3.7.6)$$

This estimate for  $\alpha$  depends greatly on the accuracy of the aerodynamic coefficients of the aircraft. However, AVL can usually provide these coefficients to a relative degree of accuracy. Therefore it is better to remove some of the lift from the wind axes ASA with an estimated  $\alpha$  rather than not at all.

### 3.8 Review of the Inner Loop Controllers

Through the implementation of all the inner loop controllers developed in this chapter, the aircraft is reduced to a point mass with its axial and normal specific accelerations as well as its roll rate being fully commandable. *All attitude dependance has been fully removed* from the system by these controllers through techniques such as the feedback linearisation of the effect of gravity in the NSA controller. A time scale decoupled outer loop controller can use these three inner loop commands to regulate the kinematics of the aircraft onto a predetermined flight trajectory. Figure 3.20 illustrates this graphically.

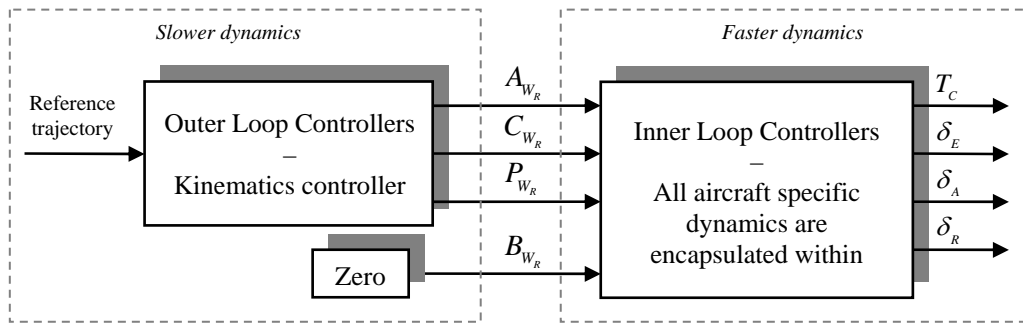


Figure 3.20: Overview of the Full Control Architecture

In order to design a time scale decoupled outer loop controller, its poles should be at least 5 times slower than the slowest inner loop pole. For the NSA and roll rate controllers it can be seen that the slowest pole is placed at 6.5 rad/s. This places an upper limit on the frequency of the outer loop poles of,

$$\omega_O \leq 1.3 \text{ [rad/s]} \quad (3.8.1)$$

where  $\omega_O$  is the frequency of any pole in the outer loop system that has to do with commanding these controllers. The axial specific acceleration controller's poles are however much slower and are located at 1.05 rad/s. They can therefore unfortunately not be time scale decoupled from the outer loop dynamics. The dynamics of the ASA controller should therefore be taken into account in the design of any outer loop control system.

### 3.9 Practical Results and Testing

The inner loop controllers developed in this chapter have to be tested practically. The CAP232 model aircraft used in this project is equipped with an OBC avionics package (see Appendix E). The controllers can be implemented discretely on the OBC using direct emulation. The 50 Hz sample frequency of the hardware is sufficiently fast for this.

**Hardware in the Loop** The functionality of these controllers can be tested using the hardware in the loop simulator developed by [15]. The HIL simulator creates simulated sensor signals using the non-linear simulator discussed in section 2.5. The avionics receives these measurements and creates actuator signals using the pre-programmed control algorithms. These actuator signals are then fed back to the simulator and the virtual aircraft then responds accordingly. This allows the developer to fully test the functionality of the controllers implemented on the OBC hardware in a software simulation environment. Once the controllers are deemed ready, practical tests can be conducted. For additional clarity on the HIL simulation setup an overview is provided in figure 3.21.

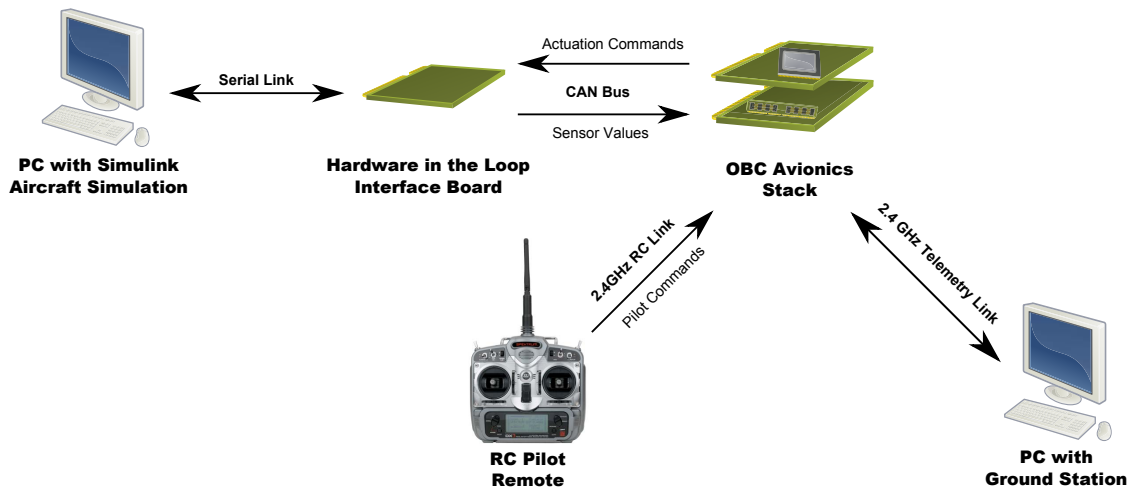


Figure 3.21: Overview of the Hardware in the Loop Simulation Setup

### 3.9.1 Axial Specific Acceleration Controller Step Response Test

For the practical testing of the ASA controller the pilot was given complete control of the aircraft's aerodynamic control surfaces, while the ASA controller was allowed to command the aircraft's throttle. The pilot was asked to fly the aircraft in a straight line while a controller command was issued from the ground station requesting an ASA step from  $0 \text{ m/s}^2$  to  $1.5 \text{ m/s}^2$ . The resulting response is shown in figure 3.22.

From the graph it can be seen that in practice the ASA controller's step response takes somewhat longer to achieve its commanded value than in the simulations. This further highlights the non-linear nature and extremely limited bandwidth of the aircraft's engine model. However, due to the robustness built into the controller design architecture, a stable and well damped response was still achieved.

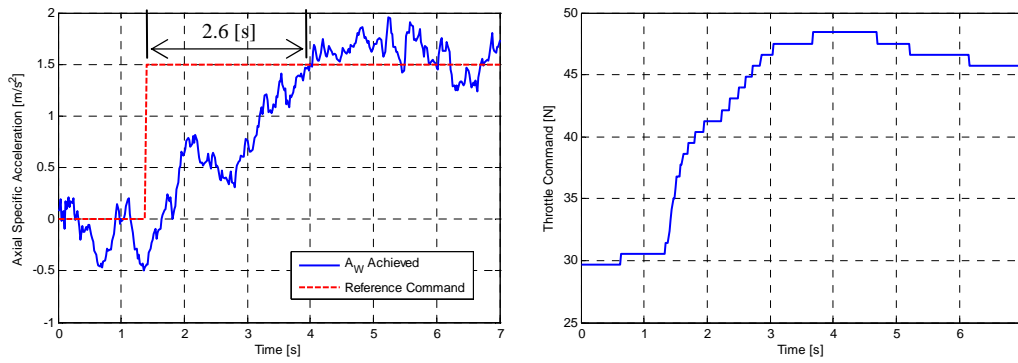


Figure 3.22: Practical Response of the ASA Controller

### 3.9.2 Normal Specific Acceleration and Roll Rate Fly-by-Wire Test

The NSA and roll rate controllers are the two most important inner loop controllers, since they will be responsible for guiding the aircraft along its flight trajectory. In order to verify their functionality, an initial test was conducted where the safety pilot was able to command these controllers via the RC remote. The control hardware was set up to intercept the signals for elevator and aileron from the RC receiver and generate NSA and roll rate commands accordingly. The pilot was therefore able to fly the plane through these inner loop controllers. In order for the aircraft to retain the same sensitivity when these controls are perturbed, the gains used to calculate the controller commands from the elevator and aileron signals were carefully calibrated during pre-flight HIL testing.

Figure 3.23 shows the results from this test. It can be seen that the commanded values were achieved relatively quickly and therefore the stability of both controllers were verified. On the NSA controller's response however, some oscillation can be observed. The cause of this effect is thought to be the extensive NSA feedback present in this control system which attempts to alter the natural frequency of the Short Period mode. This result led to the development of the pitch rate damping NSA controller – see section C.1.

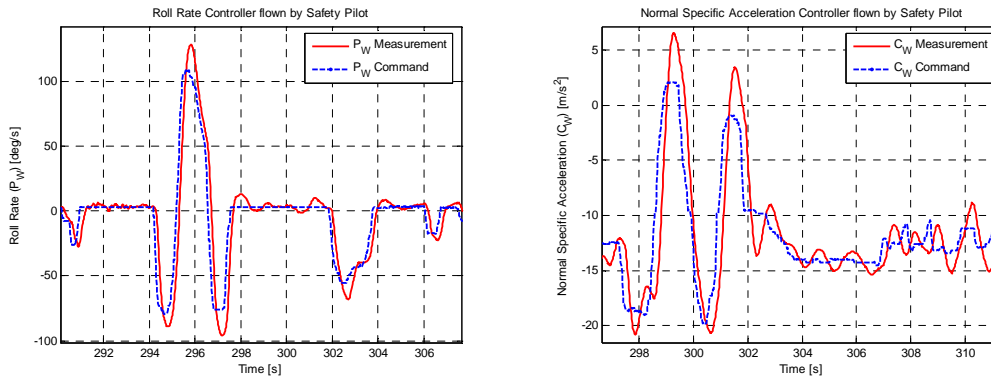


Figure 3.23: Results of Safety Pilot Commanding Inner Loop Controllers



### 3.9.3 Normal Specific Acceleration Controller Step Response Test

The goal of this test is to analyse the practical step response of the NSA controller. This analysis is paramount to the success of the aircraft control system being designed. It has already been mentioned in section 3.8 that the inner loop NSA and roll rate controllers will be time scale decoupled from any outer loop kinematic controllers that will be designed. It is therefore important to verify that the practical response speed of these controllers are similar to the simulated results obtained during their design.

For the step response of the NSA controller the safety pilot was given control of all the aircraft's actuators, except for its elevator, which was commanded by the inner loop NSA controller. The pilot was instructed to fly straight and level at the aircraft's trim velocity before a ground station command was issued to step the NSA controller from 1 g to 2 g. The aircraft therefore entered a relatively steep climb for a few seconds whereafter the safety pilot retook full control.

The result of the step response is shown in figure 3.24, where it can be seen that a very satisfactory result was obtained. In section 3.3 the simulated rise time of this controller was found to be 0.25 s at the aircraft's trim velocity. The practical results obtained from this test show a rise time of 0.3 s with minimal overshoot. The small amount of non-minimum phase present in the aircraft's NSA dynamics (see section 3.3) can also be observed in this practical step response. The high speed with which this controller is able to respond in practice verifies that any time scale decoupling assumptions made during the design of the outer loop kinematic controllers will be valid.

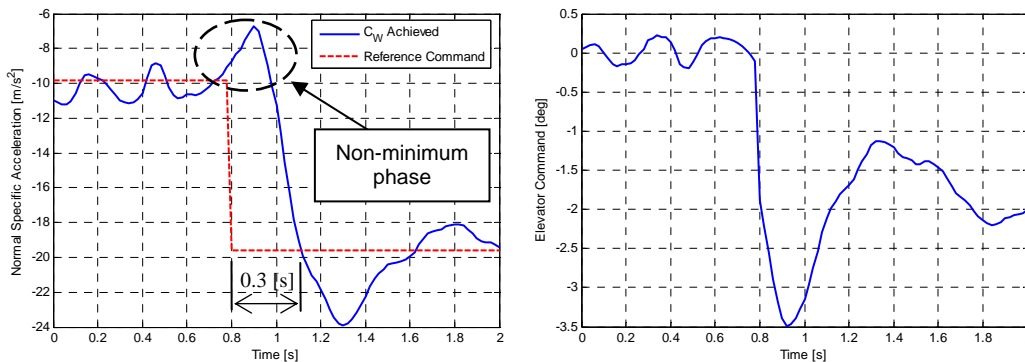


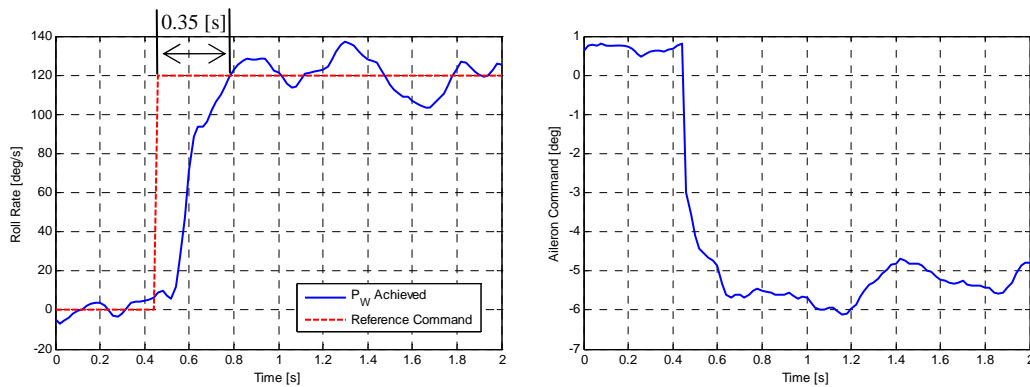
Figure 3.24: Practical Response of the Normal Specific Acceleration Controller

### 3.9.4 Roll Rate Controller Step Response Test

Similar to the NSA controller, the response speed of the roll rate controller is important in order to verify that the time scale decoupling of this controller from the outer loop kinematic controllers will be a valid assumption. In order to obtain this step response the

pilot was given control of all the aircraft's actuators, except for its ailerons, which were commanded by the roll rate controller. A ground station roll rate step of 120 deg/s was issued to this controller while the aircraft was flying in a straight and level manner.

The practical step response of the roll rate controller is shown in figure 3.25. It can be seen how a rise time of 0.35 s was attained with approximately 80% of the commanded value being achieved in under 0.2 s. This result is slightly slower than the simulated rise time of 0.17 s shown in section 3.5. It is however still fast enough to time scale decouple this controller to an outer loop control system. The robustness of this control architecture to wind and other disturbances can also be observed from this step response. Due to the high bandwidth nature of this controller, the roll rate is kept very close to the commanded value during the entire roll and a maximum error of only 18 deg/s is observed.



**Figure 3.25:** Practical Response of the Roll Rate Controller

### 3.9.5 Dutch Roll Damper Test

As stated in section 3.6, the aircraft's rudder can be used to provide additional damping to the Dutch Roll mode as well as ensure coordinated flight. The Dutch Roll damper was designed to achieve the first of these two goals. In order to verify the practical success of this controller, the open loop Dutch Roll response of the actual aircraft has to be analysed first. This mode can be excited by perturbing the aircraft's rudder and observing the resulting yaw rate and LSA responses.

Figure 3.26 shows the result obtained when this mode was excited while no controllers were active. As indicated on the graph, some overshoot can be observed in both LSA and yaw rate. This overshoot is however much less than the AVL predicted damping ratio of the Dutch Roll mode would suggest. It would however still be desirable to completely remove this overshoot from the response.

For the test of the Dutch Roll damper the rudder commands issued by the safety pilot were superimposed on the commands issued by this controller. The pilot was therefore

still able to command the aircraft's rudder, this controller simply assisted the pilot by damping any unwanted Dutch Roll effects. The aircraft's Dutch Roll Mode was therefore again excited in the same manner as before and the resulting damping of this mode was observed. Figure 3.27 shows the obtained result where it can clearly be seen that the damping of the Dutch Roll mode was significantly improved. This result verifies the functionality and success of this controller.

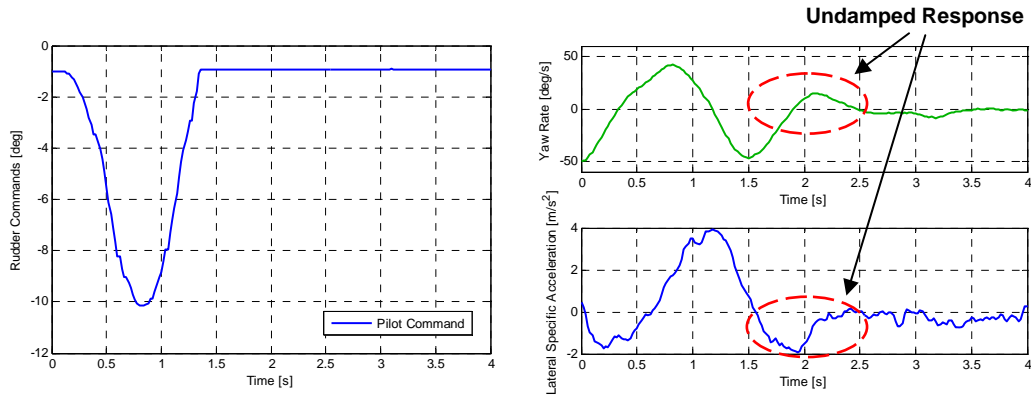


Figure 3.26: Open Loop Practical Response of the Dutch Roll Mode

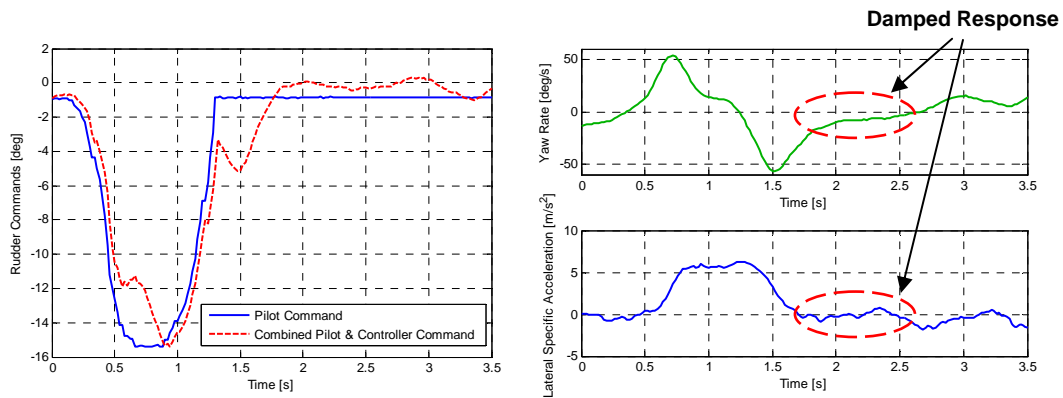


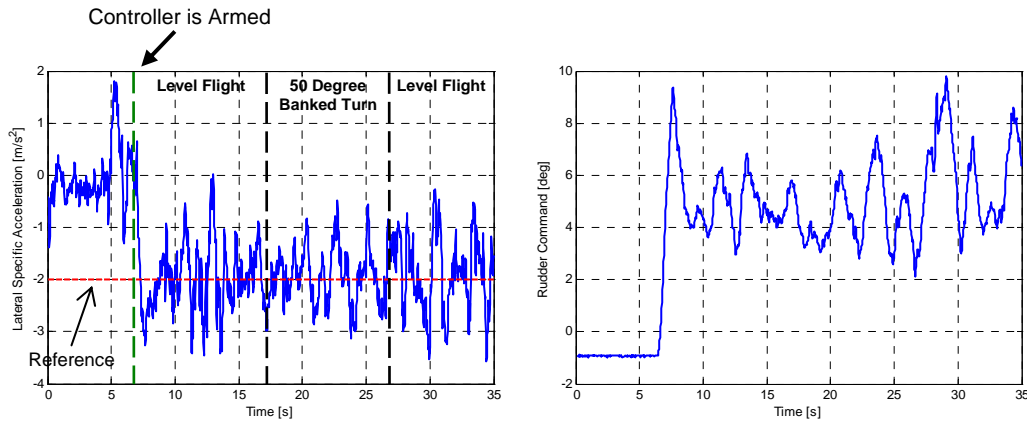
Figure 3.27: Practical Response of the Dutch Roll Damper

### 3.9.6 Lateral Specific Acceleration Regulator Test

In addition to damping the Dutch Roll mode, the aircraft's rudder can also be used to coordinate the aircraft's turns in the steady state. The LSA regulator (see section 3.6.2) was designed to obtain this goal. This controller consists of an integrator regulating LSA to zero in the steady state.

Since the aircraft is already designed to fly in a relatively coordinated manner, a practical result of this controller will show almost no change to the aircraft's LSA. In order to obtain a clear practical result, it was decided to artificially command a small offset on LSA. Instead of coordinating the aircraft's flight it was decided to command an LSA of  $-2 \text{ m/s}^2$ . Figure 3.28 shows this result where it can be seen that as soon as the controller is armed, the aircraft's rudder is perturbed in order to regulate its LSA to  $-2 \text{ m/s}^2$ . For this test the Dutch Roll damper was also armed and therefore any Dutch Roll dynamics that were excited by this controller were damped by the Dutch Roll damper.

From figure 3.28 it can be seen that steady state regulation of the aircraft's LSA was successfully achieved. During the test the pilot was instructed to bank the aircraft to a relatively large angle in order to observe the controller's ability to coordinate the aircraft's turns. It can be seen in figure 3.28 how the aircraft's LSA was regulated to its commanded value during this high bank angle turn, as well as during level flight.



**Figure 3.28:** Practical Response of the Lateral Specific Acceleration Regulator

### 3.9.7 Review

From the practical results obtained in this section, it can be concluded that the inner loop controllers all operate successfully in a real-world environment. The practical responses obtained were very similar to those seen during simulations. This both proves that these controllers function correctly as well as verifies that the non-linear model used for aircraft simulation provides a relatively accurate environment for control system testing.

The response speed of the inner loop NSA and roll rate controllers were also verified to be fast enough for successful time scale decoupling to an outer loop kinematic control system. These results therefore enable the design and analysis of various kinematic controllers which will command the aircraft through the provided inner loop acceleration and roll rate interfaces. This will be the topic of discussion for the next two chapters.

## Chapter 4

# Continuous Re-Linearisation LQR Kinematic Controller

With the inner loop controllers designed, the aircraft's dynamics have been greatly simplified from a control perspective. The problem of controlling the aircraft's attitude, position and velocity has become that of a point mass with its axial and normal specific accelerations as well as roll rate being commandable. The dynamics that describe a system such as this have been derived in section 2.3 and are non-linear in nature. These dynamics are very well modelled, as they are only of a kinematic nature and the only uncertainties that exist in this model are the gravity vector as well as the small errors stemming from the assumption that the inner loop commands are instantly attainable.

This chapter investigates the first method of kinematic control used in this project, namely *continuous re-linearisation LQR control*. It is a method whereby the system's non-linear dynamics are continually re-linearised about a reference trajectory. A linear quadratic regulator is then used to place the system's poles and create stable feedback gains.

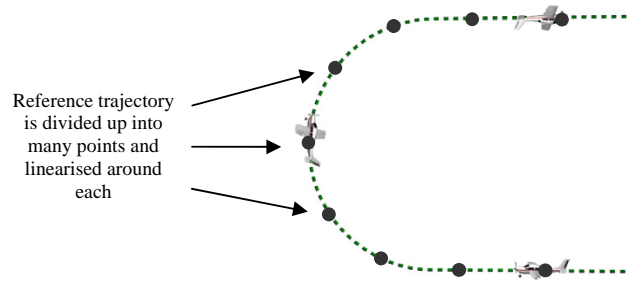
This chapter will start by discussing the general strategy used for the controller and then show the design process for this specific control problem. Various aspects of the implementation of the control system will be discussed, such as weighting matrix selection and LQR feedback gain settling. Simulation as well as practical implementation results of the controller will also be shown.

### 4.1 Design Architecture

As mentioned before, this controller will continually re-linearise the non-linear outer loop dynamics about the aircraft's current point on a predetermined reference trajectory as illustrated by figure 4.1. These linearised dynamics can then be used by a linear quadratic regulator to create stable and optimal feedback gains for a set of specified control and state weighting matrices. These feedback gains will have to be calculated at every re-linearisation interval which will become somewhat computationally demand-

ing as the solving of these gains is not a trivial operation. This is however made slightly simpler by the fact that the outer loop dynamics only model's the aircraft's slower dynamics and therefore a longer sample period will still be acceptable. Furthermore, the state and control vectors will not be as large as those used in other projects and therefore the matrix inversions will be simpler to solve – see section 7.2.

An LQR control strategy is also adopted by [16] and [15]. In both these projects the aircraft's entire model is linearised and an LQR algorithm is used to place all of the aircraft's closed loop poles. This creates difficulty in obtaining good performance from the system as both the inner and outer loop dynamics are encapsulated in one large model. All the uncertainty of the inner loop parameters are allowed to affect the system up to a guidance level. In this project an LQR controller is only used for the well modelled outer loop dynamics, with the inner loop controllers desensitising the system to model uncertainty. Due to the smaller scope of the model being controlled, the reduction in design insight that usually accompanies an LQR control architecture is reduced.



**Figure 4.1:** Continuous Re-linearisation of the Reference Trajectory

#### 4.1.1 Linearising About the Reference Trajectory

The purpose of the outer loop guidance controller is to ensure that the aircraft remains on a predetermined flight trajectory. Should the aircraft stray from this flight path, the controller will regulate it back through its closed loop error regulation dynamics. To derive a controller that does this, start by defining the state vector for the outer loop dynamics as  $\underline{x}$ . Its dynamic equations are given by,

$$\dot{\underline{x}} = f(\underline{x}, \underline{u}) \quad (4.1.1)$$

where  $\underline{u}$  is the outer loop control vector. Then suppose that  $\underline{x}$  and  $\underline{u}$  can be split into a value at a given reference trajectory and an error or deviation from this reference.

$$\underline{x} = \underline{x}_R + \underline{x}_E \quad (4.1.2)$$

$$\underline{u} = \underline{u}_R + \underline{u}_E \quad (4.1.3)$$

Equation (4.1.1) now becomes,

$$\dot{\underline{x}} = f([\underline{x}_R + \underline{x}_E], [\underline{u}_R + \underline{u}_E]) \quad (4.1.4)$$

The Taylor series expansion of equation (4.1.4) about the reference trajectory is given by,

$$\dot{\underline{x}} = f(\underline{x}_R, \underline{u}_R) + \left. \frac{df(\underline{x}, \underline{u})}{d\underline{x}} \right|_{\substack{\underline{x}=\underline{x}_R \\ \underline{u}=\underline{u}_R}} (\underline{x} - \underline{x}_R) + \left. \frac{df(\underline{x}, \underline{u})}{d\underline{u}} \right|_{\substack{\underline{x}=\underline{x}_R \\ \underline{u}=\underline{u}_R}} (\underline{u} - \underline{u}_R) + [\text{H.O.T.}] \quad (4.1.5)$$

where H.O.T. stands for the higher order Taylor series terms. Removing these higher order terms and substituting equations 4.1.2 and 4.1.3 yields,

$$\dot{\underline{x}}_R + \dot{\underline{x}}_E \simeq f(\underline{x}_R, \underline{u}_R) + \left. \frac{df(\underline{x}, \underline{u})}{d\underline{x}} \right|_{\substack{\underline{x}=\underline{x}_R \\ \underline{u}=\underline{u}_R}} \underline{x}_E + \left. \frac{df(\underline{x}, \underline{u})}{d\underline{u}} \right|_{\substack{\underline{x}=\underline{x}_R \\ \underline{u}=\underline{u}_R}} \underline{u}_E \quad (4.1.6)$$

This can become,

$$\dot{\underline{x}}_E = \mathbf{F}\underline{x}_E + \mathbf{G}\underline{u}_E \quad (4.1.7)$$

with,

$$\mathbf{F} = \left. \frac{df(\underline{x}, \underline{u})}{d\underline{x}} \right|_{\substack{\underline{x}=\underline{x}_R \\ \underline{u}=\underline{u}_R}} \quad (4.1.8)$$

$$\mathbf{G} = \left. \frac{df(\underline{x}, \underline{u})}{d\underline{u}} \right|_{\substack{\underline{x}=\underline{x}_R \\ \underline{u}=\underline{u}_R}} \quad (4.1.9)$$

if the reference trajectory dynamics satisfy the outer loop dynamic equations,

$$\dot{\underline{x}}_R = f(\underline{x}_R, \underline{u}_R) \quad (4.1.10)$$

This means that the reference trajectory should be kinematically feasible<sup>1</sup>. If however a small error in the trajectory is made and thereby the above constraint equation is violated, the resulting state errors will simply be treated as a disturbance and be removed by the closed loop error regulation dynamics. Larger trajectory errors will however start to affect controller stability.

The linear state and input matrices ( $\mathbf{F}$  and  $\mathbf{G}$ ) for the error regulation dynamics have now been determined to be the *Jacobian matrices* of the outer loop dynamics. They can now be calculated by determining the partial derivatives of all the state equations with respect to all the states and control inputs. This is a trivial task once the state equations have been defined.

---

<sup>1</sup>The reference trajectory should be defined as the result of a point mass with accelerations and roll rates applied to it.

### 4.1.2 LQR Control Algorithm

Because the open loop dynamics constantly change with every re-linearisation of the dynamic equations, a method is required to place the poles of the closed loop dynamics. A discrete linear quadric regulator can be used to place stable and optimal closed loop poles for the system. The LQR algorithm is designed to find an optimal control input,  $\underline{u}_E(k)$ , that will minimise a quadratic cost function that, for this system, is defined as,

$$J = \frac{1}{2} \sum_{k=0}^{N-1} \left[ \underline{x}_E(k)^T \mathbf{Q}_1 \underline{x}_E(k) + \underline{u}_E(k)^T \mathbf{Q}_2 \underline{u}_E(k) \right] \quad (4.1.11)$$

where  $\mathbf{Q}_1$  and  $\mathbf{Q}_2$  are diagonal, positive definite matrices that assign relative weightings to each of the states and control inputs. By increasing a diagonal element in  $\mathbf{Q}_1$  for instance, its corresponding state will generate more relative cost and will therefore be regulated to zero more aggressively. These matrices will therefore indirectly affect the resulting closed loop pole placement, which must always conform to the time scale separation constraint of equation (3.8.1). For the discrete LQR algorithm this cost function is calculated over a finite horizon of  $N$  time steps. The selection of  $N$  as well as the weighting matrices  $\mathbf{Q}_1$  and  $\mathbf{Q}_2$  will be shown in section 4.3.

The solution to this optimal control problem is given by the *Riccati difference equation*, where for any time when  $k$  is not close to the end of the horizon  $N$ , the optimal control law can be approximated by constant, linear state feedback as shown below.

$$\underline{u}_E(k) = -\mathbf{K} \underline{x}_E(k) \quad (4.1.12)$$

The steady state feedback gain matrix  $\mathbf{K}$  can be calculated by solving the Riccati difference equation backwards in time ( $k = N..0$ ) until the feedback gains have settled. The iterative algorithm is given by [9] as,

$$\mathbf{M}(k) = \mathbf{S}(k) - \mathbf{S}(k) \left[ \mathbf{Q}_2 + \mathbf{\Gamma}^T \mathbf{S}(k) \mathbf{\Gamma} \right]^{-1} \mathbf{\Gamma}^T \mathbf{S}(k) \quad (4.1.13)$$

$$\mathbf{S}(k-1) = \mathbf{\Phi}^T \mathbf{M}(k) \mathbf{\Phi} + \mathbf{Q}_1 \quad (4.1.14)$$

with initial condition,

$$\mathbf{S}(N) = \mathbf{Q}_1 \quad (4.1.15)$$

where  $\mathbf{\Phi}$  and  $\mathbf{\Gamma}$  are the discrete state and input matrices respectively. At the final time step, the feedback gain matrix  $\mathbf{K}$  can be solved using the following equation,

$$\mathbf{K}(k-1) = \left[ \mathbf{Q}_2 + \mathbf{\Gamma}^T \mathbf{S}(k) \mathbf{\Gamma} \right]^{-1} \mathbf{\Gamma}^T \mathbf{S}(k) \mathbf{\Phi} \quad (4.1.16)$$

**Eigenvector Decomposition** Another method can also be used to calculate the steady state feedback gains, namely *eigenvector decomposition*. This method, outlined in [9], in-



volves finding the eigenvalues and corresponding eigenvectors of the system's control *Hamiltonian matrix*. This is somewhat more computationally efficient than iteratively solving the Riccati difference equation backwards in time, but introduces much more complexity and constraints to finding the solution. If a similar control system is to be implemented on a platform with less computational power available to it, then this method should be investigated further.

## 4.2 Modelling

This section deals with defining the state equations that describe the outer loop kinematic system. The outer loop dynamics have already been derived in section 2.3 and those equations will simply be integrated into a full model here.

### 4.2.1 Attitude Representation

In order to obtain the state equations for the outer loop point mass dynamics, a method to represent the attitude of the aircraft's wind axes should first be selected. Of the three methods outlined in Appendix A, the DCM can be eliminated automatically as it will introduce nine state equations for a system with only three degrees of rotational freedom. Therefore the two options that should be investigated are Quaternions and Euler angles. This control system is required to accommodate all-attitude flight and therefore the following criteria should be adhered to by the attitude description that will be used,

1. The attitude of the aircraft should be controllable at any orientation. Through two of the inner loop commands available to the control system, namely NSA and roll rate, the controller is able to induce wind axes pitch, roll and yaw rates. Through these rates the aircraft's wind axes attitude dynamics will be controlled. Therefore if all three of the partial derivatives of any state representing the attitude dynamics,  $x_a$ , in terms of the wind axes angular rates are zero at the same time,

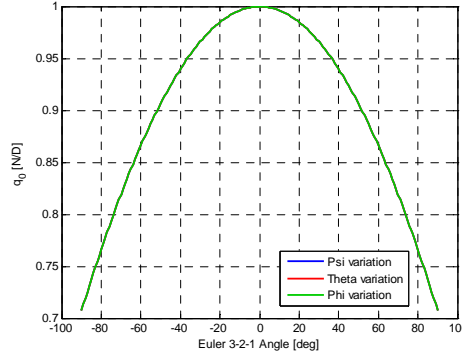
$$\frac{\partial \dot{x}_a}{\partial P_W} = 0 \quad \text{and} \quad \frac{\partial \dot{x}_a}{\partial Q_W} = 0 \quad \text{and} \quad \frac{\partial \dot{x}_a}{\partial R_W} = 0 \quad (4.2.1)$$

the state has become uncontrollable.

2. No singularities should exist in the system. Therefore if at any possible orientation,  $\dot{x}_a \rightarrow \infty$ , a singularity exists in the system at that attitude.

**Quaternions** Quaternions seem like the obvious choice to use for attitude description as they have no inherent singularities and therefore satisfy the second criterion above. However, Quaternions are very non-linear in their attitude description as illustrated by figure 4.2. The figure shows the result on  $q_0$  with a linear change being made to each of

the Euler 3-2-1 angles while the other two remain zero. It can be clearly seen that the same amount of change to an orientation angle can result in a large or small change to a Quaternion parameter depending on the current attitude. From a control perspective this will mean that the same amount of error in one of the quaternion states could imply differently sized attitude errors. This could lead to inconsistent control being applied over the range of possible attitudes since the amount of control will be directly related to the state error.



**Figure 4.2:** Resulting  $q_0$  with Variation of Different Euler 3-2-1 Attitude Angles

Another point to note from figure 4.2 is that when the aircraft's wind axes are aligned with the inertial axes, or in other words the Euler 3-2-1 attitude angles are all zero, the derivative of the  $q_0$  state is zero. This can be better seen by investigating the attitude dynamic equation for  $q_0$ . The equation, taken from section 2.3.3, states that,

$$\dot{q}_0 = \frac{1}{2} [-P_W q_1 - Q_W q_2 - R_W q_3] \quad (4.2.2)$$

The Quaternion state vector is given by,  $\mathbf{q} = [1 \ 0 \ 0 \ 0]^T$ , for the specific attitude mentioned above. Therefore at this attitude the partial derivatives of the dynamic equation with respect to all three the wind axes angular rates are zero. This means that the first criterion required for an attitude representation method is not met and that at this attitude the state has become uncontrollable. Various such situations exist for the different Quaternion parameters. The uncontrollability of these states is merely of a mathematical nature and is the result of the system's one constraint equation not being taken into account,

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1 \quad (4.2.3)$$

Incorporating this constraint into the dynamics, and thereby removing one of the dynamic equations, will solve this problem. It will also however further complicate the attitude dynamic equations as well as their linearisation process.

It can also be argued that the exact situations for uncontrollability mentioned above will only exist for a very short period of time and that the system will be controllable at all other attitudes. This is true, but at attitudes near these points of uncontrollability the matrices used for calculating the LQR feedback gains become badly scaled. The gains produced by the iterative LQR algorithm then require many more iterations to converge.

**Euler Angles** The second possibility for an attitude description method is Euler angles. They meet the first criterion for controllability at any attitude and also provide a more linear method of attitude description. However, as shown in section A.1.2, every Euler angle sequence contains a singularity in its dynamic equations. This is a violation of the second required criterion. Consider the dynamic equations for the third rotation angle  $\phi$  listed below for the Euler 3-2-1 and 2-3-1 sequences respectively,

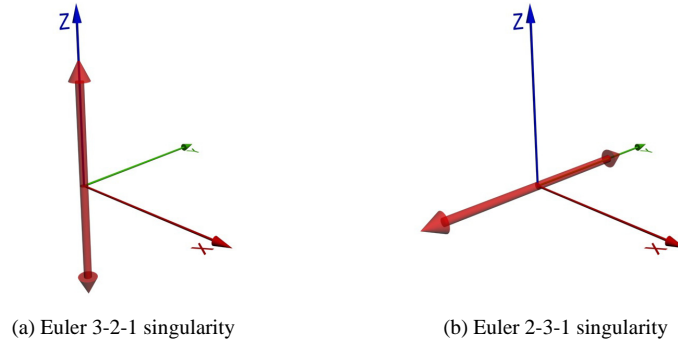
$$\dot{\phi}_{(321)} = P_W + Q_W \sin \phi \tan \theta + R_W \cos \phi \tan \theta \quad (4.2.4)$$

$$\dot{\phi}_{(231)} = P_W - Q_W \cos \phi \tan \theta + R_W \sin \phi \tan \theta \quad (4.2.5)$$

It can be seen that both sets of dynamics will become singular if,

$$\theta = \pm \frac{\pi}{2} [\text{rad}] \quad (4.2.6)$$

The second rotation angle  $\theta$  is defined differently for each of these sequences and therefore the singularity will be located at two different attitudes as shown by figure 4.3.



**Figure 4.3:** Inertial Axes Location of the Singularity for Each of the Euler Angle Sequences

**Solution** It can be seen that, except for the inherent singularity, an Euler sequence will work well for attitude representation. As shown in figure 4.3, the Euler 3-2-1 and 2-3-1 sequences have singularities at different attitudes. It is therefore decided to create two sets of outer loop dynamic equations, one for each of these Euler sequences, and then switch between them when nearing a singularity.

## 4.2.2 Dynamic Equations

As mentioned before the outer loop dynamics model a point mass that is able to rotate and translate in free space. The states therefore consist of the attitude, velocity and position of the aircraft's wind axis system and the inputs are normal and axial specific acceleration as well as the roll rate.

### 4.2.2.1 Attitude Dynamics

Substituting the algebraic constraints for  $Q_W$  and  $R_W$  of equation (2.3.6) into the Euler 3-2-1 and 2-3-1 attitude dynamics of section 2.3.3, with  $B_W = 0$  for coordinated flight, yields,

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}_{(321)} = \frac{g}{\bar{V}} \begin{bmatrix} 0 \\ -\cos \theta \\ 0 \end{bmatrix} + \frac{1}{\bar{V}} \begin{bmatrix} -\sin \phi \tan \phi \\ -\cos \phi \\ -\sin \phi \sec \theta \end{bmatrix} C_W + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} P_W \quad (4.2.7)$$

and

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}_{(231)} = \frac{g}{\bar{V}} \begin{bmatrix} \tan \theta \cos \psi \\ \sin \psi \sin \theta \\ -\cos \psi \sec \theta \end{bmatrix} + \frac{1}{\bar{V}} \begin{bmatrix} \cos \phi \tan \theta \\ -\sin \phi \\ -\cos \phi \sec \theta \end{bmatrix} C_W + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} P_W \quad (4.2.8)$$

These equations give the attitude dynamics in terms of the NSA and roll rate.

### 4.2.2.2 Velocity Dynamics

By substituting the respective Euler 3-2-1 and 2-3-1 DCM elements derived in Appendix A into equation (2.3.5) for the velocity dynamics yields,

$$\dot{\bar{V}}_{(321)} = -g \sin \theta + A_W \quad (4.2.9)$$

and

$$\dot{\bar{V}}_{(231)} = -g \cos \theta \sin \psi + A_W \quad (4.2.10)$$

### 4.2.2.3 Position Dynamics

Similar to the velocity dynamics, the position dynamics of equation (2.3.13) becomes,

$$\begin{bmatrix} \dot{P}_x \\ \dot{P}_y \\ \dot{P}_z \end{bmatrix}_{(321)} = \begin{bmatrix} \cos \psi \cos \theta \\ \sin \psi \cos \theta \\ -\sin \theta \end{bmatrix} \bar{V} \quad (4.2.11)$$

and

$$\begin{bmatrix} \dot{P}_x \\ \dot{P}_y \\ \dot{P}_z \end{bmatrix}_{(231)} = \begin{bmatrix} \cos \psi \cos \theta \\ \sin \theta \\ -\cos \theta \sin \psi \end{bmatrix} \bar{V} \quad (4.2.12)$$

when the respective DCM elements from Appendix A are substituted in.

#### 4.2.2.4 Axial Specific Acceleration Dynamics

As stated in section 3.8, the closed loop ASA dynamics are not fast enough to be time scale decoupled from the outer loop dynamics and it is therefore required to include them in the outer loop model. Because the ASA controller does not greatly affect the natural dynamics of the open loop throttle system it is decided to use the first order lag model for the throttle from section 2.4.3 for the ASA dynamics. This model is ideally suited for linearisation and state feedback as is required by the LQR control architecture. The goal of placing this model in the outer loop system is not to change the natural dynamics of the throttle, but rather to provide a model for the limited bandwidth available to the ASA command. Therefore the simple first order throttle model, given by equation (2.4.26), will be sufficient for this and can be adopted to model this effect for ASA as follows,

$$\dot{A}_W = \left[ -\frac{1}{\tau} \right] A_W + \left[ \frac{1}{\tau} \right] A_{W_C} \quad \text{with} \quad \tau = 0.75 \quad (4.2.13)$$

### 4.3 Implementation

With the dynamics modelled, the state and control vectors can be defined as follows,

$$\underline{x} = \left[ \phi \quad \theta \quad \psi \quad P_x \quad P_y \quad P_z \quad \bar{V} \quad A_W \right]^T \quad (4.3.1)$$

and

$$\underline{u} = \left[ A_{W_C} \quad C_W \quad P_W \right]^T \quad (4.3.2)$$

The linear state and input matrices given by,

$$\mathbf{F} = \left. \frac{df(\underline{x}, \underline{u})}{d\underline{x}} \right|_{\substack{\underline{x}=\underline{x}_R \\ \underline{u}=\underline{u}_R}} \quad \text{and} \quad \mathbf{G} = \left. \frac{df(\underline{x}, \underline{u})}{d\underline{u}} \right|_{\substack{\underline{x}=\underline{x}_R \\ \underline{u}=\underline{u}_R}} \quad (4.3.3)$$

can now be derived. The linearisation results are shown in Appendix C. In order to use the discrete LQR algorithm, the state and input matrices should be discretised. As long as the system's dynamics do not approach the sampling frequency, Euler's approximation

can be used to accomplish this,

$$\Phi \approx \mathbf{I} + \mathbf{F}T_{S_0} \quad (4.3.4)$$

$$\Gamma \approx \mathbf{G}T_{S_0} \quad (4.3.5)$$

where  $T_{S_0}$  is the sample period of the outer loop controller. The discrete linearised error regulation dynamics are therefore given by,

$$\underline{x}_E(k+1) = \Phi \underline{x}_E(k) + \Gamma \underline{u}_E(k) \quad (4.3.6)$$

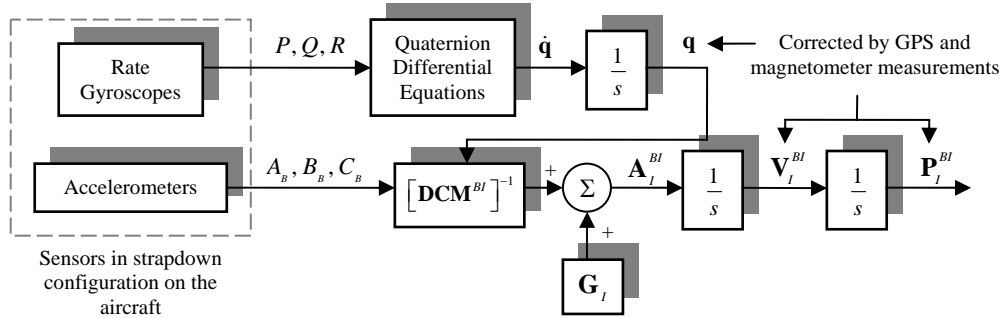
Full state feedback control is used to reject any state errors as follows,

$$\underline{u}_E(k) = -\mathbf{K} \underline{x}_E(k) \quad (4.3.7)$$

with the feedback gains  $\mathbf{K}$  recalculated at every time step by the discrete LQR algorithm.

#### 4.3.1 Wind Axes State Estimation

The states used by this control system have to be defined accurately at every sample instance. The aircraft's onboard GPS sensor provides inertial axes velocity and position updates at 4 Hz, which is much slower than the outer loop controller. A kinematic state estimator developed by [15] is used to accurately estimate these states using the combined measurements of all the aircraft's sensors as illustrated by figure 4.4.



**Figure 4.4:** Overview of the Kinematic State Estimator [15]

State	RMS State Errors	Units
Velocity ( $\tilde{V}$ )	0.22	m/s
Euler Attitude Angles	0.73	deg
Position	0.74	m

**Table 4.1:** Kinematic Estimator RMS State Errors

The theoretical accuracy of this estimator, as provided by [15], is shown in table 4.1. It can be seen that the RMS state errors are relatively small and therefore this estimator should be adequate for use with the outer loop controllers in this project.

This estimator generates the attitude of the aircraft's body axes but the outer loop model used for this control system uses wind axes attitude. Therefore a conversion is required. The angle of sideslip is continually regulated to zero by this control architecture and therefore the only significant conversion is for the angle of attack. The attitude estimate provided in Quaternion form by the state estimator can easily be converted to a DCM matrix using equation (A.2.3). This DCM can then be transformed to the wind axes DCM by multiplying it with the body to wind axes rotation matrix of equation (B.6.7) with  $\beta = 0$ .

$$\mathbf{DCM}^{WI} = \mathbf{DCM}^{WB} \mathbf{DCM}^{BI} \quad (4.3.8)$$

The estimate for the angle of attack derived in section 3.7 and given by equation (3.7.6) can be used to create this rotation matrix. The conversion back from the obtained wind axes DCM to the Euler angle states of the outer loop controller can be done using either equation (A.2.4) or (A.2.5) depending on the current Euler angle sequence being used.

### 4.3.2 Possible Uncontrollability

In order for the LQR algorithm to always be able to stabilise the closed loop system, there should not exist any scenario where the linearised  $\mathbf{F}$  and  $\mathbf{G}$  matrices become uncontrollable. One such possible scenario has been introduced into the system during the modelling process. Consider the partial derivatives for the Euler 3-2-1 heading angle  $\psi$  shown below for the various states and control inputs. These linearisation equations are taken from section C.3 with only the non-zero terms being shown.

States

$$\left. \frac{d\psi}{d\phi} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} = \frac{-C_{WR} \cos \phi_R \sec \theta_R}{\bar{V}_R} \quad (4.3.9)$$

$$\left. \frac{d\psi}{d\theta} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} = \frac{-C_{WR} \sin \phi_R \sec \theta_R \tan \theta_R}{\bar{V}_R} \quad (4.3.10)$$

$$\left. \frac{d\psi}{d\bar{V}} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} = \frac{C_{WR} \sin \phi_R \sec \theta_R}{\bar{V}_R^2} \quad (4.3.11)$$

Control Inputs

$$\left. \frac{d\psi}{dC_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} = -\frac{\sin \phi_R \sec \theta_R}{\bar{V}_R} \quad (4.3.12)$$

If there is a scenario where all of these terms become zero, the state for the angle  $\psi$  will become uncontrollable. A scenario such as this occurs when,

$$C_{W_R} = 0 \text{ [m/s}^2\text{]} \quad (4.3.13)$$

$$\phi_R = 0 \text{ [rad]} \quad (4.3.14)$$

This is when the aircraft is not rolled to a side and busy free-falling. During this instant the controller is not able to induce a yaw angle. As soon as the reference NSA,  $C_{W_R}$ , becomes non-zero, equation (4.3.9) will not result in zero anymore and the control system will roll the aircraft to induce a yaw angle. When the aircraft is banked to an angle, equation (4.3.12) also becomes non-zero and the controller will use the  $C_W$  control input to further induce a yaw angle.

However, this is a very unique scenario as a reference NSA mostly exists in order to counter the effect of gravity during level flight. Very small values of  $C_{W_R}$  will also be a problem as they will cause the system to be very close to uncontrollable and might cause sudden large control deviations. The matrices used to calculate the LQR feedback gains then become badly scaled and the algorithm takes longer to converge. However, it is seen that for the sample time and the LQR horizon that will be selected in section 4.3.4, the feedback gains converge to reasonable values and will settle suitably fast for any reference NSA where,

$$|C_{W_R}| > 2 \text{ [m/s}^2\text{]} \quad (4.3.15)$$

This problem of uncontrollability can therefore easily be overcome by not generating reference trajectories where the normal specific acceleration is this close to zero.

### 4.3.3 $A_W$ State Reference

In order to calculate the state errors for this control system, the measured values of each state are subtracted from reference values of a predefined trajectory. In order to do this for the modelled axial acceleration lag state, a sensible reference is required. A simple reference could be generated along with the reference trajectory, but because the axial acceleration model is very non-linear, most of the commanded values will be due to state errors. These commands are then not taken into account when the  $A_W$  reference from the trajectory is used. It is therefore proposed to rather use the first order Padé approximation model for the axial specific acceleration (see section 4.2.2.4) and propagate the commanded acceleration through these dynamics. This model consists of a single pole and NMP zero as given by,

$$\frac{A_{W_R}}{A_{W_C}} = \frac{-s + \tau_p}{s + \tau_p} \quad (4.3.16)$$

and will provide a reference for the  $A_W$  state.



#### 4.3.4 LQR Horizon and Discrete Sample Time Selection

For the implementation of this control system, a sample time and LQR cost optimisation horizon should be selected. By investigating equation (4.1.13) for  $\mathbf{M}(k)$ , it can be seen that a matrix inversion operation is required to solve the Riccati difference equation at every time step. This operation will require a significant amount of computation time and therefore the horizon  $N$  that the LQR cost function is minimised over, should be chosen just long enough for the feedback gains to converge to their steady state values.

Because the feedback gains represent the optimal control law over time, it stands to reason that they will require a certain amount of real time to settle to their steady state values. This has a large impact on the selection of the outer loop sample time, as real time is given by,

$$t = NT_{s_o} \quad (4.3.17)$$

If a smaller sample time is selected, it leads to more iterations  $N$  being required for the gains to converge. Figure 4.5 shows this for the  $A_{W_C}$  actuator where the feedback gains settle 2.5 times faster for a sample frequency decrease from 25 Hz to 10 Hz.

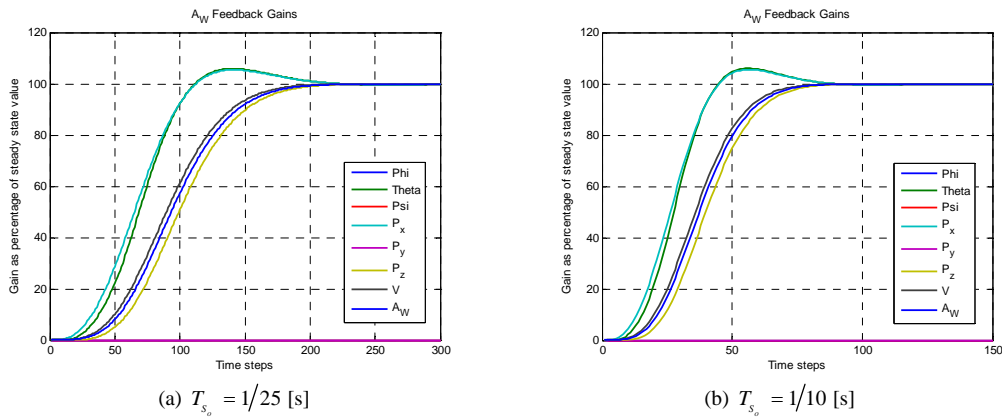


Figure 4.5: LQR Gain Settling for Different Sampling Frequencies

An increase in sample frequency will also imply that less time is available between sample instances to calculate the required iterations of the Riccati equation. Therefore the processing capabilities of the platform where the controller is implemented on should be considered when selecting the outer loop sampling frequency. Only the slower dynamics of the aircraft are however encapsulated by the outer loop model and therefore a relatively slow sampling frequency will still be sufficient.

The time scale decoupling upper limit for the outer loop closed loop poles are given by equation (3.8.1) as 1.3 rad/s. It is suggested by [9] that a sample frequency of at least 20 times faster than the system's bandwidth should be used for discrete controllers in order to obtain a relatively smooth response. This places a lower limit on the sample

frequency of,

$$f_{S_0} > 4.14 \text{ [Hz]} \quad (4.3.18)$$

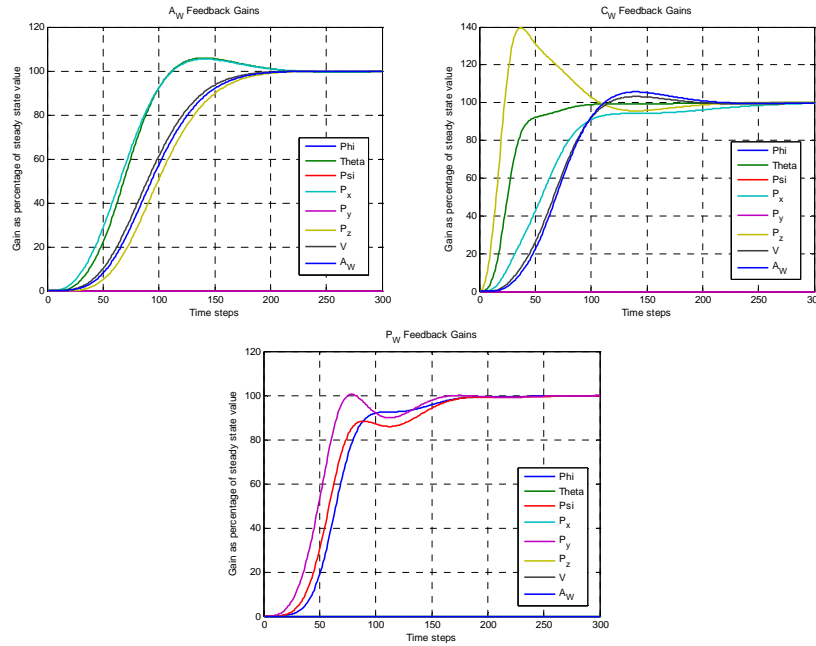
A sample frequency of 10 Hz would work very well for this controller, but because the OBC avionics used in this project has enough processing power available to it, a sampling frequency of,

$$f_{S_0} = 25 \text{ [Hz]} \quad (4.3.19)$$

is used for slightly improved performance. It can be seen from the three gain settling graphs for each of the outer loop actuators shown in figure 4.6 that for the selected sample frequency, a horizon of,

$$N = 240 \quad (4.3.20)$$

is required for the gains to fully converge. It should be noted that if the  $\mathbf{Q}_1$  or  $\mathbf{Q}_2$  weighting matrices of the LQR controller are adjusted, the closed loop dynamics of the system are affected and therefore also the settling times of the feedback gains. After any adjustment in weighting matrices, the gain settling should be rechecked.



**Figure 4.6:** Gain Conversion of the LQR Feedback Gains

### 4.3.5 Weighting Matrices Selection

The  $\mathbf{Q}_1$  and  $\mathbf{Q}_2$  weighting matrices assign relative importance to the various states and control inputs of the system. The LQR algorithm will weigh the cost of using a control to the resulting state regulation it provides and find the optimal, minimum cost result. This

will then ultimately determine where the closed loop poles of the system are placed. The upper frequency limit on the placement of these poles due to the time scale separation constraint is given by equation (3.8.1) as 1.3 rad/s. Throughout the design it should be ensured that the LQR algorithm places the poles inside this frequency constraint.

As stated before the  $\mathbf{Q}_1$  and  $\mathbf{Q}_2$  matrices are diagonal. As a starting point for selecting the state and control weights, it is suggested that the diagonal elements of these matrices be selected as follows,

$$Q_X = \frac{1}{(X_{\max})^2} \quad (4.3.21)$$

with  $X_{\max}$  being a maximum desired state or control deviation. This will assist in the state errors being weighed equally. For instance, an attitude deviation of 1 rad is much worse than a velocity deviation of 1 m/s. A global scaling factor should also be added to the  $\mathbf{Q}_2$  matrix with which the frequency of all the closed loop poles can be adjusted.

$$\mathbf{Q}_2 = p\mathbf{Q}'_2 \quad (4.3.22)$$

Here  $\mathbf{Q}'_2$  is the matrix containing the elements given by equation (4.3.21) for the maximum desired control deviations on its diagonal and  $p$  is the global scaling factor.

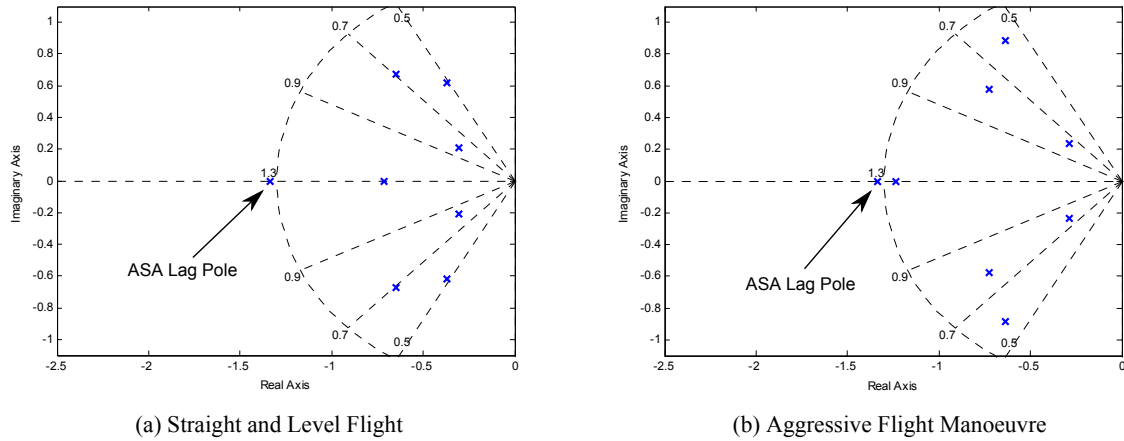
Determining the exact maximum desired deviation  $X_{\max}$  for each state and control input now becomes an iterative process whereby the system's performance is observed while these values are adjusted slightly. The relative performance of the system can be observed in two ways by:

1. Investigating the frequency and damping of the resulting closed loop poles.
2. Analysing its various step responses.

#### 4.3.5.1 Closed Loop Pole Placement

Because the poles of the open loop system change continually it is not possible to fully ensure that the LQR controller never places the closed loop poles above the frequency bound of equation (3.8.1). It can however be seen that when using the same weighting matrices the closed loop poles always end at similar frequencies. This is shown by figure 4.7 where two completely different points on a reference trajectory are linearised and used to create feedback gains. The first scenario is during straight and level flight at the aircraft's trim velocity while the other is during an aggressive flight manoeuvre where the aircraft is both pitched and rolled while commanding 3.5 g normal specific acceleration. The frequency constraint for the closed loop poles is met for both scenarios.

The first point to note from figure 4.7 is that the LQR algorithm has placed stable closed loop poles. All the complex poles are damped at a factor of 0.5 or better, while most of them are higher than critically damped. This pole placement can be deemed acceptable and the system's step response can be analysed.



**Figure 4.7:** LQR Pole Placement for Two Different Points on a Reference Trajectory

From figure 4.7 it can also be noted that a single real pole does not conform to the frequency constraint imposed on the outer loop system. This pole stems from the first order lag used to model the ASA dynamics in the outer loop model and will therefore have to do with commanding the aircraft's ASA. As these inner loop dynamics have not been time scale decoupled and are in fact included in the outer loop model, this pole need not conform to the same frequency constraint as the other poles. The effect of this pole can be seen in the velocity step response in section 4.3.5.2, which shows the amount of ASA command used to enact it.

#### 4.3.5.2 Step Responses

By allowing the LQR based outer loop controller to command the inner loop controllers developed in Chapter 3, various step responses of the complete aircraft control system can be investigated. This will validate the time scale separation assumption between the inner and outer loop controllers as well as show the performance of the outer loop control system. For each of the step responses to follow, only the states that are affected and the controls that are perturbed are shown. At the start of these simulations the aircraft starts in an initial straight and level flight condition, heading North.

**Velocity Step** For this step, the aircraft's full non-linear throttle model is used. It can be seen from figure 4.8 that ASA is mostly used to achieve the newly commanded velocity but that the LQR algorithm also commands some positive NSA. This commanded acceleration will cause the aircraft to pitch down and therefore trade some altitude for velocity. This is a faster way for the control system to gain a little airspeed but will ultimately induce an altitude error. For this step response all the position errors are set to zero, but when the aircraft is tracking a trajectory the control system will not allow this

altitude error to become very large. From figure 4.8 it can be seen that 80 per cent of the commanded velocity is achieved in about 3 s with no overshoot.

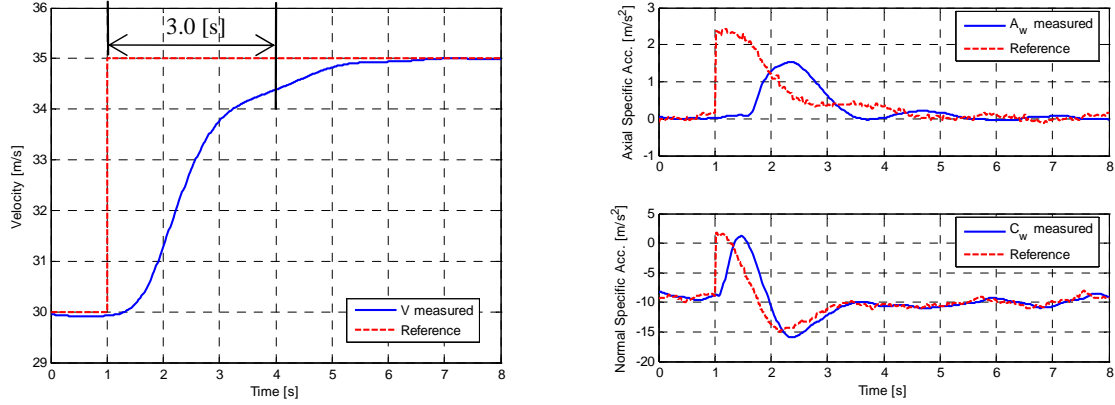


Figure 4.8: Velocity Step Response of the LQR Controller

**Pitch Angle Step** The Euler 3-2-1 pitch angle of the aircraft's wind axes can be altered by commanding a NSA, which is exactly what the LQR controller does, as shown by figure 4.9. This is one of the fastest outer loop step responses with the commanded angle being achieved after only 1.3 s.

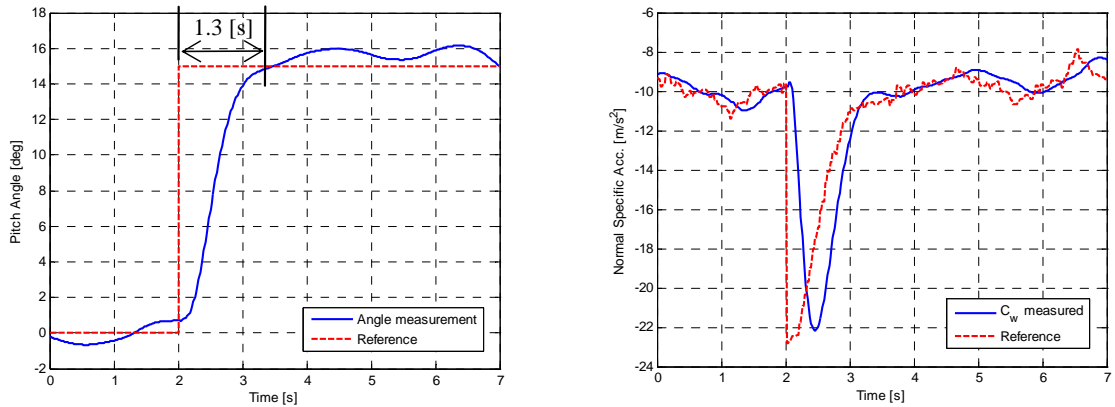


Figure 4.9: Pitch Angle Step Response of the LQR Controller

**Yaw Angle Step** The outer loop controller is unable to command the aircraft's lateral specific acceleration. Therefore the only way for the aircraft to achieve a commanded yaw angle is to command a roll rate which will induce an error in its roll angle. This will then cause the aircraft to bank and thereby the commanded yaw angle can be achieved.

For this step of 15 degrees on the yaw angle, the LQR controller induces a temporary error of 25 degrees on the roll angle in order to enact the commanded step. Figure 4.10 shows this step response where it can be seen that a yaw angle command takes the outer loop controller approximately 3 s to achieve. This is somewhat longer than the other two attitude angles due to the controller having to induce a roll angle first.

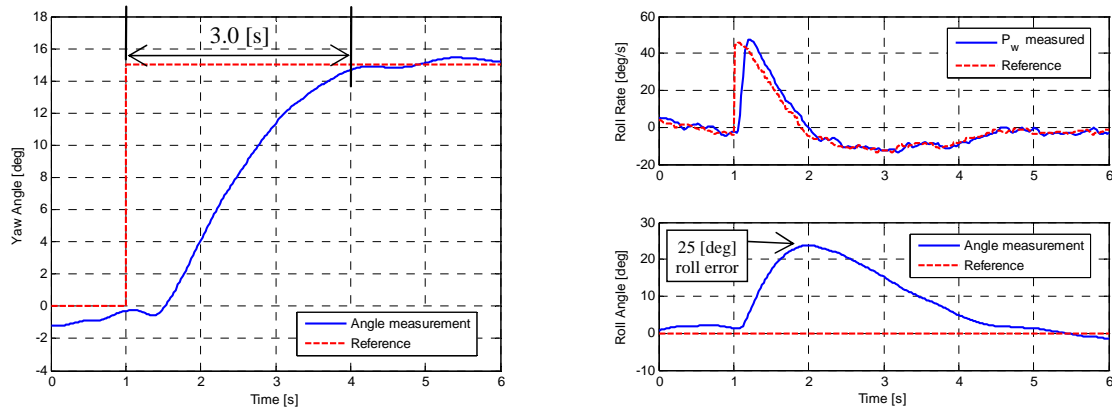


Figure 4.10: Yaw Angle Step Response of the LQR Controller

**Lateral Position Step** Before this step is commanded, the aircraft is flying directly North. This East position step will therefore show the aircraft's ability to bank and correct any lateral position errors. As expected, the LQR controller commands a roll rate which, similar to the yaw angle step, will create a roll angle and thereby cause the aircraft to bank. As soon as the commanded position is achieved the aircraft rolls back and returns to a level flight condition. Figure 4.11 shows how a lateral position step of 10 m is achieved in about 5.5 s.

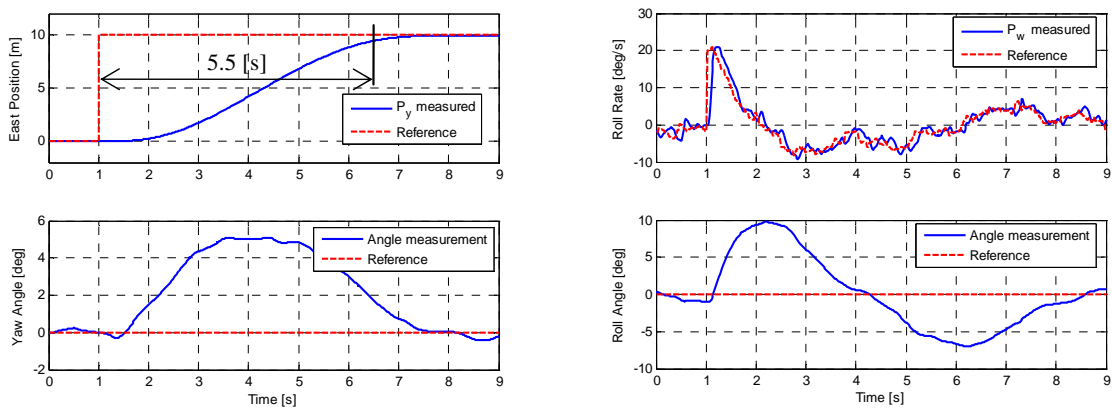
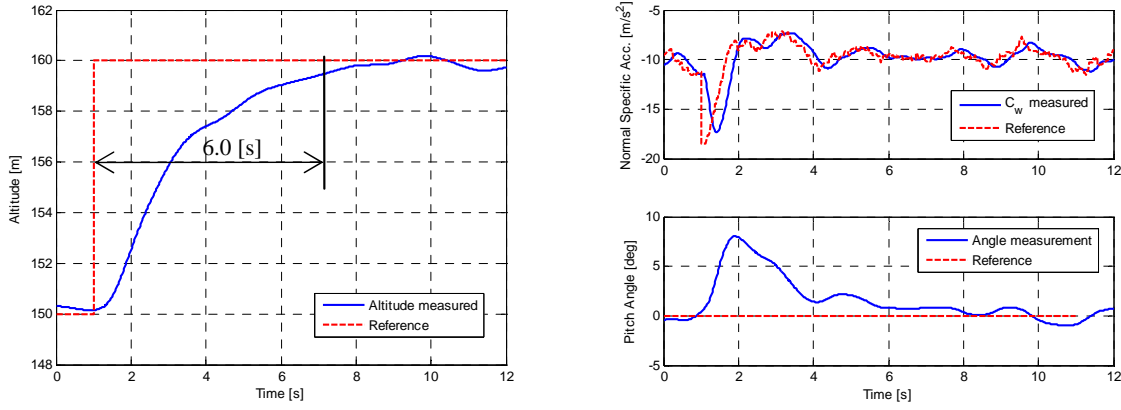


Figure 4.11: Lateral Position Step Response of the LQR Controller

**Altitude Step** In order for the aircraft to gain altitude, the controller can command a negative normal specific acceleration and thereby pitch the aircraft's nose up. Figure 4.12 shows how a small positive pitch angle error is induced in order for the aircraft to climb. The commanded 10 m step in altitude is achieved by the LQR controller in about 6.0 s.



**Figure 4.12:** Altitude Step Response of the LQR Controller

#### 4.3.5.3 Final Weighting Matrices

For the final selection for the maximum state and control deviations ( $X_{\max}$ ) that produce the diagonal elements of the weighting matrices, it is decided to weigh the three attitude states the same. This will cause attitude errors to be rejected similarly fast and minimises the transient response when the control system switches between attitude representation methods. This is also done for the three position states in order to get similar position tracking at any altitude. The selected values for these maximum deviations are given by table 4.2 and 4.3 and the value for the global scale factor is selected as,

$$p = 5000 \quad (4.3.23)$$

This selection will produce the step responses shown in the previous section and places the time scale decoupled outer loop poles shown in section 4.3.5.1.

State	Maximum Desired Deviation
Attitude Angles	6.4 [deg]
Position States	0.19 [m]
Velocity	3.5 [m/s]
Axial Specific Acceleration	1.0 [m/s²]

**Table 4.2:** Maximum Desired State Deviations

Control	Maximum Desired Deviation
Axial Specific Acceleration	1.8 [m/s <sup>2</sup> ]
Normal Specific Acceleration	12.0 [m/s <sup>2</sup> ]
Roll Rate	30.0 [deg/s]

Table 4.3: Maximum Desired Control Deviations

#### 4.4 Time Based Trajectory Generation

In the definition of the design architecture in section 4.1.1, it was assumed that the reference trajectory is a function of the outer loop dynamics thereby making it a kinematically feasible reference. Mathematically this is given by,

$$\dot{\mathbf{x}}_R = f(\mathbf{x}_R, \mathbf{u}_R) \quad (4.4.1)$$

An outer loop kinematic model simulator was therefore constructed using the outer loop dynamic equations. The ability to enter open loop reference commands is provided and the resulting effect on the aircraft's position, velocity and attitude can then be observed. These commands are generated differently for each reference trajectory and can be a function of the outer loop states. For instance, when generating a reference loop trajectory the NSA command is dependant on the current velocity as follows,

$$C_{WR} = -\frac{\bar{V}^2}{r} - g e_{33}^{WI} \quad (4.4.2)$$

where  $r$  is the radius of the loop being generated. This ensures that the radius of the loop remains constant. Figure 4.13 diagrammatically illustrates the working of this simulator. Once a suitable result for a trajectory is observed, all the states and control inputs are saved over time and can then be used by the outer loop controller as a reference trajectory.

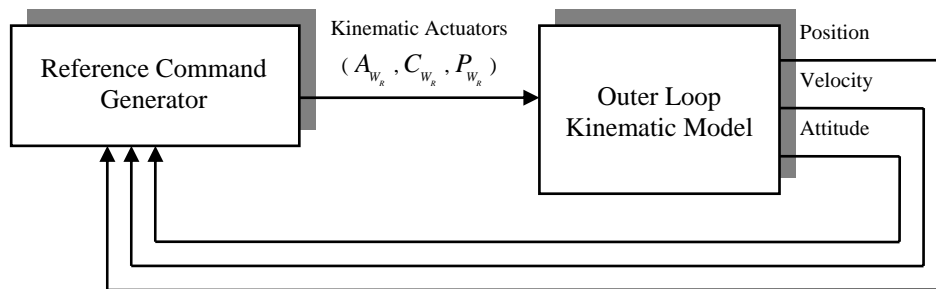


Figure 4.13: Architecture of the Reference Trajectory Generator

In order to test the full 3D flight capability of this control system, various aggressive flight trajectories were created. These trajectories, discussed in Chapter 1, are *straight and*



*level flight, an aileron roll, an aggressive climb, a high bank angle turn where the aircraft is rolled to about 60 degrees, a vertical loop and an Immelmann.*

**Attitude Representation** As mentioned before, the Euler angle sequences used in the design of the outer loop controller each contain singularities at different attitudes. The outer loop model, created for reference trajectory generation, therefore uses Quaternions in its attitude dynamics. These will then store the attitude reference and can easily be converted to DCM or Euler angle form when this is required by the controller.

#### 4.4.1 Issues with Time Based Trajectories

**Velocity Regulation** The problem of velocity regulation is one that continually presents itself for trajectories where the aircraft is required to climb aggressively. Because the maximum limit on the aircraft's throttle is not enough to counter the effect of gravity during a vertical climb, the aircraft will lose velocity during such manoeuvres. Therefore a pre-emptive command on the aircraft's ASA is required in order to build up velocity before initiating a climb. Otherwise the aircraft's velocity will be too low and it will stall during such a manoeuvre. This is intuitively correct as a pilot would also increase the aircraft's throttle in order to pick up velocity a few seconds before he executes a loop or other vertical climb manoeuvre. For these time based trajectories the pre-emptive command on the ASA will be created offline and included in the reference trajectory as an open loop feed-forward. A more elegant way of regulating the aircraft's velocity will be investigated later in this thesis when position based trajectories are introduced.

**Trajectory Runaway** This is an event that usually occurs in conjunction with the problem of velocity regulation during aggressive climbs or steep descents. Consider the scenario of a very steep climb. During this time the axial specific acceleration command will be driven into upper saturation which will result in the aircraft's throttle actuator command also being saturated. The controller now has no way of correcting a velocity or position error that stems from a reference velocity which is more than the achieved velocity. This can then result in the reference trajectory getting further and further ahead of the aircraft's actual position on the trajectory. This leads to large state errors and reference control commands being applied before they are actually required. Effective trajectory tracking is impaired by this and in certain cases, where the trajectory gets very far ahead, it can even lead to instability.

A position governor could be used to track back onto the reference trajectory, but this will cause the trajectory to no longer be an exact function of the outer loop dynamics – a constraint placed on it by equation (4.1.10). This method might minimise position errors, but will induce other state errors and can possibly jeopardise the system's stability.

The best way to prevent trajectory runaway is to provide dynamic upper and lower bounds for the ASA and not exceed these in the generation of the reference trajectory. These bounds are given by,

$$A_{W_{\max}} = \frac{T_{\max}}{m} - \frac{qS}{m} C_D \quad (4.4.3)$$

and

$$A_{W_{\min}} = \frac{T_{\min}}{m} - \frac{qS}{m} C_D \quad (4.4.4)$$

Unfortunately the aerodynamic drag and maximum thrust are two quantities that are not very accurately modelled. Also the amount of aerodynamic drag constantly changes with the actual velocity as the trajectory is flown. Therefore the effect of trajectory runaway can still be seen in some of the simulations and practical data, but has been greatly reduced through the use of the above limits. A way to prevent trajectory runaway altogether is to use position based trajectories and will be discussed in Chapter 5.

## 4.5 Simulated Trajectory Flight Results

The reference trajectories discussed in the previous section are flown in simulation using the LQR based outer loop control system to provide commands for the inner loop controllers. The more notable results are shown here with the rest placed in Appendix C. The tracking accuracy as well as the commands generated by the LQR controller are shown for each trajectory. Good tracking accuracy can be observed for most of the trajectories.

The effect of the time scale separation assumption can be seen for various trajectories by observing that certain references are overshoot slightly due the commanded rates not being immediately attained – note the roll angle for the high angle turn. Trajectory runaway can also be observed for the loop trajectory, where at the end of the loop the reference position is about 20 m ahead of the aircraft's actual position on the trajectory.

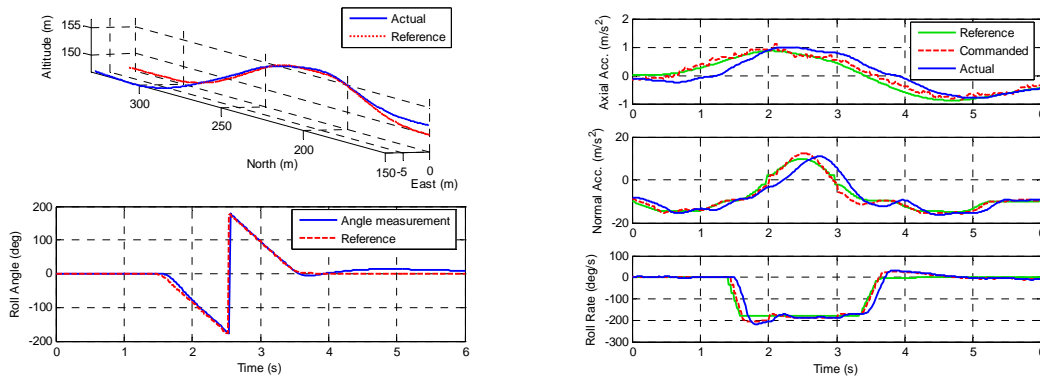


Figure 4.14: Aileron Roll Trajectory Simulation for the LQR Controller

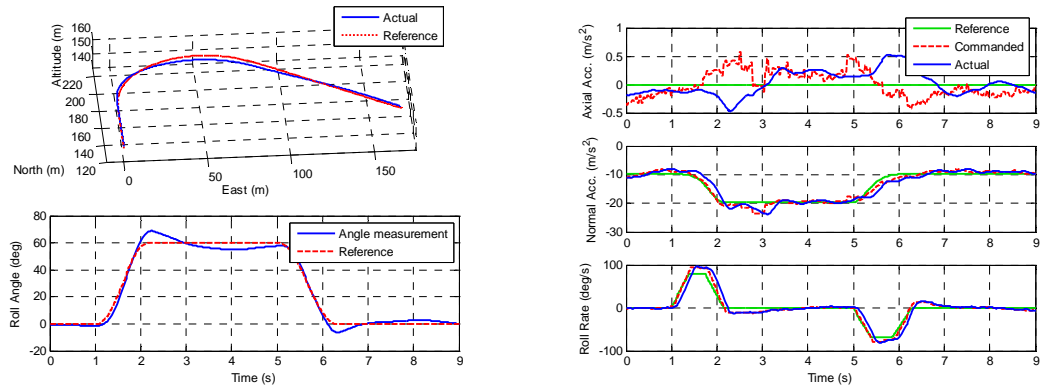


Figure 4.15: High Angle Turn Trajectory Simulation for the LQR Controller

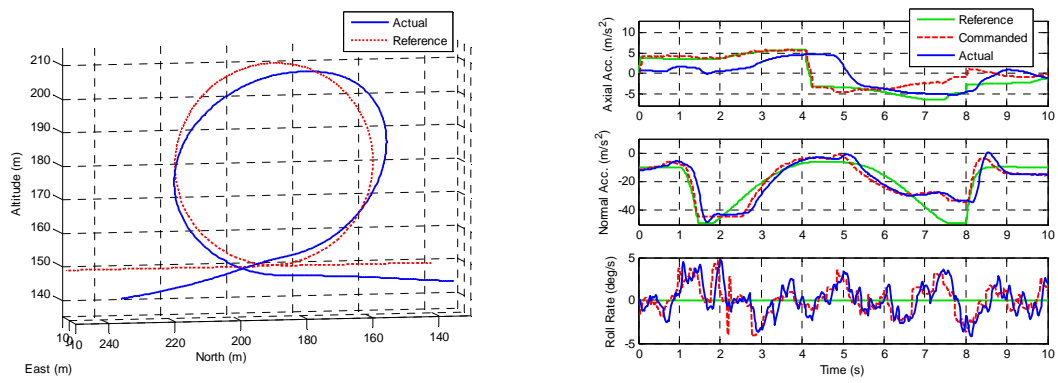


Figure 4.16: Vertical Loop Trajectory Simulation for the LQR Controller

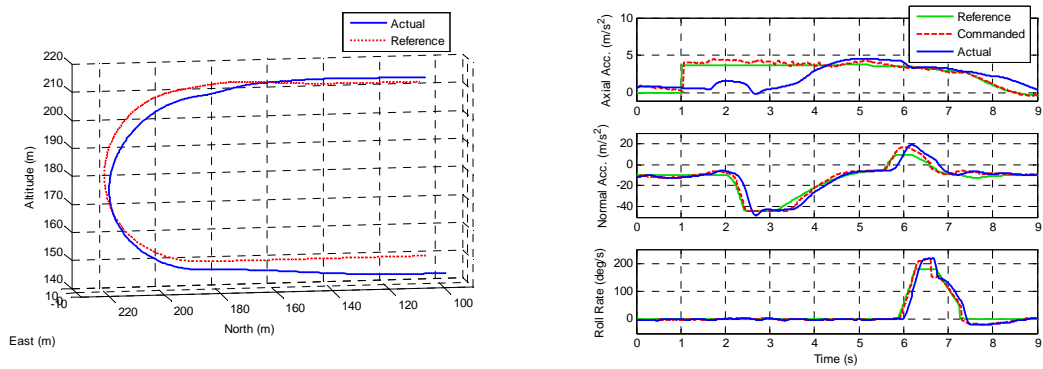


Figure 4.17: Immelmann Trajectory Simulation for the LQR Controller

## 4.6 Practical Flight Test Data

In order to practically test the outer loop LQR control system, it can be implemented on the OBC avionics package and tested using the HIL simulator discussed in section 3.9. After the algorithm is deemed ready, practical tests can be conducted.

All of the trajectories shown in section 4.5 have been practically tested and some are shown and discussed in this section. From the results it can be seen that practically the controller does not perform as well as in the simulations, but still produces acceptable results. There are various reasons for the degraded performance, such as aircraft modelling inaccuracies, actuator slew rate and accuracy, estimator drift due to GPS accuracy loss during high g flight, and trajectory runaway effects.

**Trajectory Transition** At the beginning of each trajectory the aircraft starts in a predefined state, namely straight and level at a defined velocity. Therefore, the aircraft has to be placed in this state before these trajectories can be flown. A four second transition algorithm is used to achieve this by commanding the controller to regulate all the attitude angles to zero, as well as the velocity to its pre-defined starting value with any position errors being ignored. The starting Euler 3-2-1 yaw angle of the aircraft is considered the zero reference for this angle and thereby the direction in which the manoeuvre is done can be determined by the safety pilot. This conversion of the actual attitude to the newly defined zero reference is done in DCM form using a single rotation matrix, as defined in section B.6, as follows,

$$\mathbf{DCM}_{\phi}^{WI} = \mathbf{T}^k(-\psi_i)\mathbf{DCM}^{WI} \quad (4.6.1)$$

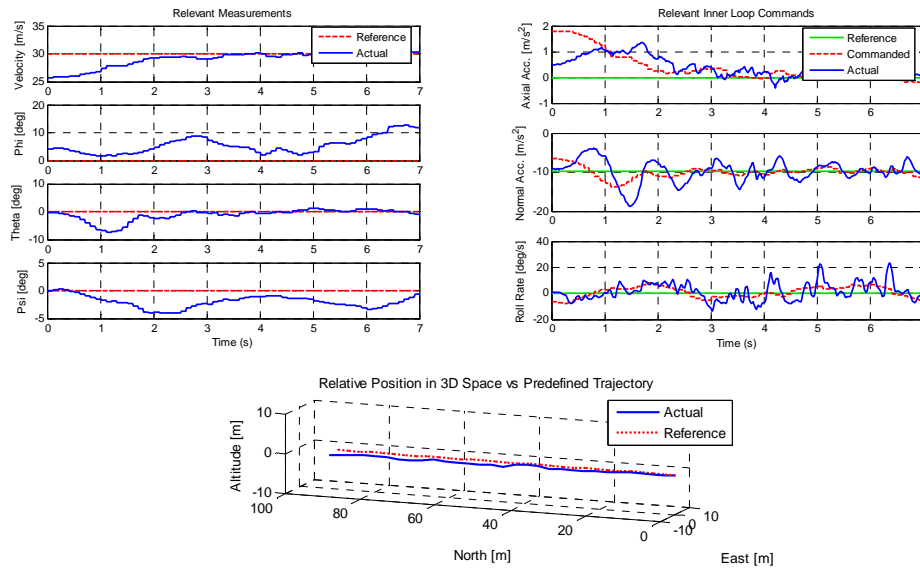
where  $\mathbf{DCM}^{WI}$  represents the wind axes attitude and is obtained from equation (4.3.8) and  $\mathbf{DCM}_{\phi}^{WI}$  represents this attitude rotated back through the initial yaw angle ( $\psi_i$ ). After this transition to the starting state of the trajectory has been done, the aircraft will start to follow the trajectory in a similar way as shown in the simulations.

### 4.6.1 Straight and Level Flight

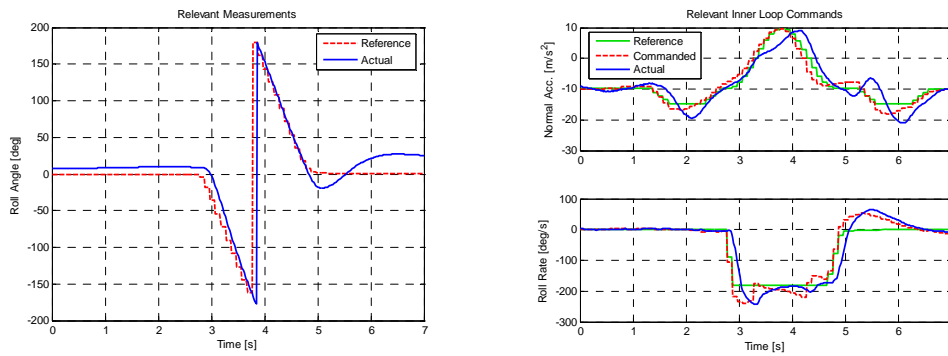
This is the first trajectory that was flown by the controller and is a good starting point in verifying whether the controller is able to stabilise the aircraft and keep it on a straight flight path. Figure 4.18 shows how the aircraft's states are regulated in order to stay on this level flight path, which verifies the functionality of this control system.

### 4.6.2 Aileron Roll

In this trajectory the aircraft has to roll through a full 360 degrees. It can be seen from the roll angle in figure 4.19 how this roll is performed, with some overshoot and correction observed at the end. A roll rate of over 200 deg/s is achieved during this manoeuvre.



**Figure 4.18:** Practical Flight Results of the Level Flight by the LQR Controller



**Figure 4.19:** Practical Flight Results of the Aileron Roll by the LQR Controller

### 4.6.3 High Angle Turn

For this trajectory the aircraft is banked to an angle of 60 degrees and 2 g's of NSA is applied in order to aggressively turn the aircraft. It can be seen from figure 4.20 that during the turn the yaw angle changes by 90 degrees in about 2 seconds. This demonstrates the lateral capabilities of this aggressive flight autopilot. Slightly more reference following error is observed for this trajectory as compared to its simulation in figure 4.15. From the results it can be seen that the aircraft achieves a slightly larger roll rate and NSA as commanded to the inner loop controllers and therefore turns a little too fast. At the end of the turn, the aircraft's heading does not match the trajectory's and the controller induces a roll angle error in order to steer the aircraft back to the reference trajectory.

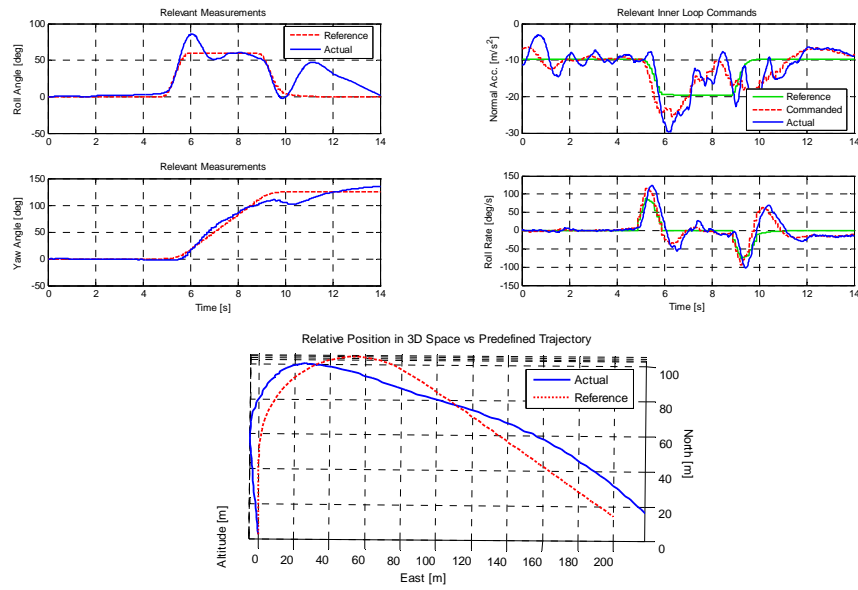


Figure 4.20: Practical Flight Results of the High Angle Turn by the LQR Controller

#### 4.6.4 Aggressive Climb

This trajectory shows off the vertical capabilities of the control system with a 60 degree climb. Figure 4.21 shows the results of this trajectory with very satisfactory position tracking results. It can be observed how the aircraft loses about 10 m/s of velocity during the 60 m climb, even when flying at full throttle.

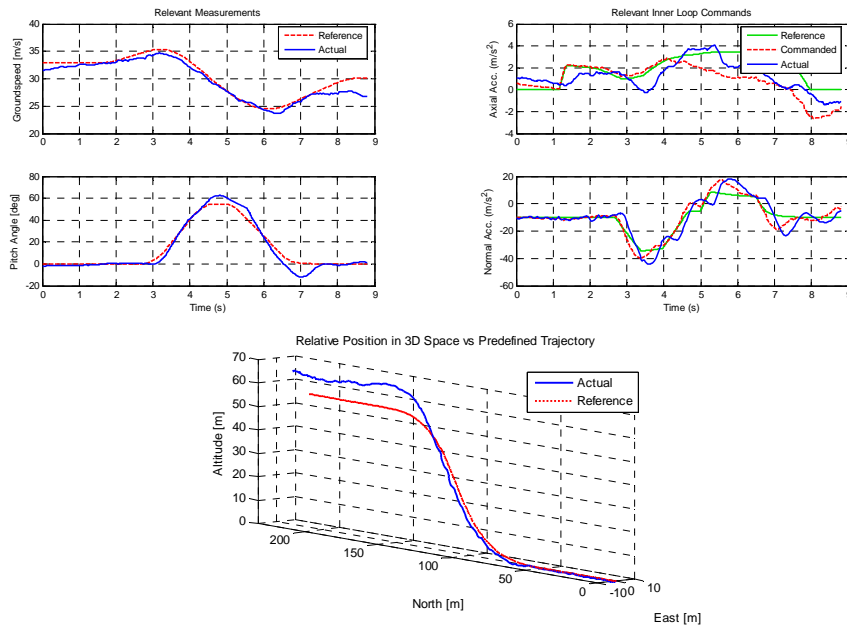


Figure 4.21: Practical Flight Results of the Aggressive Climb by the LQR Controller

### 4.6.5 Vertical Loop

The vertical loop is a more difficult trajectory for this control system to execute, as most of the time the ASA controller will command the throttle to its limits. Inaccuracies in the throttle and drag model will therefore cause trajectory runaway during the execution of this trajectory. Some discrepancies can be seen between the actual and reference velocity however the position tracking is relatively similar to the simulated results. For safety reasons the aircraft's NSA command is limited to  $\pm 4.5$  g's and this limit is achieved during the initial and final stages of the loop.

It can be seen that the loop is not fully round and that during the last section the aircraft is located below the reference trajectory. This is due to the time based trajectory that specifies the aircraft's velocity at every point on the trajectory. During the final stage of the loop the aircraft's velocity is slower than the reference, which is in fact a more desirable result. However, since the reference trajectory specifies a larger velocity, the control system commands ASA as well as induces an altitude error and thereby attempts to increase the aircraft's velocity. Position based trajectories as discussed in the next chapter can be used to correct this problem.

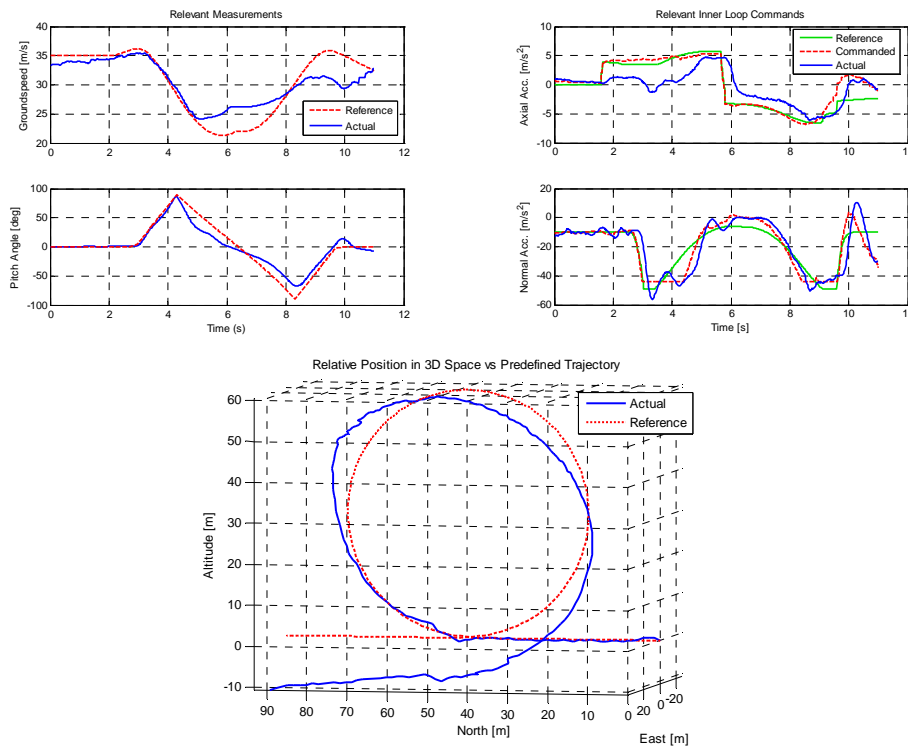


Figure 4.22: Practical Flight Results of the Vertical Loop by the LQR Controller

## 4.7 Summary

This chapter showed the theoretical derivation and practical implementation of the first outer loop guidance controller used in this project, namely continuous re-linearisation LQR control. This method was found to work relatively well, but a few negative aspects also presented themselves. These aspects can be summarised as follows:

- Because time based trajectories are used with this controller, the negative effects of trajectory runaway, as discussed in section 4.4, is sometimes observed. Trajectory based velocity regulation also degrades reference tracking performance as shown by the practical flight of the vertical loop trajectory in section 4.6.5.
- Euler singularities force the use of two sets of dynamics with the controller switching between them when nearing one of the singularities. Small transitional effects are observed with the switching. A possible solution to this problem is using relative attitude with the trajectory as a reference axis system, and then calculating the Euler 3-2-1 attitude errors of the aircraft with respect to this reference axis system. Further research on this topic is recommended as time constraints prevented this in this project.
- The linearised dynamics become uncontrollable at  $C_{W_R} = 0$ . This is a fundamental problem common to all guidance strategies and is not limited to this control architecture. This can however easily be avoided by creating trajectory references where,  $|C_{W_R}| > 2 \text{ [m/s}^2\text{]}$ .
- Solving the Riccati difference equation to obtain the LQR feedback gains is relatively computationally demanding since it requires a matrix inversion operation with every iteration. Therefore if a system is used with limited processing power available to it, this control system will be difficult to implement.

It can be seen however that despite these drawbacks, this controller still performs relatively well, both in simulation as well as in practice. Some of the shortcomings of this control architecture will be overcome by the second outer loop controller used in this project, namely the *Specific Acceleration Matching Controller*, which will be discussed in the chapter to follow.



## Chapter 5

# Specific Acceleration Matching Kinematic Controller

In the previous chapter a somewhat brute force method of continuous re-linearisation was discussed for controlling the non-linear dynamics of the aircraft's outer loop model. It is shown by [1] how additional assumptions can be made that will cause these dynamics to become linear, and how relatively simple controllers can be designed for them.

This chapter will discuss this control architecture as well as alter it for implementation on the aircraft used in this project. The architecture as proposed by [1] will be discussed first where the difficulties with implementing this controller will be highlighted. A solution will then be proposed where the core of the algorithm can still be used for guidance control. Position based trajectories will be created for this controller and their advantages will be discussed. Simulation and practical results will then be shown.

### 5.1 Overview

The idea of specific acceleration matching (SAM) as presented by [1] is shortly summarised as follows. The commandable normal and axial specific accelerations of the inner loop controllers are seen as a plane in which the aircraft is capable of providing immediate accelerations in. This is the XZ-plane of the aircraft's wind axes and its orientation is affected by the roll angle, and therefore the roll rate.

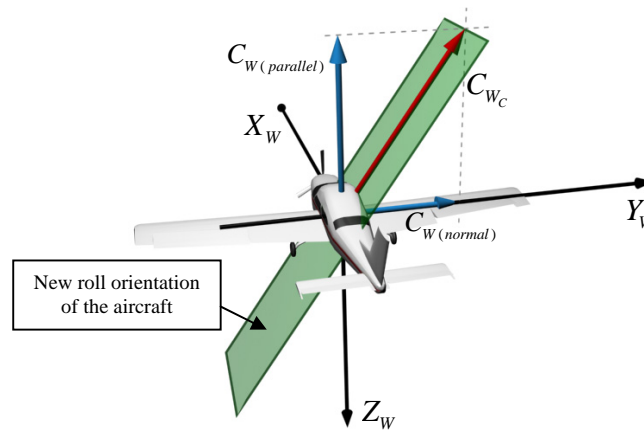
The proposed strategy is to firstly create a controller that receives the commanded orientation of this plane. In other words, it receives the commanded direction of the aircraft's NSA vector. This normal specific acceleration vector direction controller (NSAVDC) then creates a roll rate command in order to achieve this orientation. Secondly, a guidance controller is created to govern the aircraft's position and velocity vectors by calculating a resulting acceleration vector that will drive any errors to zero. This acceleration vector can then be converted to an acceleration vector in the wind axes XZ-plane as well as an orientation of this plane. For this controller the assumption is made that this ori-

entation is instantly achievable. Because of this assumption, the NSAVDC's dynamics should be placed at a higher, time scale decoupled, frequency from those of the guidance controller, thereby ensuring the correct orientation of the aircraft when accelerations are applied. The following assumptions are therefore made by this control architecture:

1. All of the inner loop controllers are much faster than the outer loop guidance dynamics and can be commanded instantly.
2. The NSAVDC's dynamics is time scale decoupled from the guidance controller and therefore an instantly commandable NSA vector direction is assumed.

For the aircraft used in this project it was shown in Chapter 3 that the aircraft's ASA controller can not be time scale decoupled from the outer loop dynamics due to the very limited bandwidth of the throttle. Therefore the SAM controller as proposed by [1] that assumes a fast, time scale decoupled ASA controller will have to be altered.

The solution proposed in this thesis is to decouple the aircraft's velocity regulation problem from its position tracking and guidance problem. Therefore ASA is completely removed from this control architecture and another separate control system is used to regulate the aircraft's velocity. The commandable plane of acceleration proposed by [1] is thereby reduced to a vector. It is therefore assumed by the guidance controller that the aircraft's NSA vector can be instantly rolled to any angle and thereby any commanded normal and lateral acceleration can be achieved instantly. Figure 5.1 illustrates this and shows that the commanded NSA vector consists of two components in the wind axes YZ-plane, one parallel and the other normal to the current direction of the NSA vector. The result is that the magnitude of this vector  $C_{Wc}$  is commanded to the inner loop NSA controller while the aircraft is rolled by the NSAVDC until its  $Z_W$ -axis is aligned with the commanded NSA vector direction. This guidance controller therefore operates in a two-dimensional fashion and ignores any axial trajectory tracking errors.



**Figure 5.1:** Specific Acceleration Matching in the Aircraft's YZ-plane

Position based trajectories can then be created which provide trajectory information based on the closest point on a smooth reference trajectory to the current location of the aircraft. These trajectories are completely independent of time and therefore velocity. When used as a reference, they remove any guidance errors that correspond to how far the aircraft has travelled along the reference trajectory. This means that the reference trajectory is never able to be ahead of, or behind the aircraft's current location. The trajectory errors will therefore only highlight the lateral and normal tracking errors of the aircraft to the reference trajectory as illustrated by figure 5.2.

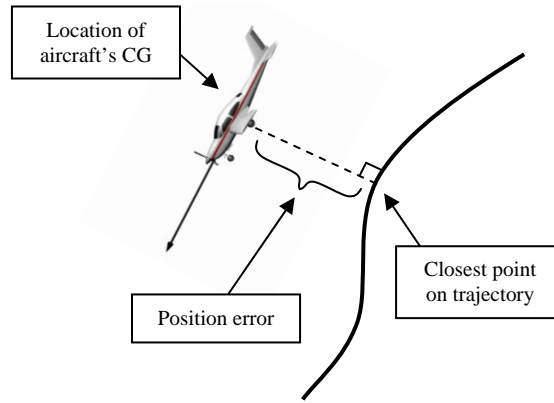


Figure 5.2: Position Based Trajectories

## 5.2 Control Architecture

### 5.2.1 Guidance Controller

Suppose the reference position and velocity vectors generated by the position based trajectories and coordinated in inertial axes are given by  $\mathbf{P}_I^{RI}$  and  $\mathbf{V}_I^{RI}$  respectively. The current trajectory tracking errors for velocity and position are then given by,

$$\mathbf{V}_I^{WR} = \mathbf{V}_I^{WI} - \mathbf{V}_I^{RI} \quad (5.2.1)$$

$$\mathbf{P}_I^{WR} = \mathbf{P}_I^{WI} - \mathbf{P}_I^{RI} \quad (5.2.2)$$

where  $\mathbf{P}_I^{WR}$  and  $\mathbf{V}_I^{WR}$  correspond to these errors. The current ( $\mathbf{V}_I^{WI}$ ) and reference ( $\mathbf{V}_I^{RI}$ ) velocity vectors are generated by combining the current velocity magnitude with the attitudes of both these axis systems. Obtaining the attitude of the aircraft's wind axes has already been discussed in section 4.3.1. The current position vector ( $\mathbf{P}_I^{WI}$ ) is obtained from the kinematic state estimator discussed in section 4.3.1 and the reference position ( $\mathbf{P}_I^{RI}$ ) corresponds to the closest point on a smooth position based reference trajectory.

The goal of this controller is to create a specific acceleration vector ( $\Sigma^{WI}$ ) that will counter any position and velocity errors. Given that the reference trajectory is kinematically feasible, the equations of motion provide the following relationships for the velocity and position error dynamics,

$$\left. \frac{d}{dt} \mathbf{V}^{WR} \right|_I = \mathbf{A}^{WR} \quad (5.2.3)$$

$$\left. \frac{d}{dt} \mathbf{P}^{WR} \right|_I = \mathbf{V}^{WR} \quad (5.2.4)$$

The velocity vector error dynamics can then be written in terms of the specific acceleration vector as follows,

$$\begin{aligned} \left. \frac{d}{dt} \mathbf{V}^{WR} \right|_I &= \mathbf{A}^{WI} - \mathbf{A}^{RI} \\ &= \Sigma^{WI} + \mathbf{G}^{WI} - \Sigma^{RI} - \mathbf{G}^{RI} \\ &= \Sigma^{WI} - \Sigma^{RI} \end{aligned} \quad (5.2.5)$$

where  $\mathbf{A}$  and  $\Sigma$  corresponds to the total and specific acceleration vectors respectively and  $\mathbf{G}$  corresponds to the gravity vector. An equipotential gravity field is assumed and therefore  $\mathbf{G}^{WI} - \mathbf{G}^{RI} = 0$ .

A control law for the specific acceleration vector, coordinated in inertial axes, is defined by [1] as,

$$\Sigma_I^{WIC} = -K_P \mathbf{P}_I^{WR} - K_V \mathbf{V}_I^{WR} + \Sigma_I^{RI} \quad (5.2.6)$$

where  $K_P$  and  $K_V$  are the position and velocity error feedback gains and  $\Sigma_I^{RI}$  is a feed-forward term for the reference specific acceleration vector obtained from the trajectory. The closed loop position dynamics can be derived by taking the second time derivative of the position vector and coordinating into inertial axes.

$$\ddot{\mathbf{P}}_I^{WR} = \dot{\mathbf{V}}_I^{WR} \quad (5.2.7)$$

By also coordinating equation (5.2.5) for the velocity dynamics into inertial axes and substituting it into equation (5.2.7) gives,

$$\ddot{\mathbf{P}}_I^{WR} = \Sigma_I^{WI} - \Sigma_I^{RI} \quad (5.2.8)$$

The velocity feedback term in the specific acceleration control law of equation (5.2.6) can be written in terms of position using the relationship of equation (5.2.4). This control law can then substituted into equation (5.2.8) to obtain the closed loop guidance dynamics,

$$\ddot{\mathbf{P}}_I^{WR} + K_V \dot{\mathbf{P}}_I^{WR} + K_P \mathbf{P}_I^{WR} = 0 \quad (5.2.9)$$

This represents a second order system with full pole placement available to it with the feedback gains given by,

$$K_P = \omega_n^2 \quad (5.2.10)$$

$$K_V = 2\zeta\omega_n \quad (5.2.11)$$

where  $\zeta$  and  $\omega_n$  correspond to the desired damping and natural frequency of the guidance dynamics.

The final step for this controller is to extract the NSA command as well as the desired NSA vector direction from the commanded specific acceleration vector. The commanded specific acceleration has been defined by [1] as acting in the aircraft's XZ-plane and can therefore be written as,

$$\boldsymbol{\Sigma}^{WI_C} = A_{W_C} \mathbf{i}^W + C_{W_C} \mathbf{k}^{W_C} \quad (5.2.12)$$

where  $A_{W_C}$  and  $C_{W_C}$  are the axial and normal specific accelerations commanded by this controller and  $\mathbf{k}^{W_C}$  is the commanded NSA vector direction. Because position based trajectories are used, very little ASA will be commanded by the guidance controller. As previously stated, this controller will not be used to regulate the aircraft's velocity magnitude and therefore this small amount of ASA can simply be removed from the commanded specific acceleration vector. The ASA can be extracted using the dot product as follows,

$$A_{W_C} = \boldsymbol{\Sigma}^{WI_C} \cdot \mathbf{i}^W \quad (5.2.13)$$

The commanded NSA vector is the remaining portion of the commanded specific acceleration vector after the ASA vector is removed.

$$\mathbf{N}^{W_C} = \boldsymbol{\Sigma}^{WI_C} - A_{W_C} \mathbf{i}^W \quad (5.2.14)$$

The respective commands for the NSA magnitude and the NSA vector direction can be derived from this vector as follows,

$$C_{W_C} = \text{sgn}(\mathbf{N}^{W_C} \cdot \mathbf{k}^R) |\mathbf{N}^{W_C}| \quad (5.2.15)$$

$$\mathbf{k}^{W_C} = \frac{1}{C_{W_C}} \mathbf{N}^{W_C} \quad \text{with} \quad C_{W_C} \neq 0 \quad (5.2.16)$$

with the  $\text{sgn}(\mathbf{N}^{W_C} \cdot \mathbf{k}^R)$  term being responsible for the sign of the commanded NSA and ensures that the aircraft does not invert when the sign of the NSA changes. The  $\mathbf{k}^R$  term in this function is the reference direction for the aircraft's NSA and is obtained from the reference trajectory.

### 5.2.2 Normal Specific Acceleration Vector Direction Controller (NSAVDC)

Using the commanded direction for the NSA vector ( $\mathbf{k}^{W_C}$ ) obtained from the guidance controller, a roll rate command can be generated in order for the aircraft to achieve the required orientation. As stated before, it is assumed by the guidance controller that this orientation can be instantly attained and therefore this controller should be time scale decoupled from the guidance controller.

As stated by [1], there are two ways in which the NSAVDC can be created. It can additionally be time scale decoupled from the inner loop roll rate controller, with its dynamics then placed faster than the guidance dynamics but slower than the roll rate dynamics. Alternatively, it can be designed in conjunction with the roll rate controller and thereby the inner loop roll rate dynamics are taken into account.

The latter method is preferable as it produces a more accurate pole placement result and eliminates the need for a second time scale decoupling in the control architecture. This second time scale decoupling would be difficult to achieve as in this project the roll rate controller's slowest dynamics are located at 6.5 rad/s and for adequate performance, the guidance dynamics will be placed close to 1 rad/s. Making two consecutive time scale decoupling assumptions in this frequency range will not be very accurate.

Therefore in order to design the combined NSAVDC, the error angle between the commanded and actual NSA vectors which is to be regulated to zero has to be defined. The definition of the dot product, shown in section B.1, gives this relationship as follows,

$$\mathbf{k}^{W_C} \cdot \mathbf{k}^W = |\mathbf{k}^{W_C}| |\mathbf{k}^W| \cos \phi = \cos \phi \quad (5.2.17)$$

The following intuitive equation is shown by [1] to hold for the dynamics of this error angle,

$$\dot{\phi} = P_{W_C} - P_W \quad (5.2.18)$$

where  $P_{W_C}$  is the roll rate required to follow the reference trajectory and  $P_W$  is the current wind axis roll rate. A linear feedback control law for the roll rate command is defined in terms of a feed-forward from the reference trajectory ( $P_{W_C}$ ) and proportional feedback from the error angle,

$$P_{W_R} = P_{W_C} + K_\phi \phi \quad (5.2.19)$$

where  $K_\phi$  is the error angle feedback gain. Combining the roll rate dynamics and augmented integrator of section 3.5 with the error angle dynamics of equation (5.2.18) yields the combined system dynamics,

$$\begin{bmatrix} \dot{P} \\ \dot{E_P} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} L_P/I_x & 0 & 0 \\ 1 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} P \\ E_P \\ \phi \end{bmatrix} + \begin{bmatrix} L_{\delta_A}/I_x & 0 \\ 0 & -1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_A \\ P_{W_R} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} P_{W_C} \quad (5.2.20)$$

By substituting the aileron control law of equation (3.5.4) and the roll rate command control law of equation (5.2.19) into the dynamics, the closed loop system can be obtained. The closed loop characteristic equation can then be calculated and is shown by [1] to be given by,

$$\alpha_c(s) = s^3 + \left( K_P \frac{L_{\delta_A}}{I_x} - \frac{L_P}{I_x} \right) s^2 + \left( K_E \frac{L_{\delta_A}}{I_x} - K_\phi \bar{N}_P \frac{L_{\delta_A}}{I_x} \right) s + \left( K_\phi K_E \frac{L_{\delta_A}}{I_x} \right) \quad (5.2.21)$$

With a desired characteristic equation for the combined system defined as,

$$\alpha_c(s) = s^3 + \alpha_2 s^2 + \alpha_1 s + \alpha_0 \quad (5.2.22)$$

the feedback gains can be obtained by solving the three equations arising from each of the terms of the characteristic equation and are given by,

$$K_P = \frac{I_x}{L_{\delta_A}} \left( \alpha_2 + \frac{L_P}{I_x} \right) \quad (5.2.23)$$

$$K_E = \frac{I_x}{L_{\delta_A}} \left( \alpha_1 + \frac{\alpha_0}{z_f} \right) \quad (5.2.24)$$

$$K_\phi = \frac{\alpha_0}{\alpha_1 + \frac{\alpha_0}{z_f}} \quad (5.2.25)$$

where  $z_f$  is the location of the feed-forward zero that can be placed using  $\bar{N}_P$  as follows,

$$\bar{N}_P = -\frac{K_E}{z_f} \quad (5.2.26)$$

### 5.2.3 Velocity Controller

Because the SAM guidance controller is only used in the aircraft's YZ-plane and not in full 3D as proposed by [1], a method of regulating the aircraft's velocity is required. For a conventional flight envelope a simple velocity controller can be designed to command the aircraft's throttle directly. This controller will easily track a trim velocity reference and thereby ensure the aircraft progresses forward along the position based trajectory.

For full 3D flight a complicating factor presents itself. The amount of gravity coupling into the system can be more than the maximum or minimum controller commands are able to counter. In order to keep the aircraft from stalling or flying too fast, an algorithm is required that looks forward in time and generates a small pre-emptive velocity error in order to counter for a large loss or gain in velocity in the future. Two algorithms such as this have been developed in this project and will be discussed in Chapter 6. For the trajectory flight results shown later in this chapter, the method of *kinetic energy analysis with logic switching* is used to create a pre-emptive command for the *velocity controller* developed in this section.

The dynamics of the throttle actuator which will be used to regulate the aircraft's velocity is modelled using the first order Padé approximation for the throttle lag, derived in section 2.4.3. This model is given by,

$$\dot{T}_s = \left[ -\frac{1}{\tau_p} \right] T_s + \left[ \frac{2}{\tau_p} \right] T_c \quad \text{with} \quad \tau_p = 0.3 \quad (5.2.27)$$

$$T = T_s - T_c \quad (5.2.28)$$

The dynamics for the velocity magnitude of the aircraft's wind axes is given by equation (2.3.5) and restated below,

$$\dot{\bar{V}} = A_W + g e_{13}^{WI} \quad (5.2.29)$$

A link is required between these two sets of dynamics. Equation (3.1.6) shows that the aircraft's wind axes ASA can be written in terms of the aerodynamic drag as well as the aircraft's thrust as follows,

$$A_W = \left[ \frac{1}{m} \right] T + \left[ -\frac{qS}{m} C_D \right] \quad (5.2.30)$$

By substituting the ASA equation into the velocity dynamics and combining this with the throttle dynamics gives,

$$\begin{bmatrix} \dot{T}_s \\ \dot{\bar{V}} \end{bmatrix} = \begin{bmatrix} -\frac{1}{\tau_p} & 0 \\ \frac{1}{m} & 0 \end{bmatrix} \begin{bmatrix} T_s \\ \bar{V} \end{bmatrix} + \begin{bmatrix} \frac{2}{\tau_p} \\ -\frac{1}{m} \end{bmatrix} T_c + \begin{bmatrix} 0 \\ -\frac{qS}{m} C_D \end{bmatrix} + \begin{bmatrix} 0 \\ g e_{13}^{WI} \end{bmatrix} \quad (5.2.31)$$

A velocity integrator is augmented to the system to ensure reference tracking as follows,

$$\dot{E}_V = \bar{V} - \bar{V}_R \quad (5.2.32)$$

where  $\bar{V}_R$  is the velocity reference command. A throttle control law is defined as follows,

$$T_C = T_{C_V} + T_{C_E} + \bar{N}_V \bar{V}_R \quad (5.2.33)$$

$$= -K_V \bar{V} - K_E E_V + \bar{N}_V \bar{V}_R \quad (5.2.34)$$

with the  $T_{C_V}$  and  $T_{C_E}$  terms corresponding to the throttle command generated due to feedback from velocity and the augmented integrator respectively. Feed-forward is also provided from the commanded velocity  $\bar{V}_R$ .

Similar to the ASA controller, drag can once again be seen as a disturbance that a fast velocity integrator should be able to counter in the steady state. The effect of the gravity coupling will also usually change relatively slowly. The large coupling generated during high pitch angles will be compensated for by the *kinetic energy* algorithm discussed in the following chapter. For the design of this controller, the effect of gravity can therefore be neglected from the model.



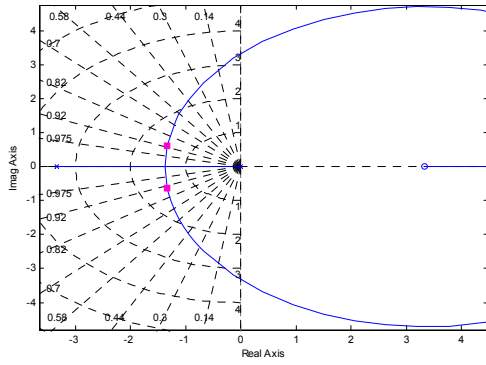
A consecutive loop closure controller design can now be done on the system for the velocity and the velocity integrator gain. The system with the control law substituted, but before each of the feedbacks are carried out, is represented below. The effects of drag and gravity have been removed from the system.

$$\begin{bmatrix} \dot{T}_s \\ \dot{\bar{V}} \end{bmatrix} = \begin{bmatrix} -\frac{1}{\tau_p} & 0 \\ \frac{1}{m} & 0 \end{bmatrix} \begin{bmatrix} T_s \\ \bar{V} \end{bmatrix} + \begin{bmatrix} \frac{2}{\tau_p} \\ -\frac{1}{m} \end{bmatrix} T_{C_V} + \begin{bmatrix} \frac{2}{\tau_p} \\ -\frac{1}{m} \end{bmatrix} T_{C_E} + \begin{bmatrix} \frac{2\bar{N}_V}{\tau_p} \\ -\frac{\bar{N}_V}{m} \end{bmatrix} \bar{V}_R \quad (5.2.35)$$

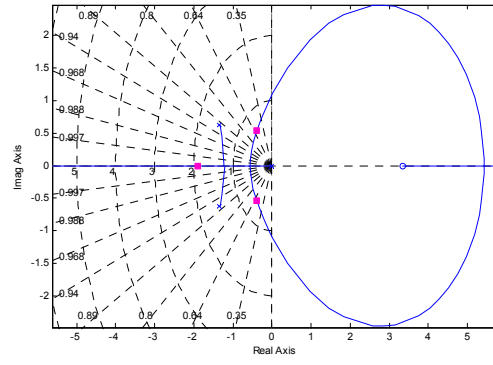
First analyse the root locus for the variation in the velocity feedback gain on the second order system without the augmented integrator as represented above. Figure 5.3(a) shows the highly damped pole placement result for a velocity feedback gain of  $K_V = 3.6$ . The integrator can now be augmented to the system and the second feedback gain can be determined. The system, after the velocity feedback has been applied, is given by,

$$\begin{bmatrix} \dot{T}_s \\ \dot{\bar{V}} \\ \dot{E}_V \end{bmatrix} = \begin{bmatrix} -\frac{1}{\tau_p} & -\frac{2}{\tau_p} K_V & 0 \\ \frac{1}{m} & \frac{1}{m} K_V & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} T_s \\ \bar{V} \\ E_V \end{bmatrix} + \begin{bmatrix} \frac{2}{\tau_p} \\ -\frac{1}{m} \\ 0 \end{bmatrix} T_{C_E} + \begin{bmatrix} \frac{2\bar{N}_V}{\tau_p} \\ -\frac{\bar{N}_V}{m} \\ -1 \end{bmatrix} \bar{V}_R \quad (5.2.36)$$

The second root locus for the variation in the integrator gain is shown in figure 5.3(b). A gain of  $K_E = 1.4$  provides two dominant complex poles with a natural frequency of  $\omega_n = 0.67$  [rad/s] and damping of  $\zeta = 0.585$ .



(a) Root locus for variation in *velocity* feedback gain



(b) Root locus for variation in *integrator* feedback gain

**Figure 5.3:** Consecutive Root Locus Plots for the Design of the Velocity Controller

### 5.3 Implementation

The specific implementation of the controllers discussed in section 5.2 will be outlined here. Their responses will be analysed and any problems with their implementation will be discussed.

### 5.3.1 Pole Placement

The closed loop pole for the error angle of the NSAVDC is selected at,

$$p_\phi = -3.5 \text{ [rad/s]} \quad (5.3.1)$$

with the roll rate and integrator pole placed at the same locations as for the roll rate controller shown in section 3.5. This error angle pole is then time scale decoupled from the guidance dynamics which have been placed at a frequency and damping of,

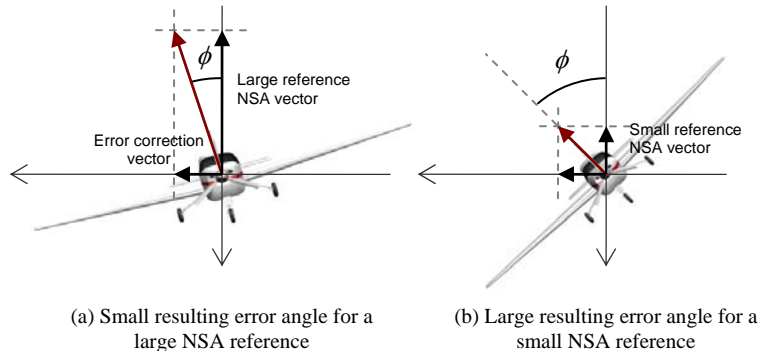
$$\omega_n = 0.75 \text{ [rad/s]} \quad (5.3.2)$$

$$\zeta = 0.8 \quad (5.3.3)$$

The separation in frequency of only 4.67 times between these two sets of dynamics was found to be enough for the time scale separation assumption to hold.

### 5.3.2 Small NSA Vector Command Problem

When the aircraft is following a reference trajectory which, during a certain section of a manoeuvre, requires very little NSA this issue will present itself. If the size of the reference NSA vector becomes very small, and a small error correction vector is added to it by the guidance controller, it could lead to a very large change in its direction and therefore also to the induced error angle of the NSAVDC. This is illustrated by figure 5.4 which shows the same required lateral specific acceleration being added to a small and large reference NSA and the resulting error angles ( $\phi$ ) that the aircraft banks to.



**Figure 5.4:** Large Bank Angle Command Generated for a Small NSA Vector

Two strategies can be used to correct this issue. The first is the skid-to-turn technique. This is where instead of rolling the aircraft so that its NSA is orientated in the correct direction, the commanded NSA vector is split into its lateral and normal components and the inner loop NSA and LSA controllers are used to achieve each.

For the second method, which is used in this project, it can be argued that when the command for the NSA vector is very small it is not very important for the aircraft to achieve this command. In fact, it would be preferable for the plane not to suddenly roll in order to apply a very small commanded NSA vector. Therefore the simplest solution is not to roll at all until the NSA vector command becomes large once again. In order to achieve this, the error angle pole of the NSAVDC is quadratically reduced in frequency as the commanded size of the NSA starts to become small.

### 5.3.3 Step Responses

#### 5.3.3.1 SAM Guidance Controller

In order to analyse the performance of the controller, simulated state errors can be induced and the resulting rejection of these errors by the control system can then be investigated. Position errors are induced on the system as they will best illustrate the controller's ability to guide the aircraft back onto the reference trajectory.

**Lateral Position Error** A lateral position error will show the controller's ability to roll the aircraft in order to turn and correct a position tracking error. Figure 5.5 shows this for a 10 m lateral position error where it can be seen that the aircraft rolls to an angle of just over 20 degrees in order to turn. Due to the fast time scale decoupled pole of the NSAVDC, this control system is much more aggressive on its roll rate as opposed to the LQR controller designed in the previous chapter. Therefore, the roll rate induced due to an error angle is limited to 40 deg/s and it can be seen how this limit is reached here. This can be compared to the maximum commanded roll rate of 20 deg/s shown in figure 4.11 for the LQR controller with the same induced lateral position error. This aggressive control leads to faster rejection of the lateral position error which occurs in about 4.5 seconds – as opposed to 5.5 seconds for the LQR controller.

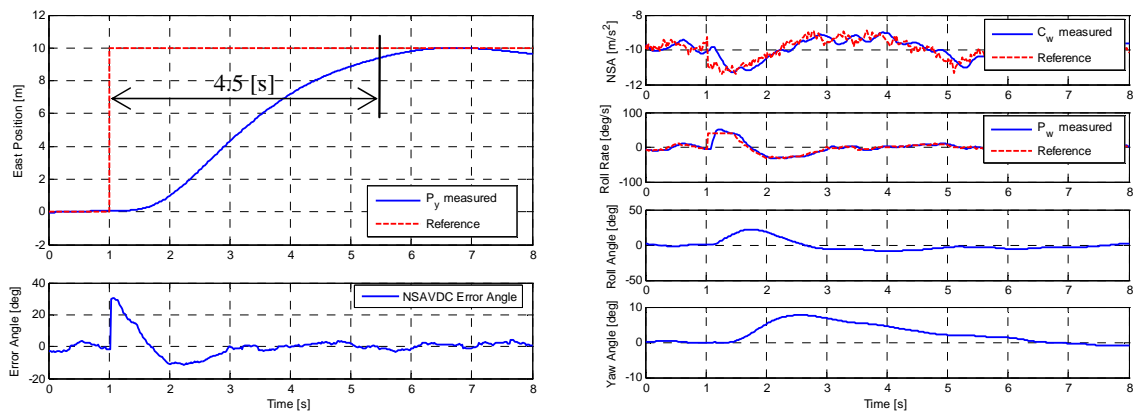
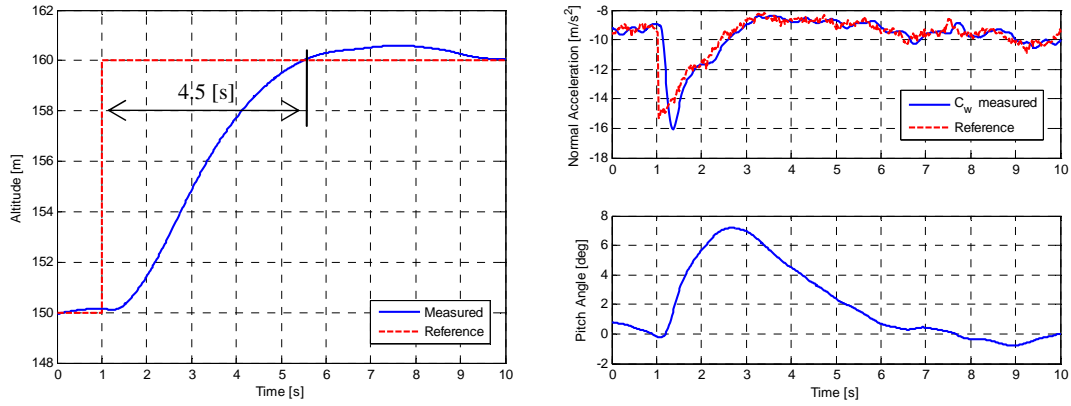


Figure 5.5: Lateral Position Step Response of the SAM Controller

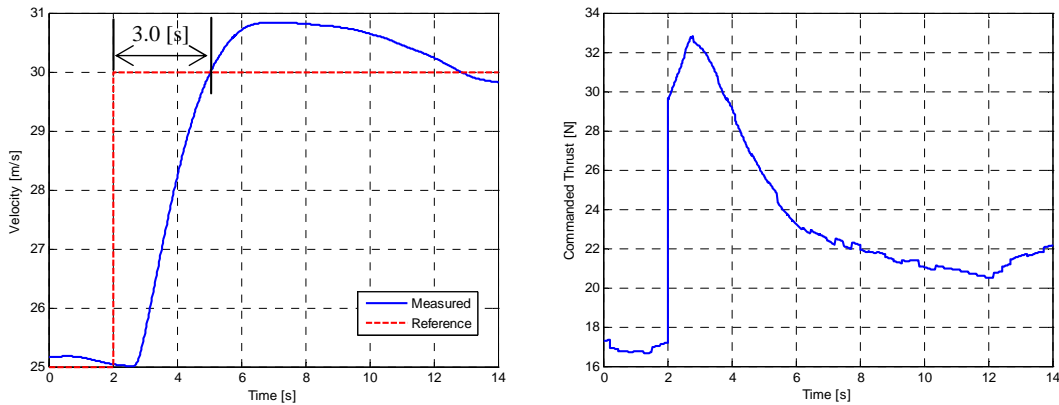
**Altitude Error** During level flight a negative altitude error should merely cause the controller to command a larger negative NSA and thereby pitch the aircraft up until it reaches the correct altitude. Figure 5.6 illustrates how this controller does exactly that and rejects a 10 m altitude error in about 4.5 seconds. This performance is an improvement on the LQR controller's response time of 6.0 seconds visible in figure 4.12.



**Figure 5.6:** Altitude Step Response of the SAM Controller

### 5.3.3.2 Velocity Controller

For this controller a step command is given to the aircraft's velocity and the resulting response is analysed. Figure 5.7 shows this response where it can be seen that the commanded velocity is achieved with some overshoot in about 3 seconds. The speed of this controller can be considered acceptable as it does not have to be time scale decoupled from any other dynamics. This is because the velocity and guidance controllers operate completely independent of each other.



**Figure 5.7:** Step Response of the Velocity Controller

## 5.4 Position Based Trajectory Generation

As previously stated, smooth position based trajectories were developed in this project for use with the SAM control architecture. Instead of containing the state and control vectors required by the controller, sampled at a given rate and thereby determining the time which the aircraft should take to complete the trajectory, these trajectories mathematically describe the path to be followed by the aircraft. Given the aircraft's current location, these trajectories are able to return the following information,

- $\mathbf{P}_I^{RI}$  – Closest point of the aircraft to the reference trajectory.
- $\mathbf{V}_I^{RI}$  – Reference velocity vector at that point.
- $P_{W_C}$  – The roll rate required to follow the reference trajectory.
- $\Sigma_I^{RI}$  – Reference specific acceleration vector.
- $\mathbf{k}_I^R$  – Reference NSA unit vector.

For these trajectories the reference specific acceleration vector will only consist of the aircraft's NSA vector and will not include any ASA data. This is due to the aircraft using an independent velocity controller. Therefore,

$$\Sigma_I^{RI} = \mathbf{N}^R \quad (5.4.1)$$

where  $\mathbf{N}^R$  is the NSA vector required to fly the reference trajectory. These reference values depend on the aircraft's current velocity. The NSA magnitude required to fly a loop for instance is given by,

$$C_{W_R} = -\frac{\bar{V}^2}{r} - g e_{33}^{WI} \quad (5.4.2)$$

where  $r$  is the radius of the loop. The current velocity of the aircraft is therefore used in conjunction with its position to generate these reference flight paths.

### 5.4.1 Basic Building Blocks

Four basic trajectory building blocks are created that can be strung together to form various complex flight trajectories. The *detailed mathematical equations* used to derive the trajectory information from the provided position and velocity are shown in Appendix D.

#### 5.4.1.1 Straight Line Flight

This is defined as a straight flight path between any two specified points in inertial space. It can be a steady climb or descent, or can simply be a level flight path. Figure 5.8 shows this trajectory with the starting and ending position indicated by  $\mathbf{P}_I^S$  and  $\mathbf{P}_I^E$  respectively.

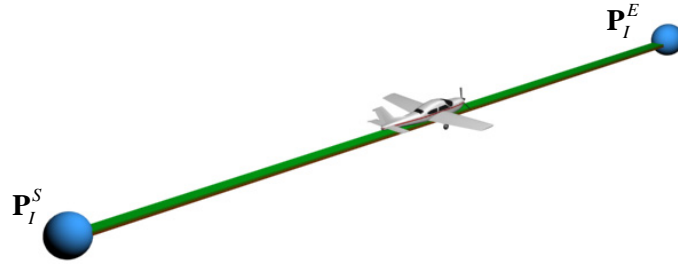


Figure 5.8: Straight Line Position Based Trajectory Building Block

#### 5.4.1.2 Aileron Roll

This trajectory is used by the aircraft to execute a roll from a specified starting position ( $\mathbf{P}_I^S$ ) and roll angle ( $\phi_S$ ) to an ending position ( $\mathbf{P}_I^E$ ) and roll angle ( $\phi_E$ ) as illustrated by figure 5.9.

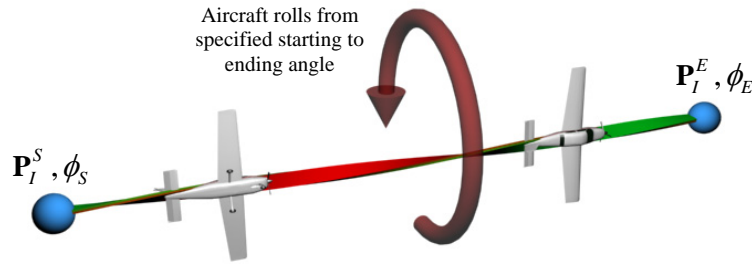


Figure 5.9: Aileron Roll Position Based Trajectory Building Block

#### 5.4.1.3 Vertical Spiral Arc

The vertical spiral arc is a trajectory where the aircraft does a loop while also moving sideways at a constant rate as shown in figure 5.10. The various parameters that define this trajectory are:

- $\mathbf{P}_I^S$  – The starting point of the trajectory.
- $\mathbf{P}_I^M$  – The midpoint of the loop.
- $\alpha_S$  and  $\alpha_E$  – The starting and ending angles of the loop.
- $\Delta l$  – The lateral distance travelled after a full loop rotation.
- $\psi_o$  – The horizontal direction of the loop, where  $\psi_o = 0$  implies a loop that is parallel to the inertial ND-plane.
- A Boolean variable that specifies whether this is an inside or outside loop.

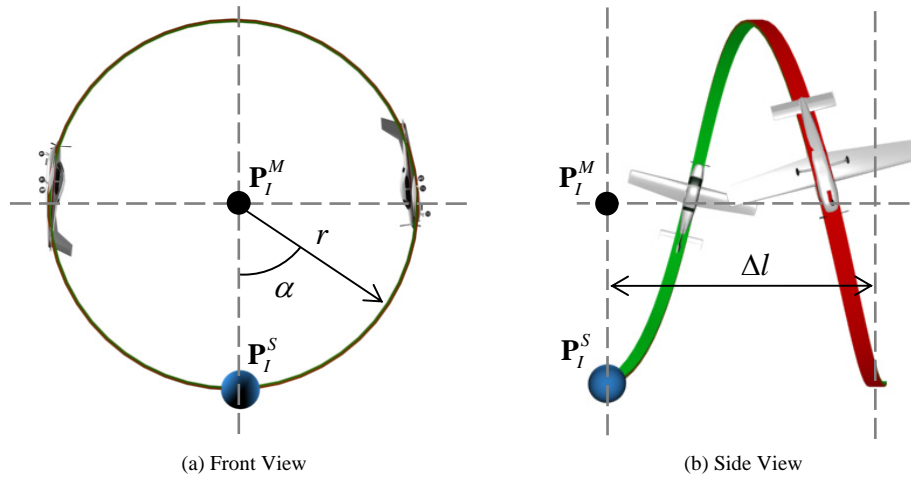


Figure 5.10: Vertical Spiral Arc Position Based Trajectory Building Block

#### 5.4.1.4 Horizontal Spiral Turn

In this trajectory the aircraft flies a coordinated turn around a specified midpoint. During this turn the aircraft is able to continually gain or loose altitude as shown in figure 5.11. The parameters that define this trajectory are:

- $P_I^S$  – The starting point of the trajectory.
- $P_I^M$  – The midpoint of the turn.
- $\alpha_t$  – The total angle through which the aircraft will turn.
- $\Delta h$  – The amount of altitude gain or loss per 360 degree turn.

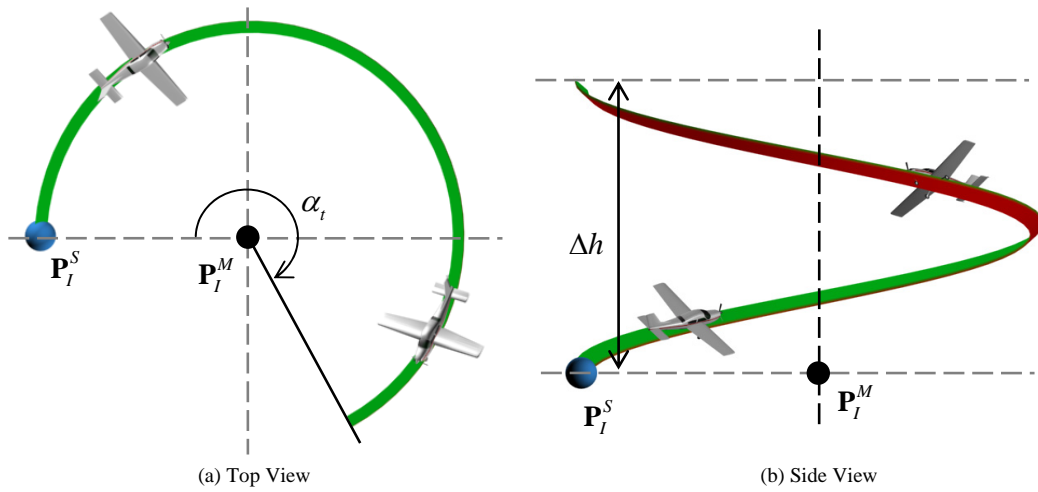


Figure 5.11: Horizontal Spiral Turn Position Based Trajectory Building Block

### 5.4.2 Complete Trajectories

By stringing these trajectory building blocks together, various flight trajectories are created for use with the SAM outer loop controller. These trajectories, discussed in Chapter 1, are *straight and level flight*, an *aileron roll*, an *aggressive climb* manoeuvre, an ascending *horizontal spiral turn* consisting of three full turns, an *Immelmann* and a *vertical loop*. An example of how these building blocks can be strung together to form a more complex trajectory is shown in figure 5.12 for the aggressive climb manoeuvre.

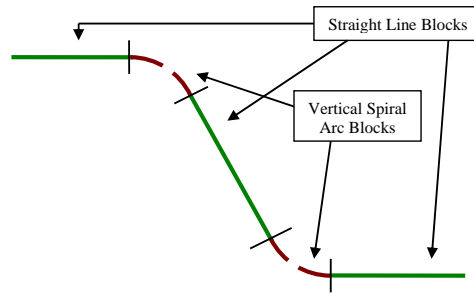


Figure 5.12: Assembly of a Complex Trajectory Using Basic Building Blocks

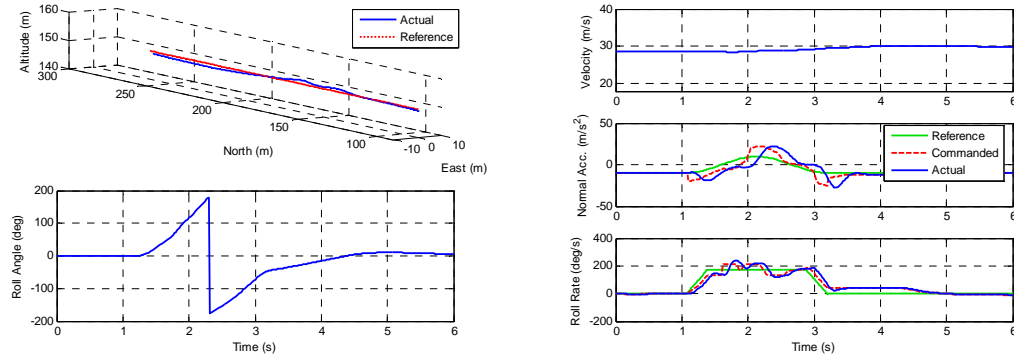
## 5.5 Simulated Trajectory Flight Results

Using the SAM controller to generate the NSA and roll rate commands and *kinetic energy analysis of the reference trajectory* (to be discussed in section 6.4) in conjunction with the velocity controller, the position based trajectories are flown in simulation by the control system. The more notable results are displayed here, while the rest of the simulations are provided in Appendix C, section C.5. Some observations can be made from these results:

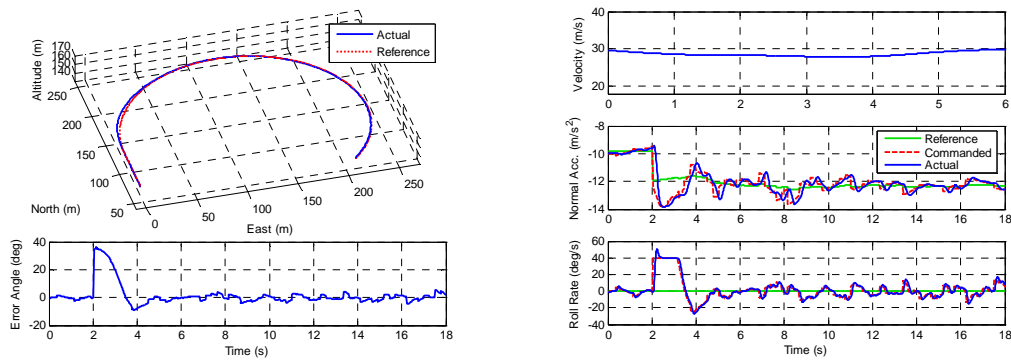
- General trajectory tracking is slightly better than that of the LQR outer loop controller. This is due to the more intuitive nature with which the poles of this control architecture can be placed. This for instance allows for the placement of a fast NSA vector error angle rejection pole without violating any time scale constraints.
- In the horizontal spiral turn trajectory of figure 5.14, a transition is directly made from level flight to a coordinated turn. It is left to the control system to roll the aircraft's NSA vector to the correct orientation. It can be observed how the error angle of the NSA vector is quickly driven to zero by the NSAVDC.
- Due to position based trajectories being used, trajectories such as the vertical loop, displayed in figure 5.15, are executed much better since the effect of trajectory run-away has been removed. The trajectory reference is generated in real time and therefore references such as the amount of NSA required to execute the loop are



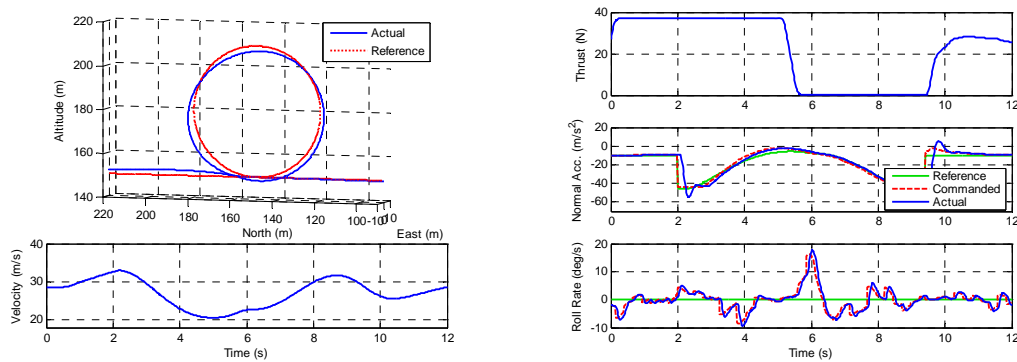
based on the aircraft's current velocity and not that of a predetermined reference. Figure 4.16 of the simulated vertical loop executed by the LQR control system can be compared to figure 5.15 to observe the improvement.



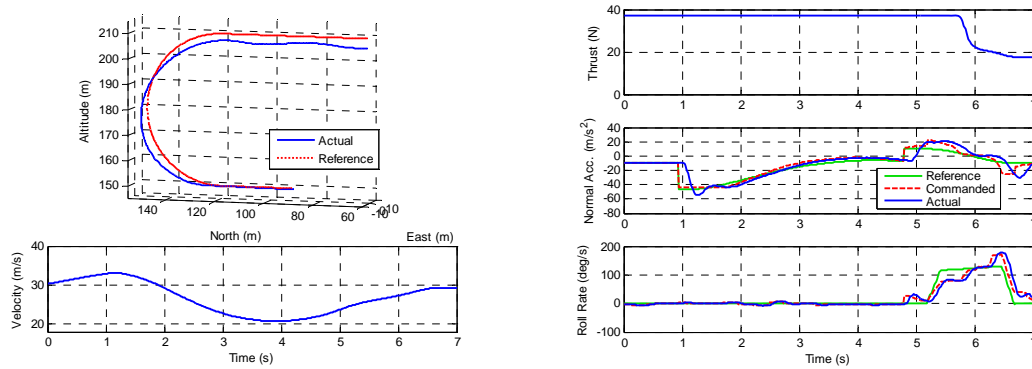
**Figure 5.13:** Simulation of the Aileron Roll Trajectory Flown by the SAM Controller



**Figure 5.14:** Simulation of the Horizontal Spiral Turn Trajectory Flown by the SAM Controller



**Figure 5.15:** Simulation of the Vertical Loop Trajectory Flown by the SAM Controller



**Figure 5.16:** Simulation of the Immelmann Trajectory Flown by the SAM Controller

## 5.6 Practical Flight Test Results

Using the CAP232 aerobatic aircraft, practical flight tests for this control system were performed for all the trajectories shown in the previous section. In order for the safety pilot to determine the direction in which the manoeuvre is performed, the same method of assigning the current yaw angle as the zero reference as used for the practical implementation of the LQR controller, is employed here. Again, the more notable results are shown in this section with the rest provided in section C.5 of Appendix C.

### 5.6.1 Straight and Level Flight

Similar to the testing of the LQR controller, the straight and level flight trajectory is used as an initial test to confirm the correct functionality of the control system. This trajectory was flown successfully with the controller able to stabilise the aircraft and keep it on the flight path. The results from this test is shown in figure C.9 in Appendix C. This was also the first test for the velocity controller discussed in the section 5.2.3. It can be seen that it was also successful at regulating the aircraft's velocity to its commanded trim of 30 m/s.

### 5.6.2 Aileron Roll

This trajectory shows the control system's ability to roll very quickly and precisely. In order for the aircraft not to lose altitude during this manoeuvre, a small positive pitch angle is requested by the reference trajectory. Figure 5.17 shows the practical execution of this manoeuvre with the aircraft rolling at more than 200 deg/s and then returning to a level flight path without any significant loss in altitude.

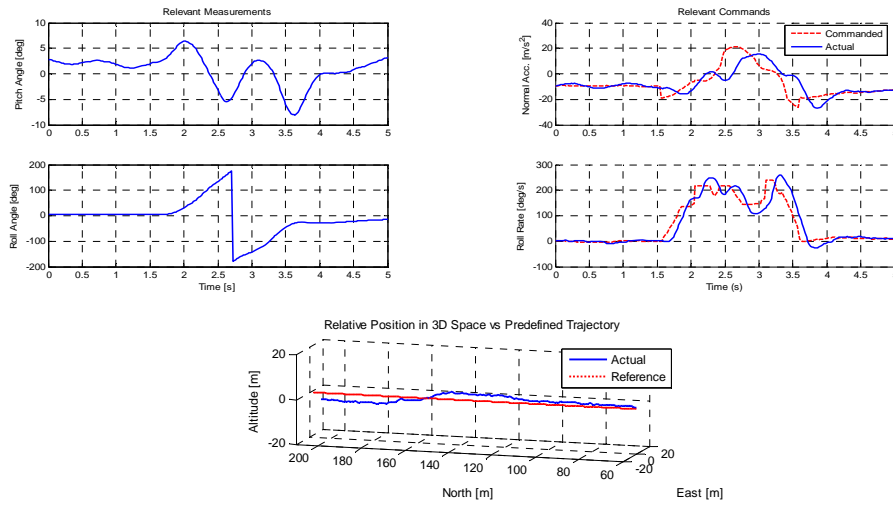


Figure 5.17: Practical Flight Results for the Aileron Roll by the SAM Controller

### 5.6.3 Horizontal Spiral Turn

With this trajectory the aircraft is allowed to make three full turns while continually banked to about 40 degrees. Similar to the simulation, the trajectory is immediately transitioned from level flight to a coordinated turn. It can be seen from the flight test results in figure 5.18 how the aircraft slightly overshoots the trajectory during this transition while the NSAVDC drives the error angle to zero and banks the aircraft to the correct angle.

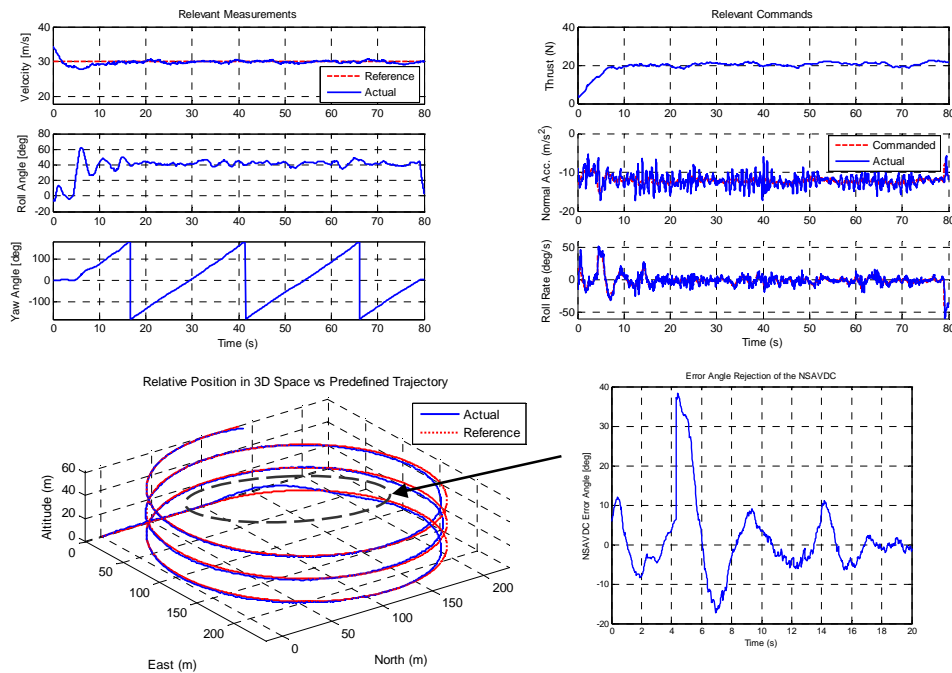
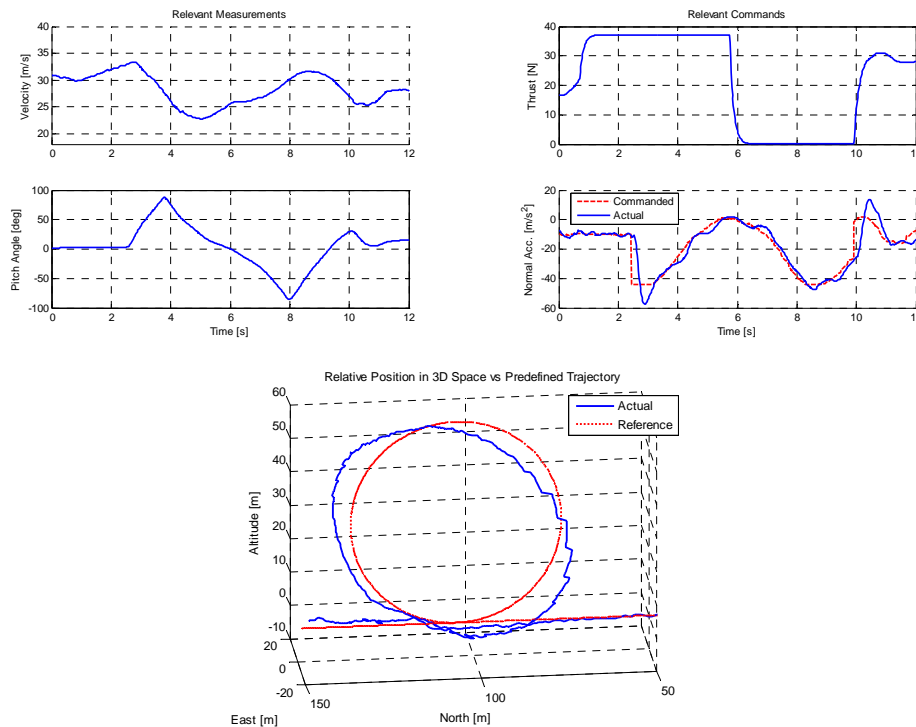


Figure 5.18: Practical Flight Results for the Horizontal Spiral Turn by the SAM Controller

### 5.6.4 Vertical Loop

As shown by figure 5.19, this controller is somewhat more successful at completing a vertical loop manoeuvre than the LQR controller of the previous chapter. The advantages of the position based trajectories discussed previously can be seen with the practical implementation of this trajectory where the correct amount of NSA is continually calculated and applied to counter any position errors before they occur. With the LQR controller, velocity errors induced altitude errors since the control system traded altitude for velocity. With this decoupled control architecture this does not occur and therefore a more circular loop can be observed. The small jumps that are visible on the actual flight path are estimator corrections due to severe GPS accuracy loss as a result of the high g manoeuvre.



**Figure 5.19:** Practical Flight Results for the Vertical Loop by the SAM Controller

### 5.6.5 Immelmann

Figure 5.20 shows an autonomous Immelmann manoeuvre being executed with the aircraft doing a half loop and then quickly rolling through 180 degrees to fly in the opposite direction. During the top section of this trajectory estimator drift can once again be observed as a result of GPS accuracy loss. This led to some position tracking errors, however the controller still managed to complete the manoeuvre and regulated the aircraft back to the flight path at the end of the trajectory.

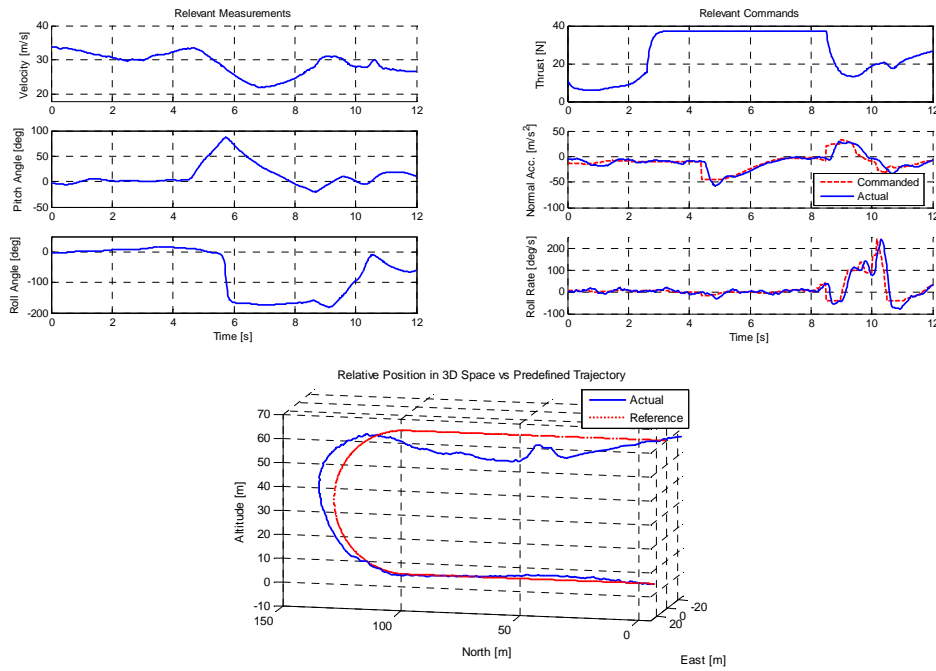


Figure 5.20: Practical Flight Results for the Immelmann by the SAM Controller

## 5.7 Summary

This chapter discussed the second method of outer loop control used in this project. Simulated as well as practical results showed how this controller improves on some of the shortcomings of the previous control architecture. These improvements are listed below:

- The control algorithms are not very complex and do not require nearly as much computational power as the LQR controller discussed in the previous chapter.
- More insight into the controller is available with the response of various aspects being easily adjustable. Roll angle, positional and directional tracking can all be adjusted through the direct pole placement available to this controller. The controller can therefore be made more aggressive than the LQR control system, without violating the time scale separation constraint.
- Position based trajectories developed for this control architecture provide improved references and aid in minimising state errors. Using the basic trajectory building blocks that were developed, new trajectories can also be created more easily.
- The problem of the aircraft's velocity regulation has been split from the guidance problem. Therefore an independent velocity controller has been developed that is simply responsible for keeping the aircraft's velocity within the allowable limits. In order to counter for the control saturation effect of gravity, an additional predictive algorithm is also required. This topic will be discussed in the chapter to follow.

## Chapter 6

# Predictive Velocity Regulation

In the previous chapter it was stated that algorithms will be developed that are able to deal with the extensive gravity coupling into the velocity dynamics. The coupling is introduced when the aircraft is pitched up or down. For small pitch angles, the coupling can still be countered by adding an integrator to the control system which will remove the error caused by the coupling effect in the steady state. If the change in pitch angle occurs faster than the bandwidth of the velocity control architecture can account for, velocity tracking errors will start to present themselves. These errors are of small consequence however as a pitch angle change will typically occur in a constant direction for a prolonged period of time until a newly desired pitch angle is reached. If this angle change occurs at a high speed, the throttle will quickly be driven into saturation and the bandwidth of the throttle control system will no longer be of any consequence.

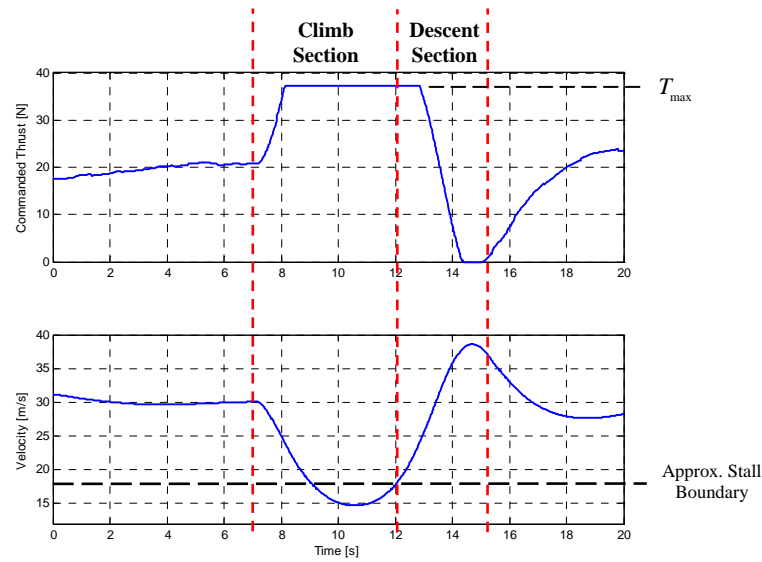
For steep climbs this occurs when the amount of gravity coupling is equal to the ASA achieved by the maximum throttle setting. Similarly, for steep descents this occurs when the braking force of the aerodynamic drag equals the amount of gravity coupling. If the pitch angle deviates beyond these points, the additional gravity coupling will quickly cause the aircraft to exit its allowable velocity range and either stall or fly too fast.

Time based trajectories used by the LQR kinematic controller provide a feed-forward command for ASA and is therefore able to (in an open loop fashion) predictively counter the effect of large gravity coupling. In order to more elegantly prevent this effect when flying position based 3D trajectories, two algorithms have been developed in this project that are able to use the limited dynamic range of the aircraft's throttle along with prior knowledge of the flight path provided by the reference trajectory, to create a pre-emptive velocity error which will counter a large gain or loss in velocity that is about to follow. The first method of *optimal control* provides the best solution to this problem but does so at the cost of increased computational burden. The second method of *kinetic energy analysis with logic switching* does not require nearly as much computational power but does not provide an optimal solution. It merely attempts to prevent large losses or gains in velocity using a decision making structure to provide a pre-emptive velocity command.

## 6.1 The Velocity Regulation Problem

The vertical loop trajectory (refer to figure 1.6) will be used throughout this chapter to illustrate the functionality of these algorithms as it provides a very steep initial climb which is directly followed by a steep descent. This will test the algorithm's ability to provide the initial excess velocity required to vertically climb without stalling and then also show if it will decrease the aircraft's throttle at the correct point at the top of the loop in order to prevent the aircraft from picking up too much speed on the way down.

Figure 6.1 shows a failed initial simulated attempt at the vertical loop trajectory using the SAM guidance controller and velocity controller discussed in Chapter 5. No predictive algorithms are used and the aircraft's velocity is simply commanded to trim.



**Figure 6.1:** Initial Attempt at the Vertical Loop Trajectory without Predictive Velocity Algorithms

A proportional increase in the throttle command can be seen at the start of the loop as soon as the velocity starts to decrease. At its lowest point the velocity drops below 15 m/s. The aerodynamic model used for this simulation does not include the effects of stall, but it is known that practically the aircraft will start to stall from about 17 m/s. This will especially be the case when it is inducing a large angle of attack in order to create the required NSA to execute a vertical loop. This result proves the necessity for a predictive velocity regulation algorithm and can be compared to the results obtained later in this chapter in order to observe the improvement. The absolute minimum and maximum bounds for the aircraft's velocity that the predictive velocity regulation algorithms should assist the aircraft in avoiding, are therefore defined as follows,

$$\bar{V}_{\min} = 18 \text{ [m/s]} \quad \text{and} \quad \bar{V}_{\max} = 40 \text{ [m/s]} \quad (6.1.1)$$

## 6.2 Modelling

This section shows the mathematical models used to describe the effects of drag, thrust and gravity on the aircraft's velocity at any point on the trajectory. These models will assist in predicting future velocity changes using the predetermined flight trajectory.

Equation (2.3.5) provides a model for the aircraft's velocity dynamics in terms of ASA and gravity.

$$\dot{\mathbf{V}} = [\mathbf{ge}_{13}^{WI}] + [\mathbf{A}_W] \quad (6.2.1)$$

The wind axes axial specific acceleration in the above equation can be written as,

$$\mathbf{A}_W = \mathbf{A}_{W_T} + \mathbf{A}_{W_D} \quad (6.2.2)$$

with  $\mathbf{A}_{W_T}$  and  $\mathbf{A}_{W_D}$  corresponding to the ASA due to the aircraft's thrust and drag respectively. With reference to equation (3.1.6) these two terms can then be expanded as follows,

$$\mathbf{A}_{W_T} = \frac{1}{m} [\mathbf{T}] \quad (6.2.3)$$

and

$$\mathbf{A}_{W_D} = \left[ -\frac{qS}{m} C_D \right] \quad (6.2.4)$$

with the drag coefficient given by equation (2.4.17) as,

$$C_D = C_{D_0} + \frac{C_L^2}{\pi A e} \quad (6.2.5)$$

The lift term in the above equation can be obtained by assuming that lift is mainly generated by the angle of attack and thereby simplifying equation (2.4.18) as follows,

$$C_L = C_{L_0} + C_{L_\alpha} \alpha \quad (6.2.6)$$

The angle of attack can then in turn be written in terms of the NSA and dynamic pressure using equation (3.7.3) and adding the effect of  $C_{L_0}$  to create a general solution.

$$\alpha = -\frac{1}{C_{L_\alpha}} \left[ C_W \frac{m}{qS} + C_{L_0} \right] \quad (6.2.7)$$

If the velocity at the prediction point is known, then the amount of NSA required to fly the position based reference trajectory can be obtained using the algorithms shown in Appendix D. This velocity can also be used to calculate the dynamic pressure using equation (2.4.16). Therefore, using the equations shown above, the ASA due to drag term  $\mathbf{A}_{W_D}$  can be calculated at any point on the trajectory given the aircraft's current velocity at that point.



The velocity dynamics can now be written as follows,

$$\dot{\mathbf{V}} = \left[ g e_{13}^{WI} + A_{W_D} \right] + A_{W_T} \quad (6.2.8)$$

where the DCM term in this equation shows the amount of gravity coupling into the velocity dynamics and can also be obtained from the flight trajectory. For every point on the trajectory the bracketed portion of the above equation can be calculated if the aircraft's velocity at that point is known. The ASA caused by the aircraft's throttle acts as the control input to the velocity dynamics with limits defined as follows,

$$A_{W_T}^{\min} \leq A_{W_T} \leq A_{W_T}^{\max} \quad (6.2.9)$$

These limits can be calculated for the CAP232 model aircraft used in this project as follows,

$$A_{W_T}^{\min} = \frac{1}{m} T_{\min} = 0 \text{ [m/s}^2\text{]} \quad (6.2.10)$$

and

$$A_{W_T}^{\max} = \frac{1}{m} T_{\max} \approx 7 \text{ [m/s}^2\text{]} \quad (6.2.11)$$

with  $T_{\max}$  given by equation (3.2.22).

### 6.3 Optimal Control

Optimal control of a non-linear time variant system, as outlined in [7], can be described as follows. The goal is to find the optimal control  $\underline{u}^*(t)$  that causes the system defined by,

$$\dot{\underline{x}}(k) = \underline{a}(\underline{x}(t), \underline{u}(t), t) \quad \text{and} \quad \underline{x}(t_0) = \underline{x}_0 \quad (6.3.1)$$

to respond in an optimal manner and thereby minimise the performance measure (or cost function) that is defined for the system. The cost function is defined as the integral, from an initial to a final time, of a chosen function in terms of the state and control vectors as well as time.

$$J = \int_{t_0}^{t_f} g(\underline{x}(t), \underline{u}(t), t) dt \quad (6.3.2)$$

The optimal control result of the form,

$$\underline{u}^*(t) = \underline{f}(\underline{x}(t), t) \quad (6.3.3)$$

will be obtained by this algorithm. The optimal control law  $\underline{f}(\underline{x}(t), t)$  specifies the optimal control value at time  $t$  and in terms of the state vector at time  $t$ . Therefore the optimal control law can be time varying.

### 6.3.1 Strategy

The velocity dynamics derived in the previous section are restated below,

$$\dot{\bar{V}} = \left[ ge_{13}^{WI} + A_{W_D} \right] + A_{W_T} \quad (6.3.4)$$

The bracketed expression is time variant and depends on the reference trajectory. As stated before, the  $A_{W_T}$  term acts as the control input to the system. It is decided not to model the ASA lag on the control input as this will introduce a second state to the system and extra dimensionality causes the dynamic programming algorithm, which will be used to find the optimal control solution, to become much more computationally demanding as will be discussed later.

The cost function for this optimal control problem is defined as follows,

$$J = \int_{t_0}^{t_0+T} \left[ (\bar{V}(t) - \bar{V}_{Trim})^6 + k_C (A_{W_T}(t) - A_{W_T}^{nom}(t, \bar{V}))^2 \right] dt \quad (6.3.5)$$

where  $t_0$  is the starting time,  $T$  is the time interval (or horizon time) over which the cost function is to be minimised and  $k_C$  is an adjustable variable for weighing state or control deviations more strongly. Furthermore,  $\bar{V}_{Trim}$  signifies the desired trim velocity which the actual velocity is to be regulated to and  $A_{W_T}^{nom}(t, \bar{V})$  is the nominal amount of ASA produced by the thrust that will counter the effect of gravity and drag at any given time. The equation for  $A_{W_T}^{nom}(t, \bar{V})$  is given by,

$$A_{W_T}^{nom}(t, \bar{V}) = - \left[ ge_{13}^{WI}(t) + A_{W_D}(t, \bar{V}) \right] \quad (6.3.6)$$

It can be seen that velocity state deviations from trim are weighed in the cost function at a sixth order level. This will cause large velocity deviations, such as those nearing the absolute minimum or maximum bounds, to create very large costs and will therefore be aggressively avoided by the resulting optimal control law while small deviations will largely be ignored.

### 6.3.2 Dynamic Programming

Using the known velocity dynamics as well as the defined cost function, the method of dynamic programming, as presented by [7], can be used to find the time varying optimal control input for the system. The first step is to define upper and lower limits for the system's states and control inputs as well as discretise the entire problem. The velocity state limits can be defined by the absolute minimum and maximum bounds shown in section 6.1 and the acceleration control input limits are given by equations (6.2.10) and (6.2.11). These state and control ranges can be broken up into discrete intervals by choosing a

discretising step size for each as follows,

$$\Delta \bar{V} = 0.1 \text{ [m/s]} \quad \text{and} \quad \Delta A_{W_T} = 0.2 \text{ [m/s}^2\text{]} \quad (6.3.7)$$

A discrete time range can also be defined over which the cost function will be minimised as follows,

$$t_0 \leq t \leq t_0 + 20 \text{ [s]} \quad (6.3.8)$$

where  $t_0$  is the starting time and with a discretising step size chosen as,

$$\Delta t = 0.1 \text{ [s]} \quad (6.3.9)$$

Equation (6.3.4) for the system's continuous velocity dynamics and equation (6.3.5) for the cost function can also be converted to discrete difference equation form. The approximate discretised result for the velocity dynamics using Euler integration is given by,

$$\bar{V}(k+1) = a_D(\bar{V}(k), A_{W_T}(k)) \quad (6.3.10)$$

$$= \bar{V}(k) + \Delta t (c(k, \bar{V}) + A_{W_T}(k)) \quad (6.3.11)$$

with

$$c(k, \bar{V}) = ge_{13}^{WI}(k) + A_{W_D}(k, \bar{V}) \quad (6.3.12)$$

Discretising the system's cost function of equation (6.3.5) yields,

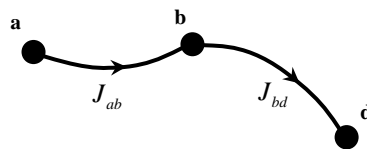
$$J = \sum_{k=0}^{N-1} g_D(\bar{V}(k), A_{W_T}(k)) \quad (6.3.13)$$

$$= \sum_{k=0}^{N-1} \left[ (\bar{V}(k) - \bar{V}_{Trim})^6 + k_C (A_{W_T}(k) - A_{W_T}^{nom}(k, \bar{V}))^2 \right] \quad (6.3.14)$$

with

$$A_{W_T}^{nom}(k, \bar{V}) = - \left[ ge_{13}^{WI}(k) + A_{W_D}(k, \bar{V}) \right] \quad (6.3.15)$$

Starting at the final time step when  $k = N$ , the optimal control input to the system can be solved for backwards in time for every allowable combination of state values. The *principle of optimality* is used to accomplish this. To understand this principle consider figure 6.2 representing a multistage decision process.



**Figure 6.2:** Optimal Path in a Multistage Decision Process

The principle states that if it is known that  $a-b-d$  is the optimal path to get from  $a$  to  $d$ , then  $b-d$  is the optimal path from  $b$  to  $d$ . Therefore the minimum cost to proceed from  $a$  to  $d$  is given by,

$$J_{ad}^* = J_{ab} + J_{bd}^* \quad (6.3.16)$$

This principle can be applied to create an algorithm that generates a matrix of optimal control solutions for any given current state of the system over the entire range of time steps. As this system only consists of one state, namely velocity, the optimal control matrix will be an  $L \times M$  matrix where  $L$  is the total amount of time steps and  $M$  is the total discretised velocity intervals.

Using the principle of optimality, the following recurrence equation is obtained for the minimum cost at time step  $k - N$ ,

$$J_{N-k,N}^*(\bar{V}(N-k)) = \min \left\{ g_D(\bar{V}(N-k), A_{W_T}(N-k)) + J_{N-(k-1),N}^*(a_D(\bar{V}(N-k), A_{W_T}(N-k))) \right\} \quad (6.3.17)$$

with initial condition,

$$J_{N,N}^*(\bar{V}(N)) = 0 \quad (6.3.18)$$

This equation states that the optimal control to apply is the control where the cost generated at the current time step added to the minimum cost required to get from the resulting state value to the final time step is a minimum. By iteratively calculating the total cost generated by every allowable control input, the minimum cost solution can be found at every allowable velocity interval. By then solving the recurrence equation back from the final to the initial time step and storing the optimal control at every allowable state value and time interval, a matrix of optimal control solutions  $A_{W_T}^*(k, \bar{V})$  is obtained. This method of finding the optimal control is a great improvement to direct enumeration where an exhaustive search for the optimal trajectory is carried out forwards in time.

**Interpolation** When calculating the resulting state value, it might not lead to an exact discretised state interval. In order to obtain the resulting cost  $J_{N-(k-1),N}^*$  linear interpolation is used between the costs of the upper and lower state intervals that the resulting state value lies within.

**Dimensionality** Adding any extra states to the system, such as ASA lag, introduces an additional dimension to the iterative solution and thereby greatly adds to the computational burden of finding a solution. This method of optimal control can therefore only be used for very simple systems such as the single state single input system used here.

**Trajectory Discretisation** This algorithm is calculated at discrete time intervals, therefore the position based trajectory has to be divided into time based segments. Because

the velocity over the trajectory is not yet known, it has to be assumed that the trajectory is flown at the aircraft's trim velocity. As the goal of this algorithm is to regulate the aircraft's velocity to trim, this assumption will not introduce significant inaccuracies. It should be noted that this assumption is only used to discretise the trajectory. The calculation of the aerodynamic drag term  $A_{W_D}(k, \bar{V})$  still uses the correct state value for  $\bar{V}$ .

### 6.3.3 Implementation

Using the dynamic programming algorithm, the optimal ASA due to throttle perturbation ( $A_{W_T}^*$ ) can be calculated. The inner loop ASA controller is able to regulate the aircraft's total ASA to a commanded value. Therefore by adding the ASA due to drag term  $A_{W_D}$ , which gets calculated by the dynamic programming algorithm, to the optimal  $A_{W_T}^*$  result, the inner loop ASA command as a function of time steps and velocity is obtained as follows,

$$A_{W_C}(k, \bar{V}) = A_{W_T}^*(k + 6, \bar{V}) + A_{W_D}(k - 3, \bar{V}) \quad (6.3.19)$$

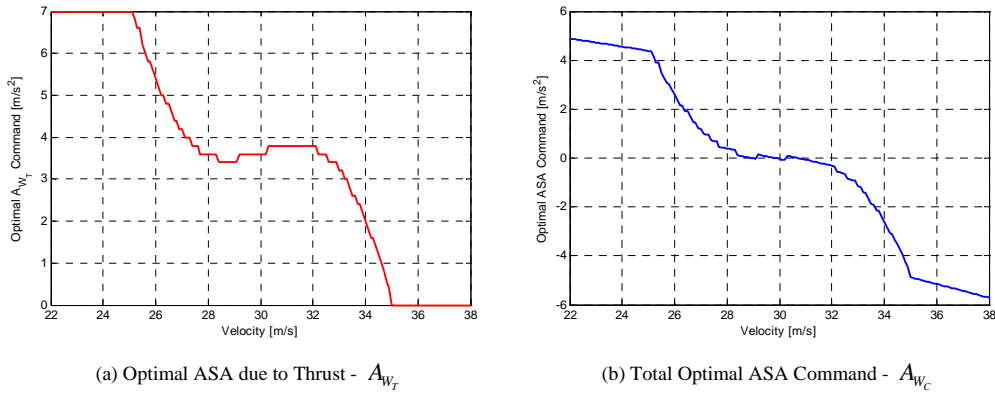
To achieve the most accurate result, the ASA due to thrust calculated for 0.6 s in the future and the ASA due to drag pertaining to 0.3 s in the past is used. The reasoning behind this is that an unmodelled 0.6 s time lag exists on the ASA command and that the average step response time of the NSA controller (which affects the dynamic drag) is 0.3 s.

This look-up table of commands can be used as a reference to the ASA controller by using the current velocity and time step information to extract the correct command. This combined control architecture will then be responsible for regulating the aircraft's velocity over any given trajectory. As ASA lag has not been included in the model, the controller's stability has to be analysed.

This controller should be able to regulate the aircraft's velocity back to a predetermined optimal value as well as provide the pre-emptive velocity required to execute aggressive 3D flight manoeuvres. The latter point is achieved by weighing velocity perturbations from trim at a sixth order level in the cost function of equation (6.3.5). However, without also weighing control perturbations, a bang-bang optimal control solution is obtained and the ability for the controller to smoothly regulate the aircraft's velocity is not achieved. By setting the control weighting gain in the cost function of equation (6.3.5) to,

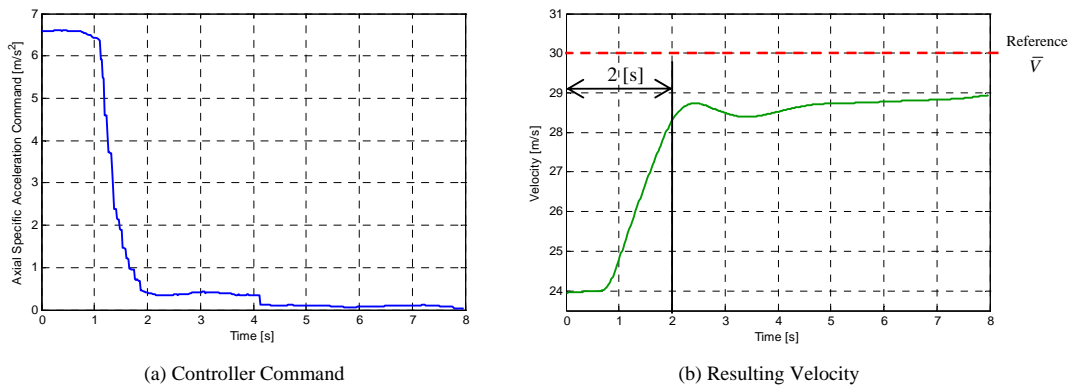
$$k_C = 500 \quad (6.3.20)$$

an interesting non-linear velocity control law is obtained. Figure 6.3(a) shows the resulting optimal  $A_{W_T}^*$  commands for a velocity range about the trim of 30 m/s. This result is for a level flight trajectory and therefore the controller will attempt to regulate the velocity to trim at all times. Figure 6.3(b) shows the resulting ASA command over this velocity range. It can be seen that for small velocity errors the controller will respond very slowly. However, for larger errors a considerable amount of ASA will be commanded.



**Figure 6.3:** Velocity Regulation Control Law Obtained From the Optimal Control Algorithm

Figure 6.4 shows a simulated 6 m/s velocity error. It can be seen that this leads to an aggressive correction of this error through a large ASA command. However, as soon as the velocity enters within a range of 1.5 m/s of the desired value, the controller command becomes very small. This can be deemed an acceptable velocity regulation result as large velocity errors will cause effects such as stall, but very small errors will be of no consequence to guidance control.



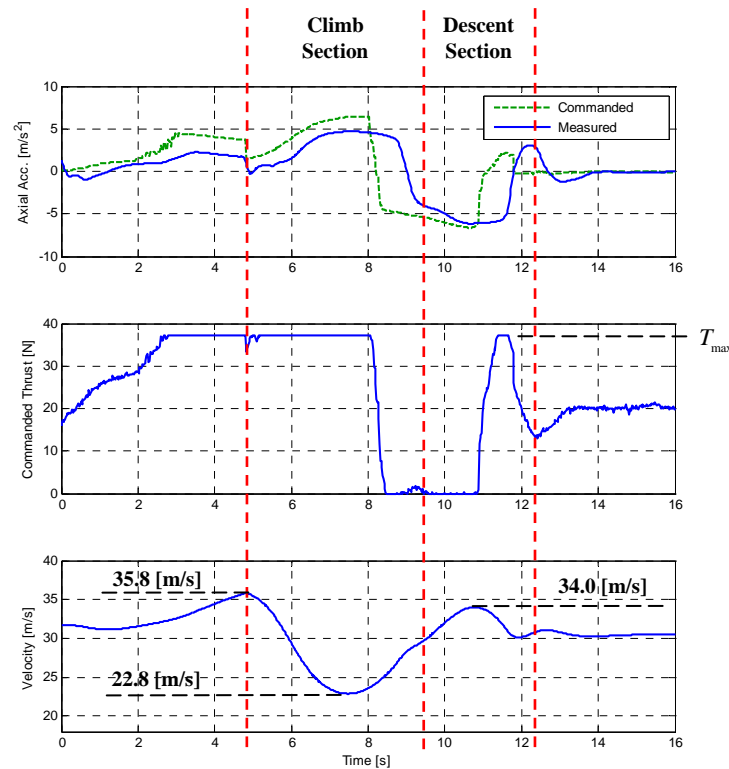
**Figure 6.4:** Error Rejection of the Optimal Velocity Regulation Control Algorithm

**Computational Efficiency** The dynamic programming algorithm calculates the optimal control result over a large time range, in this case 20 s. In order to preserve its predictive nature, the first 10 s of this result is used before the optimal control matrix is recalculated. The advantage of dynamic programming is that none of the aircraft's current states are required to calculate the optimal control solution and therefore those 10 s can be used to calculate the next optimal control matrix. This makes the algorithm practically feasible on an avionics package with some computational power available to it such as the OBC used in this project – see Appendix E.

### 6.3.4 Simulation Results

As previously stated, the vertical loop trajectory is an ideal test for the control algorithms developed in this chapter. It shows the controller's ability to both avoid stalling during the loop's vertical climb section as well as prevent the aircraft's velocity from becoming excessively large during the descent.

Figure 6.5 shows the simulated flight of the loop trajectory using the optimal control algorithm to command the ASA controller and thereby regulate the aircraft's velocity. It can be seen that a pre-emptive ASA command is generated by the algorithm more than 2 seconds before the start of the loop. This leads to the aircraft's velocity increasing to a maximum of 35.8 m/s and thereby providing the kinetic energy required by the aircraft to climb. Almost a second before the aircraft reaches the top of the loop this command is decreased to its minimum value in order to prevent the aircraft's velocity from becoming too large on the descent. At the top the aircraft reaches a minimum velocity of 22.8 m/s which is more than 5 m/s above the approximate stall speed. This result can be compared with figure 6.1 to observe the significant improvement provided by the optimal velocity controller. This simulation proves the functionality and success of this algorithm with the final result greatly resembling the predictive control inputs that an actual pilot would provide.



**Figure 6.5:** Simulation of the Loop Trajectory using the Optimal Velocity Regulation Algorithm

## 6.4 Kinetic Energy Analysis

Kinetic energy analysis is provided as a more computationally efficient alternative to the optimal control algorithm of the previous section. Simply put, this algorithm looks forward in time over the trajectory and searches for scenarios when the amount of kinetic energy loss or gain will be much more than the maximum or minimum throttle can account for. Based on the amount of excess or deficit in kinetic energy, logic switching is then used to command the velocity controller shown in section 5.2.3. This will add or remove kinetic energy from the system in order to counter the large loss or gain that is about to occur.

### 6.4.1 Energy Calculation

In order to divide the position based trajectory into time based intervals, it is once again assumed that the trajectory is flown at trim velocity. As stated before, this will not cause significant errors in the calculations. Two of these intervals are shown in figure 6.6, where the distance between the two points,  $d$ , and the change in altitude,  $\Delta h$ , can be obtained from the reference trajectory.

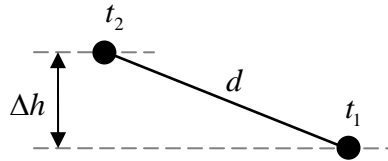


Figure 6.6: Trajectory Discretisation Intervals

Because these intervals are relatively close to each other, it is assumed that the aircraft travels in a straight line between them. As shown in section 6.2, the amount of ASA caused by the drag can be calculated at any point on the trajectory, given the aircraft's velocity at that point. By then again assuming that the aircraft is travelling at its trim velocity over the entire trajectory, the energy change between two points due to the drag is given by,

$$E_D = mA_{W_D}d \quad (6.4.1)$$

Because of this trim velocity assumption, the drag calculations will not be as accurate as the results obtained by the optimal control algorithm. They will however be accurate enough to predict a large loss or gain in kinetic energy. The second energy calculation required is for the amount of gravitational potential energy lost due to a change in altitude. This is given by,

$$E_{\Delta h} = mg(\Delta h) \quad (6.4.2)$$



The aircraft's throttle is also able to add energy to the system. The maximum and minimum energy that can be added is given by,

$$E_T^{\min} = mA_{W_T}^{\min}d \quad (6.4.3)$$

and

$$E_T^{\max} = mA_{W_T}^{\max}d \quad (6.4.4)$$

with equations (6.2.10) and (6.2.10) providing the calculated values for  $A_{W_T}^{\min}$  and  $A_{W_T}^{\max}$ .

The energy deficit or excess between each of the discretised points can be calculated as follows,

$$E_k = \begin{cases} E_D - E_{\Delta h} + E_T^{\min} & \text{if } (E_D - E_{\Delta h} + E_T^{\min} > 0) \\ E_D - E_{\Delta h} + E_T^{\max} & \text{if } (E_D - E_{\Delta h} + E_T^{\max} < 0) \\ 0 & \text{if neither are true} \end{cases} \quad (6.4.5)$$

The total energy loss or gain that the aircraft's throttle is not able to counter for can be obtained over a given prediction horizon,  $N$ , by adding the deficit or excess energy between each of the discretised points.

$$E_{total} = \sum_{k=0}^N E_k \quad (6.4.6)$$

## 6.4.2 Logic Switching

This energy total can be used in various ways to add or remove energy from the system in a pre-emptive manner and thereby keep the aircraft's velocity as close to trim as possible. The following method of logic switching is one possible solution and is very well suited for the aerobatic trajectories flown by the aggressive flight autopilot developed in this thesis. This does not provide a general optimal solution like the optimal control algorithm developed in the previous section, but rather a computationally efficient, but sub-optimal alternative, that will be sufficient in most cases.

The flow diagram in figure 6.7 illustrates the logic used to command the velocity controller developed in section 5.2.3. The goal of this control logic is to prevent the aircraft's velocity from reaching a point that is very far from its trim value. The algorithm is run at every sample instance and is able to switch between three modes, each providing a different velocity command. The energy excess and deficit modes respectively provide very small and very large velocity commands in order to saturate the aircraft's throttle and thereby quickly remove or add kinetic energy to the system. In contrast to this, the normal mode will command the aircraft's trim velocity and thereby allow the linear controller to regulate the velocity.

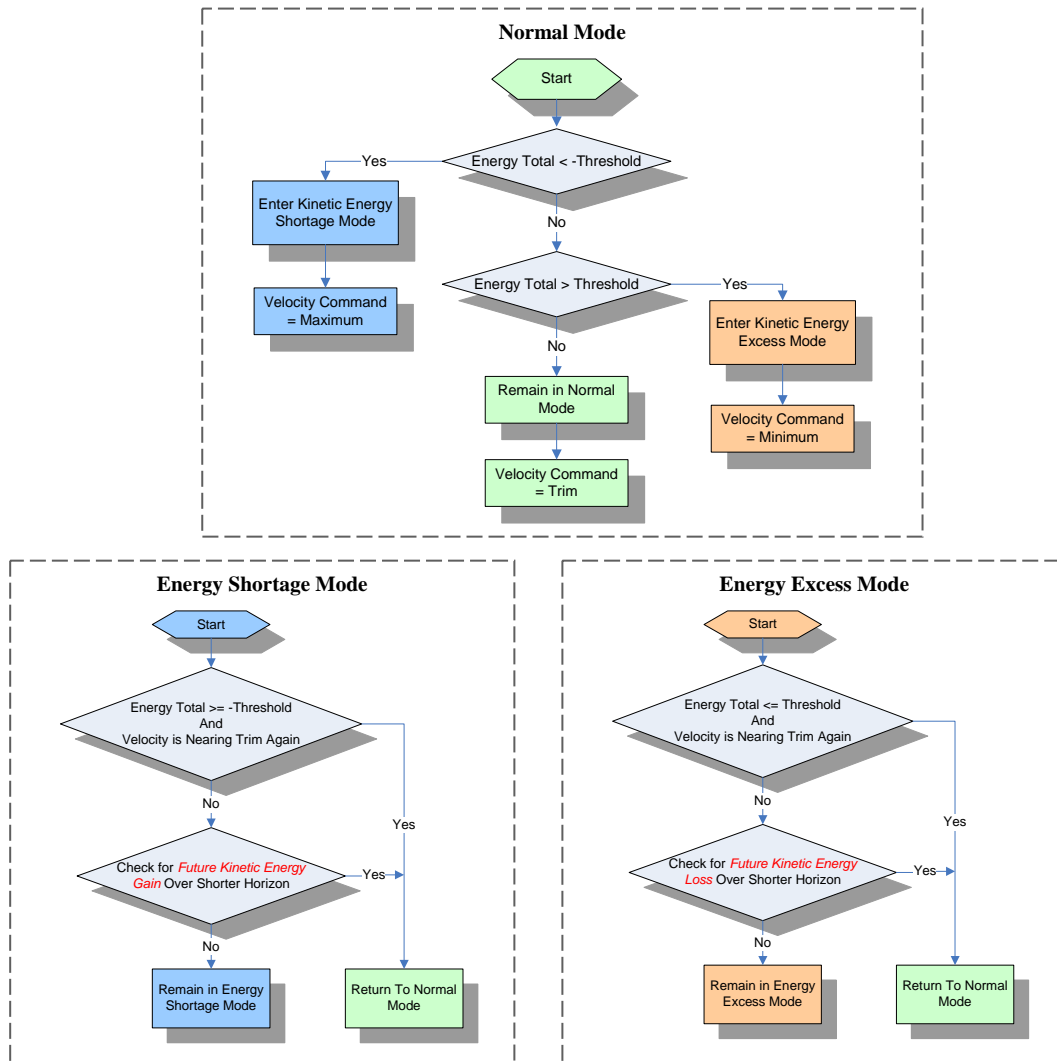


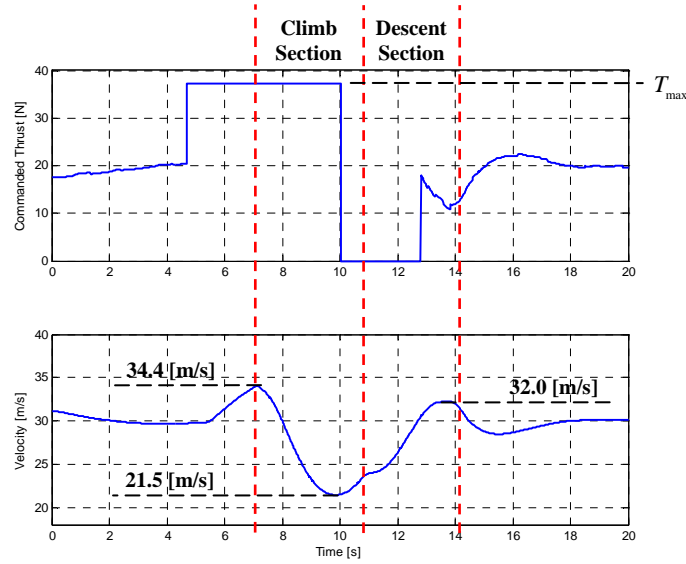
Figure 6.7: Logic Switching used to Command the Velocity Controller

### 6.4.3 Simulation Results

As before, the vertical loop trajectory is used to test the algorithm as it provides a relatively extreme test case for the velocity regulation problem. Figure 6.8 shows the velocity and commanded thrust of the aircraft when using the kinetic energy algorithm. It can be seen how the algorithm enters its *energy shortage mode* more than 2 seconds before the loop starts and thereby commands the throttle to its maximum setting. The aircraft's velocity is therefore increased causing the minimum velocity achieved during this trajectory to be 21.5 m/s, which is well within the minimum bound. It can also be seen that because the throttle is cut just before the top of the loop, the increase in velocity during the descent section never exceeds 32 m/s. A small transient can be observed at the end of the trajectory as the algorithm exits its *energy excess mode* and allows the linear velocity

controller to regulate the aircraft's velocity back to trim once again.

Even though this algorithm does not yield the optimal control result derived in the previous section, it still manages to successfully regulate the velocity to within acceptable bounds and can be used in most scenarios.



**Figure 6.8:** Simulation of the Loop Trajectory using the Kinetic Energy Analysis Algorithm

## 6.5 Summary

In conclusion, it can be seen that both the algorithms developed in this chapter are able to provide the required predictive velocity regulation. In order to decide which one should be implemented for a specific application, the results obtained with the kinetic energy algorithm should first be investigated as it provides the simplest and most computationally efficient solution. If these results prove to be inadequate, then the general optimal solution provided by the optimal control algorithm should be used.

## Chapter 7

# Comparison and Conclusion

In this project two aggressive all-attitude flight control architectures were investigated and implemented. These control systems were tested both in simulation as well as through practical flight tests. The detailed design process was shown for each of these control systems, highlighting the theoretical approach as well as the practical pole placement and implementation choices that were made. Various supporting tools such as *position based trajectories* and *predictive velocity regulation algorithms* were also developed for these control systems.

The end goal of this project is to evaluate both of these control architectures and determine the most efficient and successful control system that can be used for all-attitude aggressive flight control. In this chapter a brief summary of the control architectures investigated in this thesis will be provided. These control systems will be compared with each other as well as with other aggressive flight control strategies, such as the one developed by [15]. In conclusion the most successful aggressive flight control architecture will be selected based on the results of the control system comparison. Recommendations for future improvements and further research will then be made.

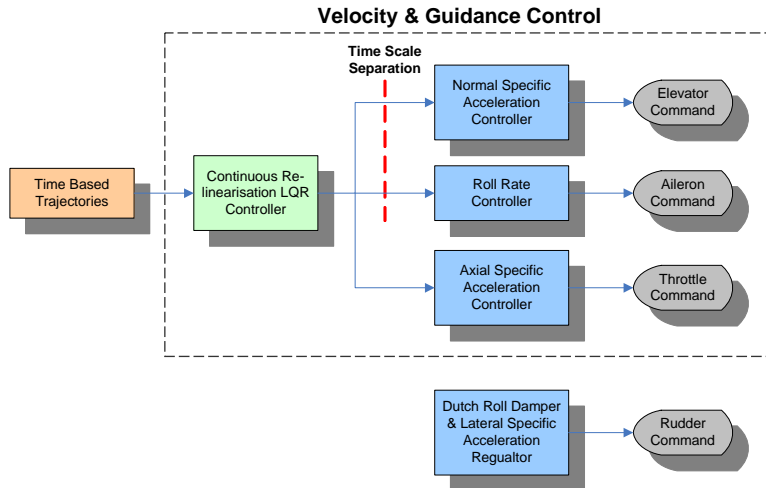
### 7.1 Control System Summaries

#### 7.1.1 Continuous Re-Linearisation LQR Outer Loop Control System

The first control architecture investigated in this thesis was the continuous re-linearisation LQR control system. For this control strategy the aircraft model was split into its fast inner loop and slower outer loop dynamics. Linear inner loop controllers were designed, as shown in Chapter 3, enabling the control of the aircraft's axial and normal specific acceleration as well as its roll rate.

For outer loop kinematic control, the strategy was to linearise the non-linear point mass dynamics of the outer loop system at every sample instance about the aircraft's current position on a time based reference trajectory. An LQR algorithm was then used

to place optimal and stable closed loop poles. The outer loop control system was then allowed to command the aircraft's inner loop ASA, NSA and roll rate controllers. The fast poles of the NSA and roll rate controllers were ignored by this controller through the principle of TSS. In this control architecture the velocity regulation and guidance problems were combined. The single LQR control system was used to regulate the aircraft's outer loop dynamics and thereby enable the aircraft to follow a pre-defined time based reference trajectory. Figure 7.1 provides an overview of this architecture.



**Figure 7.1:** Continuous Re-Linearisation LQR Control System Architecture

### 7.1.2 SAM Control System

The second control system investigated in this thesis was the specific acceleration matching controller proposed by [1]. It was found that this control system could not be implemented practically in the three dimensional fashion as was proposed – the bandwidth of the aircraft's throttle was not fast enough for this. This architecture was therefore altered to provide a two-dimensional guidance solution and thereby split the aircraft's velocity regulation and guidance problems.

Aircraft guidance was achieved by using the outer loop SAM guidance controller to command the inner loop NSA controller as well as an augmented roll rate controller that is able to achieve any commanded NSA vector direction. The guidance and NSA vector direction controllers were time scale decoupled from each other and thereby their dynamics became linear and simple controllers could be designed for each.

Trajectory guidance was therefore controlled independently of velocity and a separate control system was designed for velocity regulation. This architecture was ideally suited to the use of position based trajectories which provide a mathematical description of the intended flight path without specifying the velocity with which it should be traversed.

Successful regulation of the aircraft's velocity over an aggressive flight envelope with the limited dynamic range available to the aircraft's throttle was achieved using one of two proposed methods. Both of these methods were based on the idea that the velocity regulator should provide error correction when the velocity is not at the desired value, as well as predictively command velocities that will ensure the future preservation of velocity over an aggressive flight trajectory. The first method used an optimal control algorithm to generate a command for the aircraft's ASA controller. The second method commands a velocity controller through the analysis of potential future gains or losses in kinetic energy. Figure 7.2 provides an overview of the SAM control architecture.

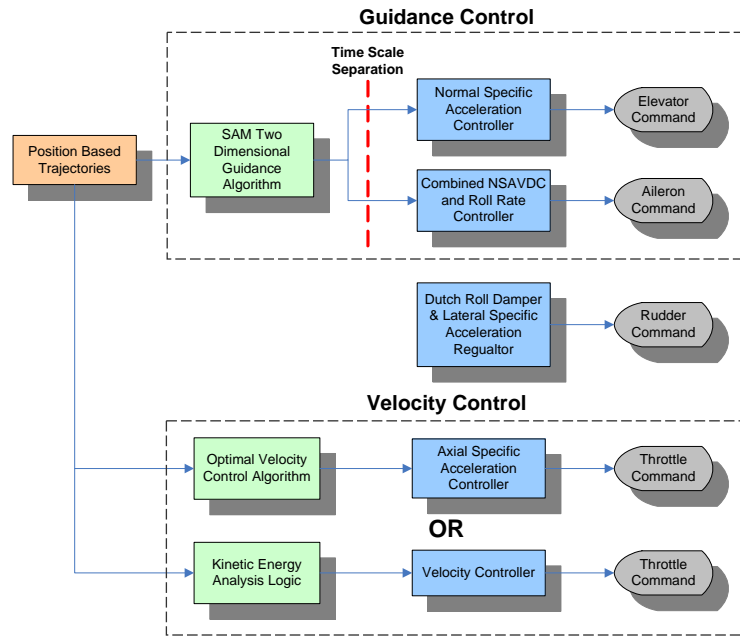


Figure 7.2: SAM Control System Architecture

## 7.2 Control System Comparison and Analysis

The two control architectures proposed in this thesis can now be compared to each other as well as with other aggressive flight controllers such as the *complete aircraft model LQR control architecture* developed by [15]. Many factors can be considered to determine the degree of success of each of these control systems. Therefore this comparison will investigate each of these controllers based on the following criteria:

- Assumptions Used
- Pole Placement Ability and Performance

- Types of Reference Trajectories
- Stability Concerns
- Computational Efficiency

### 7.2.1 Assumptions Used

In the design of both the SAM and the LQR control architectures used in this project, certain assumptions had to be made. These assumptions greatly simplified the control problem and aided in the creation of various independent controllers working together to achieve successful flight control. These assumptions include the decoupling of the inner loop model to create separate inner loop control systems as well as the time scale decoupling of these controllers to an outer loop kinematic control architecture.

For the SAM controller the time scale separation (TSS) exists between the outer loop guidance controller and the inner loop NSA and NSA vector direction controllers. The LQR outer loop control architecture uses similar assumptions with the TSS existing between the outer loop controller and the inner loop NSA and roll rate controllers.

In comparison to this, the LQR control of the full aircraft model employed by [15] uses no decoupling or TSS assumptions. Therefore the full coupled dynamic model of the aircraft is taken into account for the control system design. Some of the coupling in the aircraft model is however so negligibly small that it can not be used practically for control purposes. The complexity of this coupled system is however so great that some design insight into the performance of the system will be lost.

An additional advantage of splitting the aircraft model is that the high bandwidth inner loop controllers desensitise the system to model uncertainty. As long as it is ensured that the TSS assumption is valid, the outer loop pole placement will be relatively accurate. With the full LQR control system proposed by [15] the resulting pole placement of the algorithm can be practically very different due to inaccuracies in the aircraft model. With large modelling inaccuracies, unexpected results such as instability could even occur.

### 7.2.2 Pole Placement Ability and Performance

The ability to specify the performance of various aspects of the control system is paramount in creating a successful and efficient aggressive flight controller. The abilities of the various control architectures to place closed loop poles should therefore be compared and the resulting performance should be investigated.

The inner loop controllers used by the LQR control architecture have all been simplified to linear control architectures with full pole placement available to them. The performance of the aircraft's specific accelerations and roll rate can therefore be specified. The outer loop LQR control system does however not provide the ability for direct

pole placement. The control and state weightings can be set and the resulting pole placement for a specific reference trajectory condition can be investigated. This provides some insight into the system's closed loop dynamics, but the resulting pole placement varies with changes in the reference trajectory conditions. Therefore some design insight is lost with this outer loop control strategy and the TSS constraint on the outer loop pole placement can not be mathematically enforced.

The SAM control architecture further simplifies the control problem and enables the design of linear guidance and NSA vector direction controllers with full pole placement available to them. Critically damped guidance poles can easily be placed and the accuracy of this placement will mostly depend on the time scale separation between the guidance and NSA vector direction controllers. This TSS can easily be enforced and in this project a frequency separation of 4.67 times is achieved, leading to a well damped guidance response.

A key difference between these control architectures is that with the SAM control system the roll rate controller is augmented with an error angle pole. By placing this pole at a rather high frequency it is both time scale decoupled from the guidance controller and provides a fast NSA vector direction response. This response assists the aircraft to roll very quickly and thereby aggressively correct trajectory tracking errors. In comparison, a roll angle pole placed by the LQR outer loop controller will have to be much slower as it commands the time scale decoupled inner loop roll rate controller. This creates a less aggressive roll angle response and therefore certain trajectory errors are corrected less aggressively. This key difference therefore allows the SAM control system to better correct any trajectory tracking errors.

### 7.2.3 Types of Reference Trajectories

Two types of reference trajectories were investigated in this project, namely time and position based trajectories. Time based trajectories provide the aircraft's position, velocity, attitude and possibly some additional states at discretised time intervals. The LQR outer loop controller used in this project as well as the full aircraft model LQR architecture developed by [15] uses this type of trajectory reference. By using time based trajectories with a single outer loop controller, the aircraft's guidance and velocity control is treated as is single problem and regulated together. Velocity regulation of the aircraft onto a trajectory can potentially degrade trajectory tracking performance. The vertical loop trajectory in figure 4.22 carried out by the LQR outer loop control architecture is an excellent example of this, where it can be seen how the control system creates an altitude error in order to increase the aircraft's velocity.

An advantage of the time based trajectories used in this project over those employed by [15] is that they are only required to describe the better modelled outer loop kinematic states of the aircraft over time. Since [15] used an LQR control strategy for the entire



aircraft model, the reference trajectories have to be defined in terms of all the states of the aircraft. Reference trajectories such as these can prove to be inaccurate as they are susceptible to aircraft model inaccuracies.

With the SAM control architecture an improvement on the time based trajectories was then provided with the design and implementation of position based trajectories. These trajectories describe the aircraft's path through space mathematically without specifying the velocity with which it is should be flown. This architecture assists in separating the aircraft's guidance and velocity control problems. The guidance controller is responsible for guiding the aircraft onto this flight path while a separate velocity control system is responsible for regulating the aircraft's velocity. The trajectory references are generated on-line using the aircraft's current velocity to create more accurate reference commands and thereby improve tracking accuracy. These trajectories are also relatively easy to create as only a few well modelled kinematic variables are required by the control architecture.

#### 7.2.4 Attitude Representation

Attitude representation is approached differently for each of the control architectures that were investigated.

The full aircraft model LQR architecture developed by [15] uses Quaternions for its attitude dynamics. As shown in section 4.2.1, this could lead to inconsistent control being applied over the entire attitude range. In certain circumstances it can also cause the matrices used by the iterative LQR algorithm to become badly scaled and prevent a steady state solution from being found.

In the LQR outer loop control strategy used in this project, the decision was made to create two separate sets of dynamics that each use two different Euler angle sequences for attitude description. As Euler angles provide a more linear method of attitude description, more consistent control was achieved. The singularities in the Euler attitude dynamics were avoided by switching between the two different sets of dynamics when the aircraft's current attitude neared an Euler singularity.

Since the SAM control architecture further simplified the outer loop dynamics to a model based solely on the dynamics of the aircraft's wind axes velocity and position vectors, it avoided the need to switch between different sets of dynamics.

#### 7.2.5 Stability Concerns

Stability is a critical factor for any control system. Therefore any factors that could lead to instability in the control architecture should be investigated.

Since the LQR outer loop dynamics are linearised about the current position on a reference trajectory, the linearisation assumption becomes less accurate the more the aircraft's actual states differ from those of the reference trajectory. If this difference becomes too large, instabilities could present themselves in the control system. Therefore should

a very large wind gust severely change the aircraft's attitude, the control system could potentially become unstable.

Another stability concern that applies to both of the control architectures investigated in this project is that of the time scale decoupling assumption between the outer and inner loop controllers. If for some reason an inner loop controller's performance is degraded to such an extent where its dynamics is not significantly faster than the outer loop dynamics that commands it, the assumption that the inner loop command can be immediately attained becomes invalid. Since the outer loop controller does not take the inner loop dynamics into account, the resulting control system can become unstable. This could occur practically if severe modelling inaccuracies resulted in the inner loop poles being much slower than expected. Therefore the performance of the inner loop controllers have to be confirmed through practical flight tests (as done in section 3.9) before allowing the outer loop controllers to command them.

### 7.2.6 Computational Efficiency

The computational efficiency of a control system determines its ability to be implemented on various platforms. If a significant amount of processing power is required, the implementation of the control system on small embedded microcontrollers will not be possible. This will necessitate the use of larger and more power consuming avionics packages.

When an LQR algorithm is employed to control a non-linear system by linearising its dynamics at every sample instance, the control gains constantly have to be recalculated. The solution to these control gains are provided by iteratively solving the Riccati difference equation backwards in time until the gains stabilise to their steady state values. Every iteration requires the inversion of a matrix with the same dimension as the number of controls in the system [9]. It has been shown in section 4.3.4 that the amount of iterations required for the LQR gains to stabilise are directly related to the selected sample time. This is due to the gain convergence being linked to the system's dynamic response which takes a certain amount of real time to stabilise. A larger sample time will therefore lead to less iterations being required for the gains to stabilise.

If an LQR control architecture is used to control the aircraft's entire non-linear model as done by [15], the sample period of this control system will have to be selected relatively short, e.g. 20ms. This is due to the high bandwidth inner loop dynamics contained in the model, requiring a fast sample frequency (50 Hz in this case) for smooth digital control. Four control inputs are available to the full aircraft model, namely the three aerodynamic control surfaces and the aircraft's throttle. This implies that a significant amount of  $4 \times 4$  matrix inversions have to be carried out every 20 ms.

In the LQR outer loop control architecture used in this project, the linear inner loop controllers require very little computational power to execute. Only the LQR algorithm used to control the non-linear outer loop dynamics will play a role in the control system's

computational efficiency. A longer sample time can be selected for this controller since its model exclusively describes the aircraft's slower dynamics. It was shown in section 4.3.4 how a sample frequency as low as 5 Hz could still be considered acceptable. Therefore this will reduce the amount of Riccati iterations required for the gains to converge by a factor 10. Furthermore, this will also provide 10 times more real time for the processor to compute the steady state LQR gains.

The controls available to the LQR outer loop system are the aircraft's ASA, NSA and roll rate. Therefore the matrix to be inverted at every iteration of the algorithm will be a  $3 \times 3$  matrix which reduces the amount of computational power required even further. By using the computationally efficient method of LU decomposition to solve the matrix inversion, the approximate amount of floating point calculations required to complete an inversion is given by  $4/3n^3$ , where  $n$  is the size of the square matrix [12]. This formula can be used to quantify the computational improvement of reducing the size of the matrix to be inverted from 4 to 3. This result gives an improvement of 2.37 times. The total computational improvement is the product of all the improvements attained by this control architecture and when calculated yields a result of 237 times. Therefore even though the method of LQR outer loop control used in this project still requires some computational power, it is potentially up to 237 times more efficient than using an LQR control strategy for the entire aircraft model.

The second control architecture investigated in this project was the SAM control system, which was discussed in Chapter 5. This kinematic control architecture further simplified the outer loop dynamics and enabled the design of linear kinematic controllers. In comparison to the LQR control system the computational power required by these linear controllers is almost negligible. These controllers will be relatively simple to implement on most embedded microcontrollers without overstressing their computational abilities. The SAM kinematic controller employed in this project was only used to guide the aircraft back onto its flight path and required a velocity regulation control system to operate in conjunction with it. Of the velocity regulation control architectures that were investigated, only the optimal control solution required a significant amount of computational power. Therefore if the method of kinetic energy analysis with logic switching is used, the entire control architecture will place very little computational strain on the platform it is implemented on.

### 7.2.7 Summary

The control architecture comparison discussed in this section can shortly be summarised in table 7.1 below.

	<b>SAM Control Architecture</b>	<b>LQR Outer Loop Control Architecture</b>	<b>Full Aircraft Model LQR Control [15]</b>
<b>Assumptions Used</b>	Inner loop model decoupling. Outer loop TSS to inner loop NSA and NSA vector direction controllers.	Inner loop model decoupling. Outer loop TSS to inner loop NSA and roll rate controllers.	None.
<b>Pole Placement Ability and Performance</b>	Full pole placement of inner loop aircraft-specific dynamics as well as outer loop guidance dynamics.	Full inner loop pole placement. Outer loop dynamics selected by choosing state and control weighting values.	Full aircraft model performance selected by choosing state and control weighting values.
<b>Types of Reference Trajectories</b>	Position based trajectories – Improves tracking accuracy.	Time based trajectories – More difficult to track the reference.	Time based trajectories – More difficult to track the reference.
<b>Attitude Representation</b>	Vectors are used in this model which avoid any singularities in attitude dynamics.	Two different Euler angle sequences. Controller switches between them to avoid singularities.	Quaternions lead to inconsistent control. In certain situations LQR gains does not converge.
<b>Stability Concerns</b>	TSS pole constraints.	TSS pole constraints. Linearisation of outer loop dynamics.	Linearisation of full aircraft dynamics.
<b>Computational Efficiency</b>	Excellent	Average	Bad

Table 7.1: Comparison Between Various Aggressive Flight Control Architectures

### 7.3 Conclusion

From the comparison provided in the previous section it can clearly be seen that the second control architecture using the two-dimensional SAM guidance controller, a separate predictive velocity regulation control system and position based reference trajectories provide the best results. This conclusion is based on the following observations:

- Through the assumptions made in the derivation of the SAM control architecture, linear controllers can be designed for both the inner loop aircraft-specific as well the outer loop kinematic systems. These controllers all have full pole placement of the aircraft's dynamics available to them and therefore provide excellent insight into the performance of the control system.
- Since all the controllers used in the SAM architecture are linear, very little computational power is required to implement them. This enables the control system to

be implemented on almost any platform.

- The position based trajectories that were developed for this control architecture provide improved trajectory tracking as the velocity regulation and aircraft guidance problems are decoupled from each other.
- The predictive velocity control algorithms developed for the position based trajectories provide an online method of ensuring that the aircraft's velocity remains within its required bounds as opposed to the offline feed forward provided by the time based trajectories.
- No linearisation assumptions are required by this control architecture and therefore less stability concerns present themselves at a guidance level.
- The need to switch between two outer loop models, due to Euler singularities, is eliminated through the use of vectors in this control architecture.

## 7.4 Recommendations

Much potential for future research still exists on the topic of aggressive flight control and navigation. The recommendations are divided into suggestions about improving estimator performance as well as further control system research.

### 7.4.1 Estimator Research

The following aspects can be investigated in order to improve the current kinematic state estimator. By improving the accuracy and performance of this estimator, any control system using these kinematic estimates will also perform much better.

- Research on an aircraft-specific estimator could improve aircraft state estimates. This estimator will use the aircraft's aerodynamic model in conjunction with its kinematic model to better estimate its kinematic states.
- Replacing the current u-Blox GPS receiver with a module capable of sending the raw pseudo range, carrier phase and Doppler data directly to the OBC will enable the development of an improved kinematic estimator. This estimator will use the raw GPS data in conjunction with the accelerometer, gyroscope and magnetometer data to create an improved GPS augmented inertial navigation system. The direct use of this data with a fast processor eliminates the approximate 0.31 s time delay introduced when the GPS chip processes the data and will yield significantly more accurate results. This estimator will also be able to better handle the large accelerations experienced during aggressive flight manoeuvres as most commercial GPS modules are designed with low g applications in mind.

- Replacing the current OBC used in this project with an alternative model will also improve GPS (and therefore kinematic estimator) performance. Tests on the current OBC using a spectrum analyser showed that it generates a significant amount of EM noise in the GPS L1 band (1.575 GHz). This noise interferes with GPS accuracy and lock and thereby degrades the measurements received from the module.

#### 7.4.2 Control Systems Research

The following suggestions for future work will build on the research done in this thesis and can further contribute to the field of aggressive flight control systems.

- Further research into the concept of a guidance algorithm operating independently of a velocity regulation controller by possibly attempting to introduce less simplification assumptions into the design process could yield an improved guidance control architecture. A smaller LQR controller that is able to command the aircraft's NSA and NSA vector direction could be investigated for this task. The lag introduced by the NSAVDC's error angle pole should be included in this model.
- The pitch rate damping NSA controller developed in section C.1 needs to be practically tested and its performance should be evaluated against the current NSA controller developed by [1].
- Additional research on the topic of optimal predictive velocity regulation by investigating less computationally demanding optimal control strategies could yield very useful results.
- The position based trajectories developed in this thesis can be used by an object avoidance algorithm to create online aggressively evasive manoeuvres and thereby create a UAV capable of navigating through a maze of obstacles such as buildings or other objects.
- Additional position based trajectories which also be developed that can be combined to create even more complex and spectacular aggressive flight manoeuvres.

# Appendices

## Appendix A

# Attitude Representation Methods

Various methods exist of describing the orientation of a rotating axis system relative to a fixed inertial reference frame, each with its own unique characteristics. In this appendix, each of the methods used in this project are discussed and conversions between these different attitude representations are shown.

### A.1 Attitude Representation Methods

The three methods used for attitude representation in this project are the direction cosine matrix, Euler angles and Quaternions.

#### A.1.1 Direction Cosine Matrix (DCM)

The direction cosine matrix is a  $3 \times 3$  matrix that can be used to describe the attitude of an axis system. The rows of the  $\mathbf{DCM}^{BA}$  matrix consist of the unit vectors of axis system B, coordinated in axis system A. Thereby the attitude of B relative to A is described. Using these unit vectors, the DCM can coordinate a vector from axis system A to B.

$$\mathbf{V}_B = \left[ \mathbf{DCM}^{BA} \right] \mathbf{V}_A \quad (\text{A.1.1})$$

Because the DCM consists of three orthogonal unit vectors, it is an orthogonal matrix which implies that its inverse is given by its transpose, as stated by [11]. The reverse transform from axis system B to A is therefore given by,

$$\mathbf{V}_A = \left[ \mathbf{DCM}^{BA} \right]^{-1} \mathbf{V}_B = \left[ \mathbf{DCM}^{BA} \right]^T \mathbf{V}_B \quad (\text{A.1.2})$$

As a method to represent attitude, the DCM provides a very complete representation but might not be the most elegant method, as it represents a system with 3 degrees of rotational freedom with 9 variables and therefore 6 constraints. For more information on DCM rotation matrices see sections B.5 and B.6 in Appendix B.



### A.1.2 Euler Angles

Euler angles provide a much more intuitive way of representing the attitude of an axis system by using three rotation angles with each angle representing a rotation around a specific unit vector of an axis system. To use Euler angles to transform axis system A to B, and thereby describe the attitude of B relative to A, the following sequence of rotations should be carried out in order [16], with the notation,  $\mathbf{u}_i^A$  representing the  $i^{\text{th}}$  (first, second or third) unit vector of axis system A.

1. Rotate A through the  $\psi$  angle about its  $\mathbf{u}_a^A$  unit vector to obtain axis system  $A_1$ .
2. Rotate  $A_1$  through the  $\theta$  angle about its  $\mathbf{u}_b^{A_1}$  unit vector to obtain axis system  $A_2$ .
3. Rotate  $A_2$  through the  $\phi$  angle about its  $\mathbf{u}_c^{A_2}$  unit vector to obtain axis system B.

The selected values for  $a$ ,  $b$  and  $c$  determine the rotational axes for each of the three rotations. By selecting different values for these variables, various Euler rotation sequences can be defined, with the only restriction being that two consecutive rotations can not be about the same unit vector [8]. The sequence most commonly used is Euler 3-2-1. Referring to the above axis system transformation, it can be seen that 3-2-1 will lead to a rotation sequence of yaw, pitch and then roll.

By analysing this axis system transformation method in more detail it can be seen that the direction of the next unit vector to be rotated about is determined in part by the previous rotation. If this is investigated more carefully, it will be found that each sequence of Euler angles has an inherent singularity at a certain attitude. This occurs when the second rotation aligns the first and third unit vectors with each other and therefore also aligns the first and third rotation angles. This implies that two of the rotational degrees of freedom have now become one and the same. The result is that an infinite amount of possible first and third rotation angles can now describe the same orientation. This scenario is known as gimbal lock and occurs at different attitudes for different Euler sequences.

### A.1.3 Quaternions

Quaternions provide another way of describing the attitude of an object. They use Euler's theorem of rotation [2] that states that the attitude of axis system B relative to A can be described by rotating A through a specified angle about a given vector. Consider the unit vector  $\mathbf{n}$  with rotation angle  $\varepsilon$  shown in figure A.1. The Quaternion parameters are defined as,

$$\mathbf{q} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} \cos(\varepsilon/2) \\ \sin(\varepsilon/2) \cdot n_1 \\ \sin(\varepsilon/2) \cdot n_2 \\ \sin(\varepsilon/2) \cdot n_3 \end{bmatrix}, \quad \text{with} \quad \mathbf{n} = \begin{bmatrix} n_1 \\ n_2 \\ n_3 \end{bmatrix} \quad (\text{A.1.3})$$

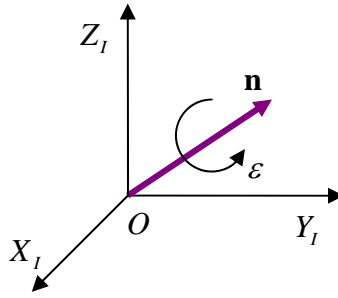


Figure A.1: Quaternion Definition

The advantage of using Quaternions to describe attitude is that they have no singularities and therefore work at any attitude. However, they describe attitude in a non-linear fashion. Therefore the same amount of variation in one Quaternion parameter can translate into a large or small change in attitude. The other disadvantage is that they describe a system with 3 degrees of freedom using four variables and therefore also have one constraint equation.

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1 \quad (\text{A.1.4})$$

## A.2 Attitude Conversions

This section provides various conversions between the attitude representation methods discussed in the previous section.

### A.2.1 Euler Angles to DCM

Using the single rotation matrices shown in section B.6, the DCM can be written in terms of any Euler sequence. It is listed below for the two Euler sequences used in this project.

Euler 3-2-1

$$\begin{aligned} \text{DCM}^{WI} &= \mathbf{T}^k(\psi)\mathbf{T}^j(\theta)\mathbf{T}^i(\phi) \\ &= \begin{bmatrix} \cos\psi \cos\theta & \sin\psi \cos\theta & -\sin\theta \\ \cos\psi \sin\theta \sin\phi - \sin\psi \cos\phi & \sin\psi \sin\theta \sin\phi + \cos\psi \cos\phi & \cos\theta \sin\phi \\ \cos\psi \sin\theta \cos\phi + \sin\psi \sin\phi & \sin\psi \sin\theta \cos\phi - \cos\psi \sin\phi & \cos\theta \cos\phi \end{bmatrix}_{(321)} \end{aligned} \quad (\text{A.2.1})$$

Euler 2-3-1

$$\begin{aligned} \mathbf{DCM}^{WI} &= \mathbf{T}^j(\psi) \mathbf{T}^k(\theta) \mathbf{T}^i(\phi) \\ &= \begin{bmatrix} \cos \psi \cos \theta & \sin \theta & -\cos \theta \sin \psi \\ \sin \psi \sin \phi - \sin \theta \cos \psi \cos \phi & \cos \theta \cos \phi & \sin \psi \sin \theta \cos \phi + \cos \psi \sin \phi \\ \cos \psi \sin \theta \sin \phi + \sin \psi \cos \phi & -\cos \theta \sin \phi & \cos \psi \cos \phi - \sin \psi \sin \theta \sin \phi \end{bmatrix}_{(231)} \end{aligned} \quad (\text{A.2.2})$$

### A.2.2 Quaternions to DCM

It is shown by [2] that the DCM can also be written in terms of Quaternions as follows,

$$\mathbf{DCM}^{WI} = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 + q_0q_3) & 2(q_1q_3 - q_0q_2) \\ 2(q_1q_2 - q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 + q_0q_1) \\ 2(q_1q_3 + q_0q_2) & 2(q_2q_3 - q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \quad (\text{A.2.3})$$

### A.2.3 DCM to Euler Angles

The reverse conversions from the DCM to the Euler angle sequences can be obtained by investigating the definitions of each of the Euler angles. A right angled triangle can be created with elements of the DCM unit vectors that contain the specified Euler angle. Inverse trigonometric functions can then be used to calculate the angle. For the Euler 3-2-1 and 2-3-1 sequences used in this project the conversions are given by,

Euler 3-2-1

$$\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}_{(321)} = \begin{bmatrix} \arctan \left( \frac{e_{23}^{WI}}{e_{33}^{WI}} \right) \\ -\arcsin \left( e_{13}^{WI} \right) \\ \arctan \left( \frac{e_{12}^{WI}}{e_{11}^{WI}} \right) \end{bmatrix} \quad (\text{A.2.4})$$

Euler 2-3-1

$$\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}_{(231)} = \begin{bmatrix} \arctan \left( -\frac{e_{32}^{WI}}{e_{22}^{WI}} \right) \\ \arcsin \left( e_{12}^{WI} \right) \\ \arctan \left( -\frac{e_{13}^{WI}}{e_{11}^{WI}} \right) \end{bmatrix} \quad (\text{A.2.5})$$

where  $e_{xy}^{WI}$  refers to the element of the  $\mathbf{DCM}^{WI}$  matrix located in row  $x$  and column  $y$ .

## Appendix B

# Mathematical Principles and Equations

### B.1 Dot and Cross Products

**Dot Product** The dot product of two vectors provides the projection of one vector onto the other and is defined as,

$$\mathbf{J} \cdot \mathbf{K} = |\mathbf{J}| |\mathbf{K}| \cos \theta \quad (\text{B.1.1})$$

with  $\theta$  defined as the angle between the two vectors.

**Cross Product** The cross product of two vectors is defined as,

$$\mathbf{J} \times \mathbf{K} = [|\mathbf{J}| |\mathbf{K}| \sin \theta] \mathbf{u}_{JK} \quad (\text{B.1.2})$$

where  $\theta$  is defined as the angle between the two vectors and  $\mathbf{u}_{JK}$  is a unit vector perpendicular to both  $\mathbf{J}$  and  $\mathbf{K}$ . The direction of the unit vector is determined by the right hand rule when  $\mathbf{J}$  is rotated to  $\mathbf{K}$  through the angle  $\theta$ .

### B.2 Cross Product Transformation Matrix

When two vectors  $\mathbf{J}$  and  $\mathbf{K}$  are coordinated in the same axis system  $A$ , their cross product can be written in terms of their coordinate vectors as follows,

$$\mathbf{J}_A \times \mathbf{K}_A = \mathbf{S}_{JA} \mathbf{K}_A \quad (\text{B.2.1})$$

The  $\mathbf{S}_{J_A}$  matrix is defined as,

$$\mathbf{S}_{J_A} = \begin{bmatrix} 0 & -Z_A & Y_A \\ Z_A & 0 & -X_A \\ -Y_A & X_A & 0 \end{bmatrix} \quad (\text{B.2.2})$$

with  $X_A$ ,  $Y_A$  and  $Z_A$  the components of  $\mathbf{J}$  coordinated in axis system A.

### B.3 Derivative of a Vector in a Rotating Reference Frame

When two axis systems exist where one is able to rotate relative to the other, the time derivative of a vector differs with respect to each of these axis systems. The equation of Coriolis provides the relationship necessary to relate the time derivative of a vector from one of these axis systems to the other. In words this relationship is stated as follows: An object's motion, as viewed from an axis system A, is equal to its motion as viewed from axis system B, which can rotate relative to A, plus the resulting motion caused by the angular velocity of the rotation of B relative to A. Mathematically this can be expressed as follows [3],

$$\left. \frac{d}{dt} \mathbf{R} \right|_A = \left. \frac{d}{dt} \mathbf{R} \right|_B + \boldsymbol{\omega}^{BA} \times \mathbf{R} \quad (\text{B.3.1})$$

### B.4 Moment of Inertia

An object's moment of inertia describes how difficult it is for the object to change its angular momentum and can be seen as a rotational analog to mass. When an object's moment of inertia is referenced to its centre of mass and coordinated in its body axis system, it is given by the following integral over the volume,  $V$ , occupied by the object [2],

$$\mathbf{I}_B = \int_V \begin{bmatrix} y^2 + z^2 & -xy & -xz \\ -xy & x^2 + z^2 & -yz \\ -xz & -yz & x^2 + y^2 \end{bmatrix} dm \quad (\text{B.4.1})$$

where  $x$ ,  $y$  and  $z$  are the distances of a small mass element,  $dm$ , to the object's centre of mass. Because the mass distribution of an object does not change when viewed from its body axis system, this integral simplifies to,

$$\mathbf{I}_B = \begin{bmatrix} I_x & -I_{xy} & -I_{xz} \\ -I_{xy} & I_y & -I_{yz} \\ -I_{xz} & -I_{yz} & I_z \end{bmatrix} \quad (\text{B.4.2})$$

Due to the symmetry of an aircraft the  $I_{xy}$  and  $I_{yz}$  cross products of inertia will be zero. The  $I_{xz}$  cross product, will not be zero however, but for most aircraft it will be relatively

small and can also be ignored. The moment of inertia tensor for most conventional aircraft is therefore given by,

$$\mathbf{I}_B = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \quad (\text{B.4.3})$$

## B.5 Rotation Matrices

When given a vector  $\mathbf{V}_A$  coordinated in axis system A, it is often useful to relate this to the same vector coordinated in another axis system, B. A rotation matrix or direction cosine matrix provides this relationship.

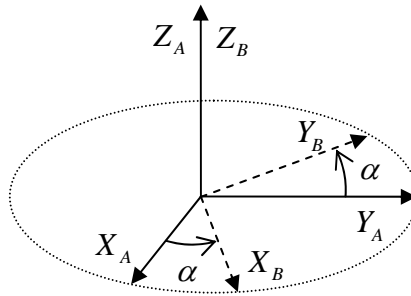
$$\mathbf{V}^B = \mathbf{T}^{BA} \mathbf{V}^A \quad (\text{B.5.1})$$

To create this rotation matrix, the unit vectors of axis system A, coordinated in axis system B, are used as the columns for the rotation matrix [1].

$$\mathbf{T}^{BA} = \begin{bmatrix} \mathbf{i}_B^A & \mathbf{j}_B^A & \mathbf{k}_B^A \end{bmatrix} \quad (\text{B.5.2})$$

A simple example of a single rotation is shown in figure B.1. For this example the rotation matrix is given by,

$$\mathbf{T}^{BA} = \begin{bmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{B.5.3})$$



**Figure B.1:** Single Rotation Example

When two or more rotation matrices are multiplied together, the result is a rotation matrix that will provide the total resulting rotation. For example, if given that

$$\mathbf{T}^R = \mathbf{T}^M \mathbf{T}^N \quad (\text{B.5.4})$$

Then the resulting rotation matrix  $\mathbf{T}^R$  will rotate a vector through the rotation matrix  $\mathbf{T}^N$  and then rotate that result using rotation matrix  $\mathbf{T}^M$ .

## B.6 Special Rotation Matrices

The rotation matrix theory explained in section B.5 will now be applied to create various rotation matrices that will be used in the text.

**Single Rotation Matrices** These rotation matrices are used when two axis systems only differ by one rotation around a single shared unit vector. The example in the previous section shows this for a positive rotation around a shared Z-axis. The results for all three of the axes are listed below with the superscript indicating the common unit vector.

$$\mathbf{T}^i(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & \sin \alpha \\ 0 & -\sin \alpha & \cos \alpha \end{bmatrix} \quad (\text{B.6.1})$$

$$\mathbf{T}^j(\alpha) = \begin{bmatrix} \cos \alpha & 0 & -\sin \alpha \\ 0 & 1 & 0 \\ \sin \alpha & 0 & \cos \alpha \end{bmatrix} \quad (\text{B.6.2})$$

$$\mathbf{T}^k(\alpha) = \begin{bmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{B.6.3})$$

**Body to Wind and Wind to Body Axes Conversions** It is often necessary to convert between the aircraft's body and wind axes with the angles of attack ( $\alpha$ ) and sideslip ( $\beta$ ) known. The rotation matrix for the wind to body axis conversion is shown below. It consists of two single rotations stringed together using equation (B.5.4).

$$\mathbf{DCM}^{BW} = \mathbf{T}^j(\alpha)\mathbf{T}^k(-\beta) \quad (\text{B.6.4})$$

$$\begin{aligned} &= \begin{bmatrix} \cos \alpha & 0 & -\sin \alpha \\ 0 & 1 & 0 \\ \sin \alpha & 0 & \cos \alpha \end{bmatrix} \begin{bmatrix} \cos \beta & -\sin \beta & 0 \\ \sin \beta & \cos \beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ \mathbf{DCM}^{BW} &= \begin{bmatrix} \cos \alpha \cos \beta & -\cos \alpha \sin \beta & -\sin \alpha \\ \sin \beta & \cos \beta & 0 \\ \sin \alpha \cos \beta & -\sin \alpha \sin \beta & \cos \alpha \end{bmatrix} \end{aligned} \quad (\text{B.6.5})$$

The inverse conversion from body to wind axes can be obtained using equation (A.1.2) which states that the inverse of the DCM matrix is simply its transpose.

$$\mathbf{DCM}^{WB} = \left[ \mathbf{DCM}^{BW} \right]^T \quad (\text{B.6.6})$$

$$\mathbf{DCM}^{WB} = \begin{bmatrix} \cos \alpha \cos \beta & \sin \beta & \sin \alpha \cos \beta \\ -\cos \alpha \sin \beta & \cos \beta & -\sin \alpha \sin \beta \\ -\sin \alpha & 0 & \cos \alpha \end{bmatrix} \quad (\text{B.6.7})$$

## B.7 Small Angle Assumptions

In various sections during the text simplifications are made to equations using small angle assumptions. Consider the small angles  $\alpha$  and  $\beta$ . Approximations can be made to mathematical formulae with terms containing these angles and are listed below.

$$\cos \alpha \approx 1 \quad (\text{B.7.1})$$

$$\sin \alpha \approx \alpha \quad (\text{B.7.2})$$

$$\alpha \beta \approx 0 \quad (\text{B.7.3})$$



## Appendix C

# Additional Control System Design Details and Results

This appendix provides additional information regarding the design and testing of the controllers used in this project.

### C.1 Pitch Rate Damping NSA Controller

The design to follow is an alternative to the normal specific acceleration (NSA) controller proposed by [1] – see section 3.3. This controller does not use feedback from NSA to change the natural frequency of the Short Period mode dynamics. Through pitch rate feedback the damping of the Short Period mode can be selected and by feeding back the NSA integrator state, any steady state errors can be removed in the closed loop system. This reduces the controller’s sensitivity to noise on the NSA measurement since only the low pass filtered result of the NSA integrator is used for feedback control.

This control architecture is ideally suited for an aircraft with stable Short Period mode poles already located at a sufficiently high frequency. It adds robustness to the control system as unmodelled effects such as actuator slew rate and control system delays greatly affect the desired pole placement of the NSA dynamics. It is shown in section 3.3.2 how large attempted changes in the system’s natural frequency can result in undesired pole placement results when a single sample period of unmodelled delay is introduced into the system. The natural frequency of the Short Period mode is mainly altered through NSA feedback and therefore feedback from this state is not used in the proposed design.

#### C.1.1 Design Architecture

Altering the elevator control law of equation (3.3.12) to remove NSA feedback yields,

$$\delta_E = -K_Q Q - K_E E_C + \bar{N}_C C_{W_R} + \delta_{E_G} \quad (\text{C.1.1})$$

Equation (3.3.15) for the closed loop characteristic equation then becomes,

$$\alpha_c(s) = s^3 + s^2 \left( \frac{M_{\delta_E}}{I_y} K_Q + \frac{\bar{L}_\alpha}{m\bar{V}} - \frac{M_Q}{I_y} \right) + s \left( \frac{\bar{L}_\alpha}{m\bar{V}} \frac{M_{\delta_E}}{I_y} K_Q - \frac{\bar{L}_\alpha M_Q}{m\bar{V} I_y} - \frac{M_\alpha}{I_y} \right) - \frac{\bar{L}_\alpha M_{\delta_E}}{m I_y} K_E \quad (\text{C.1.2})$$

The desired closed loop characteristic equation for the system is then defined as,

$$\alpha_c(s) = (s^2 + 2\zeta\omega_n s + \omega_n^2) (s + a) \quad (\text{C.1.3})$$

with the first and second terms in brackets corresponding to the complex pole pair of the Short Period mode and the single real pole of the closed loop integrator respectively. Simplifying this equation yields,

$$\alpha_c(s) = s^3 + (2\zeta\omega_n + a) s^2 + (2\zeta\omega_n a + \omega_n^2) s + \omega_n^2 a \quad (\text{C.1.4})$$

By selecting the damping of the Short Period mode ( $\zeta$ ) as well as the integrator pole ( $a$ ), the resulting natural frequency can then be solved. This is done by solving for the pitch rate feedback gain  $K_Q$  in the first and second order terms of the characteristic equation.

$$K_Q = \frac{I_y}{M_{\delta_E}} \left( 2\zeta\omega_n + a - \frac{\bar{L}_\alpha}{m\bar{V}} + \frac{M_Q}{I_y} \right) \quad (\text{C.1.5})$$

$$K_Q = \frac{m\bar{V}}{\bar{L}_\alpha} \frac{I_y}{M_{\delta_E}} \left( 2\zeta\omega_n a + \omega_n^2 + \frac{\bar{L}_\alpha M_Q}{m\bar{V} I_y} + \frac{M_\alpha}{I_y} \right) \quad (\text{C.1.6})$$

Setting these equations equal to each other yields a quadratic equation for the natural frequency,

$$0 = \omega_n^2 + \omega_n \left( 2\zeta a - 2\zeta \frac{\bar{L}_\alpha}{m\bar{V}} \right) + \frac{M_\alpha}{I_y} + \left( \frac{\bar{L}_\alpha}{m\bar{V}} \right)^2 - a \frac{\bar{L}_\alpha}{m\bar{V}} \quad (\text{C.1.7})$$

Solving for positive values of  $\omega_n$  the closed loop natural frequency of the Short Period mode is given by,

$$\omega_n = \zeta\eta + \sqrt{(\zeta\eta)^2 - \frac{M_\alpha}{I_y} - \frac{\bar{L}_\alpha}{m\bar{V}}\eta} \quad (\text{C.1.8})$$

with

$$\eta = \frac{\bar{L}_\alpha}{m\bar{V}} - a \quad (\text{C.1.9})$$

This result can be used in conjunction with equation (C.1.5) or (C.1.6) to calculate the pitch rate feedback gain. The integrator feedback gain can be solved from the final term of the closed loop characteristic equation and is given by,

$$K_E = -a \frac{\omega_n^2 m I_y}{\bar{L}_\alpha M_{\delta_E}} \quad (\text{C.1.10})$$

The integrator's response can be removed from the reference input by placing a feed-forward zero as follows,

$$\bar{N}_C = -\frac{K_E}{z_f} \quad (\text{C.1.11})$$

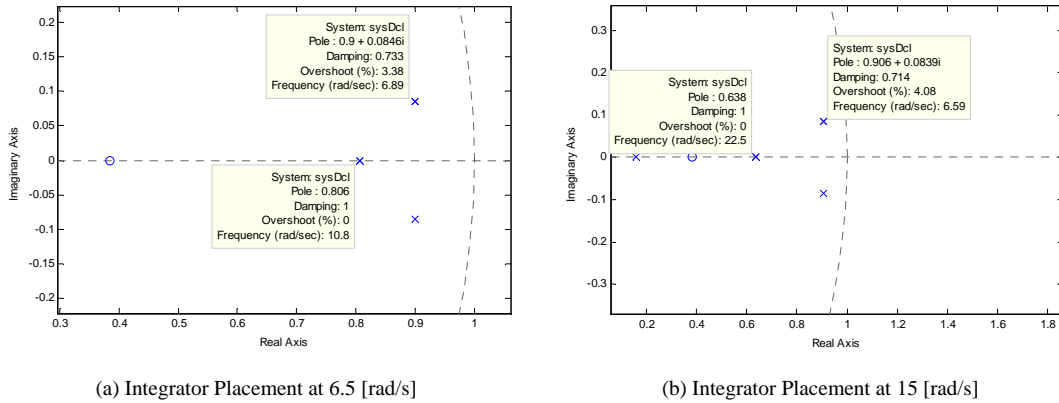
where  $z_f$  is the location of the desired zero.

### C.1.2 Implementation Example

A design example using the CAP232 model aircraft at its trim velocity will be shown for the above controller. At this velocity its open loop Short Period mode poles are given by,

$$\omega_n = 12.8 \text{ [rad/s]}, \quad \zeta = 0.794 \quad (\text{C.1.12})$$

With a conservative damping of 0.8 selected, two selections of integrator pole placement are investigated. Figure C.1(a) shows the effect of control system delay (see section 3.3.2) when an integrator pole of 6.5 rad/s is placed. It can be seen how control system delay still has a negative effect on the pole placement resulting in the damping of the Short Period mode being reduced to 0.733 and the integrator pole being shifted to 10.8 rad/s. The robustness of this control architecture becomes apparent when a higher frequency integrator pole of 15 rad/s is placed. Figure C.1(b) shows this result where it can be seen that the Short Period mode poles are still very well damped at 0.714 and the integrator pole has shifted to a frequency of 22.5 rad/s. In comparison, an integrator placement at that frequency with the NSA controller discussed in section 3.3 will lead to a lightly damped pole pair when the Short Period mode poles are placed at their open loop frequency.



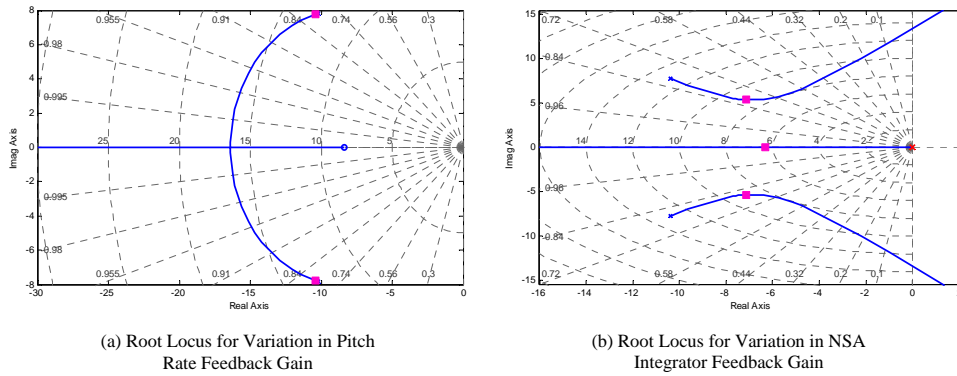
**Figure C.1:** Effect of Control System Delay on Closed Loop Pole Placement

Further investigation of the pole placement scenario with the integrator pole placed at 6.5 rad/s yields the consecutive root locus plots of pitch rate and NSA integrator feed-back. Figure C.2(a) shows how pitch rate feedback provides damping to the Short Period

mode poles. The aircraft used in this project already has well damped poles and therefore very little pitch rate feedback is required. Figure C.2(b) shows the root locus for the variation in integrator feedback gain where it can be seen how the natural frequency of the Short Period mode poles are slightly decreased through this feedback. The resulting natural frequency is given by,

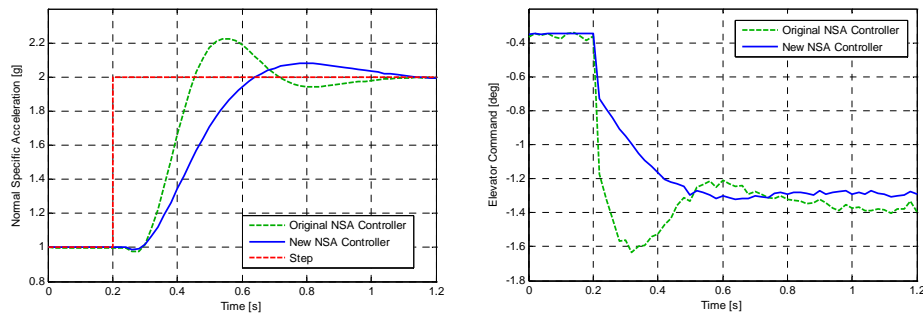
$$\omega_n = 8.92 \text{ [rad/s]} \quad (\text{C.1.13})$$

which is still be fast enough for time scale decoupling to an outer loop controller.



**Figure C.2:** Consecutive Root Locus Design of the Control System

Finally the controller's step response can be analysed. A zero is placed in the closed loop system, similar to the design of the original NSA controller. The zero is again placed at a frequency 60% higher than the integrator pole to avoid excessive feed-forward. The resulting step response is shown in figure C.3 where it is compared to the results of the original NSA controller. As expected the response is slightly slower due to the natural frequency being reduced. The expected damping is however achieved by the response. An interesting point to note is that the amount of noise on the elevator control signal has been reduced as direct feedback from a noisy NSA signal is not used. This signal is low pass filtered through an integrator before it is used in the feedback control system.



**Figure C.3:** Step Response of the Pitch Rate Damping NSA Controller

## C.2 Feedback Linearisation of the Directional Coupling

In order to design an independent roll rate controller, the roll rate and LSA dynamics had to be decoupled. A small amount of coupling does however exist between these two systems and feedback linearisation can be used in the aileron control law to assist in removing these unwanted effects. Start by adding the effect of rudder deflection to equation (3.4.1) for the roll rate dynamics before any decoupling assumptions have been made.

$$\dot{P} = \left[ \frac{L_\beta}{I_x} \right] \beta + \left[ \frac{L_P}{I_x} \right] P + \left[ \frac{L_R}{I_x} \right] R + \left[ \frac{L_{\delta_R}}{I_x} \right] \delta_R + \left[ \frac{L_{\delta_A}}{I_x} \right] \delta_A \quad (\text{C.2.1})$$

with

$$P_W = P \quad (\text{C.2.2})$$

Derive an equation for the angle of sideslip ( $\beta$ ) in terms of lateral acceleration, yaw rate and rudder deflection by rearranging equation (3.4.2) for  $B_W$ .

$$\beta = \left[ \frac{m}{\bar{Y}_\beta} \right] B_W - \left[ \frac{\bar{Y}_R}{\bar{Y}_\beta} \right] R - \left[ \frac{\bar{Y}_{\delta_R}}{\bar{Y}_\beta} \right] \delta_R \quad (\text{C.2.3})$$

Substituting the aileron control law of equation (3.5.4) as well as equation (C.2.3) for the angle of sideslip ( $\beta$ ) into the full roll rate dynamics of equation (C.2.1) yields,

$$\begin{aligned} \dot{P}_W = & \left[ \frac{L_\beta}{I_x} \frac{m}{\bar{Y}_\beta} \right] B_W + \left[ \frac{L_P}{I_x} - \frac{L_{\delta_A}}{I_x} K_P \right] P_W + \left[ -\frac{L_{\delta_A}}{I_x} K_E \right] E_P + \left[ \frac{L_{\delta_A}}{I_x} \bar{N} \right] P_{W_R} \\ & + \left[ \frac{L_R}{I_x} - \frac{L_\beta}{I_x} \frac{\bar{Y}_R}{\bar{Y}_\beta} \right] R + \left[ \frac{L_{\delta_R}}{I_x} - \frac{L_\beta}{I_x} \frac{\bar{Y}_{\delta_R}}{\bar{Y}_\beta} \right] \delta_R + \left[ \frac{L_{\delta_A}}{I_x} \right] \delta_{A_F} \end{aligned} \quad (\text{C.2.4})$$

In order to remove the coupling into the roll rate system, the feedback linearisation term should cancel the unwanted directional dynamics as well as the rudder input. Therefore,

$$\left[ \frac{L_\beta}{I_x} \frac{m}{\bar{Y}_\beta} \right] B_W + \left[ \frac{L_R}{I_x} - \frac{L_\beta}{I_x} \frac{\bar{Y}_R}{\bar{Y}_\beta} \right] R + \left[ \frac{L_{\delta_R}}{I_x} - \frac{L_\beta}{I_x} \frac{\bar{Y}_{\delta_R}}{\bar{Y}_\beta} \right] \delta_R + \left[ \frac{L_{\delta_A}}{I_x} \right] \delta_{A_F} = 0 \quad (\text{C.2.5})$$

Solving this provides an equation for the feedback linearisation control law,

$$\Rightarrow \delta_{A_F} = \frac{1}{L_{\delta_A}} \left( \left[ -L_\beta \frac{m}{\bar{Y}_\beta} \right] B_W + \left[ L_\beta \frac{\bar{Y}_R}{\bar{Y}_\beta} - L_R \right] R + \left[ L_\beta \frac{\bar{Y}_{\delta_R}}{\bar{Y}_\beta} - L_{\delta_R} \right] \delta_R \right) \quad (\text{C.2.6})$$

In the above feedback linearisation control law each term models a different aspect of the coupling from the directional dynamics and rudder perturbation to the roll rate. Therefore it can be used in part or in full depending on aspects such as which of the derivatives are more accurately known or which sensors have less noise on them. Removing terms from the feedback linearisation control law will simply result in a small amount of cou-

pling to exist between the two systems, however it is shown in section 3.5.2 that this will not greatly affect the roll rate controller.

### C.3 Linearisation of the LQR State and Input Matrices

In this section the results of the linearisation of both sets of dynamics used by the LQR outer loop controller are shown. For conciseness only the non-zero terms will be shown.

#### C.3.1 Euler 3-2-1 Dynamics

##### C.3.1.1 State Matrix

The linearised state matrix is given by,

$$\mathbf{F}_{(321)} = \left. \frac{df_{(321)}(\underline{x}, \underline{u})}{d\underline{x}} \right|_{\substack{\underline{x}=\underline{x}_R \\ \underline{u}=\underline{u}_R}} = \begin{bmatrix} \left. \frac{\partial \dot{\phi}}{\partial \phi} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} & \left. \frac{\partial \dot{\phi}}{\partial \theta} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} & \cdots & \left. \frac{\partial \dot{\phi}}{\partial A_W} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} \\ \left. \frac{\partial \dot{\theta}}{\partial \phi} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} & \left. \frac{\partial \dot{\theta}}{\partial \theta} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} & \cdots & \left. \frac{\partial \dot{\theta}}{\partial A_W} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} \\ \vdots & \vdots & \ddots & \vdots \\ \left. \frac{\partial \dot{A}_W}{\partial \phi} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} & \left. \frac{\partial \dot{A}_W}{\partial \theta} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} & \cdots & \left. \frac{\partial \dot{A}_W}{\partial A_W} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} \end{bmatrix} \quad (C.3.1)$$

with,

$$\begin{aligned} \left. \frac{\partial \dot{\phi}}{\partial \phi} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} &= \frac{-C_{W_R} \tan \theta_R \cos \phi_R}{\bar{V}_R}, & \left. \frac{\partial \dot{\phi}}{\partial \theta} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} &= \frac{-C_{W_R} \sin \phi_R \sec^2 \theta_R}{\bar{V}_R} \\ \left. \frac{\partial \dot{\phi}}{\partial \bar{V}} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} &= \frac{C_{W_R} \tan \theta_R \sin \phi_R}{\bar{V}_R^2}, & \left. \frac{\partial \dot{\theta}}{\partial \phi} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} &= \frac{C_{W_R} \sin \phi_R}{\bar{V}_R}, & \left. \frac{\partial \dot{\theta}}{\partial \theta} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} &= \frac{g \sin \theta_R}{\bar{V}_R} \\ \left. \frac{\partial \dot{\theta}}{\partial \bar{V}} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} &= \frac{g \cos \theta_R + C_{W_R} \cos \phi_R}{\bar{V}_R^2}, & \left. \frac{\partial \dot{\psi}}{\partial \phi} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} &= \frac{-C_{W_R} \cos \phi_R \sec \theta_R}{\bar{V}_R} \\ \left. \frac{\partial \dot{\psi}}{\partial \theta} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} &= \frac{-C_{W_R} \sin \phi_R \sec \theta_R \tan \theta_R}{\bar{V}_R}, & \left. \frac{\partial \dot{\psi}}{\partial \bar{V}} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} &= \frac{C_{W_R} \sin \phi_R \sec \theta_R}{\bar{V}_R^2} \\ \left. \frac{\partial \dot{P}_x}{\partial \theta} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} &= -\bar{V}_R \cos \psi_R \sin \theta_R, & \left. \frac{\partial \dot{P}_x}{\partial \psi} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} &= -\bar{V}_R \sin \psi_R \cos \theta_R, & \left. \frac{\partial \dot{P}_x}{\partial \bar{V}} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} &= \cos \psi_R \cos \theta_R \\ \left. \frac{\partial \dot{P}_y}{\partial \theta} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} &= -\bar{V}_R \sin \psi_R \sin \theta_R, & \left. \frac{\partial \dot{P}_y}{\partial \psi} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} &= \bar{V}_R \cos \psi_R \cos \theta_R, & \left. \frac{\partial \dot{P}_y}{\partial \bar{V}} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} &= \sin \psi_R \cos \theta_R \\ \left. \frac{\partial \dot{P}_z}{\partial \theta} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} &= -\bar{V}_R \cos \theta_R, & \left. \frac{\partial \dot{P}_z}{\partial \bar{V}} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} &= -\sin \theta_R, & \left. \frac{\partial \dot{\gamma}}{\partial \theta} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} &= -g \cos \theta_R, & \left. \frac{\partial \dot{\gamma}}{\partial A_W} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} &= 1 \\ & & \left. \frac{\partial \dot{A}_W}{\partial A_W} \right|_{\substack{\underline{x}_R \\ \underline{u}_R}} &= -\tau_p \end{aligned}$$

### C.3.1.2 Input Matrix

The linearised input matrix is given by,

$$\mathbf{G}_{(321)} = \left. \frac{df_{(321)}(\mathbf{x}, \mathbf{u})}{d\mathbf{u}} \right|_{\substack{\mathbf{x}=\mathbf{x}_R \\ \mathbf{u}=\mathbf{u}_R}} = \begin{bmatrix} \left. \frac{\partial \dot{\phi}}{\partial A_{WC}} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} & \left. \frac{\partial \dot{\phi}}{\partial C_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} & \left. \frac{\partial \dot{\phi}}{\partial P_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} \\ \left. \frac{\partial \dot{\theta}}{\partial A_{WC}} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} & \left. \frac{\partial \dot{\theta}}{\partial C_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} & \left. \frac{\partial \dot{\theta}}{\partial P_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} \\ \vdots & \vdots & \vdots \\ \left. \frac{\partial \dot{A}_W}{\partial A_{WC}} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} & \left. \frac{\partial \dot{A}_W}{\partial C_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} & \left. \frac{\partial \dot{A}_W}{\partial P_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} \end{bmatrix}_{(321)} \quad (\text{C.3.2})$$

with,

$$\begin{aligned} \left. \frac{\partial \dot{\phi}}{\partial C_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= -\frac{\sin \phi_R \tan \theta_R}{\bar{V}_R}, & \left. \frac{\partial \dot{\phi}}{\partial P_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= 1, & \left. \frac{\partial \dot{\theta}}{\partial C_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= -\frac{\cos \phi_R}{\bar{V}_R} \\ \left. \frac{\partial \dot{\psi}}{\partial C_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= -\frac{\sin \phi_R \sec \theta_R}{\bar{V}_R}, & \left. \frac{\partial \dot{A}_W}{\partial A_{WC}} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= \tau_p \end{aligned}$$

### C.3.2 Euler 2-3-1 Dynamics

#### C.3.2.1 State Matrix

The linearised state matrix is given by,

$$\mathbf{F}_{(231)} = \left. \frac{df_{(231)}(\mathbf{x}, \mathbf{u})}{d\mathbf{x}} \right|_{\substack{\mathbf{x}=\mathbf{x}_R \\ \mathbf{u}=\mathbf{u}_R}} = \begin{bmatrix} \left. \frac{\partial \dot{\phi}}{\partial \phi} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} & \left. \frac{\partial \dot{\phi}}{\partial \theta} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} & \cdots & \left. \frac{\partial \dot{\phi}}{\partial A_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} \\ \left. \frac{\partial \dot{\theta}}{\partial \phi} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} & \left. \frac{\partial \dot{\theta}}{\partial \theta} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} & \cdots & \left. \frac{\partial \dot{\theta}}{\partial A_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} \\ \vdots & \vdots & \ddots & \vdots \\ \left. \frac{\partial \dot{A}_W}{\partial \phi} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} & \left. \frac{\partial \dot{A}_W}{\partial \theta} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} & \cdots & \left. \frac{\partial \dot{A}_W}{\partial A_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} \end{bmatrix}_{(231)} \quad (\text{C.3.3})$$

with,

$$\begin{aligned} \left. \frac{\partial \dot{\phi}}{\partial \phi} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= \frac{-C_{W_R} \sin \phi_R \tan \theta_R}{\bar{V}_R}, & \left. \frac{\partial \dot{\phi}}{\partial \theta} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= \frac{\sec^2 \theta_R (g \cos \psi_R + C_{W_R} \cos \phi_R)}{\bar{V}_R} \\ \left. \frac{\partial \dot{\phi}}{\partial \psi} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= -\frac{g \tan \theta_R \sin \psi_R}{\bar{V}_R}, & \left. \frac{\partial \dot{\phi}}{\partial \bar{V}} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= \frac{-g \tan \theta_R \cos \psi_R - C_{W_R} \cos \phi_R \tan \theta_R}{\bar{V}_R^2} \\ \left. \frac{\partial \dot{\theta}}{\partial \phi} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= -\frac{C_{W_R} \cos \phi_R}{\bar{V}_R}, & \left. \frac{\partial \dot{\theta}}{\partial \theta} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= \frac{g \sin \psi_R \cos \theta_R}{\bar{V}_R}, & \left. \frac{\partial \dot{\theta}}{\partial \psi} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= \frac{g \cos \psi_R \sin \theta_R}{\bar{V}_R} \\ \left. \frac{\partial \dot{\theta}}{\partial \bar{V}} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= \frac{C_{W_R} \sin \phi_R - g \sin \psi_R \sin \theta_R}{\bar{V}_R^2}, & \left. \frac{\partial \dot{\psi}}{\partial \phi} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= \frac{C_{W_R} \sin \phi_R \sec \theta_R}{\bar{V}_R} \end{aligned}$$

$$\begin{aligned}
\left. \frac{\partial \dot{\psi}}{\partial \theta} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= \frac{-\sec \theta_R \tan \theta_R (g \cos \psi_R + C_{W_R} \cos \phi_R)}{\bar{V}_R}, & \left. \frac{\partial \dot{\psi}}{\partial \psi} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= \frac{g \sin \psi_R \sec \theta_R}{\bar{V}_R} \\
\left. \frac{\partial \dot{\psi}}{\partial \bar{V}} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= \frac{\sec \theta_R (g \cos \psi_R + C_{W_R} \cos \phi_R)}{\bar{V}_R^2}, & \left. \frac{\partial \dot{P}_x}{\partial \theta} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= -\bar{V}_R \cos \psi_R \sin \theta_R \\
\left. \frac{\partial \dot{P}_x}{\partial \psi} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= -\bar{V}_R \sin \psi_R \cos \theta_R, & \left. \frac{\partial \dot{P}_x}{\partial \bar{V}} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= \cos \psi_R \cos \theta_R, & \left. \frac{\partial \dot{P}_y}{\partial \theta} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= \bar{V}_R \cos \theta_R \\
\left. \frac{\partial \dot{P}_y}{\partial \bar{V}} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= \sin \theta_R, & \left. \frac{\partial \dot{P}_z}{\partial \theta} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= \bar{V}_R \sin \theta_R \sin \psi_R, & \left. \frac{\partial \dot{P}_z}{\partial \psi} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= -\bar{V}_R \cos \theta_R \cos \psi_R \\
\left. \frac{\partial \dot{P}_z}{\partial \bar{V}} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= -\cos \theta_R \sin \psi_R, & \left. \frac{\partial \bar{V}}{\partial \theta} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= g \sin \theta_R \sin \psi_R, & \left. \frac{\partial \bar{V}}{\partial \psi} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= -g \cos \theta_R \cos \psi_R \\
\left. \frac{\partial \bar{V}}{\partial A_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= 1, & \left. \frac{\partial \dot{A}_W}{\partial A_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= -\tau_p
\end{aligned}$$

### C.3.2.2 Input Matrix

The linearised input matrix is given by,

$$\mathbf{G}_{(231)} = \left. \frac{df_{(231)}(\mathbf{x}, \mathbf{u})}{d\mathbf{u}} \right|_{\substack{\mathbf{x}=\mathbf{x}_R \\ \mathbf{u}=\mathbf{u}_R}} = \begin{bmatrix} \left. \frac{\partial \dot{\phi}}{\partial A_{W_C}} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} & \left. \frac{\partial \dot{\phi}}{\partial C_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} & \left. \frac{\partial \dot{\phi}}{\partial P_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} \\ \left. \frac{\partial \dot{\theta}}{\partial A_{W_C}} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} & \left. \frac{\partial \dot{\theta}}{\partial C_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} & \left. \frac{\partial \dot{\theta}}{\partial P_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} \\ \vdots & \vdots & \vdots \\ \left. \frac{\partial \dot{A}_W}{\partial A_{W_C}} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} & \left. \frac{\partial \dot{A}_W}{\partial C_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} & \left. \frac{\partial \dot{A}_W}{\partial P_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} \end{bmatrix}_{(231)} \quad (\text{C.3.4})$$

with,

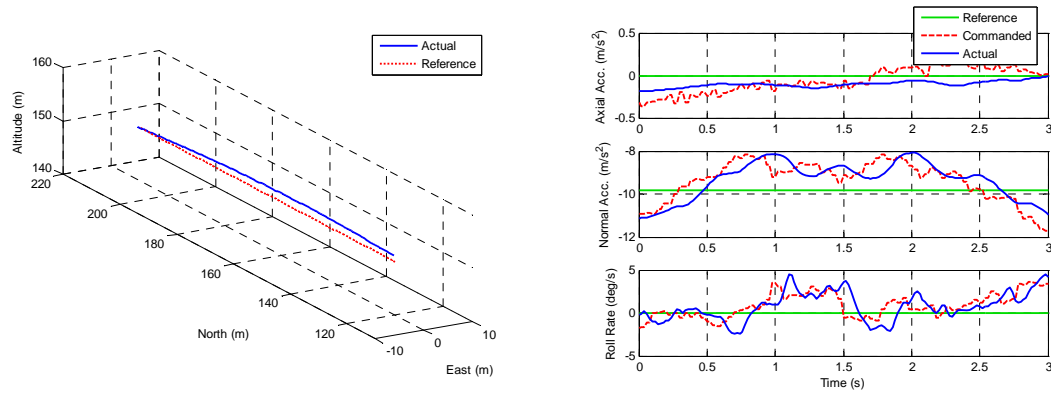
$$\begin{aligned}
\left. \frac{\partial \dot{\phi}}{\partial C_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= \frac{\cos \phi_R \tan \theta_R}{\bar{V}_R}, & \left. \frac{\partial \dot{\phi}}{\partial P_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= 1, & \left. \frac{\partial \dot{\theta}}{\partial \text{partial} C_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= -\frac{\sin \phi_R}{\bar{V}_R} \\
\left. \frac{\partial \dot{\psi}}{\partial C_W} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= -\frac{\cos \phi_R \sec \theta_R}{\bar{V}_R}, & \left. \frac{\partial \dot{A}_W}{\partial A_{W_C}} \right|_{\substack{\mathbf{x}_R \\ \mathbf{u}_R}} &= \tau_p
\end{aligned}$$



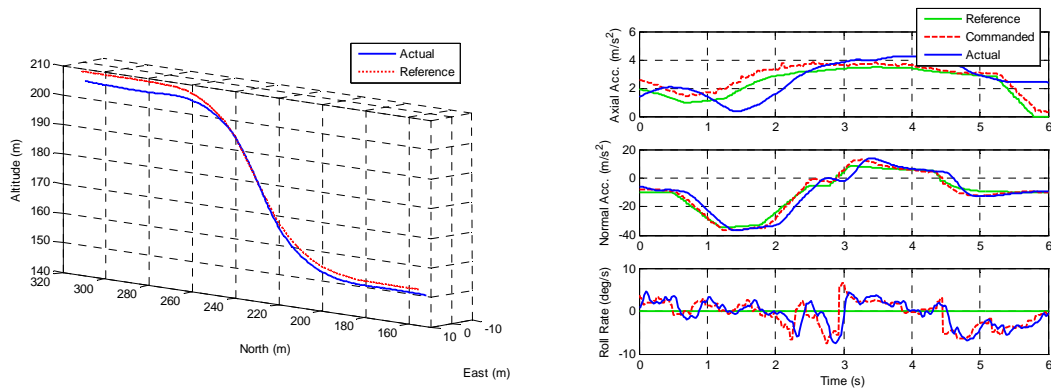
## C.4 Additional LQR Control System Test Results

Any additional results pertaining to the LQR outer loop controller developed in Chapter 4 are shown in this section.

### C.4.1 Trajectory Flight Simulations



**Figure C.4:** Straight and Level Flight Trajectory Simulation for the LQR Controller



**Figure C.5:** Aggressive Climb Trajectory Simulation for the LQR Controller

## C.5 Additional SAM Control System Test Results

Any additional results pertaining to the SAM outer loop controller developed in Chapter 5 are shown in this section.

### C.5.1 Trajectory Flight Simulations

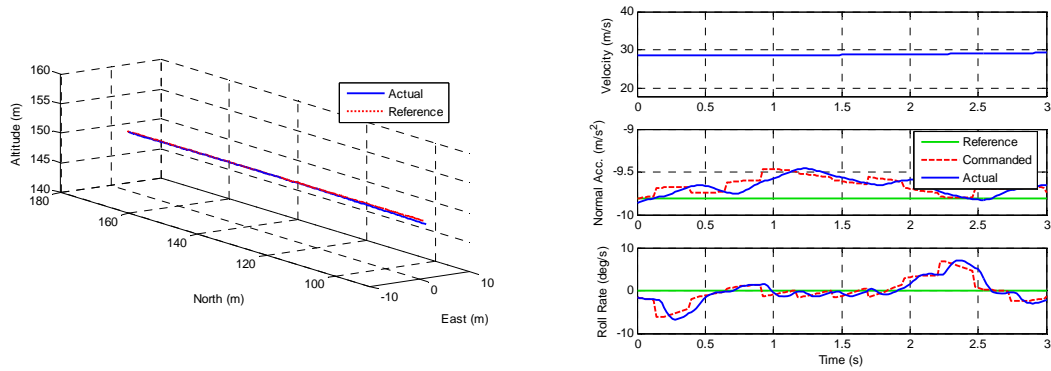


Figure C.6: Straight and Level Flight Trajectory Simulation for the SAM Controller

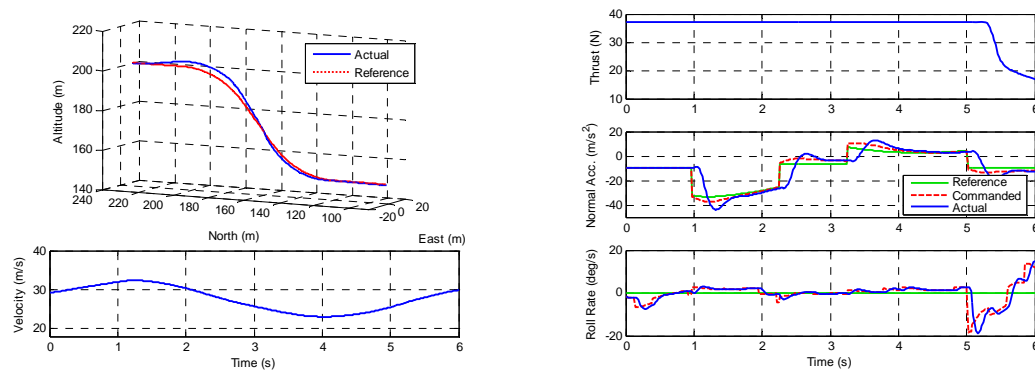


Figure C.7: Aggressive Climb Trajectory Simulation for the SAM Controller

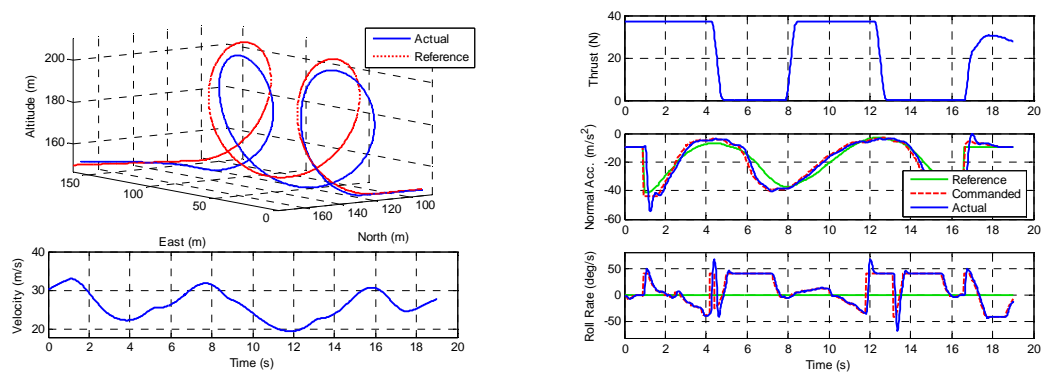
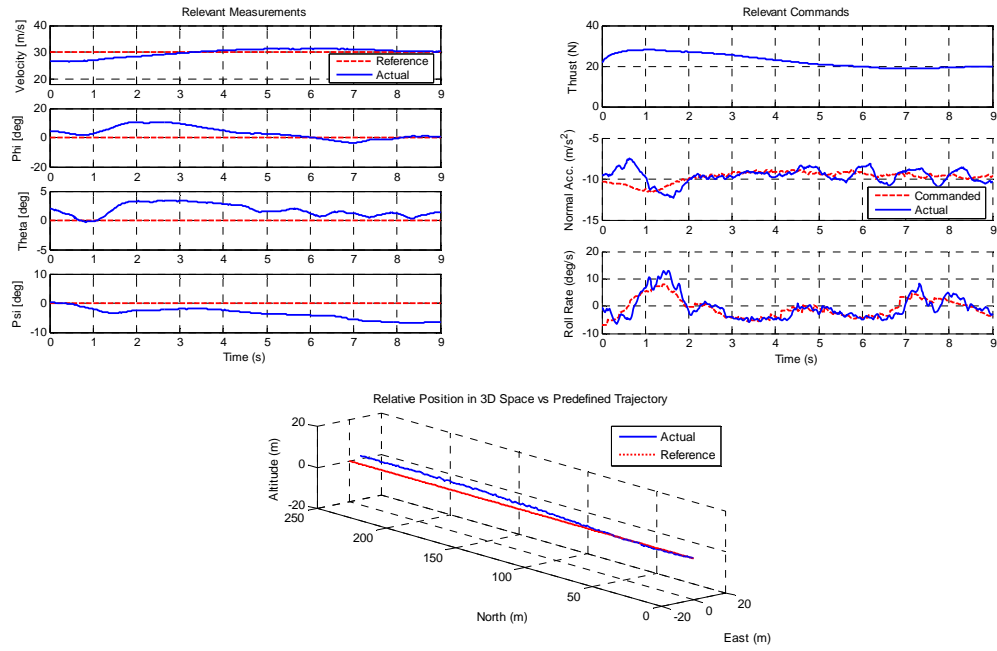
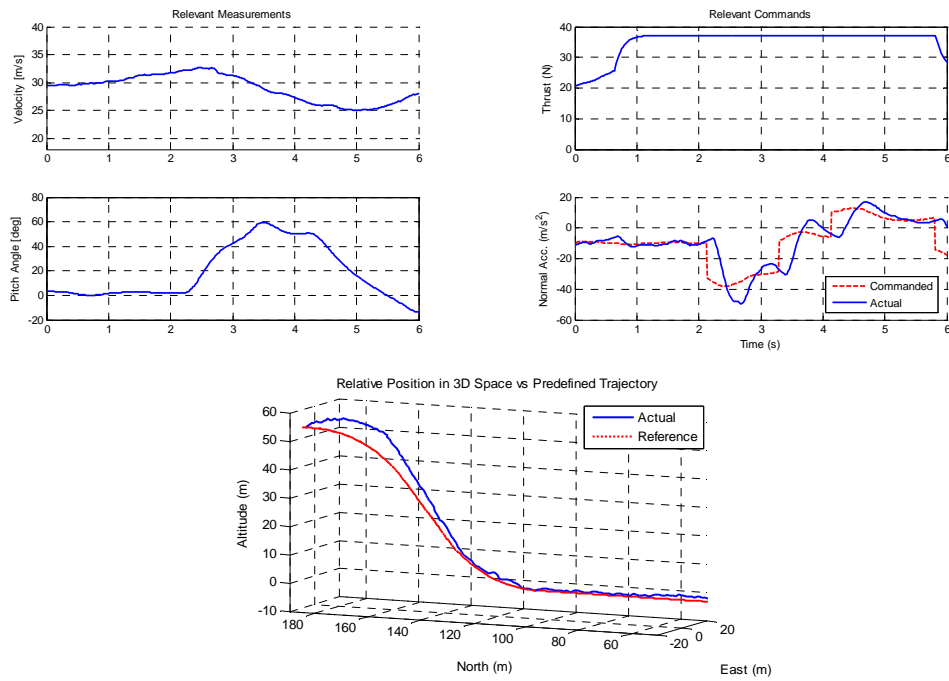


Figure C.8: Barrel Roll Trajectory Simulation for the SAM Controller

### C.5.2 Practical Trajectory Flight Results



**Figure C.9:** Practical Flight Results of the Level Flight by the SAM Controller



**Figure C.10:** Practical Flight Results of the Aggressive Climb by the SAM Controller

## Appendix D

# Position Based Trajectory Design

The development of the mathematical equations for the position based trajectory building blocks used by the SAM guidance controller in Chapter 5 is discussed in this appendix. Each of the trajectory building blocks describe a mathematical path in inertial space. Given the aircraft's current position vector ( $\mathbf{P}_I^{WI}$ ) and velocity magnitude ( $\bar{V}$ ), they are able to return the following information required by the guidance controller,

- $\mathbf{P}_I^{RI}$  – Closest point of the aircraft to the reference trajectory.
- $\mathbf{V}_I^{RI}$  – Reference velocity vector at that point.
- $\boldsymbol{\Sigma}_I^{RI}$  – Reference required specific acceleration vector.
- $\mathbf{k}_I^R$  – Reference NSA unit vector.
- $P_{Wc}$  – The roll rate required to follow the reference trajectory.

As stated before the reference specific acceleration vector will only represent the NSA vector that is required for the aircraft to remain on the trajectory. Any required ASA is not included in the reference as these trajectories are independent of velocity. Therefore,

$$\boldsymbol{\Sigma}^{RI} = \mathbf{N}^R \quad (\text{D.0.1})$$

where  $\mathbf{N}^R$  is the required NSA vector.

### D.1 Straight Line

In this trajectory the aircraft will fly level on a straight line path as shown in section 5.4.1.1. This trajectory is defined with a starting ( $\mathbf{P}_I^S$ ) and ending ( $\mathbf{P}_I^E$ ) point in 3D space and is illustrated in figure D.1.

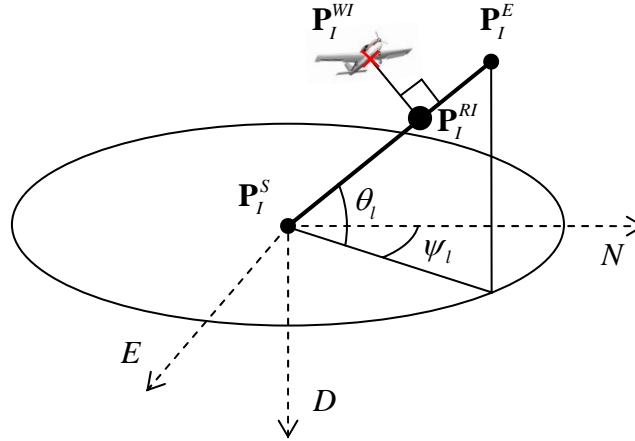


Figure D.1: Straight Line Position Based Trajectory Design

### D.1.1 Reference Position Vector

The following algorithm calculates the nearest point on the trajectory from  $\mathbf{P}_I^{WI}$ , the aircraft's current position. In order to achieve this, the aircraft's position vector can be coordinated in the direction of the line. The section of this vector that runs parallel with the line will then correspond to the closest point.

Define a vector that points from the start to the end of the line,

$$\mathbf{P}_I^{SE} = \mathbf{P}_I^E - \mathbf{P}_I^S \quad (\text{D.1.1})$$

Determine the two angles that describe the direction of the line,

$$\psi_l = \tan^{-1} \left( \frac{E_{SE}}{N_{SE}} \right) \quad (\text{D.1.2})$$

$$\theta_l = \tan^{-1} \left( \frac{-D_{SE}}{\sqrt{N_{SE}^2 + E_{SE}^2}} \right) \quad (\text{D.1.3})$$

with  $N_{SE}$ ,  $E_{SE}$  and  $D_{SE}$  corresponding to the coordinates of the  $\mathbf{P}_I^{SE}$  vector. The closest point of the aircraft to the line can be calculated as a distance along the line as follows,

$$x_R = (N_W - N_S) \cos \psi_l \cos \theta_l + (E_W - E_S) \sin \psi_l \cos \theta_l - (D_W - D_S) \sin \theta_l \quad (\text{D.1.4})$$

where the subscripts  $S$  and  $W$  correspond to the starting point of the line and current location of the aircraft respectively. The unit vector running along the line can also be calculated,

$$\mathbf{u}_I^{SE} = \frac{\mathbf{P}_I^{SE}}{|\mathbf{P}_I^{SE}|} \quad (\text{D.1.5})$$

The closest reference point on the line ( $x_R$ ) can be transformed back to inertial space using this unit vector as follows,

$$\mathbf{P}_I^{RI} = \mathbf{P}_I^S + x_R \mathbf{u}_I^{SE} \quad (\text{D.1.6})$$

### D.1.2 Reference Velocity Vector

The reference velocity vector can be obtained from the calculated by multiplying the current velocity magnitude with the line unit vector.

$$\mathbf{V}_I^{RI} = \bar{V} \mathbf{u}_I^{SE} \quad (\text{D.1.7})$$

### D.1.3 Reference NSA Vector Direction

The following two constraints are placed on the reference NSA vector.

- It lies in the plane that the straight line makes with the vertical axis.
- It is perpendicular to the line.

Therefore using the third inertial unit vector and the line unit vector, the reference NSA vector direction is given by,

$$\mathbf{k}_I^R = \frac{(\mathbf{u}_I^{SE} \times \mathbf{u}_I^K) \times \mathbf{u}_I^{SE}}{|(\mathbf{u}_I^{SE} \times \mathbf{u}_I^K) \times \mathbf{u}_I^{SE}|} \quad (\text{D.1.8})$$

with

$$\mathbf{u}_I^K = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (\text{D.1.9})$$

The special case where  $N_E - N_S = 0$  and  $E_E - E_S = 0$  represents the situation where the aircraft is flying straight upward or downward. In this case the direction of the normal vector needs to be specified as it is not required for the plane to point in any specific direction as its NSA vector will not be used to counter gravity.

### D.1.4 Reference Specific Acceleration Vector

For a straight line the NSA vector is only used to counter the effect of gravity. Therefore,

$$\boldsymbol{\Sigma}_I^{RI} = -\mathbf{k}_I^R (\mathbf{G}_I \cdot \mathbf{k}_I^R) \quad (\text{D.1.10})$$

where  $\mathbf{G}_I$  corresponds to the gravity vector.

### D.1.5 Reference Roll Rate

No roll rate is required to fly a straight line. Therefore,

$$P_{WC} = 0 \quad (D.1.11)$$

## D.2 Aileron Roll

For the aileron roll trajectory, shown in section 5.4.1.2, the starting and ending roll angles ( $\phi_S$  and  $\phi_E$ ) have to be specified in addition to all the variables required for the straight line trajectory. The same reference position and velocity equations from the straight line trajectory can be used for the aileron roll.

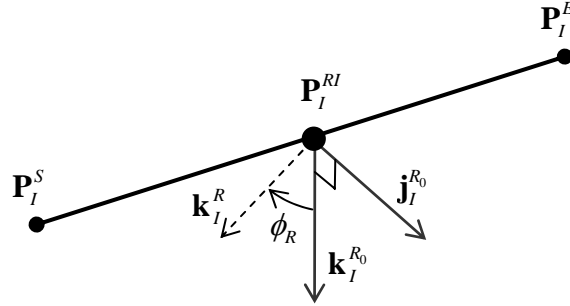


Figure D.2: Aileron Roll Position Based Trajectory Design

### D.2.1 Reference NSA Vector Direction

During a roll the reference NSA unit vector is rotated around the direction of motion from the starting angle ( $\phi_S$ ) to the ending angle  $\phi_E$  with the current roll angle being given by,

$$\phi_R = \phi_S + (\phi_E - \phi_S) \frac{|\mathbf{P}_I^{WI} - \mathbf{P}_I^S|}{|\mathbf{P}_I^E - \mathbf{P}_I^S|} \quad (D.2.1)$$

In order to obtain the rotated NSA unit vector, two base vectors have to be defined. These two unit vectors, shown in figure D.2, span the plane that the NSA unit vector will be rotating in and are obtained as follows,

$$\mathbf{j}_I^{R_0} = \frac{\mathbf{u}_I^K \times \mathbf{u}_I^{SE}}{|\mathbf{u}_I^K \times \mathbf{u}_I^{SE}|} \quad (D.2.2)$$

and

$$\mathbf{k}_I^{R_0} = \mathbf{u}_I^{SE} \times \mathbf{j}_I^{R_0} \quad (D.2.3)$$

where  $\mathbf{u}_I^{SE}$  is unit vector pointing from the start to the end of the line as defined in section D.1 and  $\mathbf{u}_I^K$  is the inertial down vector defined by equation (D.1.9). The rotated reference NSA unit vector can then be written in terms of these two base vectors,

$$\mathbf{k}_I^R = -\mathbf{j}_I^{R_0} \sin \phi_R + \mathbf{k}_I^{R_0} \cos \phi_R \quad (\text{D.2.4})$$

### D.2.2 Reference Specific Acceleration Vector

The component of the normal vector orientated in the vertical axis will be used to counter the effect of gravity.

$$\Sigma_I^{RI} = -\mathbf{k}_I^R (\mathbf{G}_I \cdot \mathbf{k}_I^R) \quad (\text{D.2.5})$$

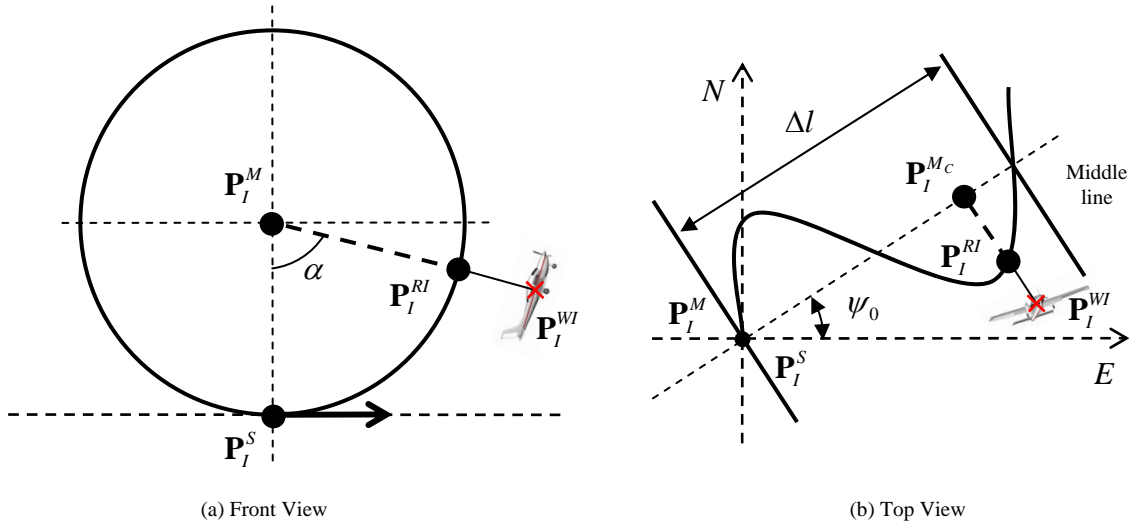
### D.2.3 Reference Roll Rate

The roll rate required to execute the aileron roll is determined by dividing the total roll angle by the time required to complete the roll.

$$P_{WC} = (\phi_E - \phi_S) \frac{\bar{V}}{|\mathbf{P}_I^{SE}|} \quad (\text{D.2.6})$$

## D.3 Vertical Spiral Arc

The vertical spiral arc trajectory, shown in section 5.4.1.3, has the aircraft doing a loop while also moving sideways at a constant rate as illustrated in figure D.3.



**Figure D.3:** Vertical Spiral Arc Position Based Trajectory Design



The parameters that define this trajectory are given by,

- $\mathbf{P}_I^S$  – The starting position of the trajectory.
- $\mathbf{P}_I^M$  – The midpoint of the loop.
- $\alpha_S$  and  $\alpha_E$  – The starting and ending angles of the loop.
- $\Delta l$  – The lateral distance travelled after a full loop rotation.
- $\psi_o$  – The horizontal direction of the loop.

### D.3.1 Reference Position Vector

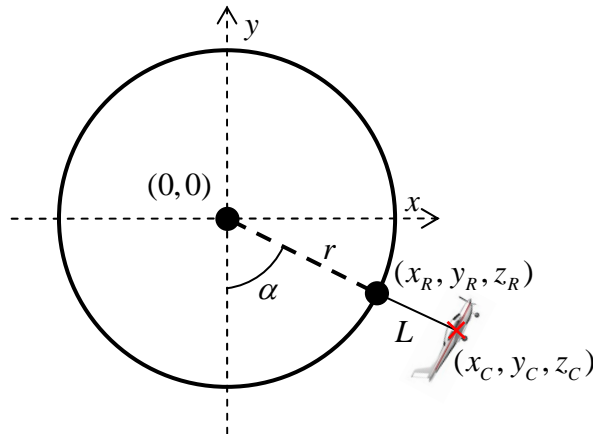
In order to calculate the nearest point on the trajectory from the aircraft's current position,  $\mathbf{P}_I^{WI}$ , the position vectors need to be rotated into the loop's main axes and referenced to its centre point. The specified loop direction angle,  $\psi_o$ , in the NE-plane will be used for this. Three new variables, as shown in figure D.4, can be used to define the trajectory as follows,

$$y = -(d - D_M) \quad (D.3.1)$$

$$x = (n - N_M) \cos(\psi_0) + (e - E_M) \sin(\psi_0) \quad (D.3.2)$$

$$z = -(n - N_M) \sin(\psi_0) + (e - E_M) \cos(\psi_0) \quad (D.3.3)$$

where  $x$  and  $y$  are the variables in the loop's main axes and  $z$  represents the lateral distance travelled while executing the loop. The aircraft's current position coordinates can be substituted into  $n$ ,  $e$  and  $d$  to obtain the aircraft's coordinates  $(x_C, y_C$  and  $z_C)$  in the newly defined axis system.



**Figure D.4:** Vertical Spiral Arc Coordinated Into the Loop's Plane

The three variables that describe the aircraft's position can be written in terms of the current loop angle as follows,

$$y = -r \cos \alpha \quad (\text{D.3.4})$$

$$x = r \sin \alpha \quad (\text{D.3.5})$$

$$z = \Delta l \frac{\alpha}{2\pi} \quad (\text{D.3.6})$$

with the loop's radius given by,

$$r = \left| \mathbf{P}_I^S - \mathbf{P}_I^M \right| \quad (\text{D.3.7})$$

An equation for the length of the aircraft to the trajectory,  $L$ , is given by,

$$L^2 = (x - x_C)^2 + (y - y_C)^2 + (z - z_C)^2 \quad (\text{D.3.8})$$

By substituting equations (D.3.4), (D.3.5) and (D.3.6) and then simplifying yields,

$$L^2 = r^2 + r(2y_C \cos \alpha - 2x_C \sin \alpha) + \alpha^2 \left( \frac{\Delta l}{2\pi} \right)^2 - 2\alpha z_C \left( \frac{\Delta l}{2\pi} \right) + x_C^2 + y_C^2 + z_C^2 \quad (\text{D.3.9})$$

To obtain the minimum length and therefore the closest point of the aircraft to the trajectory, the angle where the derivative in terms of  $\alpha$  is zero has to be found,

$$0 = -r(y_C \sin \alpha_R + x_C \cos \alpha_R) + \alpha_R \left( \frac{\Delta l}{2\pi} \right)^2 - z_C \left( \frac{\Delta l}{2\pi} \right) \quad (\text{D.3.10})$$

The above non-linear equation for  $\alpha_R$  does not have a trivial solution and can be solved numerically to find the angle corresponding to the closest point on the trajectory.

However if iterative numerical techniques want to be avoided, the loop's plane can be used to find the two dimensional closest point with a very simple equation, and thereby ignoring the lateral distance ( $z_C$ ). The reference loop angle can then be determined as follows,

$$\alpha_R = \tan^{-1} \left( -\frac{x_C}{y_C} \right) \quad (\text{D.3.11})$$

The closest position is then given by,

$$y_R = -r \cos \alpha_R \quad (\text{D.3.12})$$

$$x_R = r \sin \alpha_R \quad (\text{D.3.13})$$

$$z_R = \frac{\Delta l}{2\pi} \alpha_R \quad (\text{D.3.14})$$

By providing these position references to the control system, it can be assumed that the lateral position will progress similarly to the position of the aircraft in the loop's plane. These coordinates can then be transformed from the loop's axes back to inertial space as

follows,

$$\mathbf{P}_I^{RI} = \begin{bmatrix} \cos(\psi_0) & 0 & -\sin(\psi_0) \\ \sin(\psi_0) & 0 & \cos(\psi_0) \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} x_R \\ y_R \\ z_R \end{bmatrix} + \mathbf{P}_I^M \quad (\text{D.3.15})$$

### D.3.2 Reference Velocity Vector

For the reference velocity vector direction the lateral heading angle needs to be determined. This is the angle that the aircraft's velocity vector makes with the loop's plane. After a full loop rotation, the lateral distance travelled is given by  $\Delta l$ . Therefore the lateral heading angle can be calculated as follows,

$$\sin \psi = \frac{\Delta l}{2\pi R} \quad (\text{D.3.16})$$

The reference velocity vector now can be calculated as follows,

$$\mathbf{V}_L^{RI} = \bar{V} \begin{bmatrix} \cos \psi \cos \alpha_R \\ \cos \psi \sin \alpha_R \\ \sin \psi \end{bmatrix} \quad (\text{D.3.17})$$

where the subscript  $L$  indicates that the vector is coordinated in the loop's axis system. Similar to the position vector, the velocity vector can now be transformed back to inertial space as follows,

$$\mathbf{V}_I^{RI} = \begin{bmatrix} \cos(\psi_0) & 0 & -\sin(\psi_0) \\ \sin(\psi_0) & 0 & \cos(\psi_0) \\ 0 & -1 & 0 \end{bmatrix} \mathbf{V}_L^{RI} \quad (\text{D.3.18})$$

### D.3.3 Reference NSA Vector Direction

The aircraft's NSA vector should always point towards the middle of the loop. Define a midpoint that is inline with the aircraft's current position in the spiral as shown in figure D.3.

$$\mathbf{P}_I^{Mc} = \mathbf{P}_I^M + z_R \begin{bmatrix} -\sin(\psi_0) \\ \cos(\psi_0) \\ 0 \end{bmatrix} \quad (\text{D.3.19})$$

The reference NSA vector direction can then be calculated as follows,

$$\mathbf{k}_I^R = \frac{\mathbf{P}_I^{Mc} - \mathbf{P}_I^{RI}}{\|\mathbf{P}_I^{Mc} - \mathbf{P}_I^{RI}\|} \quad (\text{D.3.20})$$

$$= \frac{\mathbf{P}_I^{Mc} - \mathbf{P}_I^{RI}}{r} \quad (\text{D.3.21})$$

### D.3.4 Reference Specific Acceleration Vector

During a loop the NSA vector should counter the effect of gravity and create a circular loop. Therefore the reference specific acceleration vector is given by,

$$\Sigma_I^{RI} = -\mathbf{k}_I^R \left( \mathbf{G}_I \cdot \mathbf{k}_I^R + \frac{(\bar{V} \cos \psi)^2}{r} \right) \quad (\text{D.3.22})$$

### D.3.5 Reference Roll Rate

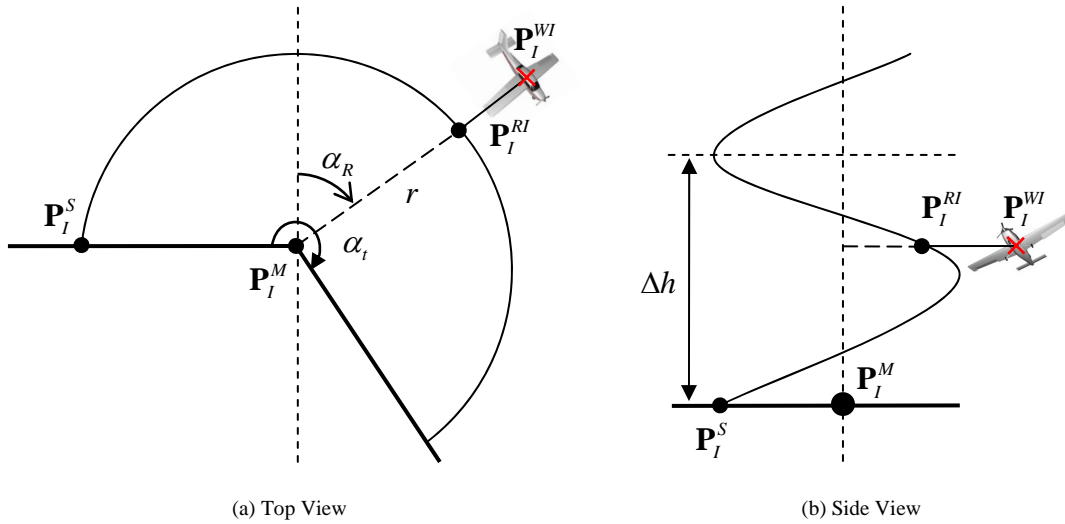
No reference roll rate is required to execute this trajectory. Therefore,

$$P_{WC} = 0 \quad (\text{D.3.23})$$

## D.4 Horizontal Spiral Turn

The horizontal spiral turn trajectory consists of a coordinated turn along with an upward or downward spiral as shown in figure D.5. It is defined by the following parameters,

- $\mathbf{P}_I^S$  – The starting position of the trajectory.
- $\mathbf{P}_I^M$  – The midpoint of the turn (at the same height as the starting point).
- $\alpha_t$  – The total angle the aircraft's heading angle will change by (positive for clockwise spirals and negative for counterclockwise spirals).
- $\Delta h$  – The amount of altitude gain or loss per 360 degree turn.



**Figure D.5:** Horizontal Spiral Turn Position Based Trajectory Design

### D.4.1 Reference Position Vector

For similar reasons as the vertical spiral turn, the horizontal spiral turn's nearest point to the trajectory will only be calculated as seen from the top view. This simplifies the problem greatly and will help to guard against the accidental skipping of an entire turn for trajectories with small values for  $\Delta h$ . Starting by determining the starting and ending turn angles,

$$\alpha_S = \tan^{-1} \left( \frac{E_S - E_M}{N_S - N_M} \right) \quad (\text{D.4.1})$$

$$\alpha_E = \alpha_S + \alpha_t \quad (\text{D.4.2})$$

The current rotation angle of the aircraft on the trajectory can then be calculated using the aircraft's current position as follows,

$$\alpha_R = \tan^{-1} \left( \frac{E_W - E_M}{N_W - N_M} \right) \quad (\text{D.4.3})$$

For multiple turns a check should be added to add (or subtract)  $2\pi$  to the angle after every completed turn. Finally the closest position on the trajectory, coordinated in inertial space, can be calculated using the current rotation angle,

$$\mathbf{P}_I^{RI} = \begin{bmatrix} r \cos \alpha_R \\ r \sin \alpha_R \\ -\Delta h \frac{\alpha_R - \alpha_S}{2\pi} \end{bmatrix} + \mathbf{P}_I^M \quad (\text{D.4.4})$$

with the radius of the turn given by,

$$r = \sqrt{(E_S - E_M)^2 + (N_S - N_M)^2} \quad (\text{D.4.5})$$

### D.4.2 Reference Velocity Vector

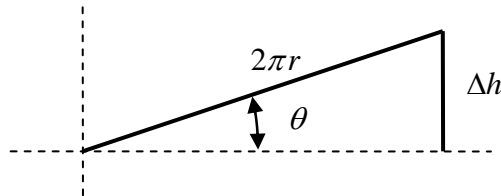


Figure D.6: Horizontal Spiral Turn Climb Angle

For the reference velocity vector, the angle of climb needs to be determined. Figure D.6 shows one full turn with the NE displacement flattened into a straight line. The climb

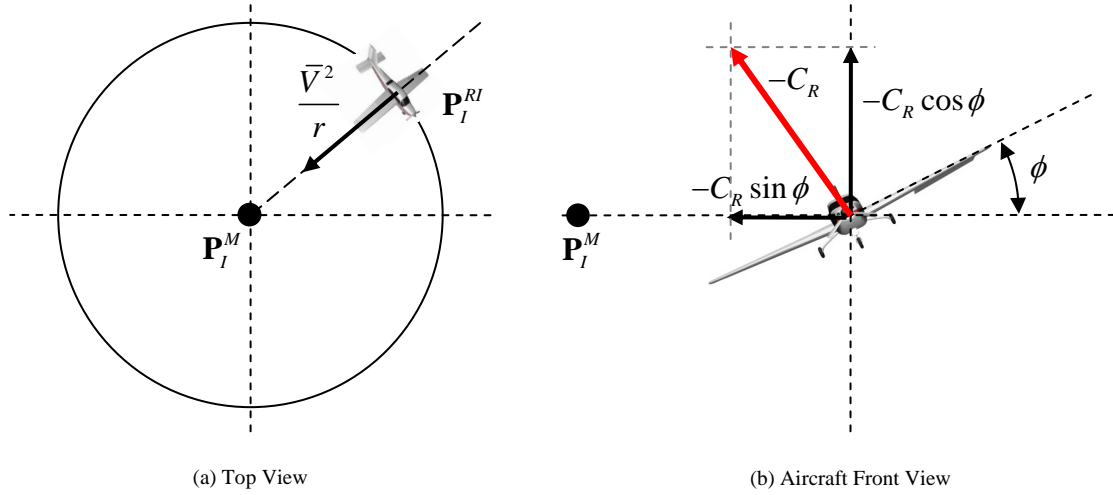
angle can therefore be written as,

$$\sin \theta = \frac{\Delta h}{2\pi R} \quad (\text{D.4.6})$$

The reference velocity vector can now be calculated. The sgn function for the north and east velocity reference is to determine if the turn is in a positive or negative direction.

$$\mathbf{V}_I^{RI} = \bar{V} \begin{bmatrix} -\text{sgn}(\alpha_t) \cdot \cos \theta \sin \alpha_R \\ \text{sgn}(\alpha_t) \cdot \cos \theta \cos \alpha_R \\ -\sin \theta \end{bmatrix} \quad (\text{D.4.7})$$

#### D.4.3 Reference NSA Vector Direction and Specific Acceleration Vector



**Figure D.7:** Horizontal Spiral Turn NSA Vector Analysis

During a coordinated turn the aircraft's NSA vector is used to counter the effect of gravity as well as create the turn as illustrated in figure D.7. Therefore the following equations have to hold,

$$-C_R \cos \phi = g \cos \theta \quad (\text{D.4.8})$$

$$-C_R \sin \phi = \frac{\bar{V}^2}{r} \quad (\text{D.4.9})$$

Solving these gives,

$$\phi = \tan^{-1} \left( \frac{\bar{V}^2}{r g \cos \theta} \right) \quad (\text{D.4.10})$$

$$C_R = -\sqrt{\left( \frac{\bar{V}^2}{r} \right)^2 + (g \cos \theta)^2} \quad (\text{D.4.11})$$

To get the direction of the NSA vector, the reference DCM can be created. Two of the three Euler orientation angles have already been calculated. The third, the aircraft's yaw angle, is given by,

$$\psi = \tan^{-1} \left( \frac{\text{sgn}(\alpha_t) \cdot \cos \alpha_R}{-\text{sgn}(\alpha_t) \cdot \sin \alpha_R} \right) \quad (\text{D.4.12})$$

In the above function the  $\text{sgn}(\alpha_t)$  term is not cancelled out due to it showing the direction the aircraft is pointing in. If the c-function  $\text{atan2}$  is used to solve the equation it provides the heading angle with the correct sign.

Using equation (A.2.2),  $\mathbf{DCM}^{RI}$  can now be created from the three Euler angles. All that remains is to coordinate the aircraft's NSA vector from its reference axis system to inertial space.

$$\mathbf{N}_I^R = [\mathbf{DCM}^{RI}]^{-1} \mathbf{N}_R^R \quad (\text{D.4.13})$$

$$\mathbf{N}_I^R = [\mathbf{DCM}^{RI}]^{-1} \begin{bmatrix} 0 \\ 0 \\ C_R \end{bmatrix} \quad (\text{D.4.14})$$

$$\mathbf{N}_I^R = C_R \begin{bmatrix} \cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi \\ \cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi \\ \cos \theta \cos \phi \end{bmatrix} \quad (\text{D.4.15})$$

As stated before the reference specific acceleration vector only consists of the NSA vector and is therefore given by,

$$\boldsymbol{\Sigma}_I^{RI} = \mathbf{N}_I^R \quad (\text{D.4.16})$$

The NSA vector direction is simply the NSA unit vector.

$$\mathbf{k}_I^R = \frac{\mathbf{N}_I^R}{|\mathbf{N}_I^R|} \quad (\text{D.4.17})$$

#### D.4.4 Reference Roll Rate

If this trajectory is entered into from a straight line, an initial roll rate will be required to attain the correct roll angle. Because the NSAVDC has a relatively fast pole it can be used to correct the aircraft's orientation without any reference feed-forward. During the trajectory however, no roll rate is required.

$$P_{W_C} = 0 \quad (\text{D.4.18})$$

## Appendix E

# Avionics and Ground Station

In this appendix a brief overview of the OBC avionics system will be provided. The ground station interface developed in this project that is used to communicate with this avionics and provide control system commands as well as receive aircraft telemetry is also discussed. Figure E.1 provides an overview of the entire system.

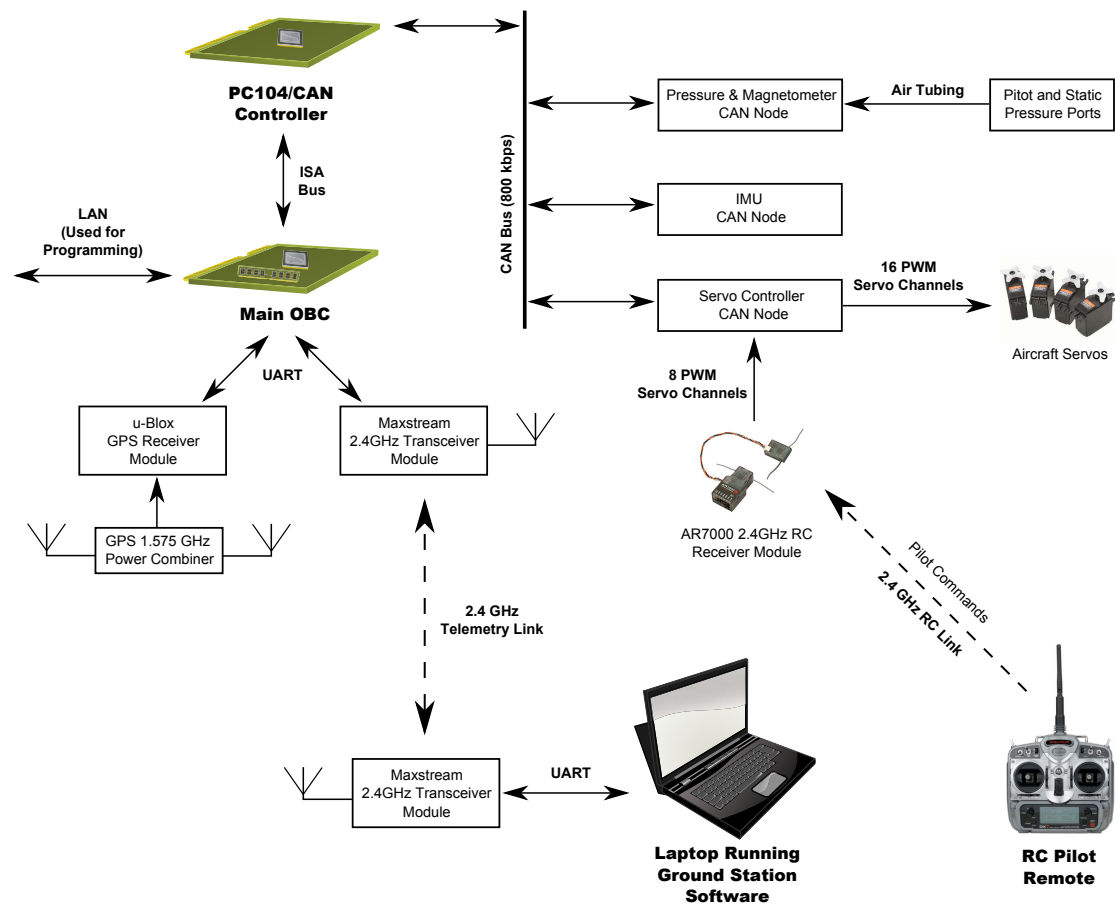


Figure E.1: Avionics System and Ground Station Overview



## E.1 Avionics Overview

In order to practically test the algorithms developed in this thesis, avionics hardware is required that is capable of providing the necessary computational power for their implementation. The OBC avionics hardware developed by [18] and [19] is ideally suited to this task and is used in this project.

### E.1.1 OBC

The system's main processor is embedded in a PC104 form factor. For this project the Kontron MOPS1cd7 is used as it offers a 300 MHz Pentium III with adequate floating point processing power available to it. Flight data is stored on a 32 Mb solid state drive and can be downloaded via LAN to a computer for further analysis. The system runs a Linux based operating system which provides a platform for implementing the control system software.

In order to communicate with the other components in the system, the OBC interfaces with a PC104/CAN controller card developed by [18]. This card acts as a bridge between the processor's ISA bus and the CAN bus that is used for communication by the aircraft's sensor and actuator nodes.

### E.1.2 Can Nodes

**Servo Controller [18]** The servo controller node interfaces with the aircraft's RC receiver as well as its servos. It samples the pilot's servo commands and transmits them via the CAN bus to the OBC. It also sends the relevant servo commands to the aircraft's actuators. By using one of the RC servo channels as an *autopilot switch* the servo controller is able to route either the pilot commands or the autopilot commands received from the OBC to the aircraft's actuators.

**Inertial Measurement Unit (IMU) [19]** The IMU CAN node is responsible for measuring the aircraft's angular rates and specific accelerations in all three axes. It uses MEMS gyroscopes and accelerometers to achieve this. By employing analogue anti-aliasing as well as digital noise filters, relatively low noise sensor data is provided to the OBC via the CAN bus.

**Pressure and Magnetometer Node [15]** This CAN node is used to provide static and dynamic pressure sensor data to the OBC. This data can be used to determine the aircraft's airspeed and altitude. It also supports a three axis magnetometer that is able to measure the earth's magnetic field vector in the aircraft's body axes. This measurement can be used by an estimator to determine the aircraft's attitude.

### E.1.3 UART Devices

Using the OBC's two UART interfaces two additional devices are connected to the avionics system.

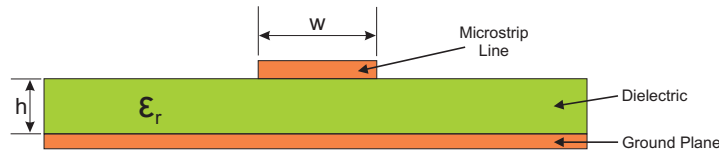
**RF Link** The first is a 2.4 GHz RF link provided by a Maxstream RF modem. This link has a range of a few kilometres, depending on the antenna being used. This interface is used to stream aircraft telemetry to a ground station as well as send controller commands to the aircraft.

**GPS Receiver** The second UART device connected to the OBC is a u-Blox GPS receiver module capable of providing measurements of the aircraft's current position and velocity. As the goal of this project is all-attitude flight, this receiver is connected to a GPS power combiner that combines the measurements of two GPS antennas that are directed both upward and downward and thereby providing GPS measurements at any attitude.

### E.1.4 GPS Power Combiner

This power combiner was developed in this project and provides the correct matching for the two signals received from the GPS antennas. In order to achieve this matching the BP2G+ power combiner module from Mini-Circuits is used which provides 2 way 0 degree  $50 \Omega$  matching in a frequency range from 1.42 GHz to 1.66 GHz. This frequency range is chosen since the GPS L1 band is located at 1.575 GHz and therefore the correct matching will be attained.

During the design of this combiner the characteristic impedance of the PCB tracks leading to this module had to be matched to that of the rest of the system at  $50 \Omega$ . If this was not done then the exact matching provided by the power combiner would be negated by the mismatch occurring as soon as the signals left the combiner module. In order to accomplish this, consider figure E.2 of a microstrip line.



**Figure E.2:** Microstrip Line Illustration

According Bahl's equation provided by [13], the approximate characteristic impedance of a microstrip line, when  $(\frac{w}{h}) \geq 1$ , is given by,

$$z_0 = \frac{120\pi}{\sqrt{\epsilon_{eff}} \left[ \frac{w}{h} + 1.393 + \frac{2}{3} \ln \left( \frac{w}{h} + 1.444 \right) \right]} \quad (\text{E.1.1})$$

with the effective dielectric constant  $\epsilon_{eff}$  being somewhat less than the substrate's relative permittivity as some of the fields from the microstrip conductor exist in air. According to [13] this effective dielectric constant can be calculated for the  $(\frac{w}{h}) \geq 1$  case as follows,

$$\epsilon_{eff} = \frac{\epsilon_r + 1}{2} + \frac{\epsilon_r - 1}{2} \left[ 1 + 12 \left( \frac{h}{w} \right) \right]^{-1/2} \quad (\text{E.1.2})$$

The substrate used in this design is FR4 with a thickness of,

$$h = 0.8 \text{ [mm]} \quad (\text{E.1.3})$$

and a relative permittivity of,

$$\epsilon_r = 4.3 \text{ [F/m]} \quad (\text{E.1.4})$$

Using equations (E.1.1) and (E.1.2) the line width required to match the microstrip line to a characteristic impedance of  $z_0 = 50 \Omega$  is calculated as,

$$w = 1.57 \text{ [mm]} \quad (\text{E.1.5})$$

The PCB was therefore designed using the above width for the tracks leading to the power combiner module and thereby a matched system was created.

## E.2 Ground Station

A ground station interface has been developed from a framework already put in place by [15]. This ground station provides updated telemetry from the aircraft's sensors, its estimator and control systems. It also acts as an interface with the control system and provides the ability to configure various aspects of the controllers and command different flight trajectories. Each tab in the ground station has its own unique function and is briefly discussed below.

**Ground Station** This is the main tab and provides the ability to run the control program on the OBC as well as download flight telemetry to a computer for analysis. The ability to enable the estimator and arm the autopilot is also available. OBC status telemetry is displayed on this page providing amongst others the ability to monitor the main battery voltage and ensure it does not drop below acceptable levels. This tab is shown in figure E.3.

**Control Setup** On this page the ability to configure the low level settings of the aircraft's control system is provided. Functionality includes the ability to change inner loop pole placements as well as change actuator limits.

**LQR Controller** From here the first outer loop control system developed in this project, namely the LQR control system, can be commanded. Various trajectories can be selected and the ability of the aircraft to track them is then displayed visually as shown in figure E.5. The ability to alter the LQR weighting matrices is also provided as well as a telemetry display showing the current state errors.

**Ctrl Testing** This page is used to test various individual controllers used by the aircraft. It can be seen in figure E.4 how the ability is provided to step individual inner loop controllers and the resulting response is then sent through the telemetry link and displayed on this page. The inner loop controllers can also be configured here to receive their commands from the safety pilot. This can enable the safety pilot to fly the aircraft through the inner loop controllers and thereby test their functionality.

**SAM Controller** The second outer loop control system developed in this project can be commanded from this tab. The linear SAM control system can be commanded to fly various position based trajectories. The telemetry display plots the aircraft's actual path along with the nominal flight path of the selected trajectory and can be used to visually see how well a trajectory is being tracked as shown in figure E.6. Reference tracking of the controller's states is also shown and the ability to configure the controller's pole placement is also provided.

**Sensors** The current measurements of all the aircraft's sensors is shown here. The ability to zero the sensors as well as configure the aircraft's GPS module is provided.

**Actuators** This page shows the current pilot commands sampled from the RC receiver as well as the commands sent to the aircraft's servos from the OBC. Trim servo positions can also be set here that will be used by the inner loop controllers as a zero reference.

**Estimator** The kinematic state estimator used on this aircraft can be initialised and configured on this page. The ability to change measurement and process noise covariances is also provided and the current state estimates are displayed.

**Servo Setup** This page is used to configure the aircraft's servos with the ability to upload various mechanical gains and offsets being provided. It enables the engineer to ensure that the commanded servo values are attained and that no offsets exist on the actuators.

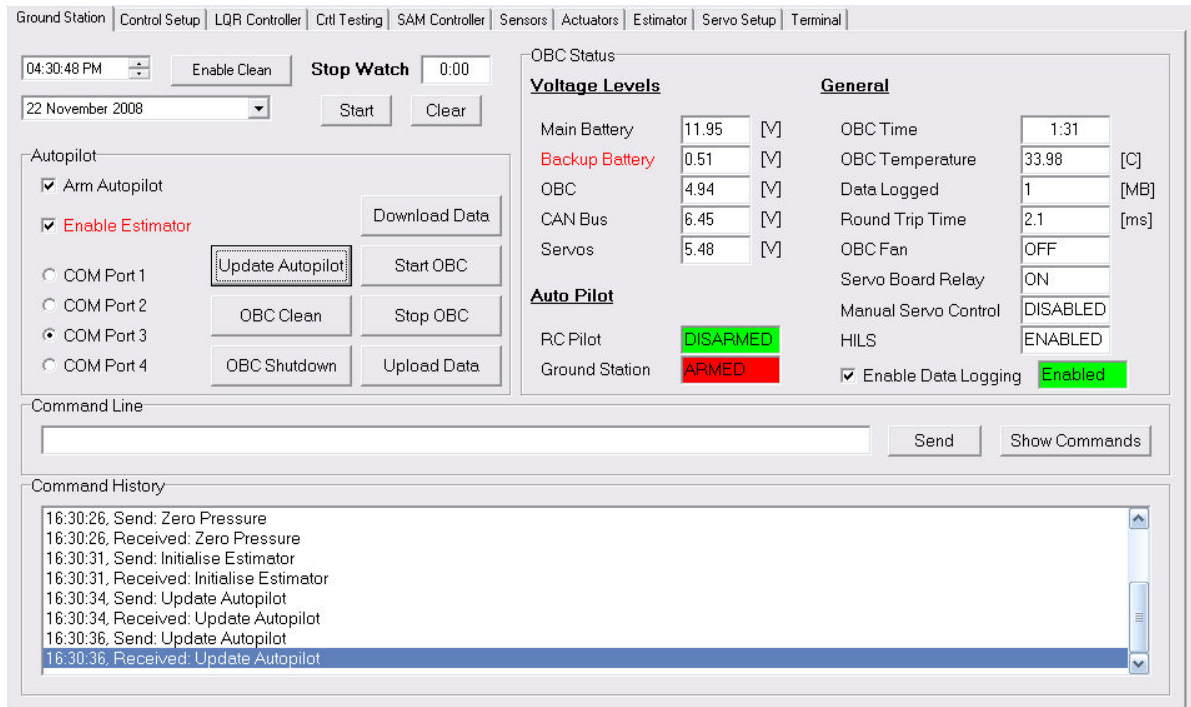


Figure E.3: Main Ground Station Tab in the Ground Station Interface

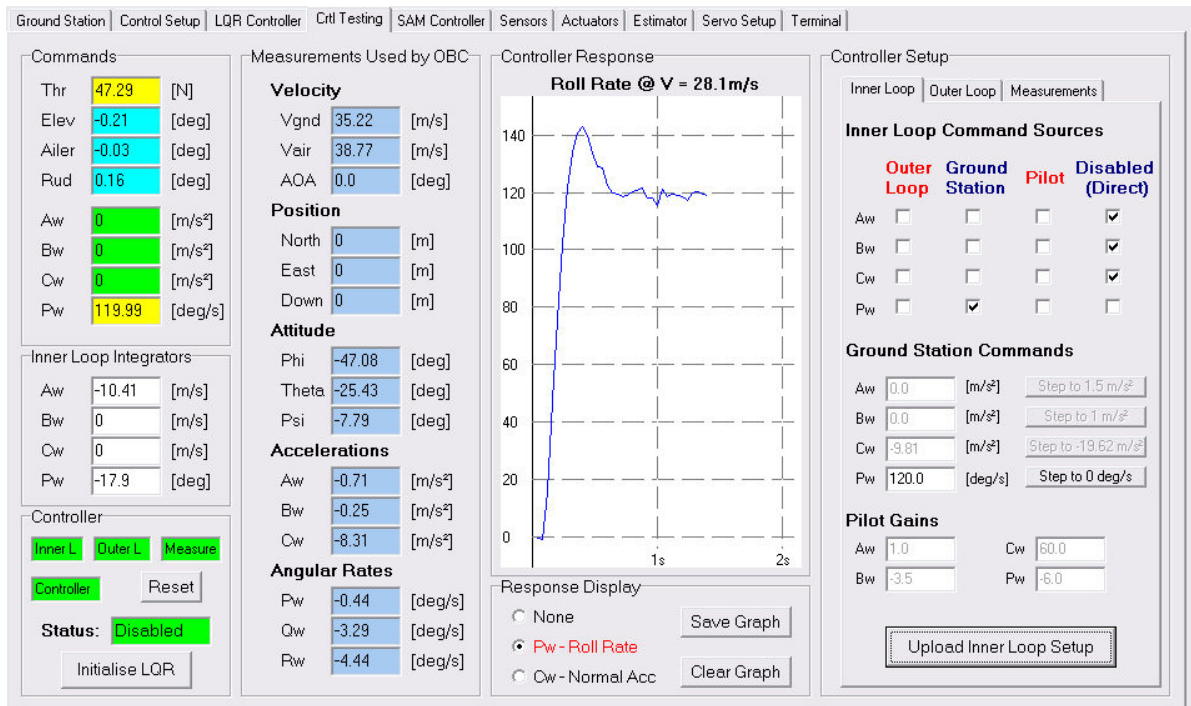


Figure E.4: Controller Test Tab in the Ground Station Interface

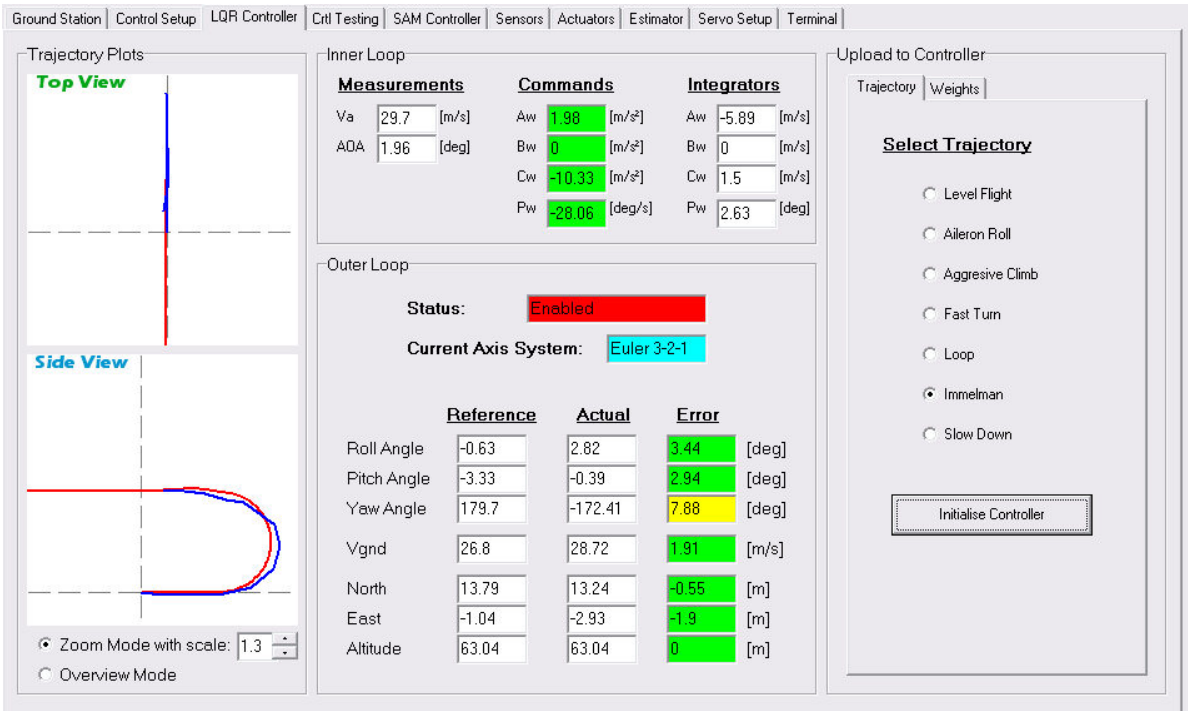


Figure E.5: LQR Control System Command Tab in the Ground Station Interface

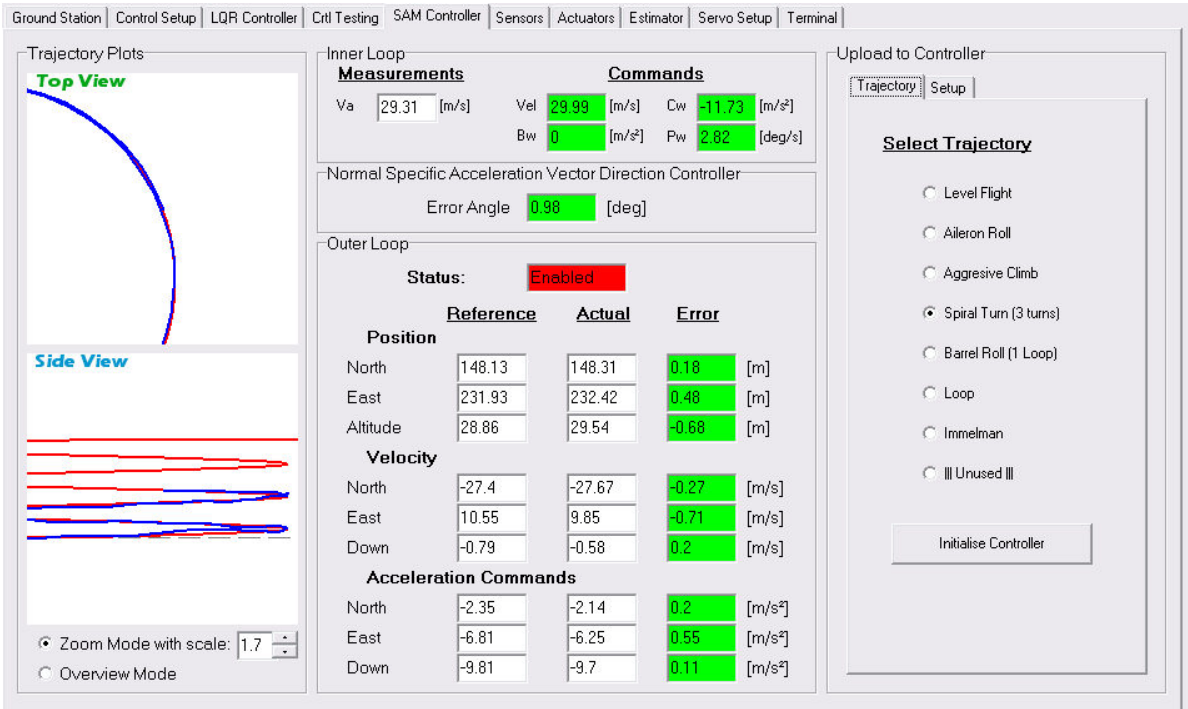


Figure E.6: SAM Control System Command Tab in the Ground Station Interface

## Appendix F

# Aircraft Specifications and Modelling

The aircraft used to test the aggressive flight controllers developed in this project is a CAP232 0.90 size model aerobatic aircraft. It has been fitted with a GMS 1.20 cubic inch methanol engine which provides a large amount of thrust in order to carry out the aggressive flight manoeuvres in this project. This appendix will summarise the aerodynamic and physical characteristics of this aircraft which are used in the non-linear model discussed in section 2.5 as well as in the design of the aircraft's control systems.

This aircraft was already used in a previous project by [15] and therefore the aerodynamic modelling as well as moment of inertia measurements has already been carried out and are merely restated here for convenience.

### F.1 Modelling Constants

Certain constants used in the aircraft model are defined in this section

#### F.1.1 Air density

The density of air is considered a constant in the aerodynamic model used in this project. For the small altitude ranges used in this project this assumption is valid. Therefore the air density at sea level is used and is given by,

$$\rho = 1.225 \text{ [kg/m}^3\text{]} \quad (\text{F.1.1})$$

If it is desired to fly over a larger altitude range, then the value for  $\rho$  can merely be updated in the gain calculations of the inner loop controllers developed in Chapter 3, since air density changes much slower than the aircraft's inner loop dynamics.

### F.1.2 Gravitational Acceleration

Through similar arguments as for air density, the earth's gravitational acceleration at sea level is used and is given by,

$$g = 9.81 \text{ [m/s}^2\text{]} \quad (\text{F.1.2})$$

## F.2 Physical Specifications

### F.2.1 Mass

The aircraft's mass with the avionics installed has been measured as,

$$m = 5.5 \text{ [kg]} \quad (\text{F.2.1})$$

### F.2.2 Moment of Inertia

The moment of inertia tests were carried out by [15] for this aircraft. The double pendulum method was used as also employed by [14]. For this method the aircraft is suspended by two thin strings that run parallel with the axis of interest. The moment of inertia of that axis is then determined by inducing a small moment and observing the resulting period of oscillation. For more information on this method see [15]. The measured moment of inertia results for each of the aircraft's axes are listed in table F.1.

Inertia	Value	Unit
$I_x$	0.2	kg.m <sup>2</sup>
$I_y$	0.36	kg.m <sup>2</sup>
$I_z$	0.525	kg.m <sup>2</sup>

**Table F.1:** Moment of Inertia Measurements [15]

### F.2.3 Engine Thrust

Through a static thrust test the maximum thrust of the aircraft's engine has been determined as,

$$T_{\max}^S = 60 \text{ [N]} \quad (\text{F.2.2})$$

However as stated in section 3.2.6, this thrust changes significantly when the aircraft is flying at its trim airspeed of 30 m/s. A thrust reduction factor has been calculated at this speed and is given by,

$$C^{DS} = 0.62 \quad (\text{F.2.3})$$

The engine's dominant dynamic response has been measured as a time lag of approximately 0.6 s, as stated in section 2.4.3. This can however be roughly approximated by a



first order lag model with a time constant of,

$$\tau = 0.75 \text{ [s]} \quad (\text{F.2.4})$$

### F.3 Aerodynamic Model

#### F.3.1 Airfoil Specifications

The properties of the CAP232's airfoil are listed in table F.2. Some of the parameters listed in this table are pictured in figure F.1. The wing's mean aerodynamic chord, span and area are used as non-dimensionalising coefficients in the calculation of the aircraft's stability and control derivatives in the sections to follow.

Parameter	Symbol	Value	Unit
Root Chord	$c_r$	0.38	m
Tip Chord	$c_t$	0.2	m
Span	$b$	1.73	m
Area	$S$	0.5017	m <sup>2</sup>
Mean Aerodynamic Chord	$\bar{c}$	0.2993	m
Aspect Ratio	$A$	5.9655	ND
Oswald Efficiency Factor	$e$	0.85	ND

Table F.2: Airfoil Properties [15]

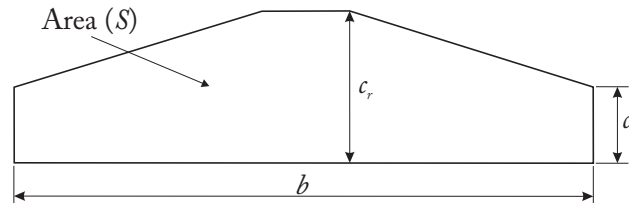


Figure F.1: Airfoil Illustration

#### F.3.2 Static Coefficients

##### F.3.2.1 Zero Angle of Attack Lift and Pitching Moment Coefficients

Due to the CAP232 model aircraft's wing having symmetrical camber these coefficients can be approximated as zero,

$$C_{L_0} = 0 \quad (\text{F.3.1})$$

$$C_{m_0} = 0 \quad (\text{F.3.2})$$

### F.3.2.2 Parasitic Drag Coefficient

Various methods exist with which to calculate the parasitic drag coefficient such as the empirical method outlined in [6] and used by [15]. Practically however it is difficult to achieve accurate results as many factors contribute to the parasitic drag. The aircraft used in this project has non-retractable landing gear as well as an external Pitot tube. All of these as well as the aircraft's body and lifting surfaces contribute to the parasitic drag. For this reason practical tests were used to determine the approximate parasitic drag coefficient of this aircraft as,

$$C_{D_0} = 0.07 \quad (\text{F.3.3})$$

### F.3.3 Non-Dimensional Stability and Control Derivatives

In order to calculate the aircraft's stability and control derivatives, the computational fluid dynamics *vortex-lattice method* is used. The aircraft's lifting surfaces are assumed to be infinitely thin and thereby the influence of thickness on the aerodynamic forces is ignored. This method produces derivatives with a relative degree of accuracy however other methods such as wind tunnel tests or more complete CFD calculations such as those from *Fluent* will yield more accurate results. For a statically stable aircraft, such as the one used in this project, the vortex-lattice derivatives will provide a sufficiently accurate aerodynamic model and robustness in the control system architecture will remove any errors induced due to modelling inaccuracies.

The vortex-lattice program used to calculate the derivatives of the CAP232 model aircraft is called *Athena Vortex Lattice* and has been developed at MIT's department of Aeronautics and Astronautics. The aircraft's geometry is specified using a text file and can also be plotted to ensure that it resembles the actual airframe, as shown in figure F.2. This modelling was done by [15] and the stability and control derivative results are provided in tables F.3, F.4 and F.5.

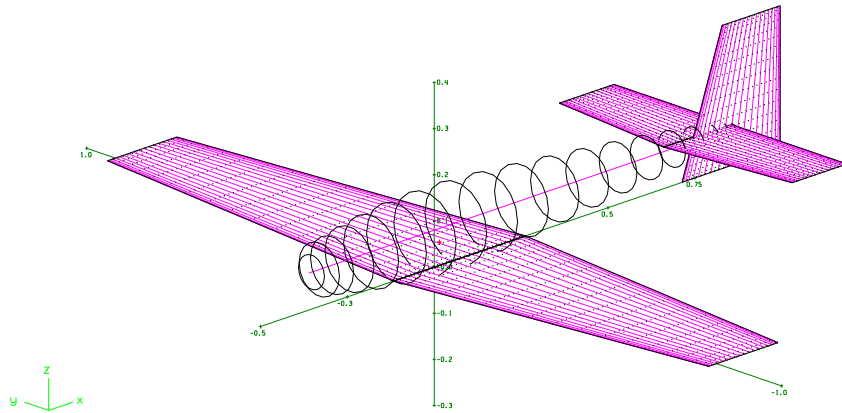


Figure F.2: AVL Geometry Plot of the Aircraft

Angle of Attack ( $\alpha$ )	Value	Pitch Rate ( $Q$ )	Value
$C_{L_\alpha}$	5.1309	$C_{L_q}$	7.7330
$C_{m_\alpha}$	-0.2954	$C_{m_q}$	-10.2807

**Table F.3:** Longitudinal Stability Derivatives [15]

Angle of Sideslip ( $\beta$ )	Value	Roll Rate ( $P$ )	Value	Yaw Rate ( $R$ )	Value
$C_{y_\beta}$	-0.2777	$C_{y_p}$	0.0102	$C_{y_r}$	0.212231
$C_{l_\beta}$	-0.0331	$C_{l_p}$	-0.4248	$C_{l_r}$	0.045011
$C_{n_\beta}$	0.0860	$C_{n_p}$	-0.0251	$C_{n_r}$	-0.124994

**Table F.4:** Lateral Stability Derivatives [15]

Elevator ( $\delta_E$ )	Value	Aileron ( $\delta_A$ )	Value	Rudder ( $\delta_R$ )	Value
$C_{L_{\delta_E}}$	0.71266	$C_{y_{\delta_A}}$	-0.0077	$C_{y_{\delta_R}}$	0.2303
$C_{l_{\delta_E}}$	-1.5853	$C_{l_{\delta_A}}$	-0.3731	$C_{l_{\delta_R}}$	0.0080
$C_{n_{\delta_E}}$		$C_{n_{\delta_A}}$	-0.0065	$C_{n_{\delta_R}}$	-0.1129

**Table F.5:** Control Derivatives [15]

### F.3.4 Trim Airspeed

The aircraft's trim airspeed has been selected at,

$$V_0 = 30 \text{ [m/s]} \quad (\text{F.3.4})$$

as this speed is approximately in the middle of the aircraft's allowable velocity range – refer to equation (6.1.1).

## F.4 Dimensional Stability and Control Derivative Notation

The notation for the dimensional stability and control derivatives used in certain sections of this thesis is defined as follows,

$$A_x = qSl \frac{\partial C_A}{\partial x'} n_c \quad (\text{F.4.1})$$

with the length  $l$  being given by  $\bar{c}$  for pitch moment derivatives,  $b$  for roll and yaw moment derivatives and unity for force derivatives. The normalising coefficient  $n_c$  refers

to equation (2.4.21). For additional clarity all of the dimensional stability and control derivatives are provided in tables F.6 and F.7.

Due to	Lift Forces	Sideslip Forces
Angle of Attack ( $\alpha$ )	$\bar{L}_\alpha = qSC_{L_\alpha}$	$\bar{Y}_\beta = qSC_{y_\beta}$
Angle of Sideslip ( $\beta$ )		$\bar{Y}_P = qS\frac{b}{2V_a}C_{y_P}$
Roll Rate ( $P$ )		
Pitch Rate ( $Q$ )	$\bar{L}_Q = qS\frac{\bar{c}}{2V_a}C_{L_Q}$	$\bar{Y}_R = qS\frac{b}{2V_a}C_{y_R}$
Yaw Rate ( $R$ )		
Elevator Deflection ( $\delta_E$ )	$\bar{L}_{\delta_E} = qSC_{L_{\delta_E}}$	$\bar{Y}_{\delta_A} = qSC_{y_{\delta_A}}$
Aileron Deflection ( $\delta_A$ )		$\bar{Y}_{\delta_R} = qSC_{y_{\delta_R}}$
Rudder Deflection ( $\delta_R$ )		

**Table F.6:** Dimensional Stability and Control Derivatives (Forces)

Due to	Roll moments	Pitch moments	Yaw moments
Angle of Attack ( $\alpha$ )		$M_\alpha = qS\bar{c}C_{m_\alpha}$	$N_\beta = qSbC_{n_\beta}$
Angle of Sideslip ( $\beta$ )	$L_\beta = qSbC_{l_\beta}$		$N_P = qSb\frac{b}{2V_a}C_{n_P}$
Roll Rate ( $P$ )	$L_P = qSb\frac{b}{2V_a}C_{l_P}$		
Pitch Rate ( $Q$ )		$M_Q = qS\bar{c}\frac{\bar{c}}{2V_a}C_{m_Q}$	$N_R = qSb\frac{b}{2V_a}C_{n_R}$
Yaw Rate ( $R$ )	$L_R = qSb\frac{b}{2V_a}C_{l_R}$		
Elevator Deflection ( $\delta_E$ )		$M_{\delta_E} = qS\bar{c}C_{m_{\delta_E}}$	
Aileron Deflection ( $\delta_A$ )	$L_{\delta_A} = qSbC_{l_{\delta_A}}$		$N_{\delta_A} = qSbC_{n_{\delta_A}}$
Rudder Deflection ( $\delta_R$ )	$L_{\delta_R} = qSbC_{l_{\delta_R}}$		$N_{\delta_R} = qSbC_{n_{\delta_R}}$

**Table F.7:** Dimensional Stability and Control Derivatives (Moments)

# Bibliography

- [1] I.K. Peddle. *Acceleration Based Manoeuvre Flight Control System for Unmanned Aerial Vehicles*. PhD dissertation, Stellenbosch University, 2008.
- [2] P.H. Zipfel. *Modelling and Simulation of Aerospace Vehicle Dynamics*. AIAA Education Series, 2000.
- [3] J. H. Blakelock. *Automatic Control of Aircraft and Missiles, 2nd ed.* John Wiley & Sons, 1991.
- [4] B. Etkin, L.D. Reid. *Dynamics of Flight, Stability and Control, 3rd Ed.* John Wiley & Sons, 1996.
- [5] M.V. Cook. *Flight Dynamics Principles*. Elsevier Butterworth-Heinemann, 1997.
- [6] R.S. Shevell. *Fundamentals of Flight*. Prentice-Hall, 1983.
- [7] D.E. Kirk. *Optimal Control Theory - An Introduction*. Prentice-Hall, 1970.
- [8] J. Diebel. *Representing Attitude: Euler Angles, Unit Quaternions and Rotation Vector*. Stanford University, 2006.
- [9] G.F. Franklin, J.D. Powell, M.L. Workman. *Digital Control of Dynamic Systems*. Addison Wesley Longman, 1998.
- [10] G.F. Franklin, J.D. Powell, A. Emami-Naeini. *Feedback Control of Dynamic Systems*. Prentice Hall, 2002.
- [11] D.C. Lay. *Linear Algebra and Its Applications, 3rd Ed.* Pearson Education Inc, 2003.
- [12] W.Y. Yang, W. Cao, T. Chung, J. Morris. *Applied Numerical Methods Using MATLAB*. Wiley-IEEE, 2005.
- [13] I.J. Bahl, D.K. Trivedi. *A Designer's Guide to Microstrip Line, Microwaves*, 1977.

- [14] I.K. Peddle. *Autonomous Flight of a Model Aircraft*. Masters thesis, Stellenbosch University, 2005.
- [15] W.J. Hough. *Autonomous Aerobatic Flight of a Fixed Wing Unmanned Aerial Vehicle*. Masters thesis, Stellenbosch University, 2007.
- [16] S.C. Kriel. *A Comparison of Control Systems for the Flight Transition of VTOL Unmanned Aerial Vehicles*. Masters thesis, Stellenbosch University, 2008.
- [17] L. Rossouw. *Autonomous Flight of an Unmanned Helicopter*. Masters thesis, Stellenbosch University, 2008.
- [18] J. Venter. *Development of an Experimental Tilt-Wing VTOL Unmanned Aerial Vehicle*. Masters thesis, Stellenbosch University, 2006.
- [19] S. Groenewald. *Development of a Rotary-Wing Test Bed for Autonomous Flight*. Masters thesis, Stellenbosch University, 2006.
- [20] D. Blaauw. *Flight Control System for a Variable Stability Blended-Wing-Body Unmanned Aerial Vehicle*. Masters thesis, Stellenbosch University, To be submitted.
- [21] R. de Hart. *Advanced Take-off and Flight Control Algorithms for Fixed-Wing Unmanned Aerial Vehicles*. Masters thesis, Stellenbosch University, To be submitted.
- [22] M.C. Koen. *Modelling and Simulation of an RPV for Flight Control System Design Purposes*. Department of Electronic and Computer Engineering, University of Pretoria, 2006.
- [23] A. Budiyo. *Design and Development of Autonomous Uninhabited Air Vehicles at ITB: Challenges and Progress Status*. Paper presented at the Aerospace Indonesia Meeting, 2005.
- [24] Z. Sarris. *Survey of UAV Applications in Civil Markets*. Paper presented at Mediterranean Conference on Control and Automation, 2001.
- [25] M.C. Silberbauer. *Simulation Visualisation System*. Undergraduate Final Year Project, Stellenbosch University, 2005.
- [26] NASA History web page. <http://history.nasa.gov/SP-367/appendc.htm>. 2008.
- [27] Teal Group 2008 UAV Market Profile and Forecast. <http://www.tealgroup.com/content/view/31/16/>. 2008.