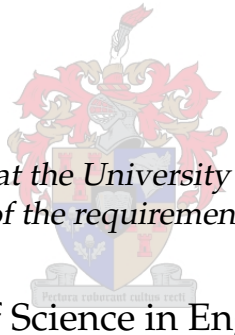
UNIVERSITEIT·STELLENBOSCH·UNIVERSITY
jou kennisvennoot • your knowledge partner

Full State Control of a Fury X-Cell Unmanned Helicopter

by

Carlo van Schalkwyk



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

Master of Science in Engineering

Department of Electronic Engineering
University of Stellenbosch
Matieland, South Africa

Study leaders:

Prof G Milne
Prof T Jones

December 2008

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.

Abstract

This thesis describes the successful development of an autopilot for an unmanned radio controlled helicopter. It presents a non-linear helicopter model. An adaptive linearised model is derived and used to design a controller. The adaptive full state controller is tested in various ways, including two aerobatic manoeuvres. A number of analyses are performed on the controller, including its robustness to parameter changes, noisy estimates, wind and processing power. The controller is compared with a non-adaptive counterpart, which leads to the conception, design and analysis of a much improved control structure. Practical flight test results are presented and analysed.

In some instances available literature was reworked and re-derived to produce a generic model-controller package that can easily be adapted for helicopters of any make, model and size.

Opsomming

Hierdie dokument beskryf die suksesvolle ontwikkeling van 'n outoloods vir 'n onbemande radiobeheer helikopter. Dit beskryf 'n nie-lineêre helikoptermodel en hoe 'n aanpasbare geliniariseerde weergawe van hierdie model ontwikkel is om die beheerder te ontwerp. Die aanpasbare voltoestandbeheerder word op verskillende maniere getoets, onder andere met twee aërobatiese maneuvers. Die beheerder se robuustheid teenoor parameter veranderinge, onakkurate afskattings en windsteurings word geanaliseer. Verder word die benodigde verwerkingskrag en ander praktiese kwessies bespreek. Die beheerder word vergelyk met 'n nie-aanpasbare beheerder. Die ontwerp en toetsing van 'n verbeterde beheertegniek word getoon. Praktiese vlugtoetsresultate word voorgelê.

Gedeeltes van die huidige literatuur word her afgelei om 'n generiese modelbeheerder pakket te vorm, wat maklik aangepas kan word om met enige model of grote helikopter te werk.

Acknowledgements

I would like to express my sincere gratitude to the following people and organisations who have contributed to making this multi-year project a great success:

- To the University of Stellenbosch ESL funding partners for providing much needed UAV funding that makes projects like this possible. Also to the National Research Foundation for providing personal funding, as well as funding in general for skills development in engineering.
- Nicol Carstens and Stephanus Groenewald, for their contribution to the initial development of helicopter platforms at the University.
- Our pilots, the late Mike Davis and Kyle Davis, for their advice, practical help and hours spent flying - usually at very short notice.
- All my colleagues and friends in the ESL, for your support and interest. Special thanks to Emile Rossouw, Steven Kriel, John Wilson and AM de Jager for your help, advice and enjoyable hours of intense arguments over ideas which eventually steered the course of this project.
- My parents, Carel and Louna van Schalkwyk, for your unequivocal love and support during the course of this project and many years before. Thank you Dad, for the many hours spent turning my ideas and designs into reality - almost always at the shortest possible notice.
- My first supervisor, the late Prof. Garth Milne. I have yet to find another man with such great vision, vast experience and unbelievable passion for what he does. He was a great engineer and an even better person - I would have loved for him to see this project succeed and being developed even further.
- My second supervisor , Thomas Jones, for taking the helm in difficult conditions. Thank you also for the excellent job you are doing at managing UAV activities at the University and securing sustainable funding for many more future projects.

To my loving sister - it would have been great to share this with you.

-

I miss you!

Contents

Nomenclature

Subscripts:

B	Body Aligned
BL	Main Blade
col	Collective
e	Engine Component
E	Earth Aligned
emp	Empennage
fus	Fuselage Component
g	Gravitational Component
lat	Lateral
$long$	Longitudinal
mr	Main Rotor Component
$rudd$	Rudder
SB	Stabiliser Bar
tr	Tail Rotor Component
i_*	Longitudinal Component (In Earth or Body Axis)
j_*	Lateral Component (In Earth or Body Axis)
k_*	Normal Component (In Earth or Body Axis)

Vehicle Specific Variables:

α_*	Blade Angle of Attack (Main Blade or Stabilizer Bar)
β_*	Instantaneous Blade Flapping Angle (Main Rotor or Stabilizer Bar)
β_{BL}^{const}	Main Blades Coning Angle
β_*^{lat}	Blade Lateral Flapping Angle (Main Rotor or Stabilizer Bar)
β_*^{long}	Blade Longitudinal Flapping Angle (Main Rotor or Stabilizer Bar)
C_*^T	Blade Lift Coefficient (Main Rotor or Tail Rotor)
C_{Q_*}	Rotor Drag Coefficient (Main Rotor or Tail Rotor)
δ_{col}	Collective Actuator

δ_{lat}	Lateral Actuator
δ_{long}	Longitudinal Actuator
δ_{rudd}	Rudder Actuator
F_I^i, F_B^i	Skid Contact Point "i" Runway Forces in Inertial and Body Frame Respectively
γ_*	Rotor Lock Number (Main Rotor or Stabilizer Bar)
λ_*	Rotor Induced Airflow Velocity, Normalised with Blade Tip Speed (Main Rotor or Tail Rotor)
K_λ	Factor of Main Rotor Airflow Exposure to Tail Rotor
L_*	Roll Moment Component
M_B^i	Skid Contact Point "i" Runway Moments
μ_*	In-plane Rotor Approaching Airflow Velocity, Normalised With Blade Tip Speed (Main Rotor or Tail Rotor)
μ_*^z	Normal Rotor Approaching Airflow Velocity, normalised With Blade Tip Speed (Main Rotor or Tail Rotor)
M_*	Pitch Moment Component
N_*	Yaw Moment Component
Ω_*	Rotor Speed (Main Rotor or Tail Rotor)
$\dot{\Omega}_{k_B}^{shaft}$	Main Rotor Shaft Acceleration in Body Coordinates
ϕ_w	Angle of Vertical Approaching Airflow Component to Blade Normal
ψ_w	Angle of Horizontal Approaching Airflow Component to Blade Normal
ψ_*	Instantaneous Blade Rotation Angle (Main Blade or Stabilizer Blade)
Q_{e_*}	Rotor Shaft Torque (Main Rotor or Tail Rotor)
S_β	Main Blade Stiffness Number
τ_*	Rotor Flapping Time Constant (Main Rotor or Stabilizer Bar)
θ_*	Instantaneous Blade Pitch Angle (Main Blade or Stabilizer Blade)
T_*	Rotor Thrust (Main Rotor or Tail Rotor)
u_a	Longitudinal Body Orientated Nett Airflow
u_w	Longitudinal Body Orientated Wind Component
$V_{*_{induced}}$	Rotor Induced Airflow Velocity (Main Rotor or Tail Rotor)
v_a	Lateral Body Orientated Nett Airflow
v_w	Lateral Body Orientated Wind Component
V_{∞_*}	Total Incoming Airflow Velocity
w_a	Normal Body Orientated Nett Airflow
w_{tr}	Vertical Component of Tail Rotor In-Plane Airflow Velocity
w_w	Normal Body Orientated Wind Component
X_*	Body Orientated Longitudinal Force Component
Y_*	Body Orientated Lateral Force Component

Z_*	Body Orientated Normal Force Component

Vehicle Specific Parameters:

a_*	Slope of Lift Curve (Main Blade, Tail Blade or Stabilizer Bar)
c_*	Blade Chord Length (Main Blade, Tail Blade or Stabilizer Bar)
$C^*_{D_0}$	Blade Profile Drag Coefficient (Main Blade, Tail Blade or Stabilizer Bar)
CF_{mr}	Main Rotor Wake Contraction Factor
η_w	Main Rotor Airflow Efficiency
f_t	Tail Rotor Thrust Blocking Factor
G_{BL}	Swash Plate to Main Blade Pitch Gain
h_*	Rotor Position Above CG (Main Rotor or Tail Rotor)
$\boldsymbol{I}$	Moment of Inertia Tensor
$I_{B_{mr}}$	Single Main Blade Axial Inertia
$I_{B_{tr}}$	Single Tail Blade Axial Inertia
I_{eng}	Axial Inertia of Engine Rotating Parts
I_{rot}	Effective Axial Inertia of All Rotating Parts, Translated to the Main Rotor Shaft
I_{xx}, I_{yy}, I_{zz}	Principle Moment of Inertia (About x, y or z Axis Respectively)
K_β	Hub Torsional Stiffness
K_{SB}	Stabilizer Bar Flapping Angle to Mail Blade Pitch Gain
K_{gov}	Engine Governor Gain
l_*	Rotor Position Behind CG (Main Rotor or Tail Rotor)
R_{SB2}, R_{SB1}	Stabilizer Bar Paddle Outer Radius and Inner Radius Respectively
R_*	Rotor Radius (Main Rotor or Tail Rotor)
S_{vf}	Effective Area of Vertical Stabilizer Fin
S_{ht}	Effective Area of Horizontal Stabilizer Fin
S^{fus}_*	Fuselage Drag Area (in x, y or z Axis)
σ_*	Blade Solidity Ratio (Main Rotor or Tail Rotor)
τ_{eng}	Engine Time Constant

Natural Constants:

g	Earth's Gravitational Acceleration
ρ	Air Density
R_e	Earth's Mean Radius

Position and Rotation Variables:

a_x, a_y, a_z	Forward, Lateral and Normal Body Accelerations Respectively
$\alpha_x, \alpha_y, \alpha_z$	Roll, Pitch and Yaw Angular Accelerations Respectively
D	Down Inertial Runway Offset
E	East Inertial Runway Offset
N	North Inertial Runway Offset
ϕ	Euler 321 Roll Orientation Angle
θ	Euler 321 Pitch Orientation Angle
ψ	Euler 321 Yaw Orientation Angle
ψ_I	Integrated Euler 321 Yaw Orientation Angle Error
p	Roll Rate
p_{int}	Integrated Roll Rate Error
q_1, q_2, q_3, q_4	Quaternions
q	Pitch Rate
q_{int}	Integrated Pitch Rate Error
r	Yaw Rate
r_{int}	Integrated Yaw Rate Error
$\boldsymbol{T}$	Inertial to Body Frame Transformation Matrix
u	Forward Body Translation Velocity
u_{int}	Integrated Forward Body Translation Velocity Error
v	Lateral Body Translation Velocity
v_{int}	Integrated Lateral Body Translation Velocity Error
w	Normal Body Translation Velocity
w_{int}	Integrated Normal Body Translation Velocity Error
X	Forward Body Position Offset
X_I	Integrated Forward Body Position Offset
Y	Lateral Body Position Offset
Y_I	Integrated Lateral Body Position Offset
Z	Normal Body Position Offset
Z_I	Integrated Normal Body Position Offset

Controller Variables:

$\boldsymbol{A}$	Continuous State Space System Matrix
$\boldsymbol{B}$	Continuous State Space Input Matrix
$\boldsymbol{\Gamma}$	Discrete State Space Input Matrix
$\boldsymbol{\Gamma_A}$	Discrete State Space Input Matrix, with Augmented Integrator States
$\boldsymbol{J}$	Total Cost of Cost Function

$\boldsymbol{M}$	Input-State Cross Weighting for Cost Function Matrix
$\boldsymbol{\Phi}$	Discrete State Space System Matrix
$\boldsymbol{\Phi_A}$	Discrete State Space System Matrix, with Augmented Integrator States
$\boldsymbol{Q}$	State Weighting for Cost Function Matrix
$\boldsymbol{R}$	Input Weighting for Cost Function Matrix

Acronyms:

6DOF	6 Degrees of Freedom
bps	Bits per Second
BR	Barrel Roll
CAN	Controller Area Network
CG	Centre of Gravity
CLC	Consecutive Loop Closure
CPLD	Complex Programmable Logic Device
DCM	Direct Cosine Matrix
DGPS	Differential Global Positioning System
EKF	Extended Kalman Filter
ESL	Electronic Systems Laboratory
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GUI	Graphical User Interface
HIL	Hardware in the Loop
IEEE	Institute for Electrical and Electronic Engineers
IMU	Inertial Measurement Unit
ISA	Industry Standard Architecture (Communication Bus)
LiPo	Lithium Polymer
LTI	Linear Time Invariant
LQR	Linear Quadratic Regulator
MIMO	Multi-Input Multi-Output
MIT	Massachusetts Institute Of Technology
NASA	National Aeronautics and Space Administration
NiCad	Nickel Cadmium
OBC	Onboard Computer
PC-104	Embedded Computer Standard Controlled by the PC/104 Consortium
PCB	Printed Circuit Board
RC	Radio Controlled

RPM	Revolutions per Minute
RTCM	Radio Technical Commission for Maritime Services
SISO	Single Input Single Output
ST	Stall Turn
TPP	Tip Path Plane
VT	Vertical Takeoff and Landing
UAV	Unmanned Aerial Vehicle

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

In recent years the commercial and military potential for unmanned aerial vehicles (UAVs) has soared. With the addition of cheaper, more powerful electronics, better sensors and more affordable vehicles, these systems constitute the key research areas of a large number of universities worldwide.

While fixed wing implementations offer great speed, efficiency, endurance and payload capabilities, rotary wing aircraft offer the advantage of stationary hovering, precision manoeuvring and vertical take-off and landing (VTOL). Unfortunately these vehicles are generally unstable and difficult to control. This project aims to address this problem by designing a fully autonomous system that can be flown without any ground based assistance.

1.2 Background

Research into the automation of rotor vehicles at the ESL (Electronic Systems Laboratory of the University of Stellenbosch) began in 1996 with a probe by Mooren [1] into the feasibility of building a counter rotating, coaxial mounted, dual rotor vehicle. In 1998 a prototype was built and tested by Calitz [2], but the project was abandoned due to mechanical problems.

Acting on Mooren's initial recommendation that a quad rotor design would be more feasible, Carstens [3] attempted to build such a vehicle in 2003. His attempt was later abandoned in favour of an off the shelf, JR Voyager E, electrically powered helicopter. This was the first tail rotor design and signalled the start of the automation of the helicopters at the ESL. Carstens fitted the vehicle with an array of sensors and microcontrollers. As the vehicle had a limited payload capability, a telemetry link was added to relay sensor data to a ground based computer, on which control laws were executed. Carstens successfully regulated vehicle heading, altitude and longitudinal motion.

After the proven success of the helicopter platform, an X-Cell Model 60 radio controlled helicopter, with a 70 size methanol engine was purchased in 2004. This larger methanol powered platform addressed the limited payload issues and allowed the addition of on-board processing, with the goal of developing a fully independent autonomous system. In the same year Groenewald [4] started to adapt the vehicle for autonomous flight. He fitted the X-Cell with a 300 MHz PC-104, Intel-based on-board computer (OBC). A CAN-based communication bus architecture was conceived to interlink different electronic components on the vehicle. Groenewald developed the *CANSens board* which samples signals from low cost 3-axis gyros, accelerometers and magnetometers and passes on this data via the CAN bus, to the OBC. He also integrated an ultrasonic range sensor.

The CAN system was further expanded by Venter [5], through the *PC104 CAN* system, as well as the *Servo CAN* controller board. The PC104 CAN system interfaces the CAN bus with the OBC, through the OBC's ISA bus, while the Servo Controller interfaces the servos and remote control signals with the CAN bus. In 2006, Hough [6], while working on a fixed wing UAV project, extended the CAN hardware by developing a *hardware in the loop interface board* to be used for simulations.

The combination of all these electronic systems created a platform well suited for the start of this project. In 2006, together with Emile Rossouw [7], the goal was set to fully automate the X-Cell as part of two separate Masters theses under the guidance of Prof. Garth Milne. After the untimely death of Prof. Milne in 2007, the project was overseen by Prof. Thomas Jones.

1.3 Task Description and Strategy

The main goal of this project was set to design a control system to automate an X-Cell radio controlled helicopter. The control system had to cover a large area of the flight envelope, including hover, high speed forward flight, backwards flight, tight turns and high climb/descent rates. Additionally the system should provide for fail safe control by any inexperienced human pilot.

The controller should be robust, with high margins of stability, while still being easily adaptable to new manoeuvres and flight conditions. The feasibility of using the system to perform various aerobatic manoeuvres should also be explored.

To achieve these goals, an adaptive full state optimal control strategy would be used, together with an acceleration limited pilot interface. For controller verification, extensive testing would be done using *hardware in the loop* (HIL) simulations. Final verification would be done through practical flight tests.

1.4 Achievements

The primary project goal of designing a robust and versatile controller for a radio controlled helicopter was successfully achieved and partially verified by practical flight tests. In achieving this goal the following additional achievements were realised:

- The existing helicopter platform was partially redesigned to achieve a 40% weight saving of the flight avionics payload. The vehicle's engine, blades and landing skid system were upgraded for additional performance, safety and flight stability.
- The on-board sensors were calibrated and tested, together with the vehicle's swash plate, tail rotor and gyro.
- The existing X-Cell model was re-derived from first principles and changes were made to produce a general genetic non-linear model that can be easily adapted for any small to medium scale helicopter platforms. Many additions were made, including a runway, ground effect and wind model.
- The vast non-linear model was linearised with great effort and discretised into a 16-state model. Both were coded in MATLAB and C.
- A full non-linear hardware in the loop simulation was constructed to facilitate proper controller testing.
- A full state adaptable optimal controller was designed. The controller was extensively tested in HIL and a vast array of stability analyses was performed to verify controller performance.
- Two aerobatic manoeuvres (a 180° stall turn and a barrel roll) were constructed and successfully tested in HIL.
- A ground control station was designed and the controller was partially flight tested with very positive results.
- An alternative to the traditional LQR control structure was designed and simulated, with very positive results.

1.5 Document Outline

This thesis describes the process to design and implement a control system for a helicopter. **Chapter 2** gives full insight into the non-linear dynamics of helicopters and details the process to derive such a model, given a specific airframe. A model linearised around hover trim conditions will be listed and the relevant dynamics during hover will be discussed and analysed.

Chapter 3 evaluates different control methodologies considered for this project and describes the concept of optimal control. It shows the process of designing an

optimal controller and setting up trajectories. The implications of practical implementation of LQR are discussed and full parameter and estimator sensitivity analysis is done and discussed. Finally a novel alternative control structure is derived, analysed and discussed.

Chapter 4 discusses the advantages of using an adaptive controller and evaluates it by comparing results with those of a non-adapting classical counterpart. It discusses the HIL process and how it is used to evaluate and approve a UAV for flight testing. HIL results are listed and evaluated, together with results from practical flight tests.

Conclusions and recommendations are discussed in **Chapter 5**.

The **Appendices** list additional information on the corresponding chapters, together with mathematical derivations and formulas. They also list complete simulation results and analysis data for a large range of flight conditions.

Chapter 2

Modelling

Several institutions (see [8], [9] and [10]) have developed models for small scale helicopters. Unfortunately all of these models are model/size specific, as they make use of lumped parameters. These lumped parameters must be identified for a specific helicopter through a number of expensive and time-consuming methods. These include wind tunnel testing, practical flight tests and sting balance[1] experiments.

Padfield [11] and Bramwell [12] presented methods on the complete modelling of full sized helicopters. However, these helicopters do not make use of a stabilizer bar. This Chapter will detail how the modelling techniques of full scale helicopters are applied to model small scale helicopters from first principles. It will also list the complete formulas for the lumped parameters that are used in other small scale models. The model of Gavrilets [10] was used where possible and complementary modelling was done where needed.

2.1 Overview

The aim of this chapter is to give the reader insight into the non-linear mathematical model used to describe the dynamics of a small scale helicopter. It will describe the entire process needed to obtain a vehicle model which can be used in the simulation and controller design of such a vehicle. Sections of the vehicle model are based on work done at MIT by Gavrilets et al. [10]. Alternative modelling methods were used where the Gavrilets modelling was insufficient for application to this project. A runway and ground effect model are derived. This model is assumed valid for a flight envelope with an advanced ratio[2] of less than 0.15.

[1] A device used for measuring forces and moments produced by a body.

[2] The advance ratio of a helicopter is defined as the vertical translational airspeed as a factor of blade tip speed - $\mu_{mr} = \frac{\sqrt{(u-u_w)^2+(v-v_w)^2}}{\Omega_{mr} R_{mr}}$.

2.2 Axis Systems

Similar to most other self propelled vehicles, helicopters use body fixed actuators to produce forces and moments, enabling them to fly. By their nature, these effects tend to be aligned with the body of the vehicle. External effects, such as gravity, gyroscopic and centrifugal forces, act upon any mass, irrespective of their body orientation and are best described in an earth-fixed or inertial coordinate system. Thus the need arose for the use of more than one axis system.

2.2.1 Body Reference Frame

The body axis system is defined with its origin at the vehicle's centre of mass and stays fixed to the vehicle (rotates and translates with the vehicle). The positive x-axis is defined as pointing forward (parallel with the tail). The positive z-axis points vertically downward, parallel with the main rotor drive shaft. The positive y-axis points to the right of the vehicle, perpendicular to the plane that spans the x- and z-axis. Figure 2.1 shows the axes definitions, as well as the positive rotation directions about these axes. The unit vectors in the X, Y and Z body axes are defined as $\boldsymbol{i}_B$, $\boldsymbol{j}_B$ and $\boldsymbol{k}_B$ respectively.

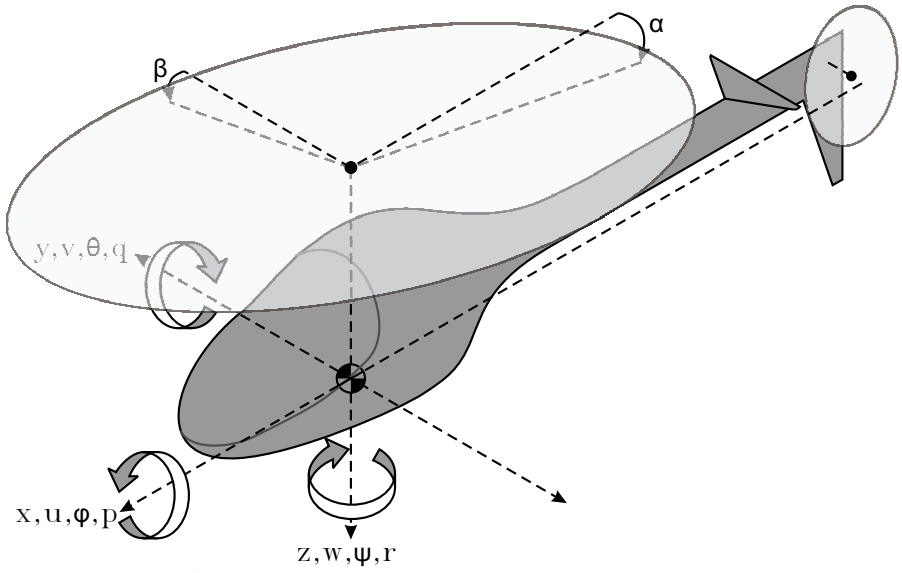


Figure 2.1: Body Axis System Definition

2.2.2 Earth Reference Frame

To simplify the modelling equations, the earth is assumed to be flat, non-rotating and non-translating. Since the vehicle will not fly long distances from the take-off position, this is a reasonable assumption. Flight will also be limited to relatively

low altitudes and relatively low velocities. These assumptions allow for the approximation that the earth fixed frame is inertially aligned, so there would be no need to distinguish between the two.

The earth reference frame is defined with its origin at some convenient location (normally on the runway). Its positive x-axis is defined as pointing North from this point, while the positive y-axis points East. The z-axis points vertically downwards and is perpendicular to the plane that spans the x- and y-axis. Figure 2.2 illustrates these axes. Unit vectors in the X, Y and Z earth axes are defined as $\boldsymbol{i}_E$, $\boldsymbol{j}_E$ and $\boldsymbol{k}_E$ respectively.

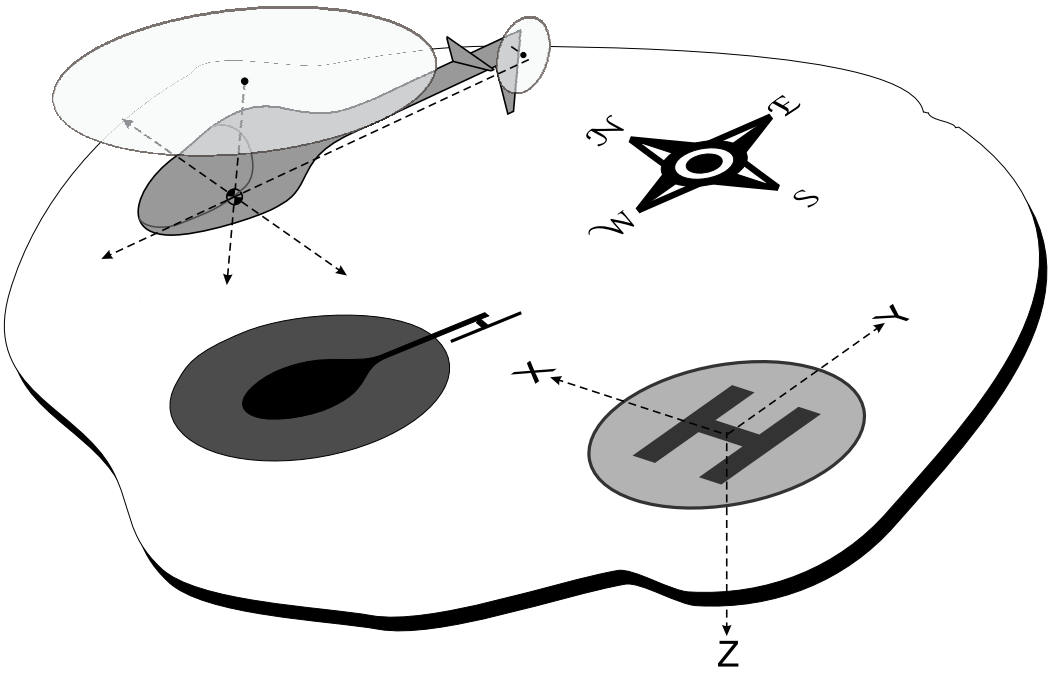


Figure 2.2: Earth Axis System Definition

2.3 Attitude Description

There are many different methods by which an object's attitude in inertial space can be described. Some of these representations are referred to as *Leonhard Euler's angles*. This method describes an object's attitude by way of 3 rotations (one about each axis of the reference frame) [13]. Euler does not specify in which order these rotations need to occur - this results in a total of 6 different methods with which attitude can be described using Euler angles. It will be shown that when using these angles to convert inertially specified quantities to body coordinated quantities, each Euler method has a discontinuity at two attitudes.

To avoid the Euler discontinuities, the use of a different method to specify attitude is needed. One such method is called *Quaternions*. Quaternions define atti-

tude by rotating the reference frame about a unit vector defined in inertial space. Although the method stays the same, there are different ways to represent the rotations. One such method is shown:

$$\begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \begin{bmatrix} e_x \sin(\frac{\mu}{2}) \\ e_y \sin(\frac{\mu}{2}) \\ e_z \sin(\frac{\mu}{2}) \\ \cos(\frac{\mu}{2}) \end{bmatrix} \tag{2.3.1}$$

where e_x, e_y, e_z denote the components of the unit vector coordinated in inertial space and μ is the rotation angle of the reference frame about this vector. Due to their non-linear nature, one problem with the Quaternion representation is that the terms q_{1-4} differ in sensitivity with a change in attitude. This creates a problem in non-adaptive control systems as feedback from these quaternions varies with sensitivity, as the attitude changes.

Although the proposed control system for this project is adaptive in nature, the weightings used to calculate the cost of errors of quaternions will vary in sensitivity at different attitudes. For this reason it was decided to use the *Euler 3-2-1* angle representation in the controller. The discontinuity of this representation is at $\pm 90^\circ$ in pitch, which is outside the proposed normal flight envelope of the vehicle. Quaternions are used throughout the rest of the system, so if the flight envelope should ever enter this region of discontinuity, it is proposed that there be switched to a different Euler representation. This method was implemented for one of the aerobatic manoeuvres and works in conjunction with the normal Euler representation, so that at every single attitude at least one representation is valid.

2.4 Kinematics

Kinematics describes how the position of an object changes over time, without consideration of forces, moments, mass and inertia. Translational kinematics describes linear movement using position, velocity and acceleration, where rotational kinematics describes angular movement using attitude, angular velocity and angular acceleration.

Most sensors on the vehicle are body orientated and fixed, thus the majority of the kinematic states were chosen to be orientated in the body axis to simplify the estimation process. Vehicle position is denoted by N, E and D, measured as the offsets from the earth frame origin and coordinated in this frame. These offsets can also be coordinated into the body axis and are then denoted as X, Y and Z. Translational velocity and acceleration are measured with respect to the Earth and inertial origins respectively, but are coordinated in the body axes as u, v, w and a_x, a_y, a_z. Angular velocity and acceleration are both coordinated in the body axis system and are denoted by p, q, r and α_x, α_y, α_z respectively.

Special care has to be taken when calculating the differential equations of states that are coordinated in a rotating axis system such as the body reference frame, because of the Corriolis effect. Corriolis [14] stated that when a coordinate system is rotating with respect to the inertial frame, a fictitious force arises when differentiating vectors that are coordinated in this frame. Corriolis characterized the

relationship between a derivative in the inertial frame and one in a rotating frame as:

$$\left[\left(\frac{\partial}{\partial t}\right)_I \boldsymbol{R}\right]_B = \left[\left(\frac{\partial}{\partial t}\right)_B \boldsymbol{R}\right]_B + \boldsymbol{\omega}_B \times [\boldsymbol{R}]_B \tag{2.4.1}$$

where $\boldsymbol{R}$ is an arbitrary vector and $\boldsymbol{\omega}_B$ is the rate of rotation of the rotating frame (in this example the body reference frame). $\left(\frac{\partial}{\partial t}\right)_I$ denotes the time derivative with respect to the inertial frame and $[\,]_B$ denotes that the encapsulated value is coordinated in the body reference frame.

If a vector can be coordinated in both the body and earth axes, there exists a unique transformation matrix ($\boldsymbol{T}$) [13], so that:

$$\begin{aligned} [\boldsymbol{V}]_B &= \boldsymbol{T}\,[\boldsymbol{V}]_E \\ [\boldsymbol{V}]_E &= \boldsymbol{T}^{-1}\,[\boldsymbol{V}]_B \end{aligned}$$

$\boldsymbol{T}$ is a function of the difference in attitude between the axis systems and is referred to as the *direct cosine matrix*, or *DCM*. Transformation matrices are orthogonal [13] and thus $\boldsymbol{T}^{-1} = \boldsymbol{T}^T$. For Euler 3-2-1 and Quaternions the DCM follows:

$$\boldsymbol{T}(\phi,\theta,\psi) = \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\cos\theta \\ \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\cos\theta \end{bmatrix}_{321} \tag{2.4.2}$$

$$\boldsymbol{T}(q_1,q_2,q_3,q_4) = \begin{bmatrix} q_4^2+q_1^2-q_2^2-q_3^2 & 2(q_4q_3+q_1q_2) & 2(q_1q_3-q_4q_2) \\ 2(q_1q_2-q_4q_3) & q_4^2-q_1^2+q_2^2-q_3^2 & 2(q_4q_1+q_2q_3) \\ 2(q_4q_2+q_1q_3) & 2(q_2q_3-q_4q_1) & q_4^2-q_1^2-q_2^2+q_3^2 \end{bmatrix} \tag{2.4.3}$$

The following equations list the differential equations for the kinematics of a rigid body (note that the equations for attitude only holds for the Euler 3-2-1 con-

version):

$$\begin{bmatrix} \dot{N} \\ \dot{E} \\ \dot{D} \end{bmatrix} = \boldsymbol{T}^T \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{2.4.4}$$

$$\boldsymbol{T} \begin{bmatrix} \dot{N} \\ \dot{E} \\ \dot{D} \end{bmatrix} = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} + \boldsymbol{\omega}_B \times \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = \begin{bmatrix} u \\ v \\ w \end{bmatrix} + \begin{bmatrix} rY - qZ \\ pZ - rX \\ qX - pY \end{bmatrix} \tag{2.4.5}$$

$$\boldsymbol{T} \begin{bmatrix} \ddot{N} \\ \ddot{E} \\ \ddot{D} \end{bmatrix} = \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} + \boldsymbol{\omega}_B \times \begin{bmatrix} u \\ v \\ w \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} a_{\boldsymbol{x}} \\ a_{\boldsymbol{y}} \\ a_{\boldsymbol{z}} \end{bmatrix} + \begin{bmatrix} rv - qw \\ pw - ru \\ qu - pv \end{bmatrix} \tag{2.4.6}$$

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}_{321} = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix}_{321} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2.4.7}$$

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \alpha_{\boldsymbol{x}} \\ \alpha_{\boldsymbol{y}} \\ \alpha_{\boldsymbol{z}} \end{bmatrix} \tag{2.4.8}$$

2.5 Dynamics

The dynamics of a rigid body can be described by the law of conservation of momentum and angular momentum. In their general form they state that:

$$\vec{F} = \frac{\partial\left(m\vec{v}\right)}{\partial t} \tag{2.5.1}$$

$$\vec{N} = \frac{\partial\left(\boldsymbol{I}\vec{\omega}\right)}{\partial t} \tag{2.5.2}$$

$$\boldsymbol{I} = \begin{bmatrix} I_{i_B i_B} & I_{i_B j_B} & I_{i_B k_B} \\ I_{j_B i_B} & I_{j_B j_B} & I_{j_B k_B} \\ I_{k_B i_B} & I_{k_B j_B} & I_{k_B k_B} \end{bmatrix} \tag{2.5.3}$$

where $\vec{F}$ is the force vector acting in on a mass m and is coordinated in the same axis as the velocity vector $\vec{v}$. $\vec{N}$ is the torque vector acting in on an object with $\boldsymbol{I}$ inertia in the axis of rotation, described by vector $\vec{\omega}$.

Equations (2.5.1) and (2.5.2) can now be applied to the differential equations of

weightless translational and angular velocity in (2.4.6) and (2.4.8):

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \frac{1}{m} \begin{bmatrix} X_F \\ Y_F \\ Z_F \end{bmatrix} + \begin{bmatrix} rv - qw \\ pw - ru \\ qu - pv \end{bmatrix} \tag{2.5.4}$$

$$\begin{aligned} \left[\left(\frac{\partial}{\partial t} \right)_I \boldsymbol{I\omega} \right]_B &= \left[\left(\frac{\partial}{\partial t} \right)_B \boldsymbol{I\omega} \right]_B + \boldsymbol{\omega}_B \times \boldsymbol{I} \left[\boldsymbol{\omega} \right]_B \\ \Rightarrow \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} &= \boldsymbol{I}^{-1} \begin{bmatrix} L \\ M \\ N \end{bmatrix} - \boldsymbol{I}^{-1} \left(\boldsymbol{\omega}_B \times \boldsymbol{I\omega}_B \right) \end{aligned} \tag{2.5.5}$$

where X_F, Y_F, Z_F and L, M, N are the forces and moments applied to the body respectively. It can be seen in Equation (2.5.5) that when the system is no longer assumed weightless, an additional cross coupling term is introduced namely the gyroscopic effect. If the vehicle is assumed to be symmetrical in mass about the rotation axis, the moment of inertia reduces to the principle moment of inertia:

$$\boldsymbol{I} = \begin{bmatrix} I_{i_B i_B} & 0 & 0 \\ 0 & I_{j_B j_B} & 0 \\ 0 & 0 & I_{k_B k_B} \end{bmatrix} \tag{2.5.6}$$

Equation (2.5.5) can then be written as:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \frac{L+qr\left(I_{j_B j_B} - I_{k_B k_B} \right)}{I_{i_B i_B}} \\ \frac{M+pr\left(I_{k_B k_B} - I_{i_B i_B} \right)}{I_{j_B j_B}} \\ \frac{N+pq\left(I_{i_B i_B} - I_{j_B j_B} \right)}{I_{k_B k_B}} \end{bmatrix} \tag{2.5.7}$$

The rest of this chapter will describe all the forces and moments that act upon the vehicle.

2.5.1 Gravitational

Gravity acts in the positive z direction, towards the centre of the earth. To model this as a body orientated force, the DCM is applied:

$$\begin{bmatrix} X_g \\ Y_g \\ Z_g \end{bmatrix} = \boldsymbol{T} \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \tag{2.5.8}$$

For Euler 321, this becomes:

$$\begin{bmatrix} X_g \\ Y_g \\ Z_g \end{bmatrix} = mg \begin{bmatrix} -\sin\theta \\ \sin\phi\cos\theta \\ \cos\phi\cos\theta \end{bmatrix} \tag{2.5.9}$$

2.5.2 Main Rotor

The main rotor blades of the vehicle act as its primary source of propulsion and are, in conjunction with the swash plate, used for steering. As a result, they account for most of the dynamics of the vehicle and consequently the utmost care should be taken when modelling them.

To ensure that the smallest number of errors are introduced and the least assumptions are made in their modelling process, the blades were analysed in their elementary form. The process is called blade element theory and begins with a description of the dynamics of a blade element (usually a cross section of the blade). These dynamics are then added to account for the whole blade(s). Figure 2.3 illustrates blade element theory as well as the terminology used throughout the process.

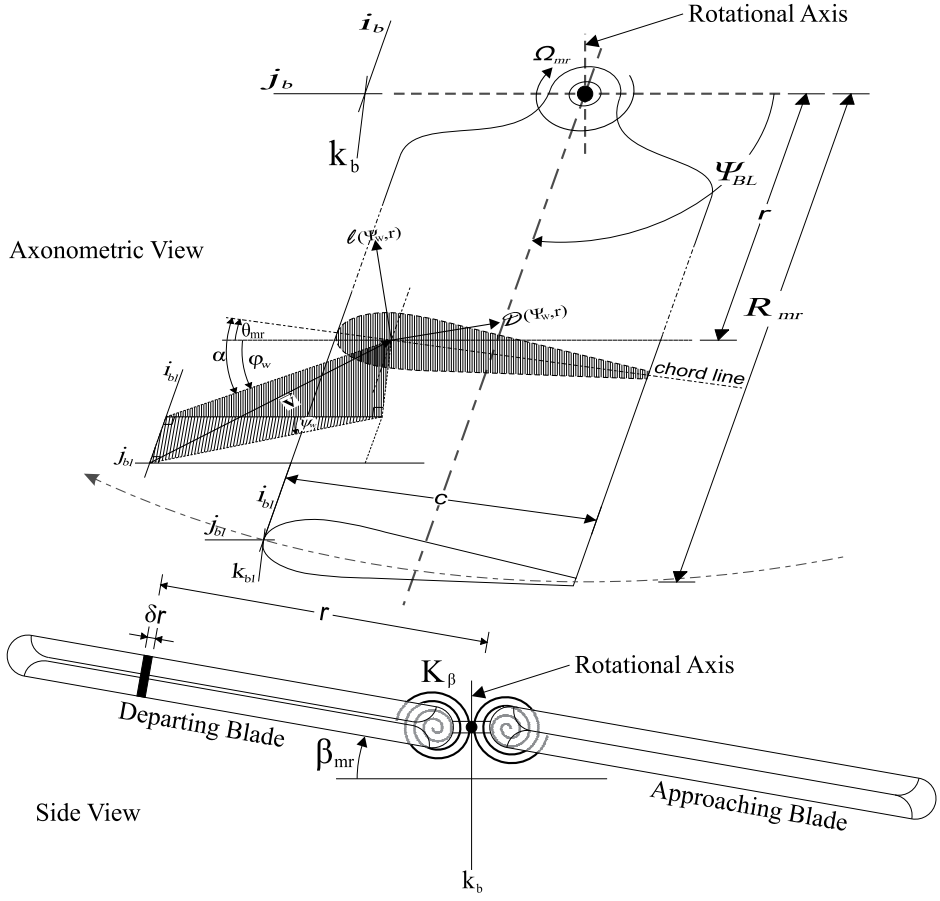


Figure 2.3: Main Blade Element Theory

Where the rotating blade axis system is defined by $\boldsymbol{i}_{bl}$ (always points along the length of the blade, inwards, towards the rotation centre), $\boldsymbol{j}_{bl}$ (always points toward the direction of rotation) and $\boldsymbol{k}_{bl}$ (normal to the rotating plane).

c is the chord length of the blade, r is the distance of the section from the rotation axis and R is the total radius per blade. θ_{mr} denotes the collective pitch angle of the blade (δ_{col}, measured from the rotating plane), while ϕ_w denotes the angle of the incoming airflow's perpendicular component to the rotating plane (in $\boldsymbol{j}_{bl}$ and $\boldsymbol{k}_{bl}$ axis).

The angle at which the in-plane non-rotating airflow approaches the blade is denoted by ψ_w. Together δ_{col} and ϕ_w sum to form the *angle of attack*, α, which is the angle from the blade chord to the incoming airflow. The angle of attack (α) and perpendicular airflow (V) can be rewritten as:

$$\alpha = \delta_{col} + \arctan\left(\frac{w - w_w - V_{mr_{induced}}}{\Omega_{mr} r + \left(\sqrt{(u-u_w)^2 + (v-v_w)^2}\right)\cos(\psi_w)}\right) \tag{2.5.10}$$

$$V = \sqrt{(w - w_w - V_{mr_{induced}})^2 + \left(\Omega_{mr} r + \left(\sqrt{(u-u_w)^2 + (v-v_w)^2}\right)\cos(\psi_w)\right)^2} \tag{2.5.11}$$

There is a general convention to scale all airflow related parameters with respect to the rotor tip speed ($\Omega_{mr} R_{mr}$). This makes helicopters of various sizes easily scalable and comparable:

$$\lambda_{mr} = \frac{V_{mr_{induced}}}{\Omega_{mr} R_{mr}} \tag{2.5.12}$$

$$\mu_{mr} = \frac{\sqrt{(u-u_w)^2 + (v-v_w)^2}}{\Omega_{mr} R_{mr}} \tag{2.5.13}$$

$$\mu^z_{mr} = \frac{(w - w_w)}{\Omega_{mr} R_{mr}} \tag{2.5.14}$$

If the assumption is now made that the horizontal component of the perpendicular airflow is much greater that the vertical component for the majority of the radius r, i.e. $(\Omega_{mr} r + \Omega_{mr} R_{mr} \mu_{mr} \cos(\psi_w)) >> \Omega_{mr} R_{mr} (\mu^z_{mr} - \lambda_{mr})$, Equations (2.5.10) and (2.5.11) can then be approximated as:

$$\alpha \approx \delta_{col} + \frac{\Omega_{mr} R_{mr} (\mu^z_{mr} - \lambda_{mr})}{\Omega_{mr} r + \Omega_{mr} R_{mr} \mu_{mr} \cos(\psi_w)} \tag{2.5.15}$$

$$V \approx \Omega_{mr} r + \Omega_{mr} R_{mr} \mu_{mr} \cos(\psi_w) \tag{2.5.16}$$

According to [15], the lift of an airfoil can be calculated as:

$$L = \frac{1}{2} \rho V^2 A C_{L_\alpha} \alpha \tag{2.5.17}$$

where V is the incoming airflow (perpendicular to the span of the wing) and A the area of the wing (approximated by the product of the chord length and the length of the wing). The air density is denoted by ρ, while C_{L_α} is the slope of the lift curve and α denotes the angle between the chord of the wing and the incoming airflow, perpendicular to the span of the wing.

If the helicopter blade is approximated to be non-twisting, Padfield [11] suggests that Equation (2.5.17) can be used to calculate the total thrust produced by the blades. Applying Equation (2.5.17) to the small section of blade described in Figure 2.3:

$$l(r, \psi_w) = \frac{1}{2}\rho V^2_{j_{bl}\boldsymbol{k}_{bl}} a_{mr} \alpha c_{mr} \delta r \cos(\beta_{BL}) \tag{2.5.18}$$

where a_{mr} is the gradient of the linear part of the lift curve. Figure 2.4 shows a typical lift coefficient curve for a blade/wing of similar shape, with the relatively linear region shown between $\pm 10^\circ$.

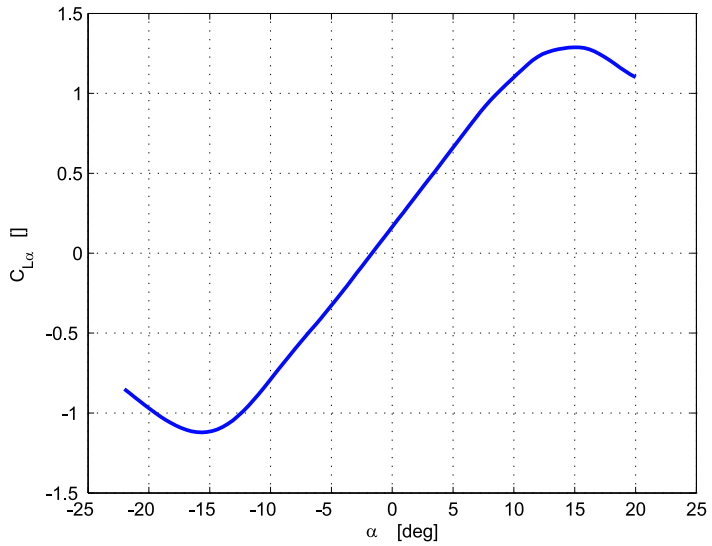


Figure 2.4: Lift Curve of a NACA 24 Airfoil, Reproduced from Abbott & von Doenhoff [16]

To calculate the total lift produced by a blade, the function in Equation (2.5.18) needs to be integrated over the entire length of the blade. This will give the total instantaneous thrust as a function of the offset wind angle ψ_w. An average is obtained by integrating the thrust over all possible offset angles from 0 to 2π and then normalising it to 2π:

$$\begin{aligned} T_{mr}(n) \quad = \quad & n\frac{\rho a_{mr} c_{mr}}{4\pi} \int_0^{2\pi} \int_0^{R_{mr}} (\Omega_{mr} r + \Omega_{mr} R_{mr} \mu_{mr} \cos(\psi_w))^2 \\ & \left(\delta_{col} + \frac{\Omega_{mr} R_{mr} (\mu^z_{mr} - \lambda_{mr})}{\Omega_{mr} r + \Omega_{mr} R_{mr} \mu_{mr} \cos(\psi_w)} \right) \cos(\beta_{BL}) \, \delta r \delta \psi_w \end{aligned} \tag{2.5.19}$$

If the flapping angle of the blade is assumed small, $\cos(\beta_{BL})$ can be approximated as being 1, so that Equation (2.5.19) simplifies to:

$$T_{mr}(n) = n\frac{1}{2}\rho(\Omega_{mr}R_{mr})^2\pi R_{mr}^2\left(\frac{2c_{mr}}{\pi R_{mr}}\frac{a_{mr}}{2}\left(\delta_{col}\left(\frac{1}{3}+\frac{\mu_{mr}^2}{2}\right)+\frac{\mu_{mr}^z-\lambda_{mr}}{2}\right)\right) \tag{2.5.20}$$

Again some of the states can be grouped and rewritten, which will make for better parameters that can be easily compared between different vehicles:

$$\sigma_{mr} = \frac{2c_{mr}}{\pi R_{mr}} \tag{2.5.21}$$

$$C_{T_{mr}} = \frac{\sigma_{mr}a_{mr}}{2}\left(\delta_{col}\left(\frac{1}{3}+\frac{\mu_{mr}^2}{2}\right)+\frac{\mu_{mr}^z-\lambda_{mr}}{2}\right) \tag{2.5.22}$$

This novel choice of variables enables the azimuth-averaged immediate thrust of a blade to be written, using the generic lift formula in Equation (2.5.17), where $C_{T_{mr}}$ is the coefficient of lift, as if the entire blade is travelling at the same speed as its tip does. The airflow is then also the blade tip speed and the area is that of the circular blade plane. For a 2-bladed vehicle the total thrust produced by the main rotor blades is:

$$T_{mr} = \rho(\Omega_{mr}R_{mr})^2\pi R_{mr}^2 C_{T_{mr}} \tag{2.5.23}$$

The formula for thrust is a function of the induced airflow λ_{mr}. This parameter can be calculated using momentum theory, through the use of Glauert's hypothesis [17], which states that the thrust produced by a spinning blade is directly proportional to the product of the induced airflow and the total airflow through the blade's rotating plane (see Figure 2.5):

$$T = 2\rho AV'w \tag{2.5.24}$$

This can be applied to the total thrust produced by the 2 main rotor blades of the vehicle (from Equation (2.5.23)):

$$\begin{aligned} T_{mr} &= 2\rho(\pi R_{mr}^2)\sqrt{(\Omega_{mr}R_{mr}\lambda_{mr}-\Omega_{mr}R_{mr}\mu_{mr}^z)^2+(\Omega_{mr}R_{mr}\mu_{mr})^2}(\Omega_{mr}R_{mr}\lambda_{mr}) \\ &= \rho(\Omega_{mr}R_{mr})^2\pi R_{mr}^2 C_{T_{mr}} \end{aligned}$$

,which reduces to:

$$\lambda_{mr} = \frac{C_{T_{mr}}}{2\eta_w\sqrt{(\lambda_{mr}-\mu_{mr}^z)^2+(\mu_{mr})^2}} \tag{2.5.25}$$

where η_w is an efficiency factor that has been introduced to compensate for the non-ideal airflow below the blades (as a result of friction on the vehicle body)[3].

[3]The efficiency can be determined by comparing practical airspeed measurements to the calculated airspeed.

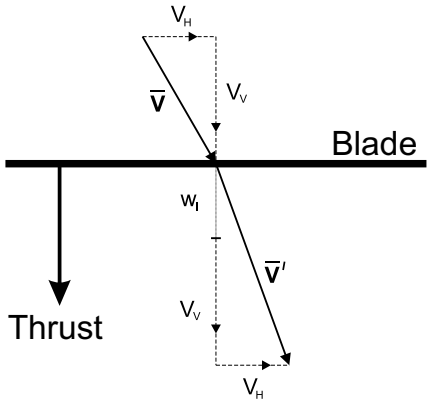


Figure 2.5: Airflow Through a Rotating Blade

In the same way, the drag produced by the main rotor can be integrated to form the main rotor torque (see Appendix A for a full derivation):

$$Q_{mr} = \rho(\Omega_{mr}R_{mr})^2\pi R_{mr}^3 C_Q^{mr} \quad (2.5.26)$$

$$C_Q^{mr} = \left(C_{T_{mr}}(\lambda_{mr} - \mu_{mr}^z) + \frac{C_{D_0}^{mr}\sigma_{mr}}{8}\left(1 + \frac{7}{3}\mu_{mr}^2\right)\right) \quad (2.5.27)$$

where $C_{D_0}^{mr}$ is the profile drag coefficient of the blades.

The main rotor blades of the helicopter are made of carbon composite material that can bend and flex. It is this bending of the blades that helps to steer the vehicle by directing the thrust in the bending directions. The swash plate plays the biggest role in directing the thrust of the vehicle, as it allows for the continuous change of the blade pitch angle as the blade rotates, on top of the constant applied blade pitch (as discussed during the thrust calculations). As the swash plate is only an open loop input to the blades, they offer very limited rate and velocity damping.

To improve these dampings, most small helicopters make use of a free flapping *flight bar*, better known as a *stabilizer bar*. The bar consists of small aerodynamic surfaces and weights and is mounted perpendicular to the main blades on the hub. The stabilizer bar is mechanically connected to the main blades in such a way that when they flap, they have the ability to change the pitch angle of the main blades (on top of the commanded angles from the swash plate). If the convention is used that the blades rotate clock-wise (when viewed from above) and that the blade azimuth angle is denoted by ψ_{BL} (measured from the most backward position), the main blade and flight bar pitch angles can be written in terms of the commanded swash plate collective, longitudinal and lateral commands, as well as the stabilizer bar flapping angle ($\beta_{SB}(\psi_{BL})$) as:

$$\theta_{BL} = \delta_{col} + G_{BL}\delta_{long}\sin(\psi_{BL}) + G_{BL}\delta_{lat}\cos(\psi_{BL}) + K_{SB}\beta_{SB}(\psi_{BL}) \quad (2.5.28)$$

$$\theta_{SB} = G_{SB}(\delta_{long}\cos(\psi_{BL}) - \delta_{lat}\sin(\psi_{BL})) \quad (2.5.29)$$

where G_{BL} and G_{SB} are the hub linkage gain from the swash plate deflection angle to the pitch angles on the main rotor blades and stabilizer bar respectively. K_{SB} is

the hub linkage gain from the out of plane deflection angle of the stabilizer to the pitch angle on the main rotor blades.

It can then be shown that the differential equations for the blade and flight bar motion are (see Appendix A for full derivation):

$$\begin{aligned}
&\ddot{\boldsymbol{\beta}}_{\boldsymbol{BL}} + \dot{\boldsymbol{\beta}}_{\boldsymbol{BL}} \left(\Omega_{mr} + \frac{4}{3} \frac{(u_a \sin(\psi_{BL}) - v_a \cos(\psi_{BL}))}{R_{mr}} \right) \frac{\gamma_{mr}}{8} + \boldsymbol{\beta}_{\boldsymbol{BL}} \left(\Omega_{mr}^2 + \frac{K_\beta}{I_\beta} \right) \\
= \; & \boldsymbol{\theta}_{\boldsymbol{mr}} \left(\Omega_{mr}^2 \tfrac{\gamma_{mr}}{8} \right) \left(1 + \tfrac{8}{3} \tfrac{(u_a \sin(\psi_{BL}) - v_a \cos(\psi_{BL}))}{\Omega_{mr} R_{mr}} + 2 \left(\tfrac{(u_a \sin(\psi_{BL}) - v_a \cos(\psi_{BL}))}{\Omega_{mr} R_{mr}} \right)^2 \right) \\
+ \; & \tfrac{(w_a - \boldsymbol{\lambda_{mr}} \boldsymbol{\Omega_{mr}} \boldsymbol{R_{mr}})}{\boldsymbol{\Omega_{mr}} \boldsymbol{R_{mr}}} \left(\Omega_{mr}^2 \tfrac{\gamma_{mr}}{8} \right) \left(\tfrac{4}{3} + 2 \tfrac{(u_a \sin(\psi_{BL}) - v_a \cos(\psi_{BL}))}{\Omega_{mr} R_{mr}} \right) \\
- \; & \tfrac{(\boldsymbol{p} \sin(\boldsymbol{\psi_{BL}}) - \boldsymbol{q} \cos(\boldsymbol{\psi_{BL}}))}{\boldsymbol{\Omega_{mr}}} \left(\Omega_{mr}^2 \tfrac{\gamma_{mr}}{8} \right) \left(1 + \tfrac{4}{3} \tfrac{(u_a \sin(\psi_{BL}) - v_a \cos(\psi_{BL}))}{\Omega_{mr} R_{mr}} \right)
\end{aligned} \tag{2.5.30}$$

$$\begin{aligned}
&\ddot{\boldsymbol{\beta}}_{\boldsymbol{SB}} + \dot{\boldsymbol{\beta}}_{\boldsymbol{SB}} \left(\Omega_{mr} \right) \frac{\gamma_{SB}}{8} + \boldsymbol{\beta}_{\boldsymbol{SB}} \left(\Omega_{mr}^2 \right) \\
= \; & \boldsymbol{\theta}_{\boldsymbol{sb}} \left(\Omega_{mr}^2 \tfrac{\gamma_{SB}}{8} \right) \left(1 + 2 \left(\tfrac{R_{mr}}{\sqrt{R_{SB2}^2 + R_{SB1}^2}} \right)^2 \left(\tfrac{(u_a \sin(\psi_{BL}) + v_a \cos(\psi_{BL}))}{\Omega_{mr} R_{mr}} \right)^2 \right) \\
+ \; & \tfrac{(w_a - \boldsymbol{\lambda_{mr}} \boldsymbol{\Omega_{mr}} \boldsymbol{R_{mr}})}{\boldsymbol{\Omega_{mr}} \boldsymbol{R_{mr}}} \left(\Omega_{mr}^2 \tfrac{\gamma_{SB}}{8} \right) \left(2 \left(\tfrac{R_{mr}}{\sqrt{R_{SB2}^2 + R_{SB1}^2}} \right)^2 \tfrac{(u_a \sin(\psi_{BL}) + v_a \cos(\psi_{BL}))}{\Omega_{mr} R_{mr}} \right) \\
- \; & \tfrac{(\boldsymbol{p} \sin(\boldsymbol{\psi_{BL}}) + \boldsymbol{q} \cos(\boldsymbol{\psi_{BL}}))}{\boldsymbol{\Omega_{mr}}} \left(\Omega_{mr}^2 \tfrac{\gamma_{SB}}{8} \right)
\end{aligned} \tag{2.5.31}$$

With the advance ratio staying below 0.15, this second order system is poorly damped for the most part ($\zeta_{BL} \approx \frac{\gamma_{mr}}{16} \approx 0.25; \zeta_{SB} = \frac{\gamma_{SB}}{16} \approx 0.05$), however the oscillation frequencies are very high and close to the blade speed ($\omega_n \approx \Omega_{mr} \approx 180$ [rad/s]). It is impractical to model such fast dynamics, so the two blade systems can both be approximated as being first order, with the same settling time as the second order systems. The time constants then become:

$$\tau_{BL} = \frac{16}{\gamma_{mr} \Omega_{mr}} \tag{2.5.32}$$

$$\tau_{SB} = \frac{16}{\gamma_{SB} \Omega_{mr}} \tag{2.5.33}$$

If it is assumed that the blade and flight bar flapping angles are of the form (ignoring any higher order terms):

$$\beta_{BL}(\psi_{BL}) = \beta_{BL}^{const} + \beta_{BL}^{long} \cos(\psi_{BL}) + \beta_{BL}^{lat} \sin(\psi_{BL}) \tag{2.5.34}$$

$$\beta_{SB}(\psi_{BL}) = \beta_{SB}^{long} \cos(\psi_{BL}) + \beta_{SB}^{lat} \sin(\psi_{BL}) \tag{2.5.35}$$

then the first order approximated solution for Equations (2.5.30) and (2.5.31) be-

comes:

$$\begin{aligned}
\beta_{BL}^{long} &= \left(\frac{1}{(\tau_{BL}s+1)}\right)\frac{1}{(1+S_\beta^2)}\Bigg[\\
&-\delta_{col}\left(\frac{8}{3}\right)\left(\frac{u_a+S_\beta v_a}{\Omega_{mr}R_{mr}}\right)-\delta_{long}G_{BL}\left(1+\left(\frac{1}{2}\right)\frac{3u_a^2+2S_\beta u_a v_a+v_a^2}{(\Omega_{mr}R_{mr})^2}\right) \\
&+\delta_{lat}G_{BL}\left(S_\beta+\left(\frac{1}{2}\right)\frac{S_\beta u_a^2+2u_a v_a+3S_\beta v_a^2}{(\Omega_{mr}R_{mr})^2}\right)+\frac{q}{\Omega_{mr}}\left(S_\beta+\frac{16}{\gamma_{mr}}\right) \\
&+\frac{p}{\Omega_{mr}}\left(1-S_\beta\frac{16}{\gamma_{mr}}\right)-\frac{(w_a-\lambda_{mr}\Omega_{mr}R_{mr})}{(\Omega_{mr}R_{mr})}(2)\left(\frac{u_a+S_\beta v_a}{\Omega_{mr}R_{mr}}\right) \\
&+\beta_{SB}^{long}K_{SB}\left((S_\beta+\left(\frac{1}{2}\right)\left(\frac{S_\beta u_a^2+2u_a v_a+3S_\beta v_a^2}{(\Omega_{mr}R_{mr})^2}\right)\right) \\
&-\beta_{SB}^{lat}K_{SB}\left((1+\left(\frac{1}{2}\right)\left(\frac{3u_a^2+2S_\beta u_a v_a+v_a^2}{(\Omega_{mr}R_{mr})^2}\right)\right)\Bigg] \qquad (2.5.36)
\end{aligned}$$

$$\begin{aligned}
\beta_{BL}^{lat} &= \left(\frac{1}{(\tau_{BL}s+1)}\right)\frac{1}{(1+S_\beta^2)}\Bigg[\\
&\delta_{col}\left(\frac{8}{3}\right)\left(\frac{S_\beta u_a-v_a}{\Omega_{mr}R_{mr}}\right)+\delta_{long}G_{BL}\left(S_\beta+\left(\frac{1}{2}\right)\frac{3S_\beta u_a^2-2u_a v_a+S_\beta v_a^2}{(\Omega_{mr}R_{mr})^2}\right) \\
&+\delta_{lat}G_{BL}\left(1+\left(\frac{1}{2}\right)\frac{u_a^2-2S_\beta u_a v_a+3v_a^2}{(\Omega_{mr}R_{mr})^2}\right)-\frac{p}{\Omega_{mr}}\left(S_\beta+\frac{16}{\gamma_{mr}}\right) \\
&+\frac{q}{\Omega_{mr}}\left(1-S_\beta\frac{16}{\gamma_{mr}}\right)+\frac{(w_a-\lambda_{mr}\Omega_{mr}R_{mr})}{(\Omega_{mr}R_{mr})}(2)\left(\frac{S_\beta u_a-v_a}{\Omega_{mr}R_{mr}}\right) \\
&+\beta_{SB}^{lat}K_{SB}\left((S_\beta+\left(\frac{1}{2}\right)\left(\frac{3S_\beta u_a^2-2u_a v_a+S_\beta v_a^2}{(\Omega_{mr}R_{mr})^2}\right)\right) \\
&+\beta_{SB}^{long}K_{SB}\left((1+\left(\frac{1}{2}\right)\left(\frac{u_a^2-2S_\beta u_a v_a+3v_a^2}{(\Omega_{mr}R_{mr})^2}\right)\right)\Bigg] \qquad (2.5.37)
\end{aligned}$$

$$\begin{aligned}
\beta_{SB}^{long} &= \left(\frac{1}{(\tau_{SB}s+1)}\right)\Bigg[-\delta_{long}G_{SB}\left(\frac{R_{mr}}{\sqrt{R_{SB2}^2+R_{SB1}^2}}\right)^2\left(\frac{u_a v_a}{(\Omega_{mr}R_{mr})^2}\right) \\
&+\delta_{lat}G_{SB}\left(1+\left(\frac{1}{2}\right)\left(\frac{R_{mr}}{\sqrt{R_{SB2}^2+R_{SB1}^2}}\right)^2\frac{u_a^2+3v_a^2}{(\Omega_{mr}R_{mr})^2}\right)-\frac{p}{\Omega_{mr}}\left(\frac{16}{\gamma_{SB}}\right) \\
&+\frac{q}{\Omega_{mr}}-\frac{v_a}{\Omega_{mr}R_{mr}}\frac{(w_a-\lambda_{mr}\Omega_{mr}R_{mr})}{(\Omega_{mr}R_{mr})}(2)\left(\frac{R_{mr}}{\sqrt{R_{SB2}^2+R_{SB1}^2}}\right)^2\Bigg] \qquad (2.5.38)
\end{aligned}$$

$$\begin{aligned}\beta_{SB}^{lat} &= \left(\frac{1}{(\tau_{SB}s+1)}\right)\left[\delta_{long}G_{SB}\left(1+\left(\frac{1}{2}\right)\left(\frac{R_{mr}}{\sqrt{R_{SB2}^2+R_{SB1}^2}}\right)^2\frac{3u_a^2+v_a^2}{(\Omega_{mr}R_{mr})^2}\right)\right.\\ &\quad -\delta_{lat}G_{SB}\left(\frac{R_{mr}}{\sqrt{R_{SB2}^2+R_{SB1}^2}}\right)^2\left(\frac{u_av_a}{(\Omega_{mr}R_{mr})^2}\right)-\frac{p}{\Omega_{mr}}-\frac{q}{\Omega_{mr}}\left(\frac{16}{\gamma_{SB}}\right)\\ &\quad \left.+\frac{u_a}{\Omega_{mr}R_{mr}}\frac{(w_a-\lambda_{mr}\Omega_{mr}R_{mr})}{(\Omega_{mr}R_{mr})}(2)\left(\frac{R_{mr}}{\sqrt{R_{SB2}^2+R_{SB1}^2}}\right)^2\right] \qquad (2.5.39)\end{aligned}$$

Where S_β is the blade stiffness factor (usually with a value of less than 0.1):

$$S_\beta = \frac{\frac{K_\beta}{I_{BL}}}{\frac{\gamma_{mr}}{8}\Omega_{mr}^2} \qquad (2.5.40)$$

From Equations (2.5.36) to (2.5.39) it can be seen that the flight bar dramatically increases the rate feedback of the flapping angles (usually by up to a factor of 7), while also increasing the effect of the cyclic inputs (usually by a factor of 3 to 5). The flight bar also reduces the forward velocity feed forward term, which greatly increases the rejection of wind gusts - depending on the amount of collective applied, the result can even be negative feedback to forward velocity.

It can be seen that the lateral and longitudinal flapping angles both have fast and slow dynamics. This will result in a fourth order differential equation. As the faster, natural blade dynamics (those without the stabilizer bar) generally contributes less than 20% of the total steady state value, the approximation that these dynamics act at the same slower rate as the flight bar dynamics, would introduce little error to the model. This slower rate assumption however greatly simplifies the flapping angle dynamics to a second order system. The flapping angle differential equations then become:

$$\begin{aligned}\dot{\beta}_{BL}^{long} &= -\beta_{BL}^{long}\left(\frac{1}{\tau_{SB}}\right)+\left(\frac{1}{\tau_{SB}}\right)\left(\frac{1}{(1+S_\beta^2)}\right)\left[\delta_{col}\left(\frac{\partial\beta_{BL}^{long}}{\partial\delta_{col}}\right)\right.\\ &\quad +\delta_{long}\left(\frac{\partial\beta_{BL}^{long}}{\partial\delta_{long}}\right)+\delta_{lat}\left(\frac{\partial\beta_{BL}^{long}}{\partial\delta_{lat}}\right)+q\left(\frac{\partial\beta_{BL}^{long}}{\partial q}\right)\\ &\quad \left.+p\left(\frac{\partial\beta_{BL}^{long}}{\partial p}\right)+(\mu_{mr}^z-\lambda_{mr})\left(\frac{\partial\beta_{BL}^{long}}{\partial\left(\mu_{mr}^z-\lambda_{mr}\right)}\right)\right] \qquad (2.5.41)\end{aligned}$$

$$\dot{\beta}_{BL}^{lat} = -\beta_{BL}^{lat}\left(\frac{1}{\tau_{SB}}\right)+\left(\frac{1}{\tau_{SB}}\right)\left(\frac{1}{(1+S_{\hat{\beta}}^2)}\right)\left[\delta_{col}\left(\frac{\partial\beta_{BL}^{lat}}{\partial\delta_{col}}\right)\right.$$
$$+\delta_{long}\left(\frac{\partial\beta_{BL}^{lat}}{\partial\delta_{long}}\right)+\delta_{lat}\left(\frac{\partial\beta_{BL}^{lat}}{\partial\delta_{lat}}\right)+q\left(\frac{\partial\beta_{BL}^{lat}}{\partial q}\right)+p\left(\frac{\partial\beta_{BL}^{lat}}{\partial p}\right)$$
$$\left.+\left(\mu_{mr}^{z}-\lambda_{mr}\right)\left(\frac{\partial\beta_{BL}^{lat}}{\partial\left(\mu_{mr}^{z}-\lambda_{mr}\right)}\right)\right] \tag{2.5.42}$$

with the flapping angle derivatives:

$$\frac{\partial\beta_{BL}^{long}}{\partial\delta_{col}} = -\left(\frac{8}{3}\right)\left(\frac{u_a+S_\beta v_a}{\Omega_{mr}R_{mr}}\right) \tag{2.5.43}$$
$$\frac{\partial\beta_{BL}^{lat}}{\partial\delta_{col}} = \left(\frac{8}{3}\right)\left(\frac{S_\beta u_a-v_a}{\Omega_{mr}R_{mr}}\right) \tag{2.5.44}$$
$$\frac{\partial\beta_{BL}^{long}}{\partial\delta_{long}} = -\left(\left(G_{BL}+K_{SB}G_{SB}\right)+\frac{\left(G_{BL}+G_{SB}K_{SB}R_{SB}^{mr}\right)}{2}\frac{\left(3u_a^2+2S_\beta u_a v_a+v_a^2\right)}{(\Omega_{mr}R_{mr})^2}\right) \tag{2.5.45}$$
$$\frac{\partial\beta_{BL}^{lat}}{\partial\delta_{long}} = \left(S_\beta\left(G_{BL}+K_{SB}G_{SB}\right)+\frac{\left(G_{BL}+G_{SB}K_{SB}R_{SB}^{mr}\right)}{2}\frac{\left(3S_\beta u_a^2-2u_a v_a+S_\beta v_a^2\right)}{(\Omega_{mr}R_{mr})^2}\right) \tag{2.5.46}$$
$$\frac{\partial\beta_{BL}^{long}}{\partial\delta_{lat}} = \left(S_\beta\left(G_{BL}+K_{SB}G_{SB}\right)+\frac{\left(G_{BL}+G_{SB}K_{SB}R_{SB}^{mr}\right)}{2}\frac{\left(S_\beta u_a^2+2u_a v_a+3S_\beta v_a^2\right)}{(\Omega_{mr}R_{mr})^2}\right) \tag{2.5.47}$$

$$\frac{\partial\beta_{BL}^{lat}}{\partial\delta_{lat}} = \left(\left(G_{BL}+K_{SB}G_{SB}\right)+\frac{\left(G_{BL}+G_{SB}K_{SB}R_{SB}^{mr}\right)}{2}\frac{\left(u_a^2-2S_\beta u_a v_a+3v_a^2\right)}{(\Omega_{mr}R_{mr})^2}\right) \tag{2.5.48}$$
$$\frac{\partial\beta_{BL}^{long}}{\partial q} = \left(\frac{\left(1+K_{SB}\right)S_\beta+\left(1+K_{SB}\frac{\gamma_{mr}}{\gamma_{SB}}\right)\left(\frac{16}{\gamma_{mr}}\right)}{\Omega_{mr}}\right) \tag{2.5.49}$$
$$\frac{\partial\beta_{BL}^{lat}}{\partial q} = \left(\frac{\left(1+K_{SB}\right)-\left(1+K_{SB}\frac{\gamma_{mr}}{\gamma_{SB}}\right)S_\beta\left(\frac{16}{\gamma_{mr}}\right)}{\Omega_{mr}}\right) \tag{2.5.50}$$
$$\frac{\partial\beta_{BL}^{long}}{\partial p} = \left(\frac{\left(1+K_{SB}\right)-\left(1+K_{SB}\frac{\gamma_{mr}}{\gamma_{SB}}\right)S_\beta\left(\frac{16}{\gamma_{mr}}\right)}{\Omega_{mr}}\right) = \left(\frac{\partial\beta_{BL}^{lat}}{\partial q}\right) \tag{2.5.51}$$
$$\frac{\partial\beta_{BL}^{lat}}{\partial p} = -\left(\frac{\left(1+K_{SB}\right)S_\beta+\left(1+K_{SB}\frac{\gamma_{mr}}{\gamma_{SB}}\right)\left(\frac{16}{\gamma_{mr}}\right)}{\Omega_{mr}}\right) = -\left(\frac{\partial\beta_{BL}^{long}}{\partial q}\right) \tag{2.5.52}$$

$$\frac{\partial \beta_{BL}^{long}}{\partial \left(\mu_{mr}^{z} - \lambda_{mr}\right)} = -(2)\left(1 + K_{SB} R_{SB}^{mr}\right)\left(\frac{u_a + S_\beta v_a}{\Omega_{mr} R_{mr}}\right) \tag{2.5.53}$$

$$\frac{\partial \beta_{BL}^{lat}}{\partial \left(\mu_{mr}^{z} - \lambda_{mr}\right)} = (2)\left(1 + K_{SB} R_{SB}^{mr}\right)\left(\frac{S_\beta u_a - v_a}{\Omega_{mr} R_{mr}}\right) \tag{2.5.54}$$

Gavrilets et al. [10] took a different approach in modelling their X-Cell's flapping dynamics. They assumed no cross coupling in the axes and gave each of the feedback terms a non-dimensionalized gain. These gains were determined using a variety of aerodynamic tests (including wind tunnel and test flights). When the same approximations are made in Equations (2.5.36) to (2.5.39), the same model structure is obtained as the one listed by Gavrilets.

The complex generic form developed in this section has the advantage that it can be applied to any vehicle, even if the hub linkages and stabiliser bar differ from those used in this project. This form enables model identification without having to go through the tedious process of aerodynamic tests.

It can be shown that a flapping blade produces a moment perpendicular to its flapping plane, as well as a force in its flapping plane. The moment and the force can be described by (per blade):

$$M_{\boldsymbol{i}_B} = \frac{1}{2} K_\beta \beta_{BL}^{lat} \tag{2.5.55}$$

$$M_{\boldsymbol{j}_B} = -\frac{1}{2} K_\beta \beta_{BL}^{long} \tag{2.5.56}$$

$$F_{\boldsymbol{i}_B} = \frac{1}{2} T_{mr} \beta_{BL}^{long} \tag{2.5.57}$$

$$F_{\boldsymbol{j}_B} = \frac{1}{2} T_{mr} \beta_{BL}^{lat} \tag{2.5.58}$$

The main rotor forces and moments about the centre of gravity follow:

$$X_{mr} = T_{mr} \beta_{BL}^{long} \tag{2.5.59}$$

$$Y_{mr} = T_{mr} \beta_{BL}^{lat} \tag{2.5.60}$$

$$Z_{mr} = -T_{mr} \tag{2.5.61}$$

$$L_{mr} = \beta_{BL}^{lat}(K_\beta + h_{mr} T_{mr}) \tag{2.5.62}$$

$$M_{mr} = -\beta_{BL}^{long}(K_\beta + h_{mr} T_{mr}) \tag{2.5.63}$$

$$N_{mr} = 0 \tag{2.5.64}$$

Note that the torque produced by the main rotor is not listed here, as it is countered by the engine and will be described as part of engine dynamics in section 2.5.5.

2.5.3 Tail Rotor

The thrust and torque produced by the tail rotor is calculated in a similar manner as for the main rotor. The main difference comes from the thrust blocking factor f_t,

which was proposed by Padfield [11], to account for the blocking of airflow by the vertical tail fin:

$$T_{tr} = f_t \rho (\Omega_{tr} R_{tr})^2 \pi R_{tr}^2 C_{T_{tr}} \tag{2.5.65}$$

$$C_{T_{tr}} = \frac{\sigma_{tr} a_{tr}}{2} \left(\delta_{rudd} \left(\frac{1}{3} + \frac{\mu_{tr}^2}{2} \right) + \frac{\mu_{tr}^z - \lambda_{tr}}{2} \right) \tag{2.5.66}$$

$$\lambda_{tr} = \frac{C_{T_{tr}}}{2\eta_w \sqrt{(\lambda_{tr} - \mu_{tr}^z)^2 + (\mu_{tr})^2}} \tag{2.5.67}$$

$$Q_{tr} = \rho (\Omega_{tr} R_{tr})^2 \pi R_{tr}^3 C_Q^{tr} \tag{2.5.68}$$

$$\sigma_{tr} = \frac{2c_{tr}}{\pi R_{tr}} \tag{2.5.69}$$

$$f_t = 1 - \frac{3}{4} \frac{S_{vf}}{\pi R_{tr}^2} \tag{2.5.70}$$

$$\Omega_{tr} = n_{tr} \Omega_{mr} \tag{2.5.71}$$

$$C_Q^{tr} = \left(C_{T_{tr}} (\lambda_{tr} - \mu_{tr}^z) + \frac{C_{D_0}^{tr} \sigma_{tr}}{8} \left(1 + \frac{7}{3} \mu_{tr}^2 \right) \right) \tag{2.5.72}$$

where n_{tr} is the gear ratio between the tail and main rotor and S_{vf} is the effective area of the vertical tail fin.

The airflow parameters for the tail rotor are slightly more complex to calculate, as the tail rotor is not located as close to the centre of gravity as the main rotor and any body rotations cause induced airflow on the tail rotor. Also, as forward speed increases, quite a significant amount of the main rotor induced airflow can be directed towards the tail rotor. Figure 2.6 illustrates this phenomenon.

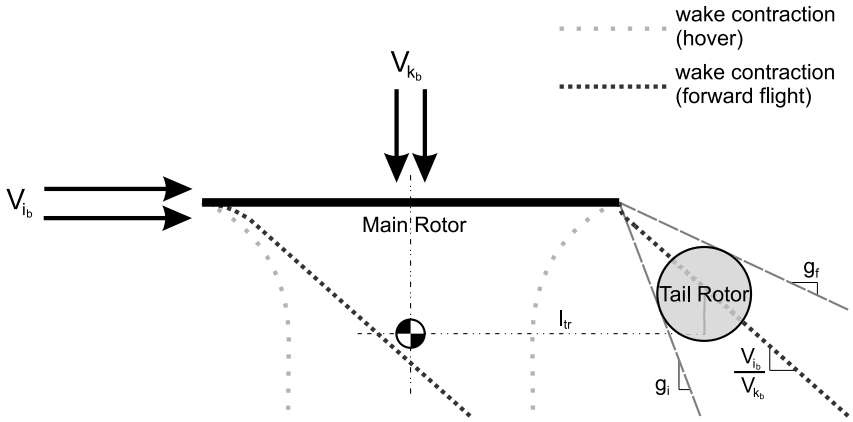


Figure 2.6: Exposure of Main Rotor Airflow to Tail Rotor

Where V_{i_B} is the forward component of the horizontal airflow $(u - u_w)$ and V_{k_B} the vertical airflow, which is the sum of the rotor induced and body induced airflow $(\Omega_{mr} R_{mr} \lambda_{mr} - (w - w_w))$.

To estimate the percentage of the total main rotor induced airflow that is directed towards the tail rotor, the gradients of the lines from the closest tip of the main rotor to the beginning and the end of the tail rotor were approximated as:

$$g_i = \frac{l_{tr} - R_{mr} - R_{tr}}{h_{mr} - h_{tr}} \tag{2.5.73}$$

$$g_f = \frac{l_{tr} - R_{mr} + R_{tr}}{h_{mr} - h_{tr}} \tag{2.5.74}$$

where h_{mr} and h_{tr} are the vertical offsets of the main and tail rotors above the centre of gravity respectively and l_{tr} is the horizontal offset of the tail rotor to the CG. The part of the main rotor airflow exposed to the tail rotor can now be approximated as:

$$K_\lambda = CF_{mr} \frac{\frac{V_{i_B}}{V_{k_B}} - g_i}{g_f - g_i} \tag{2.5.75}$$

where CF_{mr} is the wake contraction factor of the main rotor airflow at a distance from the rotors and represents the increase in airflow. Note that K_λ has an upper limit of CF_{mr} when the tail rotor is fully exposed in the wake and a lower limit of 0 if it is out of the wake. The other airflow parameters of the tail rotor can now be calculated using this partial main rotor airflow, the translational induced airflow and the body rotational induced airflow:

$$w_{tr} = w - w_w - K_\lambda(\Omega_{mr} R_{mr} \lambda_{mr}) + q l_{tr} \tag{2.5.76}$$

$$\mu_{tr} = \frac{\sqrt{(u - u_w)^2 + w_{tr}^2}}{\Omega_{tr} R_{tr}} \tag{2.5.77}$$

$$v_{tr} = v - v_w - l_{tr} r + h_{tr} p \tag{2.5.78}$$

$$\mu_{tr}^z = \frac{v_{tr}}{\Omega_{tr} R_{tr}} \tag{2.5.79}$$

Once the thrust is calculated, the forces and moments produced by the tail rotor around the centre of gravity become:

$$X_{tr} = 0 \tag{2.5.80}$$

$$Y_{tr} = -T_{tr} \tag{2.5.81}$$

$$Z_{tr} = 0 \tag{2.5.82}$$

$$L_{tr} = -h_{tr} T_{tr} \tag{2.5.83}$$

$$M_{tr} = 0 \tag{2.5.84}$$

$$N_{tr} = l_{tr} T_{tr} \tag{2.5.85}$$

2.5.4 Analog Gyro

During manual safety pilot flight, rudder control is augmented with an *analog hobby gyro*. This device is a partial yaw rate rudder controller that dampens the fast tail dynamics of the vehicle and enables the safety pilot to control this normally unflyable axis. Due to the critical importance of this analog controller to the safety of flight, it was decided not to remove this device during autopilot flight, as switchover between manual and autopilot flight might produce unforeseen results. The gyro's dynamics was rather characterized and the controller was designed taking these added dynamics into account.

The gyro output and yaw rate were measured for different yaw rate inputs. The results follow:

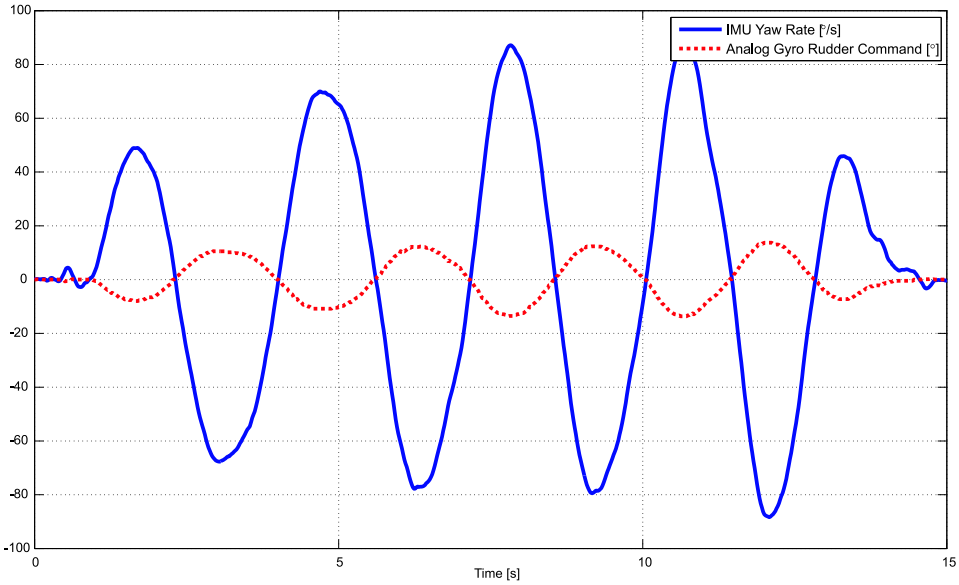


Figure 2.7: Analog Gyro Calibration Test Data

From the data in Figure 2.7 a first order model could be approximated:

$$\frac{\delta_{rudder}}{r_{true}} = -\frac{0.156}{\left(\frac{s}{30}+1\right)} \tag{2.5.86}$$

Figure 2.8 shows the IMU gyro measurements, with the above-mentioned plant applied to it, as well as the analog gyro outputs with the IMU filter plant applied to it.

2.5.5 Engine

The assumption can be made that around trim, the engine's steady state torque response to throttle commands is linear. Additionally, it can be assumed that the

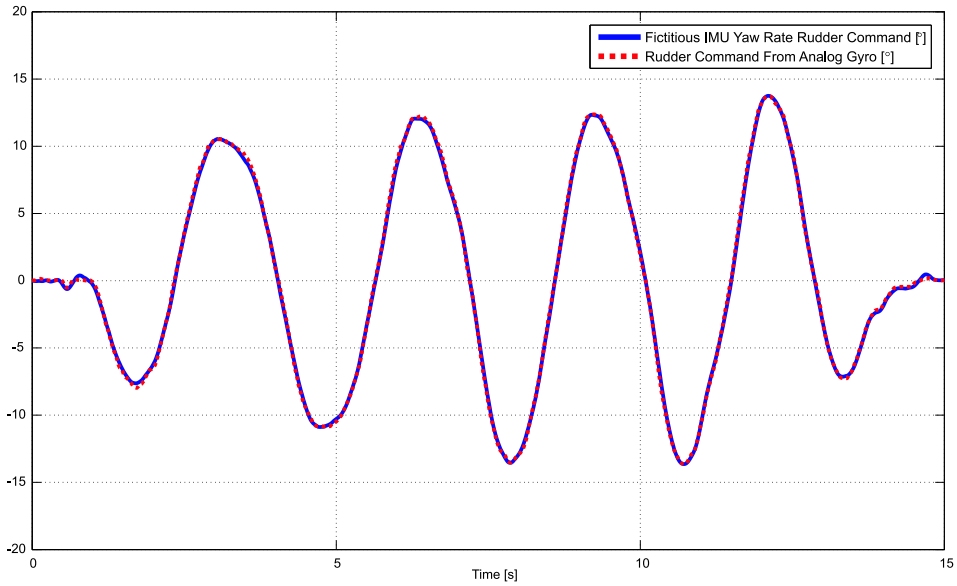


Figure 2.8: Analog Gyro Calibration Verification

lag from the commanded value to produced torque can be modelled as a first order delay, with a time constant of τ_{eng}.

$$Q_e = \left(\frac{1}{\frac{s}{\frac{1}{\tau_{eng}}} + 1} \right) \left(Q_e^0 + \delta_{throttle} \right) \tag{2.5.87}$$

where $\delta_{throttle}$ is the commanded engine torque in radians and Q_e^0 is the trim engine torque in Nm. If a governor[4] is used to control the main rotor speed, then the commanded throttle around hover conditions can be expressed as the speed offset of the main rotor blades, multiplied by the governor feedback gain (K_{gov}). The engine torque becomes:

$$Q_e = \left(\frac{1}{\frac{s}{\frac{1}{\tau_{eng}}} + 1} \right) \left(Q_e^0 + K_{gov}(\Omega_c - \Omega_{mr}) \right) \tag{2.5.88}$$

$$\dot{Q}_e = -\frac{1}{\tau_{eng}} Q_e + \frac{1}{\tau_{eng}} \left(Q_e^0 + K_{gov}(\Omega_c - \Omega_{mr}) \right) \tag{2.5.89}$$

In an effort to model the main rotor blade speed, the total angular kinetic energy of all the *body fixed* rotating parts is calculated namely that of the engine, main rotor and tail rotor:

$$E_{rotation} = \frac{1}{2} I_{mr} (\Omega_{mr})^2 + \frac{1}{2} I_{tr} (\Omega_{tr})^2 + \frac{1}{2} I_{engine} (\Omega_{engine})^2 \tag{2.5.90}$$

[4] An engine governor controls the engine and thus the blade speed to keep a fixed, predefined reference speed.

All the rotations can now be written in terms of the main rotor speed and the rotor inertia in terms of the single blade inertia:

$$E_{rotation} = \frac{1}{2}\left(2I_{\beta_{mr}} + 2I_{\beta_{tr}}n_{tr}^2 + I_{engine}n_{engine}^2\right)\Omega_{mr}^2 \quad (2.5.91)$$

From Equation (2.5.91) it can be seen that the rotational inertia referenced to the main rotor blade speed can be written as $I_{rot} = 2I_{\beta_{mr}} + 2I_{\beta_{tr}}n_{tr}^2 + I_{engine}n_{engine}^2$, where n_{tr} and n_{engine} are the gear ratios from the tail rotor and engine respectively, towards the main rotor. The main shaft angular acceleration in the body axes is written in terms of the applied moments:

$$\dot{\Omega}_{\boldsymbol{k_B}}^{shaft} = \frac{Q_e - Q_{mr} - n_{tr}Q_{tr}}{I_{rot}} \quad (2.5.92)$$

Converting the shaft acceleration to the earth frame, the differential equation for rotor speed follows:

$$\dot{\Omega}_{mr} = -\dot{r} + \frac{Q_e - Q_{mr} - n_{tr}Q_{tr}}{I_{rot}} \quad (2.5.93)$$

The total forces and moments produced by the engine around the centre of gravity of the vehicle become:

$$X_e = 0 \quad (2.5.94)$$
$$Y_e = 0 \quad (2.5.95)$$
$$Z_e = 0 \quad (2.5.96)$$
$$L_e = 0 \quad (2.5.97)$$
$$M_e = 0 \quad (2.5.98)$$
$$N_e = -Q_e \quad (2.5.99)$$

2.5.6 Fuselage

Accurately modelling the drag produced by a complex object such as the airframe of a helicopter can be quite a tedius task and would require hours of wind tunnel testing. As the net airflow approaches the airframe from only one direction, the drag coefficient and centre of pressure would have to be known from that exact angle. It was decided to avoid such tedious testing - Gavrilets et al. [10] suggested that an adequate approximation would be to model the fuselage as three flat *drag surfaces*, one normal to each of the body axes, with the centre of pressure of each surface coinciding with the centre of gravity of the vehicle. The area of each *drag surface* would become a lumped parameter of the true surface area and the drag coefficient of that area, so that the drag can be calculated as:

$$D = \frac{1}{2}\rho V_{\perp}^2 S^{fus} \quad (2.5.100)$$

where $V_{\perp}$ is the airflow normal to the surface and S^{fus} is the drag surface area. Figure 2.9 shows the position of these surfaces.

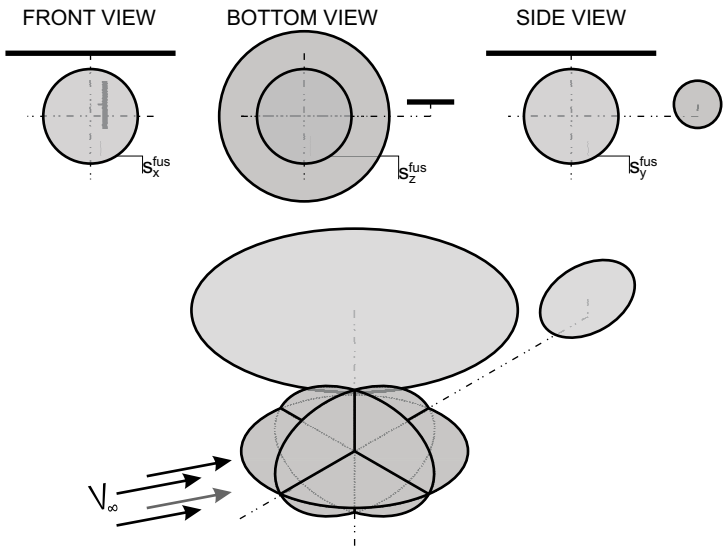


Figure 2.9: Body Drag Surfaces

The difficulty is still that the airflow on each of these surfaces will rarely be normal to that surface. It can now be assumed that the total force produced by drag can be superposed from the individual forces on the individual surfaces. The drag of each surface can then be determined by calculating the effective wetted area (area exposed to the airflow) of each surface. Figure 2.10 and Equation (2.5.104) show one way to approximate this area for the $\boldsymbol{i}_B$ axis.

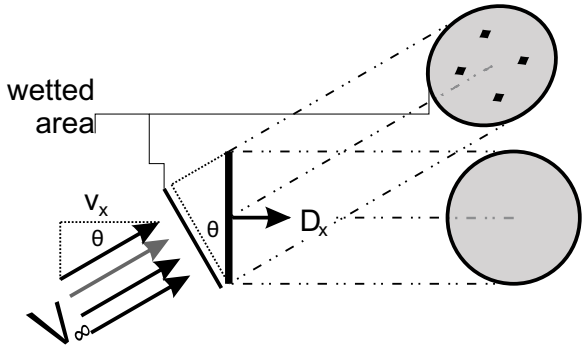


Figure 2.10: Scaled Drag Surfaces

$$V_\infty = \sqrt{(u-u_w)^2+(v-v_w)^2+(w-w_w-\Omega_{mr}R_{mr}\lambda_{mr})^2} \quad (2.5.101)$$

$$\cos(\theta) = \frac{u-u_w}{V_\infty} \quad (2.5.102)$$

$$S^{fus}_{x_{true}} \approx \cos(\theta)S^{fus}_x \quad (2.5.103)$$

$$\Rightarrow S^{fus}_{x_{true}} \approx \frac{u-u_w}{V_\infty}S^{fus}_x \quad (2.5.104)$$

where V_∞ is the total incoming airflow and θ is the angle offset, from normal to the surface, to the airflow. As the centre of pressure was assumed to coincide with the centre of gravity, no moments are produced by fuselage drag. The drag forces and moments can be written as:

$$X_{fus} = -\frac{1}{2}\rho V_\infty^2\left(\frac{u-u_w}{V_\infty}S^{fus}_x\right) \quad (2.5.105)$$

$$Y_{fus} = -\frac{1}{2}\rho V_\infty^2\left(\frac{v-v_w}{V_\infty}S^{fus}_y\right) \quad (2.5.106)$$

$$Z_{fus} = -\frac{1}{2}\rho V_\infty^2\left(\frac{w-w_w-\Omega_{mr}R_{mr}\lambda_{mr}}{V_\infty}S^{fus}_z\right) \quad (2.5.107)$$

$$L_{fus} = 0 \quad (2.5.108)$$

$$M_{fus} = 0 \quad (2.5.109)$$

$$N_{fus} = 0 \quad (2.5.110)$$

2.5.7 Empennage

The empennage of the helicopter includes horizontal and vertical fins close to the tail rotor assembly. These surfaces help to provide some longitudinal velocity and yaw angle feedback when travelling forward. Their modelling can become complicated, since at some point when accelerating forward, the airflow from the main rotor blades produces additional feedback through these surfaces. As these surfaces are actually flat, the same modelling methodology as described with fuselage drag can be used. In the case of fuselage drag no lift was generated by the fuselage and the modelling of the profile drag was sufficient for this unknown and irregular surface. Due to their fairly simple design and relatively large influence on the model, a more complex modelling methodology will be developed for the fins.

The total force on the horizontal surface can be broken up into two sections: Profile drag[5] and lift[6]. Profile drag becomes dominant when the down-wash airspeed from the main rotor blades is dominant, compared to the horizontal speed of the vehicle. However at higher speeds this factor becomes less significant and the lift produced by the surface dominates. The two forces can be expressed as:

$$F_{drag} = \frac{1}{2}\rho S_{ht}(w_{tr})^2\, sgn(w_{tr}) \quad (2.5.111)$$

$$F_{lift} = \frac{1}{2}\rho S_{ht}C^{ht}_{L\alpha}(V^{tr}_\infty)^2\alpha \approx \frac{1}{2}\rho S_{ht}C^{ht}_{L\alpha}(V^{tr}_\infty)^2\left(\frac{w_{tr}}{V^{tr}_\infty}\right) \quad (2.5.112)$$

[5] A drag force is normally in-line with airflow direction.

[6] A lift force is normally perpendicular to airflow direction.

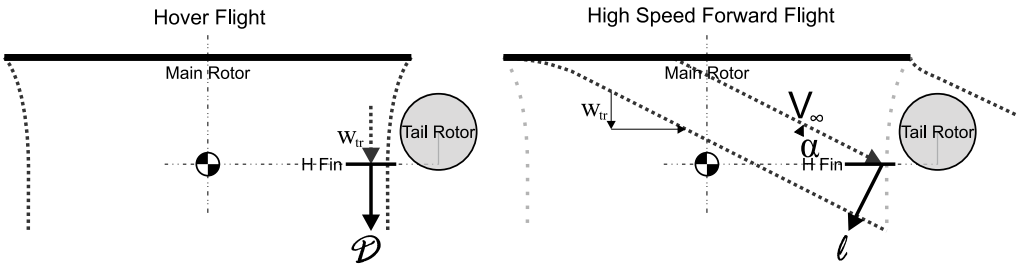


Figure 2.11: Drag (left) and Lift (right), Produced by the Empennage

It should be noted that the lift coefficient ($C^{ht}_{L\alpha}$) is only valid for angles of attack up to $\pm 18^\circ$, as the lift curve becomes non-linear after this point and the surface enters stall[7]. It was found that past this point the profile drag increases dramatically and starts to dominate this equation. It would be difficult to model the coefficient of lift past this point. An easier approach would be to limit the total drag on this surface as the maximum possible drag of ($F_{drag} = \frac{1}{2}\rho S_{ht}(V^{tr}_\infty)^2$).

A similar approach can be taken to model the forces on the vertical fin. To summarize these forces:

$$Y_{vf} = -\frac{1}{2}\rho S_{vf}\left(C^{vf}_{L\alpha}V^{tr}_\infty + |v_a|\right)v_a \tag{2.5.113}$$

$$|Y_{vf}| \leq \frac{1}{2}\rho S_{vf}\left(V^{tr}_\infty\right)^2 \tag{2.5.114}$$

$$Z_{ht} = \frac{1}{2}\rho S_{ht}\left(C^{ht}_{L\alpha}V^{tr}_\infty + |w_{tr}|\right)w_{tr} \tag{2.5.115}$$

$$|Z_{ht}| \leq \frac{1}{2}\rho S_{ht}\left(V^{tr}_\infty\right)^2 \tag{2.5.116}$$

The forces and moments created by the empennage around the centre of gravity of the vehicle can be summarized:

$$X_{emp} = 0 \tag{2.5.117}$$

$$Y_{emp} = Y_{vf} \tag{2.5.118}$$

$$Z_{emp} = Z_{ht} \tag{2.5.119}$$

$$L_{emp} = Y_{vf}h_{tr} \tag{2.5.120}$$

$$M_{emp} = Z_{vf}l_{ht} \tag{2.5.121}$$

$$N_{emp} = -Y_{vf}l_{tr} \tag{2.5.122}$$

2.5.8 Servos

There are two types of servos on the vehicle. The servos that control the swash plate are rated as being capable to rotate at a maximum rate of $300\,[^\circ/s]$ and the one for the tail rotor at $600\,[^\circ/s]$. The mechanical linkages from each servo to the

[7] See Figure 2.4.

swash plate actuators and tail rotor were characterized and the resulting maximum actuator rates were calculated[8]:

$$\left(\frac{\partial \delta_{col}}{\partial t}\right)_{max} = 62\,[^\circ/s] \quad (2.5.123)$$

$$\left(\frac{\partial \delta_{long}}{\partial t}\right)_{max} = 115\,[^\circ/s] \quad (2.5.124)$$

$$\left(\frac{\partial \delta_{lat}}{\partial t}\right)_{max} = 125\,[^\circ/s] \quad (2.5.125)$$

$$\left(\frac{\partial \delta_{rudd}}{\partial t}\right)_{max} = 260\,[^\circ/s] \quad (2.5.126)$$

The small actuator commands that controlled flight normally produces should be little affected by these relatively fast slew rates. Any linear approximations of these non-linear models will increase the pole count of the model. For this reason it was decided not to include these approximations in the controller design. Their models were however added in their full non-linear form to the non-linear simulation model, which improved the accuracy of the HIL simulation.

2.6 Ground Effect

The model that has been introduced until now was based on the assumption that the vehicle will fly in open air, far away from any objects that might cause aerodynamic disturbances. This assumption is adequate for the majority of the intended flight envelope, except close to the ground. Padfield [11] states that the effect of the ground in close proximity to the main rotor can cause its induced airflow to decrease. The net effect of this decrease in airflow is an increase in thrust.

Such large variation in the model close to the ground definitely merits further modelling, especially since the vehicle would be prepared for automated take-off and landing. Figure 2.12 shows practical measurements of how thrust varies with forward speed in close proximity to the ground (all measurements are taken at constant induced power).

In Figure 2.12 T_g and T_∞ are the measured thrust close to and far away from the ground respectively. The induced velocity is denoted by v_i and the horizontal airspeed by V. z_g is the distance of the blade plane above the ground (as a fraction of rotor radius R_{mr}).

Padfield further suggests that it would be easier to calculate the complex airflow close to the ground plane (see Figure 2.13a), by making the assumption that the ground can be approximated as being an equal, but opposite rotor on the other side of the ground plane (see Figure 2.13b). This makes it easier to estimate the induced velocity at the blades by just adding the velocity of the opposing airflow. When the rotor starts to move horizontally with respect to the ground this effect

[8]The actuator speed is calculated by scaling each servo speed by the mechanical linkage gain.

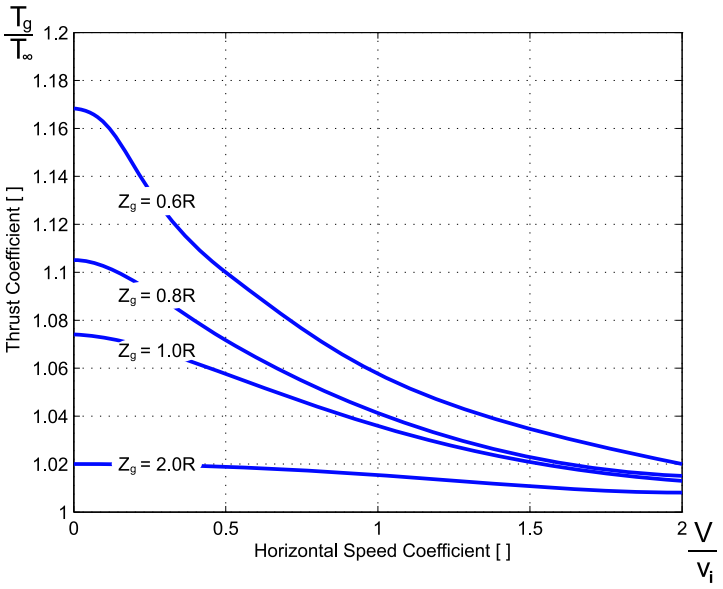


Figure 2.12: Ground Effect on Thrust [11]

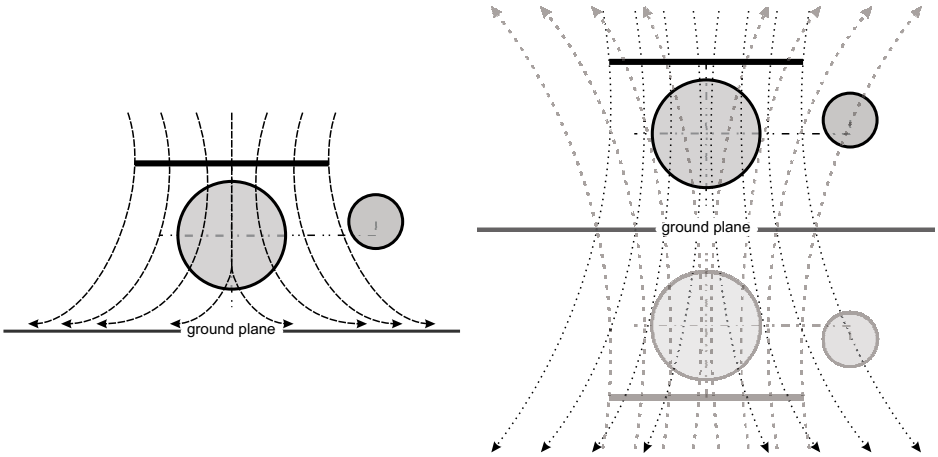


Figure 2.13: Ground Effect on Induced Velocity

diminishes quickly. An equation for the opposing airflow was proposed [18]:

$$\delta v_{i_g} = -\left(\frac{\frac{A_{BL}}{16\pi z_g^2}}{1+\left(\frac{V}{v_i}\right)^2}\right) v_{i_\infty} \tag{2.6.1}$$

$$\Rightarrow v_{i_g} = v_{i_\infty} - \left(\frac{\frac{A_{BL}}{16\pi z_g^2}}{1+\left(\frac{V}{v_i}\right)^2}\right) v_{i_\infty} \tag{2.6.2}$$

with A_{BL} being the main blade plane area.

In Section 2.5.2 as part of the process to model the thrust and induced velocity of the main rotor blades the assumption was made initially that the blades rotate in free space far away from the ground. Rather than using a different methodology and to remodel them close to the ground, Equation (2.6.2) allows for an open loop correction of these parameters to be made in the vicinity of the ground plane.

From Equations (2.5.22) and (2.5.23), the corrected thrust parameters can now be calculated:

$$\begin{aligned}
\lambda_{mr}^{g} &= \lambda_{mr}^{\infty} - \lambda_{mr}^{\infty}\left(\frac{\frac{A_{BL}}{16\pi z_g^2}}{1+\left(\frac{V}{v_i}\right)^2}\right) \\
&= \lambda_{mr}^{\infty}\left(1+\frac{\left(\frac{R_{mr}}{4z_g}\right)^2}{1+\left(\frac{V}{v_i}\right)^2}\right) \qquad (2.6.3)\\
T_{mr}^{g} &= \rho(\Omega_{mr}R_{mr})^2\pi R_{mr}^2 C_{T_{mr}}^{g} \qquad (2.6.4)\\
C_{T_{mr}}^{g} &= \frac{\sigma_{mr}a_{mr}}{2}\left(\delta_{col}\left(\frac{1}{3}+\frac{\mu_{mr}^2}{2}\right)+\frac{\mu_{mr}^z-\left(\lambda_{mr}^{\infty}-\lambda_{mr}^{\infty}\left(\frac{\left(\frac{R_{mr}}{4z_g}\right)^2}{1+\left(\frac{V}{v_i}\right)^2}\right)\right)}{2}\right) \\
&= C_{T_{mr}}^{\infty}+\frac{\sigma_{mr}a_{mr}}{64}\left(\frac{\left(\frac{R_{mr}}{z_g}\right)^2}{1+\left(\frac{V}{v_i}\right)^2}\right)\lambda_{mr}^{\infty} \qquad (2.6.5)
\end{aligned}$$

2.7 Runway Model

Another important factor when considering take-off and landing pertains to the forces that are exerted on the vehicle when making contact with the runway surface. Because of the relatively high centre of gravity, combined with the much higher main rotor blades, any vectored forces at the main rotor blades or excessive lateral momentum can cause the vehicle to topple over.

The skids have a rectangular shape and it was decided to model their four corners as four separate contact points. This will enable the modelling of skew touchdowns - as is the case with most landings. Each contact point is modelled as having a deformation zone, where it exerts spring forces and is rate damped. These forces act perpendicular to the surface of the runway. Additionally a lateral viscous friction component is modelled as a force that is proportional to the normal runway force and acts in the opposite direction to the lateral movement.

The first step is to calculate the position and velocity of each contact point to determine when it starts touching the runway and at what speed it is moving with

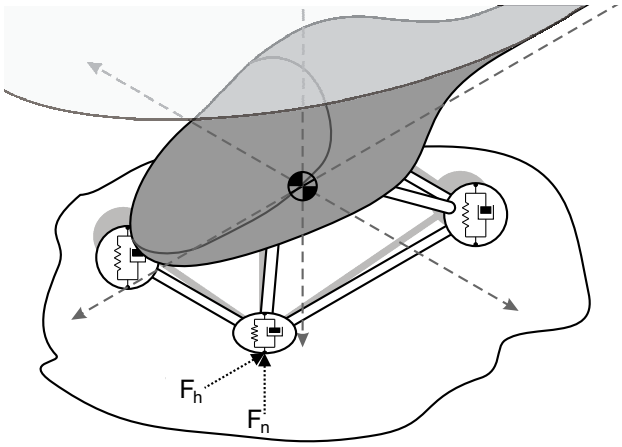


Figure 2.14: Modelling of the Skids

respect to the runway:

$$\boldsymbol{P_E^i} = \boldsymbol{T^T P_B^i} \tag{2.7.1}$$

$$\boldsymbol{V_E^i} = \boldsymbol{V_E^B} + \boldsymbol{\omega}_B \times \boldsymbol{P_B^i} \tag{2.7.2}$$

where $\boldsymbol{P_E^i}$ and $\boldsymbol{V_E^i}$ are the position and velocity of a contact point in earth coordinates, offset from the CG and $\boldsymbol{V_E^B}$ is the vehicle velocity in earth coordinates. The skid contact point number is noted by "i".

The forces can now be calculated:

$$F_Z^i = \left(P_{E_Z}^B - P_{E_Z}^i\right) K_S^i - \left(V_{E_Z}^i\right) B_D^i \tag{2.7.3}$$

$$\begin{bmatrix} F_X^i \\ F_Y^i \\ 0 \end{bmatrix} = - \begin{bmatrix} V_{E_X}^i \\ V_{E_Y}^i \\ 0 \end{bmatrix} \left|F_Z^i\right| K_V^i \tag{2.7.4}$$

$$\boldsymbol{F_E^i} = \begin{bmatrix} F_X^i \\ F_Y^i \\ F_Z^i \end{bmatrix} \tag{2.7.5}$$

where K_S^i, B_D^i and K_V^i are the spring constant, damper constant and viscous friction constant respectively. Of great importance are the moments that these forces exert on the body of the vehicle. They can be written as:

$$\boldsymbol{M_B^i} = \boldsymbol{P_B^i} \times \left(\boldsymbol{T F_E^i}\right) \tag{2.7.6}$$

2.8 Non-linear Model

Now that all the kinematics, vehicle-independent and vehicle-depended dynamics have been defined, a non-linear model is built up. It was decided to limit the number of vehicle states to 16, thus including the most important states, but also not adding unnecessary complexity.

These states are:

$$\begin{aligned}
\text{Body velocities} &: u, v, w \ [m/s] \\
\text{Body angular rates} &: p, q, r \ [rad/s] \\
\text{Body Euler 321 attitude} &: \phi, \theta, \psi \ [rad] \\
\text{Body position error offset} &: X, Y, Z \ [m] \\
\text{Main blade flapping angles} &: \beta_{BL}^{long}, \beta_{BL}^{lat} \ [rad] \\
\text{Main blade speed} &: \Omega_{mr} \ [rad/s] \\
\text{Engine torque} &: Q_e \ [Nm]
\end{aligned}$$

The four vehicle actuators are:

$$\begin{aligned}
\text{Swash plate longitudinal deflection} &: \delta_{long} \ [rad] \\
\text{Swash plate lateral deflection} &: \delta_{lat} \ [rad] \\
\text{Main blades collective pitch angle} &: \delta_{col} \ [rad] \\
\text{Tail rotor blades collective pitch angle} &: \delta_{rudd} \ [rad]
\end{aligned}$$

The non-linear state differential equations then become:

$$\dot{u} = vr - wq - g\sin(\theta) + \frac{X_{mr} + X_{tr} + X_e + X_{fus} + X_{emp}}{m} \tag{2.8.1}$$

$$\dot{v} = wp - ur + g\sin(\phi)\cos(\theta) + \frac{Y_{mr} + Y_{tr} + Y_e + Y_{fus} + Y_{emp}}{m} \tag{2.8.2}$$

$$\dot{w} = uq - vp + g\cos(\phi)\cos(\theta) + \frac{Z_{mr} + Z_{tr} + Z_e + Z_{fus} + Z_{emp}}{m} \tag{2.8.3}$$

$$\dot{p} = \frac{qr\left(I_{zz} - I_{yy}\right) + L_{mr} + L_{tr} + L_e + L_{fus} + L_{emp}}{I_{xx}} \tag{2.8.4}$$

$$\dot{q} = \frac{pr\left(I_{xx} - I_{zz}\right) + M_{mr} + M_{tr} + M_e + M_{fus} + M_{emp}}{I_{yy}} \tag{2.8.5}$$

$$\dot{r} = \frac{pq\left(I_{yy} - I_{xx}\right) + N_{mr} + N_{tr} + N_e + N_{fus} + N_{emp}}{I_{zz}} \tag{2.8.6}$$

$$\dot{\phi} = p + q\sin(\phi)\tan(\theta) + r\cos(\phi)\tan(\theta) \tag{2.8.7}$$

$$\dot{\theta} = q\cos(\phi) - r\sin(\phi) \tag{2.8.8}$$

$$\dot{\psi} = q\sin(\phi)\sec(\theta) + r\cos(\phi)\sec(\theta) \tag{2.8.9}$$

$$\dot{X} = u - qZ + rY \tag{2.8.10}$$

$$\dot{Y} = v + pZ - rX \tag{2.8.11}$$

$$\dot{Z} = w - pY + qX \tag{2.8.12}$$

$$\dot{\beta}_{BL}^{long} = -\beta_{BL}^{long}\left(\frac{1}{\tau_{SB}}\right)+\left(\frac{1}{\tau_{SB}}\right)\left(\frac{1}{(1+S_{\beta}^{2})}\right)\left(\delta_{col}\left(\frac{\partial\beta_{BL}^{long}}{\partial\delta_{col}}\right)+\delta_{long}\left(\frac{\partial\beta_{BL}^{long}}{\partial\delta_{long}}\right)\right.$$
$$\left.+\delta_{lat}\left(\frac{\partial\beta_{BL}^{long}}{\partial\delta_{lat}}\right)+q\left(\frac{\partial\beta_{BL}^{long}}{\partial q}\right)+p\left(\frac{\partial\beta_{BL}^{long}}{\partial p}\right)+(\mu_{mr}^{z}-\lambda_{mr})\left(\frac{\partial\beta_{BL}^{long}}{\partial(\mu_{mr}^{z}-\lambda_{mr})}\right)\right) \tag{2.8.13}$$

$$\dot{\beta}_{BL}^{lat} = -\beta_{BL}^{lat}\left(\frac{1}{\tau_{SB}}\right)+\left(\frac{1}{\tau_{SB}}\right)\left(\frac{1}{(1+S_{\beta}^{2})}\right)\left(\delta_{col}\left(\frac{\partial\beta_{BL}^{lat}}{\partial\delta_{col}}\right)+\delta_{long}\left(\frac{\partial\beta_{BL}^{lat}}{\partial\delta_{long}}\right)\right.$$
$$\left.+\delta_{lat}\left(\frac{\partial\beta_{BL}^{lat}}{\partial\delta_{lat}}\right)+q\left(\frac{\partial\beta_{BL}^{lat}}{\partial q}\right)+p\left(\frac{\partial\beta_{BL}^{lat}}{\partial p}\right)+(\mu_{mr}^{z}-\lambda_{mr})\left(\frac{\partial\beta_{BL}^{lat}}{\partial(\mu_{mr}^{z}-\lambda_{mr})}\right)\right) \tag{2.8.14}$$

$$\dot{\Omega}_{mr} = -\dot{r}+\frac{Q_e-Q_{mr}-n_{tr}Q_{tr}}{I_{rot}} \tag{2.8.15}$$

$$\dot{Q}_e = -\frac{1}{\tau_{eng}}Q_e+\frac{1}{\tau_{eng}}\left(Q_e^0+K_{gov}(\Omega_c-\Omega_{mr})\right) \tag{2.8.16}$$

In short, the non-linear model will be referred to as:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}\left(\boldsymbol{x},\boldsymbol{u},t\right) \tag{2.8.17}$$

with $\boldsymbol{x}$ being the state vector, $\dot{\boldsymbol{x}}$ the time derivative of the state vector and $\boldsymbol{u}$ the actuator vector.

2.9 Linearised Model

Now that the non-linear model has been concluded, it has to be linearised about a trim condition so that an LTI (linear time invariant) model can be found. An LTI model is necessary for the type of control planned for this vehicle. Linearising a non-linear model about a trim condition is not a trivial task and there are a few criteria that should be met before such a task can be performed. These are:

- The model should not deviate much from its trim condition - should it do so, the linearisation assumptions would not hold
- The trim condition must be an equilibrium point (system derivatives must be zero at this point)

If these conditions are met, the non-linear system Equation (2.8.17) can be written as a Taylor series:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}\left(\boldsymbol{x}_T,\boldsymbol{u}_T\right)+\frac{\partial\boldsymbol{f}\left(\boldsymbol{x}_T,\boldsymbol{u}_T\right)}{\partial\boldsymbol{x}}\left(\boldsymbol{x}-\boldsymbol{x}_T\right)+\frac{\partial\boldsymbol{f}\left(\boldsymbol{x}_T,\boldsymbol{u}_T\right)}{\partial\boldsymbol{u}}\left(\boldsymbol{u}-\boldsymbol{u}_T\right)$$
$$+\frac{\partial^2\boldsymbol{f}\left(\boldsymbol{x}_T,\boldsymbol{u}_T\right)}{\partial\boldsymbol{x}^2}\left(\boldsymbol{x}-\boldsymbol{x}_T\right)^2+\frac{\partial^2\boldsymbol{f}\left(\boldsymbol{x}_T,\boldsymbol{u}_T\right)}{\partial\boldsymbol{u}^2}\left(\boldsymbol{u}-\boldsymbol{u}_T\right)^2+\text{higher order terms} \tag{2.9.1}$$

with $\boldsymbol{x_T}$ and $\boldsymbol{u_T}$ being the trim state and actuator vectors respectively.

The model was assumed to be close to its trim condition, so that any powers of these deviation errors would be negligibly small. If any terms past first order are ignored:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}_T, \boldsymbol{u}_T) + \frac{\partial \boldsymbol{f}(\boldsymbol{x}_T, \boldsymbol{u}_T)}{\partial \boldsymbol{x}}(\boldsymbol{x} - \boldsymbol{x}_T) + \frac{\partial \boldsymbol{f}(\boldsymbol{x}_T, \boldsymbol{u}_T)}{\partial \boldsymbol{u}}(\boldsymbol{u} - \boldsymbol{u}_T) \tag{2.9.2}$$

The second condition for linearisation states that the trim condition must be an equilibrium point, so by definition $\boldsymbol{f}(\boldsymbol{x_T}, \boldsymbol{u_T}) = 0$. The time varying LTI system is then written as:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}(t)\boldsymbol{x}_p(t) + \boldsymbol{B}(t)\boldsymbol{u}_p(t) \tag{2.9.3}$$

with

$$\boldsymbol{x}_p(t) = \boldsymbol{x}(t) - \boldsymbol{x}_T(t) \tag{2.9.4}$$

$$\boldsymbol{u}_p(t) = \boldsymbol{u}(t) - \boldsymbol{u}_T(t) \tag{2.9.5}$$

$$\boldsymbol{A}(t) = \left[\begin{array}{cccc} \left(\frac{\partial \dot{u}}{\partial u}\right) & \left(\frac{\partial \dot{u}}{\partial v}\right) & \left(\frac{\partial \dot{u}}{\partial w}\right) & \cdots \\ \left(\frac{\partial \dot{v}}{\partial u}\right) & \left(\frac{\partial \dot{v}}{\partial v}\right) & \left(\frac{\partial \dot{v}}{\partial w}\right) & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{array}\right]\Bigg|_{\boldsymbol{x}_T(t), \boldsymbol{u}_T(t)} \tag{2.9.6}$$

$$\boldsymbol{B}(t) = \left[\begin{array}{cccc} \left(\frac{\partial \dot{u}}{\partial \delta_{col}}\right) & \left(\frac{\partial \dot{u}}{\partial \delta_{long}}\right) & \left(\frac{\partial \dot{u}}{\partial \delta_{lat}}\right) & \left(\frac{\partial \dot{u}}{\partial \delta_{rudd}}\right) \\ \left(\frac{\partial \dot{v}}{\partial \delta_{col}}\right) & \left(\frac{\partial \dot{v}}{\partial \delta_{long}}\right) & \left(\frac{\partial \dot{v}}{\partial \delta_{lat}}\right) & \left(\frac{\partial \dot{v}}{\partial \delta_{rudd}}\right) \\ \vdots & \vdots & \vdots & \vdots \end{array}\right]\Bigg|_{\boldsymbol{x}(t), \boldsymbol{u}(t)} \tag{2.9.7}$$

The linearised system would be a good approximation, granted that the vehicle does not deviate too much from the trim condition and that the trim condition is an equilibrium point. The requirement not to deviate from trim can be enforced by continuously re-linearising the non-linear model (choosing a new trim condition closer to the actual state values). This will cause the trim condition not to be an equilibrium point, but if the movements are relatively slow, the error that they introduce can be assumed small and should be corrected by the control system.

The non-linear differential Equations (2.8.1) to (2.8.16) become quite complex to differentiate, especially those for the velocities, angular rates and flapping angles. However, the fact that parameters were used throughout the modelling process helped a great deal and these differential equations were linearised partially with respect to the parameters. The parameters could then be linearised separately and substituted back into the linearised differential equations when needed. Figure 2.15 and Table 2.1 show the open loop pole locations for the vehicle model linearised about the hover trim condition.

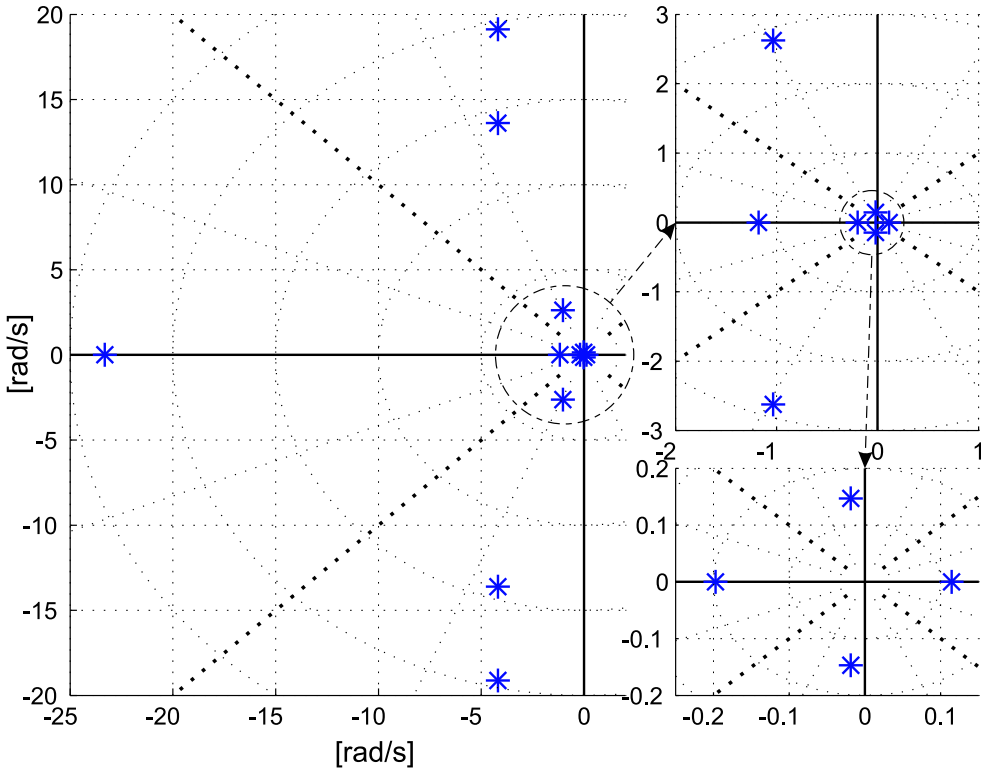


Figure 2.15: Open Loop Plant Poles

Mode	**Value** [rad/s]	ω_n [rad/s]	**Damping**
Longitudinal			
Pitch Rate	$-4.193 \pm 13.619i$	14.250	0.294
Phugoid	$-0.019 \pm 0.147i$	0.148	0.125
Lateral			
Roll Rate	$-4.196 \pm 19.127i$	19.582	0.214
Roll angle	0.114	0.114	unstable
Lateral Velocity	-0.197	0.197	1
Yaw			
Yaw Rate	-23.314	23.314	1
Heave			
Vertical Velocity	-1.179	1.179	1
Engine			
Blade Speed, Torque	$-1.035 \pm 2.623i$	2.820	0.367

Table 2.1: Hover Model Open Loop Poles

The longitudinal motion of the vehicle is dominated by four poles. The first two of these represent themselves as a very fast, but poorly damped imaginary pole pair. They describe the pitch rate response to longitudinal cyclic command,

which is very responsive, but also reasonably well damped by the workings of the stabilized bar (see Section A.1). The square of the natural frequency of these poles can be approximated as being directly proportional to the flapping angle-to-pitch angular acceleration gain, multiplied by the pitch rate-to-flapping angle gain, while the response speed of the stabilizer bar only influences the damping of the pitch rate poles:

$$\omega_{nq}^2 \approx \left(\frac{T_{mr}h_{mr}+K_\beta}{I_{YY}}\right)\left(\left(1+K_{SB}\right)S_\beta+\left(1+K_{SB}\frac{\gamma_{mr}}{\gamma_{SB}}\right)\left(\frac{16}{\gamma_{mr}}\right)\right)\left(\frac{1}{\Omega_{mr}}\right) \tag{2.9.8}$$

$$\zeta_q \approx \frac{1}{2\tau_{SB}\omega_{nq}} \tag{2.9.9}$$

The third and fourth poles of the longitudinal motion also come in the form of an imaginary pole pair. These poles are very poorly damped, but also extremely slow. Their oscillatory motion is caused by the effect where the longitudinal flapping angle decreases, with an increase in forward speed. The change in flapping angle causes a pitch rate acceleration, which eventually causes an increase in pitch rate, pitch angle and eventually a decrease in speed. This mode is normally completely unstable because of the very weak velocity feedback term of the stabilizer bar, but the addition of the horizontal stabilizer fin on the tail adds enough velocity feedback to stabilize this mode.

The lateral motion of the vehicle is also dominated by four poles, where the first two are in the form of a complex pole pair, similar to the one in longitudinal motion. Their frequency and damping can be approximated by similar equations (with the rolling moment of inertia replacing the pitching inertia). The third and fourth poles do not manifest themselves in a mode, but because of no additional velocity feedback (as is the case with the longitudinal tail fin), the roll angle is represented by a relatively slow unstable real pole. The lateral velocity is represented by a slow stable real pole, because of the feedback lateral drag offers.

The yawing motion of the vehicle is dominated by a relatively fast real pole. This fast pole dominates the yaw rate dynamics. The speed of this pole is almost exclusively caused by the rate feedback of the rate gyro[9]. The location can be approximated by the vehicle's tail command to yaw acceleration gain multiplied by the gyro gain.

$$\omega_{nr}^2 \approx l_{tr}f_t\rho(\Omega_{tr}R_{tr})^2\pi R_{tr}^2\frac{\sigma_{tr}a_{tr}}{2}\left(\frac{K_{gyro}}{3}-\frac{K_{gyro}}{2}\frac{\partial\lambda_{tr}}{\partial\delta_{rudd}}-\frac{\partial\lambda_{tr}}{\partial\delta_r}\frac{1}{2}-\frac{l_{tr}}{2\Omega_{tr}R_{tr}}\right)\frac{1}{I_{zz}} \tag{2.9.10}$$

The heave motion of the vehicle is dominated by a stable real pole. This pole describes the vehicle's vertical velocity response to a collective step. The location is almost exclusively dominated by the size of thrust increase when descending (interestingly, vertical drag plays little part in the vertical response during hover). The

[9]The dynamics of the rest of the aircraft causes less than 5% of the total rate feedback.

speed of this pole can be approximated by the gain in thrust that vertical translation causes, divided by the vehicle's weight.

$$\omega_{nw}^2 \approx \rho(\Omega_{mr}R_{mr})^2\pi R_{mr}^2\frac{\sigma_{mr}a_{mr}}{2}\frac{1}{2}\left(\frac{1}{\Omega_{mr}R_{mr}}-\frac{\partial\lambda_{tr}}{\partial w}\right)\frac{1}{m} \tag{2.9.11}$$

The dynamics of the engine is largely dominated by an under damped, complex pole pair. These poles describe the motion between the speed of the blades and the torque of the engine. Their speed and damping can both be increased, by decreasing the engine lag, which is an important factor when trying to improve engine dynamics.

$$\omega_{ne}^2 \approx \sqrt{\frac{K_{gov}}{\tau_{eng}I_{rot}}} \tag{2.9.12}$$

$$\zeta_e \approx \sqrt{\frac{I_{rot}}{4K_{gov}\tau_{eng}}} \tag{2.9.13}$$

2.10 Summary

A full non-linear dynamic model was derived for the vehicle in normal flight in Equations (2.8.1) to (2.8.16). This non-linear model was further extended to account for conditions that do not occur during normal flight, like the ground effect in Equations (2.6.3) to (2.6.5) and runway landings in Equations (2.7.5) to (2.7.6). These models are very complete and accurate and will be used to accurately simulate and test the control system.

The non-linear model was linearised in a trim independent fashion to be used with an adaptive control system.

The vehicle dynamics during hover was analysed and described.

Chapter 3

Controller

This chapter will cover the design, optimization, implementation, testing and analysis of a control system for the X-Cell helicopter. Different methods of control will be discussed. Optimal control will be explained and a solution will be derived. The resulting controller will be tested using different manoeuvres which will cover an extended range of the flight envelope. A full sensitivity analysis will also be performed.

3.1 Different Control Methods

A few controller designs were considered and each one was analysed critically.

Simple pole placement was considered, but initial trials showed that the amount of control needed to keep the closed loop poles fixed in one desired location across the entire flight envelope was not feasible. Consideration was given to the adjustment of these pole locations for different flight conditions in order to reduce the amount of actuator control needed, while still achieving the desired responses. Specifying these changing pole locations, however, is a non-exact science and would require a considerable amount of design time to cover the extensive flight envelope required.

Various other projects (see [19]) achieved success with frequency based Hardy space controllers. These included optimal H^2 and sub-optimal H^∞ controllers[1], which generally result in high bandwidth solutions. Unfortunately, to take full advantage of these types of control, well defined frequency models for plant and sensor noise need to be available. Since these noise models were not available and with the limited knowledge of this type of control at Stellenbosch University, it was eliminated as an option in this project.

Non-linear control techniques were considered[2], which usually result in excel-

[1] A frequency based optimal control technique, where the controller is optimized between higher low frequency gain to reduce steady state offsets and increased high frequency disturbance rejection.

[2] Such as feedback linearisation or even Lyapunov.

lent solutions for small non-linear plants. However the sheer size and the vast number of non-linearities in the vehicle model made this type of control impractical.

Classical control techniques (more specifically variations of successive loop closure) are the most commonly used methods for basic control of RC helicopters. These controllers make use of the weak cross-axial dynamic coupling of helicopters to design separate controllers for each axis.

The controllers work in a layered structure, with the faster inner loops fed back first, followed by the slower outer loops. The design process is usually done for each decoupled model according to a set of specifications - these can be time domain[3] or frequency domain[4] specifications. This type of control is well suited for this project, although with the varying flight conditions, degraded performance and possible instability could result. To address this problem, successive loop closure can be used in conjunction with gain scheduling or fuzzy logic techniques[5] to accommodate the varying model. However this will also be a time-consuming design process.

Time varying full state optimal control was found to be the solution that satisfies most of the requirements of this project. This type of controller offers the advantage that, with a reasonably defined model the controller adapts to the flight conditions while still offering optimal full state control with more than adequate stability margins. Least squares optimal control (LQR) was chosen as the type of control to be used for this project. It uses an intuitive cost structure and is relatively easy to design and implement.

A previous project at Stellenbosch University by Carstens[3] successfully used consecutive loop closure to design a controller for a smaller size helicopter. It was decided to use this knowledge base to adapt the controller for the X-Cell, which will then serve as a comparative control method to evaluate the performance of the LQR controller.

3.2 Introduction Into Full State Optimal Control

The idea of optimal control refers to the application of control in order to minimize a certain cost defined by some cost structure. These costs may be applied to the states, the control inputs or even to some parameters within the state equations. For a multi-input, multi-output (MIMO) linear system it is convenient to weigh the states, inputs and a combination of these. The most common cost function for such

[3] Eg. rise time, settling time, etc.

[4] In the form of phase/gain margins or desired pole locations.

[5] A technique where multi-valued logic is used to test a number of inputs to decide a output. In control systems fuzzy logic is normally used to schedule different control structures, depending on the system states.

a system is:

$$J = \frac{1}{2}\sum_{k=k_i}^{k_f} \quad \boldsymbol{x}(k)^T \boldsymbol{Q}(k)\boldsymbol{x}(k) + \boldsymbol{x}(k)^T \boldsymbol{M}(k)\boldsymbol{u}(k) + \left(\boldsymbol{x}(k)^T \boldsymbol{M}(k)\boldsymbol{u}(k)\right)^T + \boldsymbol{u}(k)^T \boldsymbol{R}(k)\boldsymbol{u}(k) \tag{3.2.1}$$

where $\boldsymbol{Q}(k)$, $\boldsymbol{R}(k)$ and $\boldsymbol{M}(k)$ are positive definite, symmetrical matrices, indicating the weighting for the states, inputs and state-input cross weightings respectively. $[k_i, k_f]$ is the discrete time interval on which the cost is evaluated.

For a system with feedback that is a linear function of the system states (i.e. $\boldsymbol{u}(k) = -\boldsymbol{C}(k)\boldsymbol{x}(k)$), the cost function in Equation (3.2.1) can be minimised on the interval $[k_i, k_f]$, with respect to the feedback function:

$$\begin{aligned}
\boldsymbol{S}(k) &= \boldsymbol{Q}(k) + \boldsymbol{\Phi}(k)^T \boldsymbol{S}(k+1)\boldsymbol{\Phi}(k) \\
&- \left(\boldsymbol{M}(k) + \boldsymbol{\Phi}(k)^T \boldsymbol{S}(k+1)\boldsymbol{\Gamma}(k)\right) \\
&\quad \left(\boldsymbol{\Gamma}(k)^T \boldsymbol{S}(k+1)\boldsymbol{\Gamma}(k) + \boldsymbol{R}(k)\right)^{-1} \left(\boldsymbol{\Gamma}(k)^T \boldsymbol{S}(k+1)\boldsymbol{\Phi}(k) + \boldsymbol{M}(k)\right) \\
\boldsymbol{C}(k) &= \left(\boldsymbol{\Gamma}(k)^T \boldsymbol{S}(k+1)\boldsymbol{\Gamma}(k) + \boldsymbol{R}(k)\right)^{-1} \left(\boldsymbol{\Gamma}(k)^T \boldsymbol{S}(k+1)\boldsymbol{\Phi}(k) + \boldsymbol{M}(k)\right)
\end{aligned} \tag{3.2.2}$$

Equation (3.2.2) takes on the form of a *Discrete Riccati Matrix Equation* [21]. In order to solve $\boldsymbol{S}$ and thus $\boldsymbol{C}$, at least one boundary condition needs to be known (for a full derivation of the *Discrete Riccati Matrix Equation*, refer to Appendix B.1). $\boldsymbol{S}$ signifies the minimum sum of the total future cost, as a coefficient of the square of the current state vector $\boldsymbol{x}(k)$, i.e.:

$$J(k_i, k_f)_{min} = \frac{1}{2}\boldsymbol{x}(k_i)^T \boldsymbol{S}(k_i, k_f)\boldsymbol{x}(k_i) \tag{3.2.3}$$

If it is assumed that k_f is sufficiently far into the future, then the lowest possible cost can be achieved for a given $\boldsymbol{x}(k)$ at time step $k = k_f$, i.e.:

$$J(k_f, k_f)_{min} = min\left\{\tfrac{1}{2}\boldsymbol{x}(k_f)^T \boldsymbol{Q}(k_f)\boldsymbol{x}(k_f) + \boldsymbol{x}(k_f)^T \boldsymbol{M}(k_f)\boldsymbol{u}(k_f) + \tfrac{1}{2}\boldsymbol{u}(k_f)^T \boldsymbol{R}(k_f)\boldsymbol{u}(k_f)\right\} \tag{3.2.4}$$

$$\Rightarrow \boldsymbol{u}(k_f) = 0 \tag{3.2.5}$$

The optimal solution in Equation (3.2.3) can be compared to this optimal final value by setting $k_i = k_f$:

$$\begin{aligned}
\frac{1}{2}\boldsymbol{x}(k_f)^T \boldsymbol{S}(k_f)\boldsymbol{x}(k_f) &= \frac{1}{2}\boldsymbol{x}(k_f)^T \boldsymbol{Q}(k_f)\boldsymbol{x}(k_f) \\
\Rightarrow \boldsymbol{S}(k_f) &= \boldsymbol{Q}(k_f)
\end{aligned} \tag{3.2.6}$$

This optimal condition in Equation (3.2.6) can now be used as the necessary boundary condition to solve the Riccati Equation backwards from $k = k_f$ to $k = k_{i+1}$. As soon as a value is available for $\boldsymbol{S}(k_{i+1})$, the current-time feedback gain $\boldsymbol{C}(k_i)$ can be solved using Equation (3.2.2).

3.3 Controller Design

As can be seen from Equation (3.2.2), to solve the current time feedback gain, a number of future values of parameters are needed. There should be enough future knowledge to allow the backward solving of the Riccati Equation to produce a stabilised answer $\boldsymbol{S}(k_i)$. The required parameters are:

1. The state, input and cross weightings, $\boldsymbol{Q}(k)$, $\boldsymbol{R}(k)$ and $\boldsymbol{M}(k)$ respectively, for the future time interval $[k_{i+1}, k_f]$
2. The discrete plant model $\boldsymbol{\Phi}(k)$ and $\boldsymbol{\Gamma}(k)$, for the future time interval $[k_{i+1}, k_f]$

Usually the first condition is easily met, as these weightings are chosen to be constant on any interval. The second condition is not as easily met, as no future information is available on the vehicle states. A good approximation of the vehicle states can be made if some form of trajectory is available, but even then there is no guarantee that the vehicle will stay on this predefined trajectory.

In a previous project done at Stellenbosch University [6], the assumption was made that the vehicle does not deviate too much from its predefined trajectory and then the trajectory linearised model was used to solve Equation (3.2.2). Reasonable results were obtained, but for this project there will be flight conditions where no trajectory can be predefined (flight path changes in real time). An alternative solution had to be found to obtain the future plant model.

When inspecting the natural bandwidth of the linearised hover system in Section 2.9 it can be seen that, with the exception of the heading plant, the other dynamics of the vehicle are relatively slow. Given the relatively slow system, it is fair to assume that the discrete system model ($\boldsymbol{\Phi}(k)$ and $\boldsymbol{\Gamma}(k)$) will not change too much in a short time interval. For the given vehicle, this assumption will hold for 15 to 20 time steps (300 to 400ms). Equation (3.2.2) can now be solved by assuming the system matrices ($\boldsymbol{\Phi}(k)$ and $\boldsymbol{\Gamma}(k)$) are constant and equal to the current time system over the interval.

3.3.1 Model Discretization

To implement the LQR controller practically, a discrete system model is required. There are many accurate ways to obtain such a system, but most of them are computationally intensive. It has been established in the previous section that the system bandwidth is much slower than the required control frequency. For the sampling time of T_s, consider the continuous system in Equation (2.9.3):

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}(t)\boldsymbol{x}_p(t) + \boldsymbol{B}(t)\boldsymbol{u}_p(t) \tag{3.3.1}$$

By making use of Euler integration the states at time $t = t_i + T$ are:

$$\begin{aligned} \boldsymbol{x}_p(t_i + T_s) &= \boldsymbol{x}_p(t_i) + T_s\dot{\boldsymbol{x}}(t_i) && (3.3.2)\\ \Rightarrow \boldsymbol{x}_p(t_i + T_s) &= \boldsymbol{x}_p(t_i) + T_s\left(\boldsymbol{A}(t_i)\boldsymbol{x}_p(t_i) + \boldsymbol{B}(t_i)\boldsymbol{u}_p(t_i)\right) && (3.3.3)\\ \boldsymbol{x}_p(t_i + T_s) &= \left(\boldsymbol{I} + \boldsymbol{A}(t_i)T_s\right)\boldsymbol{x}_p(t_i) + \left(\boldsymbol{B}(t_i)T_s\right)\boldsymbol{u}_p(t_i) && (3.3.4) \end{aligned}$$

The discrete system matrices can now be approximated as:

$$\mathbf{\Phi}(k) = \boldsymbol{I} + \boldsymbol{A}(t_k)T_s \quad (3.3.5)$$
$$\mathbf{\Gamma}(k) = \boldsymbol{B}(t_k)T_s \quad (3.3.6)$$

3.3.2 Integrators

To allow for the control system to accurately hold its reference position, given an array of environmental disturbances, integrators have to be added to the control. This will ensure that the required states converge to a zero steady state error. Adding these integrators to the faster, inner loop states is recommended since they will eliminate errors faster and increase the controller's performance.

Since the controller will be verified using practical implementation and testing on the vehicle, the available estimator structure was evaluated. It was found that none of the states from the estimator could be guaranteed to be without a constant error offset[6] and practical tests confirmed that attitude estimate errors of 4-6 degrees and velocity errors of 0.3-0.8 [m/s] occur frequently.

This led to the conclusion that a practical implementation with integrators on these states is not possible, since these estimated states cannot be guaranteed to be zero and in equilibrium at the same time. As a result, the integrators were implemented only on the states that do not form part of the model dynamics: Body position offsets (X, Y, Z) and heading angle (ψ). An alternative to this problem will be discussed in Section (3.8).

The discrete system in Equations (3.3.5) and (3.3.6) can be augmented to include these integrators, but first a discrete model for the integrators needs to be obtained. Using Euler integration, consider for X:

$$X_I(t_i + T_s) = X_I(t_i) + T_s X(t_i) \quad (3.3.7)$$

Using this, the discrete integrators can be approximated as:

$$X_I(k+1) = X_I(k) + X(k)\,T_s \quad (3.3.8)$$
$$Y_I(k+1) = Y_I(k) + Y(k)\,T_s \quad (3.3.9)$$
$$Z_I(k+1) = Z_I(k) + Z(k)\,T_s \quad (3.3.10)$$
$$\psi_I(k+1) = \psi_I(k) + \psi(k)\,T_s \quad (3.3.11)$$

[6]The estimator has no sensor bias estimation, so biases in the sensor measurements create offsets in the estimated states.

The discrete system is now augmented as follows:

$$\boldsymbol{X}_A(k) = \begin{bmatrix} u & v & w & p & q & r & \phi & \theta & \psi & X & Y & Z & \beta_{BL}^{long} & \beta_{BL}^{lat} & \Omega_{mr} & Q_e & X_I & Y_I & Z_I & \psi_I \end{bmatrix} \tag{3.3.12}$$

$$\boldsymbol{\Phi}_I(k) = \begin{matrix} \\ \\ \\ \\ \end{matrix} \begin{matrix} \quad & \psi & X & Y & Z & \quad & X_I & Y_I & Z_I & \psi_I \end{matrix} \\ \begin{bmatrix} \dots & 0 & T_s & 0 & 0 & \dots & 1 & 0 & 0 & 0 \\ \dots & 0 & 0 & T_s & 0 & \dots & 0 & 1 & 0 & 0 \\ \dots & 0 & 0 & 0 & T_s & \dots & 0 & 0 & 1 & 0 \\ \dots & T_s & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.3.13}$$

$$\boldsymbol{\Gamma}_I(k) = \boldsymbol{0} \tag{3.3.14}$$

$$\boldsymbol{\Phi}_A(k) = \begin{bmatrix} \boldsymbol{\Phi}(k) & \boldsymbol{0} \\ & \\ \boldsymbol{\Phi}_I(k) & \end{bmatrix} \tag{3.3.15}$$

$$\boldsymbol{\Gamma}_A(k) = \begin{bmatrix} \boldsymbol{\Gamma}(k) \\ \boldsymbol{0} \end{bmatrix} \tag{3.3.16}$$

3.3.3 Weightings

It might seem like quite a trivial matter to choose weighting matrices for the inputs and states, but in a practical system this can become quite a difficult task. This system alone has 24 weighting variables (diagonals of the weighting matrices). If no clear guidelines are set for the process of choosing these weightings, the task can end up being a long and tedious iterative process of choosing weightings, calculating gains, simulating the control system and analysing results.

It can be seen that the cost function is a square function of the state deviations and plant inputs. It makes sense that if this function should stay dimensionless, the diagonals of the weighting matrices should be proportional to the inverse of these squares:

$$\boldsymbol{Q}[i,i] \propto \frac{1}{(\boldsymbol{X}_c[i,i])^2} \tag{3.3.17}$$

$$\boldsymbol{R}[i,i] \propto \frac{1}{(\boldsymbol{u}_c[i,i])^2} \tag{3.3.18}$$

where $\boldsymbol{X}_c$ and $\boldsymbol{u}_c$ denote some constraint function for the states and inputs respectively. In several literary references (see [22] and [23]) it is suggested that these constraints should be the maximum allowable errors in the states and the maximum allowable control to be used. Also, some scaling between the input and state weightings is suggested to balance the importance of these two effects:

$$\boldsymbol{Q}[i,i] \approx \frac{1}{(\Delta\boldsymbol{X}_{max}[i,i])^2} \tag{3.3.19}$$

$$\boldsymbol{R}[i,i] \approx (K_{SF}) \frac{1}{(\Delta\boldsymbol{u}_{max}[i,i])^2} \tag{3.3.20}$$

where K_{SF} is a scaling factor between the state and input weightings.

As a first iteration, these matrices can be set up and the scaling factor can be adjusted until the best results are obtained. Small adjustments can then be made to further improve the results.

For the helicopter the following maximum deviations were decided upon initially:

State	$\Delta \boldsymbol{X}_{max}$	Unit	State	$\Delta \boldsymbol{X}_{max}$	Unit
u	0.5	$[m/s]$	X	0.5	$[m]$
v	0.5	$[m/s]$	Y	0.5	$[m]$
w	0.5	$[m/s]$	Z	0.5	$[m]$
p	30	$[^\circ/s]$	ϕ	20	$[^\circ]$
q	30	$[^\circ/s]$	θ	20	$[^\circ]$
r	30	$[^\circ/s]$	ψ	10	$[^\circ]$

Table 3.1: LQR Weighting Maximum Acceptable State Deviations

State	$\Delta \boldsymbol{u}_{max}$	Unit
δ_{col}	2	$[^\circ]$
δ_{long}	1	$[^\circ]$
δ_{lat}	1	$[^\circ]$
δ_{rudd}	3	$[^\circ]$

Table 3.2: LQR Weighting Maximum Acceptable Actuator Deviations

With these weightings reasonable hover results were obtained. From Table 3.3 it is clear that the closed loop model is stable with an adequate stability margin.

The process now continues to optimise these weightings for the type of flight that might be expected. Instead of creating a separate set of weightings for every different type of flight condition, the weightings will be optimised in a way that would ensure that one set will produce reasonable results throughout the entire flight envelope. This way the controller stays generic and makes the process of transitioning from one flight condition to the next much easier. Before the weightings are listed, consider the flight envelope:

The flight envelope is defined for precise hover flight, with low frequency horizontal wind conditions up to speeds of 5 m/s from any direction and vertical drafts of up to 3 [m/s]. Wind gusts of up to 3 [m/s] should also be handled. Horizontal forward flight of up to 90% of the modelling boundary (advance ratio of 0.15 - approximately 20 [m/s]). Backward flight of up to 5 [m/s] and horizontal lateral motion of up to 5 [m/s] (when at low forward/backward speed). Ascending and decending of up to a maximum rate of 4 [m/s] .

	Uncontrolled		Controlled	
Mode	ω_n [rad/s]	**Damping**	ω_n [rad/s]	**Damping**
Longitudinal				
Pitch Rate	14.250	0.294	14.837	0.269
Phugoid	0.148	0.125	1.563	0.332
Translational	0	1	0.272	0.986
Lateral				
Roll Rate	19.582	0.214	20.333	0.205
Phugoid	0.114 0.197	unstable 1	1.431	0.283
Translational	0	1	0.414 0.236	1 1
Yaw				
Yaw Rate	23.314	1	23.431	1
Rotational	0	1	0.266	0.760
Heave				
Vertical Velocity	1.179	1	1.853	1
Translational	0	1	0.201	0.654
Engine				
Blade Speed, Torque	2.820	0.367	2.787	0.371

Table 3.3: Hover Model Open Loop and Closed Loop Poles

These flight envelope boundaries form the basis for a layered control system where the user can change the required translation speeds of the vehicle in real time using the radio control remote. The inner layers of the control system stabilise the vehicle, limit the maximum accelerations and ensure that flight stays inside the intended flight envelope while trying to meet user commands. As a result, any user, however limited his piloting skills, can safely fly and steer the vehicle using high level input commands.

It was decided to perform two aerobatic manoeuvres to test the controller. The first was a barrel roll. This manoeuvre is normally started by straight and level flight, where after the vehicle performs a full 360 [°] continuous lateral roll and ends the manoeuvre by continuing straight and level flying. As little as possible altitude and forward speed should be lost during the execution of this manoeuvre. This manoeuvre (illustrated in Figure 3.1) incorporates a relatively high forward speed, high roll rate and momentary inverted flight.

The second manoeuvre was a 180 degree stall turn. This manoeuvre was normally started by straight and level flying, where after the vehicle performs a longitudinal climb in the shape of a circular path. As the vehicle stalls, the nose is turned through 180 [°] and the vehicle ascends downward along the same circular path. The vehicle should exit the circular path at the same point it entered it and continue with straight and level flight in the opposite direction. This manoeuvre (illustrated in Figure 3.2) incorporates relatively high forward speeds, high pitch and yaw rates, tail slip, vertical translation and manoeuvring during stall. Together these two manoeuvres cover a large range of the flight envelope and will

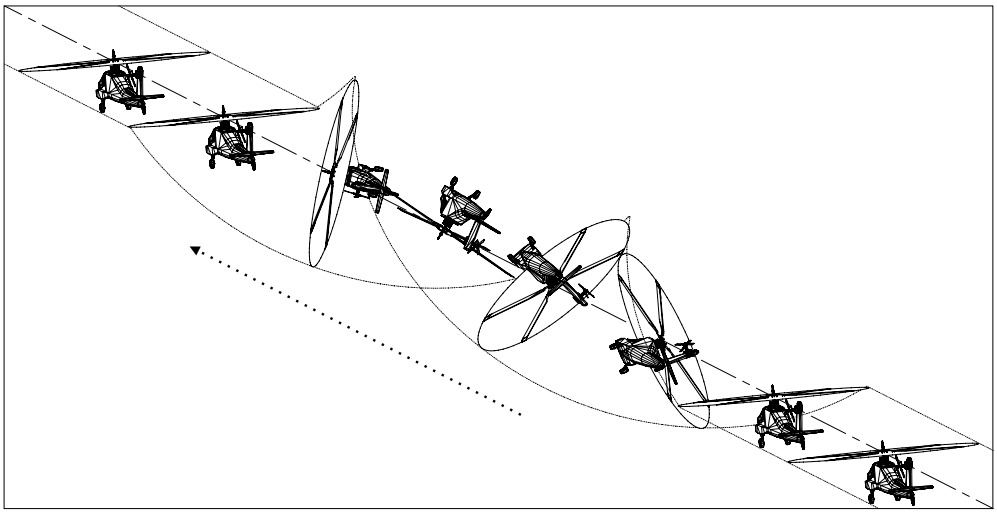


Figure 3.1: Illustration of a Barrel Roll Manoeuvre

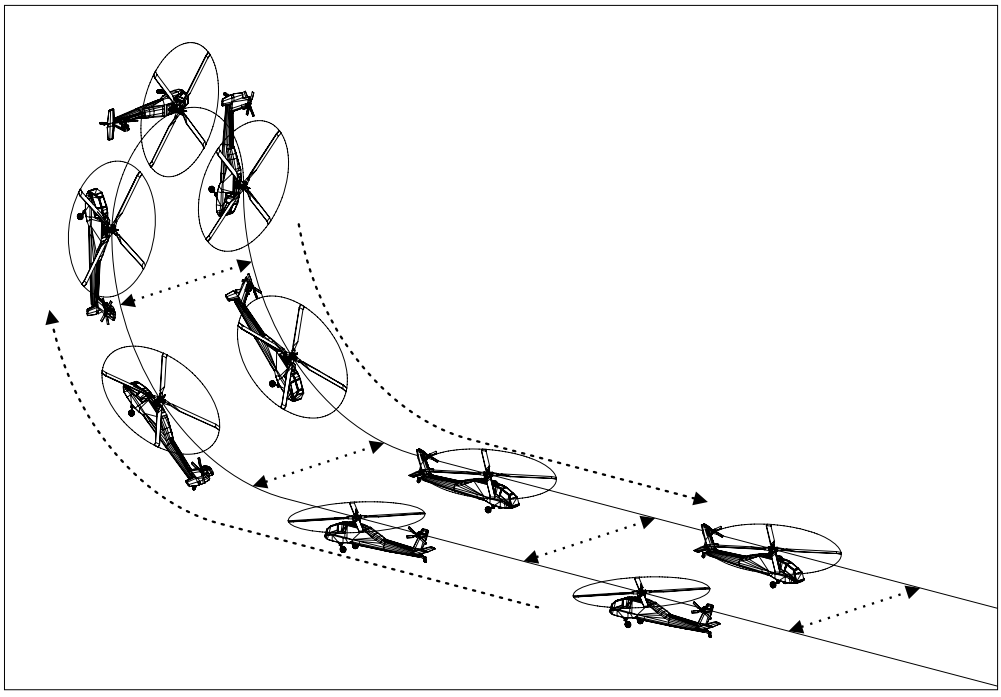


Figure 3.2: Illustration of a 180° Stall Turn Manoeuvre (Laterally Separated for Easier Viewing)

demonstrate the controller's ability to perform agile manoeuvres.

For full state feedback, flight conditions must be defined by some form of trajectory or reference structure.

3.4 Trajectories

For real-time assisted flight control it was decided not to use a real-time full dynamic trajectory as this trajectory is not analytically solvable and would require extensive iterative calculation to solve. An approximated reference structure was used, where the equilibrium values for the important states are calculated for each translational command. The acceleration of these references will be limited to prevent an amateur pilot from giving commands that will cause the vehicle to perform outside the intended flight envelope. Limiting these accelerations will cause the dynamic effects that are ignored when calculating the references to be bounded and will be managed by the controller as if it is simply plant noise. As the controller is fairly stable at low speeds, it was decided not to model the coupling between the references in different axes. At higher speeds the cross coupling are limited in any case.

For the longitudinal model the following equations hold for equilibrium:

$$\dot{u} = -g\sin(\theta) + \frac{X_{mr} + X_{fus}}{m} = 0 \tag{3.4.1}$$

$$\dot{w} = g\cos(\phi)\cos(\theta) + \frac{Z_{mr} + Z_{fus} + Z_{emp}}{m} = 0 \tag{3.4.2}$$

$$\dot{q} = \frac{M_{mr} + M_{emp}}{I_{yy}} = 0 \tag{3.4.3}$$

$$V_{forward} = \cos(\theta)\,u + \sin(\phi)\sin(\theta)\,v + \cos(\phi)\sin(\theta)\,w \tag{3.4.4}$$

$$V_{vertical} = -\sin(\theta)\,u + \sin(\phi)\cos(\theta)\,v + \cos(\phi)\cos(\theta)\,w \tag{3.4.5}$$

$$\begin{aligned}\dot{\beta}_{BL}^{long} = {} & -\beta_{BL}^{long}\left(\frac{1}{\tau_{SB}}\right) + \left(\frac{1}{\tau_{SB}}\right)\left(\frac{1}{(1+S_{\tilde{\beta}}^2)}\right)\left(\delta_{col}\left(\frac{\partial\beta_{BL}^{long}}{\partial\delta_{col}}\right) + \delta_{long}\left(\frac{\partial\beta_{BL}^{long}}{\partial\delta_{long}}\right)\right. \\ & \left. +\delta_{lat}\left(\frac{\partial\beta_{BL}^{long}}{\partial\delta_{lat}}\right) + (\mu_{mr}^{z} - \lambda_{mr})\left(\frac{\partial\beta_{BL}^{long}}{\partial\,(\mu_{mr}^{z} - \lambda_{mr})}\right)\right) = 0\end{aligned} \tag{3.4.6}$$

The 6 Equations 3.4.1 to 3.4.6 have the 6 unknowns: u, w, θ, β_{BL}^{long}, δ_{long}, δ_{col}, for any value of the commanded forward and vertical speed $V_{forward}$ and $V_{vertical}$. Substitution was used to get a non-empirical function of u, which was then solved, using the Newton Raphson [24] numerical method. The remaining variables are solved using substitution.

For the lateral model, Equations 3.4.7 to 3.4.12 hold in equilibrium:

$$\dot{v} = g\sin(\phi)\cos(\theta) + \frac{Y_{mr} + Y_{tr} + Y_{fus} + Y_{emp}}{m} = 0 \quad (3.4.7)$$

$$\dot{w} = g\cos(\phi)\cos(\theta) + \frac{Z_{mr} + Z_{fus} + Z_{emp}}{m} = 0 \quad (3.4.8)$$

$$\dot{p} = \frac{L_{mr} + L_{tr} + L_{emp}}{I_{xx}} = 0 \quad (3.4.9)$$

$$V_{lateral} = \cos(\phi)\, v - \sin(\phi)\, w \quad (3.4.10)$$

$$V_{vertical} = -\sin(\theta)\, u + \sin(\phi)\cos(\theta)\, v + \cos(\phi)\cos(\theta)\, w \quad (3.4.11)$$

$$\dot{\beta}_{BL}^{lat} = -\beta_{BL}^{lat}\left(\frac{1}{\tau_{SB}}\right) + \left(\frac{1}{\tau_{SB}}\right)\left(\frac{1}{(1+S_{\beta}^{2})}\right)\Bigg(\delta_{col}\left(\frac{\partial\beta_{BL}^{lat}}{\partial\delta_{col}}\right) + \delta_{long}\left(\frac{\partial\beta_{BL}^{lat}}{\partial\delta_{long}}\right) + \delta_{lat}\left(\frac{\partial\beta_{BL}^{lat}}{\partial\delta_{lat}}\right)$$
$$+q\left(\frac{\partial\beta_{BL}^{lat}}{\partial q}\right) + p\left(\frac{\partial\beta_{BL}^{lat}}{\partial p}\right) + (\mu_{mr}^{z} - \lambda_{mr})\left(\frac{\partial\beta_{BL}^{lat}}{\partial(\mu_{mr}^{z} - \lambda_{mr})}\right)\Bigg) = 0 \quad (3.4.12)$$

The 6 Equations 3.4.7 to 3.4.12 also have 6 unknowns: v, w, ϕ, β_{BL}^{lat}, δ_{lat}, δ_{col}, for any value of the commanded lateral and vertical speed $V_{lateral}$ and $V_{vertical}$. Again substitution and Newton Raphson [24] were used to solve the variables.

Finally, for the azimuth model, Equations 3.4.13 to 3.4.14 hold in equilibrium:

$$\dot{r} = \frac{N_{tr} + N_{e} + N_{emp}}{I_{zz}} = 0 \quad (3.4.13)$$

$$\dot{\psi} = q\sin(\phi)\sec(\theta) + r\cos(\phi)\sec(\theta) \quad (3.4.14)$$

Equations 3.4.13 to 3.4.14 have the unknown: δ_{rudd} for any value of the commanded heading rate $\dot{\psi}$. Equation (3.4.13) can be rewritten in its closed form to produce an answer for δ_{rudd}.

Figure 3.3 shows these equilibrium references for different commanded values of horizontal and vertical speeds, as well as different yaw rates.

Figure 3.3: Equilibrium Reference Values

The creation of the reference trajectories for the aerobatic manoeuvres requires more detail, as the vehicle will be driven into a much more unstable state than the partially stable model at hover. During the manoeuvre, the stability margin would most likely be decreased and a smaller error margin will be required in the trajectory specification.

It was decided to use a scheme where the important states, the ones that define the manoeuvre, are specified in such a way that their derivatives are continuous and change slow enough for the vehicle to achieve them. These states are then used as references for a differential controller. No noise is added to the system, which would result in a continuous differential signal. If the controller followed the reference commands reasonably well, the resulting states are recorded and used as the basis for the manoeuvre trajectory. Thorough testing can then be conducted

to evaluate the performance of the controller on the trajectory with noise added to the system.

For the barrel roll manoeuvre it is important to follow a set roll rate (and commanded roll angle), while losing little of the forward speed of the vehicle. These then made up the core specifications for the manoeuvre. The forward speed and roll rate were approximated by visually inspecting stunts done by a pilot, using a similar airframe. The resulting trajectory from flying the differential controller in simulation is [7]:

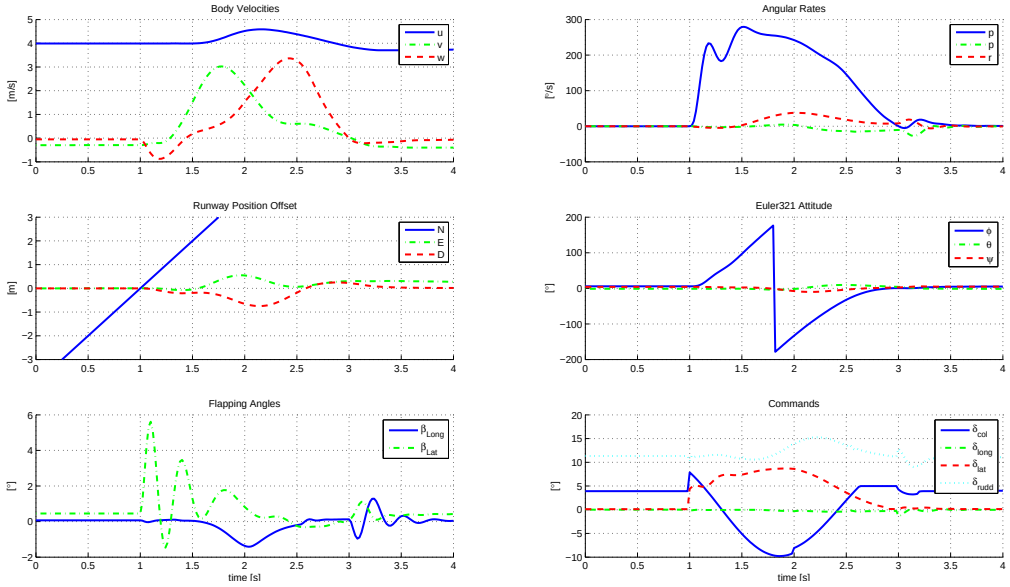


Figure 3.4: State Trajectory for Barrel Roll Manoeuvre

For the stall turn manoeuvre, it is important to follow a set arc (preferably a circular path) as the vehicle pulls up into a vertical climb. In aerobatic competitions the stunt is considered very well executed if the vehicle can exit the arc at exactly the same point in the air where it entered it (with approximately the same speed). Also, care has to be taken with the pitch rate of the vehicle, as a large pitch rate in the initial fast section of the pull-up can result in large centrifugal forces that can damage the airframe. Again the chosen pitch angle, pitch rate and longitudinal and vertical position curves were based on observations made during real life stunts done by a pilot, using a similar airframe. The resulting trajectory from flying the differential controller in simulation is [8]:

[7] For convenience of plotting the manoeuvre was executed in the northerly direction and attitude was described using the Euler 321 convention.

[8] For convenience of plotting the manoeuvre was executed in the northerly direction and attitude was described using the Euler 321 convention.

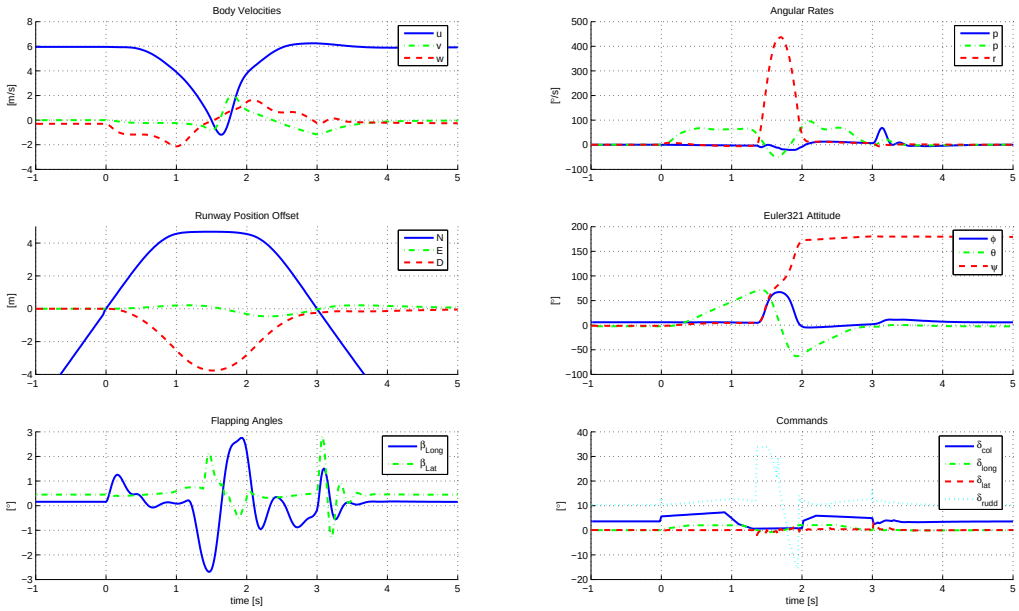


Figure 3.5: State Trajectory for Stall Turn Manoeuvre

3.5 Final LQR Weightings

Now that the reference structure has been properly defined for all intended flight conditions, the LQR weightings in Section 3.3.3 can be optimised to achieve reasonably good results under all flight conditions. It was found that the stall turn and barrel roll require aggressive attitude and rate controllers, while the assisted flight system performs best with lower bandwidth inner loop controllers. The art was in finding a balance. The resulting weightings are:

State	ΔX_{max}	Unit	State	ΔX_{max}	Unit
u	0.3	$[m/s]$	X	1	$[m]$
v	0.3	$[m/s]$	Y	1	$[m]$
w	0.3	$[m/s]$	Z	1	$[m]$
p	15	$[°/s]$	ϕ	15	$[°]$
q	15	$[°/s]$	θ	15	$[°]$
r	15	$[°/s]$	ψ	5	$[°]$
β_{BL}^{long}	25	$[°]$	Ω_{mr}	42	$[rad/s]$
β_{BL}^{lat}	25	$[°]$	Q_e	3.15	$[Nm]$
X_I	1	$[m]$	ψ_I	5	$[°]$
Y_I	1	$[m]$			
Z_I	1	$[m]$			

Table 3.4: Final LQR Weighting Maximum Acceptable State Deviations

State	$\Delta \boldsymbol{u}_{max}$	Unit
δ_{col}	1	$[^\circ]$
δ_{long}	0.5	$[^\circ]$
δ_{lat}	0.5	$[^\circ]$
δ_{rudd}	1.5	$[^\circ]$

Table 3.5: Final LQR Weighting Maximum Acceptable Actuator Deviations

The weightings for the integrator states, flapping angles, blade speed and engine torque were also added to the weighting matrix. It can be seen in Table 3.4 that the two latter weightings are relatively high, as these estimates are very inaccurate and the states relatively difficult to control. It will be seen in Section 3.7 through sensitivity analysis why it is important not to weigh states too strongly when they have poor estimates.

3.6 Gain Settling

In order to solve the Riccati differential equation in (3.2.2), a boundary condition (the future steady state optimal control solution) was used. This resulted in a numerical iterative process which converges to the current time solution. Special care has to be taken to ensure that a sufficient number of iterations are applied to converge the solution, but at the same time the process is computationally very demanding given the large matrix multiplications and inversions. An unconverged solution will result in a non-optimal and possibly unstable feedback gain.

The settling of the Riccati solution was evaluated for multiple flight conditions. Little variation was noticed for a change in flight conditions. The gains tend to settle a bit more slowly during high speed forward flight, so this was used as the worst case scenario. Figure 3.6 shows these gains as a function of the number of iterations in the calculation.

It was found that some of the gains take more than 450 iterations to settle to within 95% of their final value. Initial testing suggested that processing power would only allow for 8-12 iterations to be safely calculated every 20 [ms]. This will result in the controller to be adapted only every 1 second, in which time the model could have changed significantly. It was found that by changing certain weightings - especially those that influence the integrators, the settling time can be reduced. But with the already vast array of flight conditions and manoeuvres imposing limits on the weightings, it was decided to seek an alternative solution.

It was noted that in the iterative solution, $\boldsymbol{S}$ signifies the minimum sum of the total future cost ($J\left(k_i, k_f\right)_{min}$), as a coefficient of the square of the current state vector $\boldsymbol{x}\left(k\right)$, i.e.:

$$J\left(k_i, k_f\right)_{min} = \frac{1}{2}\boldsymbol{x}\left(k_i\right)^T \boldsymbol{S}\left(k_i, k_f\right)\boldsymbol{x}\left(k_i\right) \tag{3.6.1}$$

So, if a future solution is available for $\boldsymbol{S}$, somewhere between k_f and k_i, then by definition this value will also be optimal given the state equations of the time.

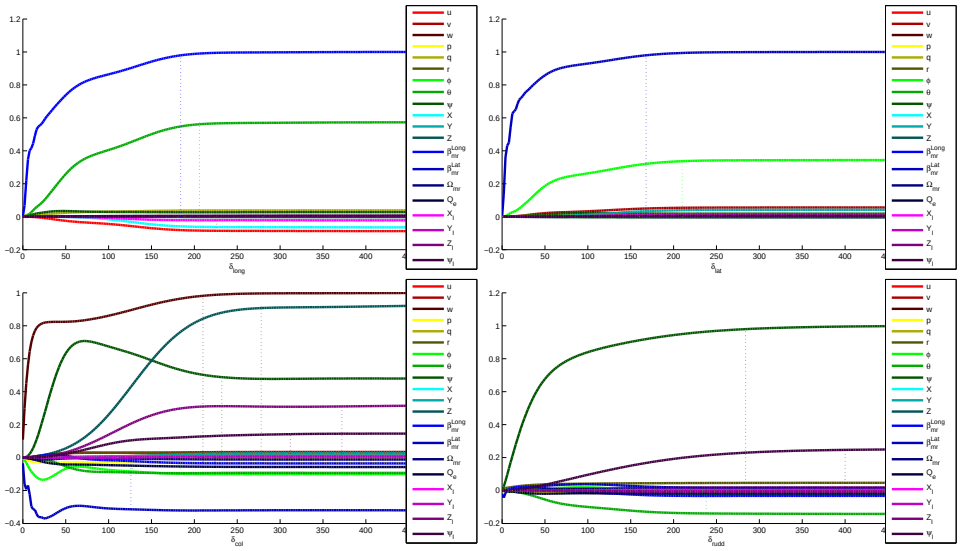


Figure 3.6: LQR Gain Calculation from $S_f = Q$

This intermediate value for $\boldsymbol{S}$ can then be used as a new boundary condition for Equation (3.2.2).

An optimal solution for $\boldsymbol{S}$ was available from calculating the previous gain. This, however is calculated from the old state equations and not from the future model. Since steady state LQR is implied, the direction of model changes is not important and $\boldsymbol{S}$ will be a valid initial condition to solve the iterative process.

It was found that by applying this method the settling time of the LQR gains was drastically reduced. The same solution as in Figure 3.6 was applied, only with the initial value of $\boldsymbol{S}$ being from a hover solution (6 [m/s] slower). Figure 3.7 show the gain settling times, when using this method.

The results showed that all the gains settled to within 95% of their final values (from their initial values) in less than 120 iterations. With the current processing power, this will allow for the controller to be comfortably updated every 100-150 [ms].

3.7 Robustness and Sensitivity

LQR is a very model dependent type of control and care should always be taken that not too much trust is placed in a potentially inaccurate model. When the LQR gains are based on an inaccurate model, or the state measurements are noisy or offset, stability can no longer be guaranteed by the LQR design.

In an effort to analyse how well the controller would fare given certain parameter errors in the model, the closed loop pole locations were calculated for a system with varying parameters, while the controller was kept constant (unaware of the model changes).

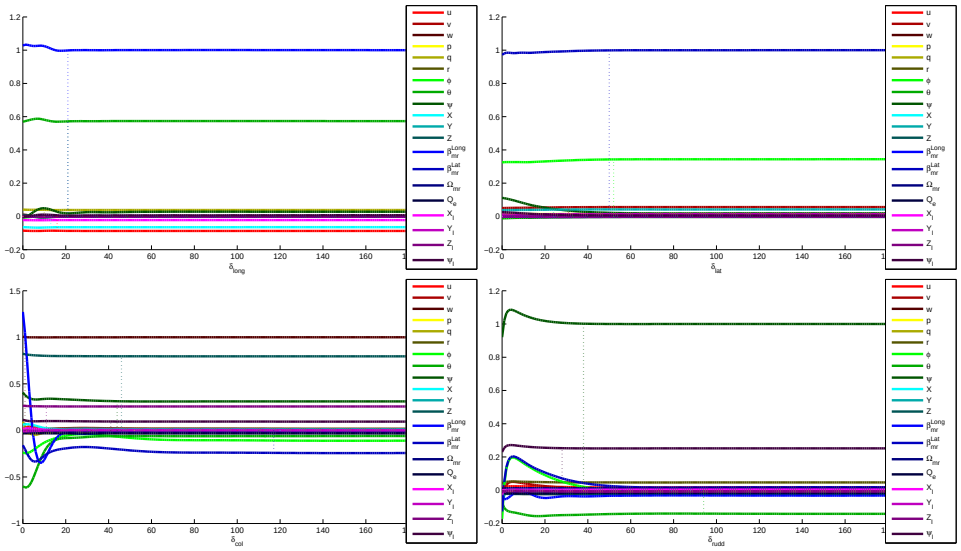


Figure 3.7: LQR Gain Calculation for $S_f = S_{k-1}$

Figure 3.8 and 3.9 show the pole movements during hover and forward flight respectively, for varying the most significant model parameters (those that affect the model most). The parameters are varied according to their predicted maximum inaccuracies.

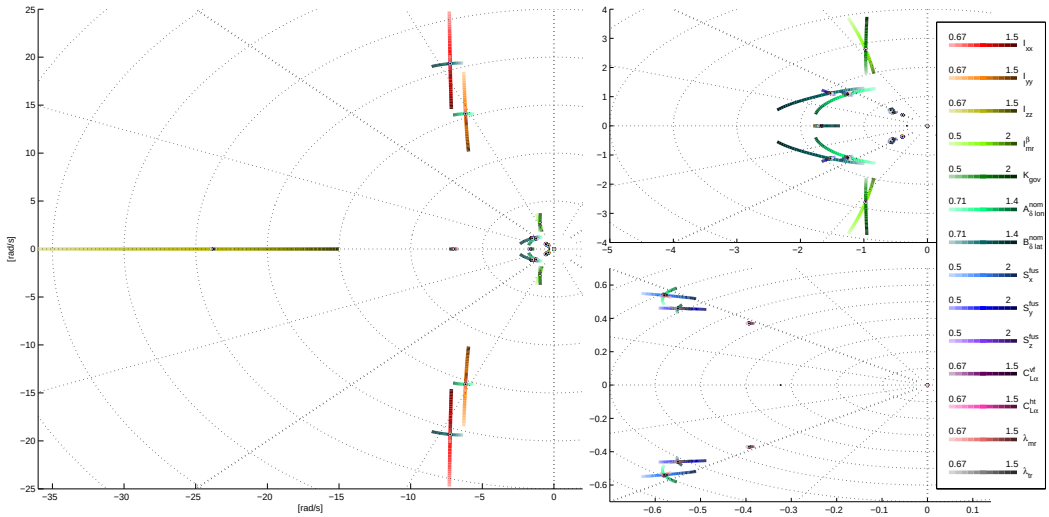


Figure 3.8: Hover Model Sensitivity to Parameter Changes

As expected, the body angular rate poles vary with a variation in inertia. An increase in inertia shows a decrease in bandwidth. The variation of the lateral and

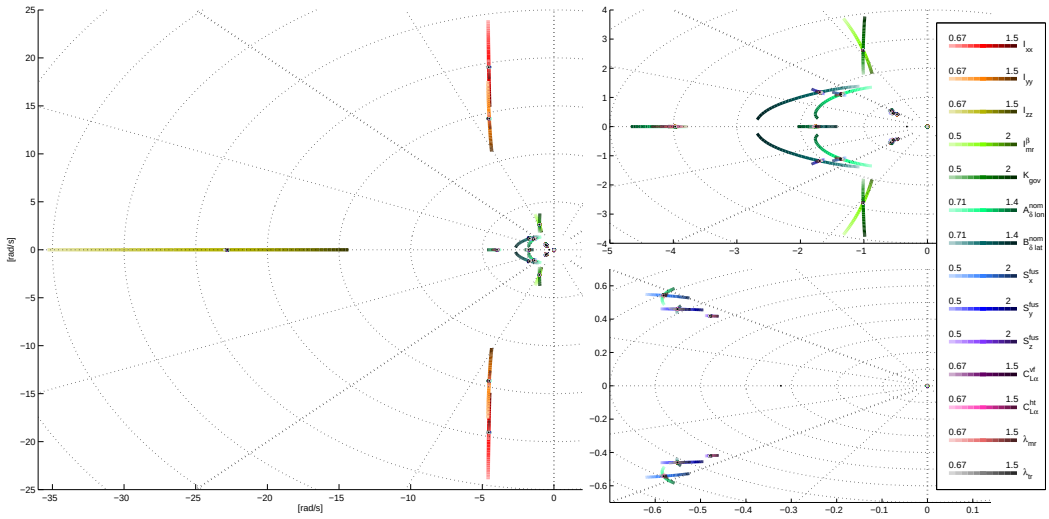


Figure 3.9: Forward Flight Model Sensitivity to Parameter Changes

longitudinal flapping angle gains also influences their rate poles, as well as their velocity counterparts.

The engine and rotor mode vary slightly in damping and speed with a change in the governor gain and blade inertia. Surprisingly, the lift characteristics of the tail stabilizer fins and the main and tail rotor airflow rates have very little effect on the model dynamics.

It can be seen that the system is stable at all times, with an adequate stability margin. During forward flight the stability margin is slightly less than during hover flight.

Other than the physical vehicle parameters, the vehicle is also exposed to varying environmental conditions. These are typically wind and changing air density. The controller was designed in such a way that these conditions can be uploaded to the OBC where the model changes will be calculated and the controller adapted accordingly. Unfortunately the current ground station has no means of measuring these variables, so sensitivity analysis was conducted on the controller being oblivious to these changes in the model. The air density was varied across the range of air densities that can be experienced at different altitudes in South Africa, with air temperatures (0-40°C).

It can be seen that the variation in air density only noticeably influences the heave velocity and tail rate modes. This is expected, since the speed of these modes is directly related to the main rotor and tail rotor responses to air inflow changes. As a decrease in air density directly reduces this inflow effect, these rotor poles are expected to be slower.

A change in wind conditions results in little noticeable changes in the closed loop plant. The tail rate pole's speed increases with an increase in head wind - this is a direct result of the increased airflow over the vertical stabilizer fin.

The roll and pitch rate poles speed up with an increase of wind in any direction,

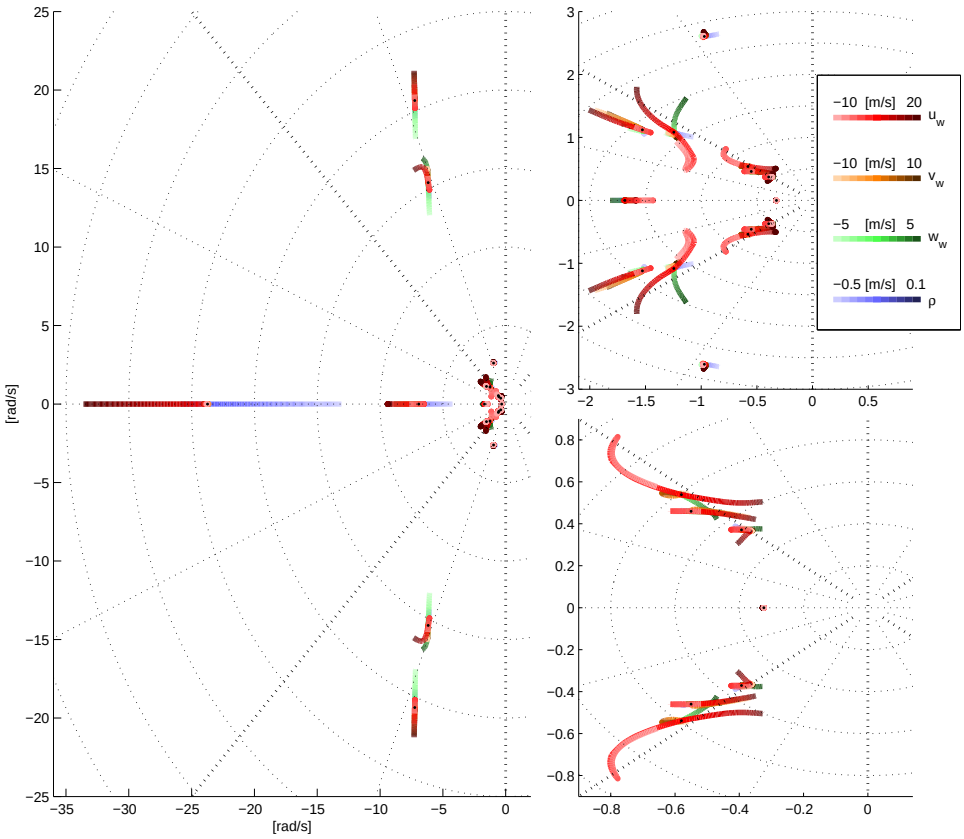


Figure 3.10: Hover Model Sensitivity to Environmental Parameter Changes

except from below. This is a direct result of the increase in control sensitivity of the main rotor blade as the inflow velocity increases.

Another vulnerability of LQR control becomes apparent when states cannot be accurately measured. In practice most sensors cannot measure the system states directly necessitating the use of a state estimator. Sensor measurements also have high levels of noise and biases on them. In an effort to analyse the influence of state estimate inaccuracies on the stability of the controller, the controller gains are calculated using a virtual vehicle model, as calculated from the noisy or offset estimates. These controller gains are then applied to the true model to calculate the resultant closed loop pole movements.

Figures 3.11 and 3.12 show the pole movements during hover and forward flight respectively for varying errors in state estimates. The variation for each state is based on a maximum expected probable error in the estimation process of those states. These maximums were observed from practical flight test recorded data.

As expected, the state variations have very little or no effect on the fast system poles as these are predominantly fixed by the vehicle's natural dynamics. Control errors mostly influence the slower, more controllable poles.

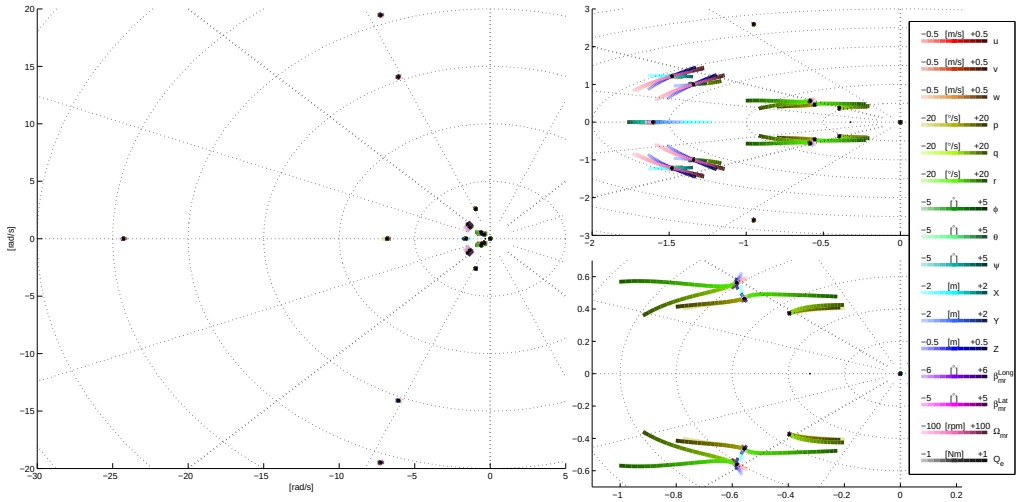


Figure 3.11: Hover Model Sensitivity to Estimate Errors

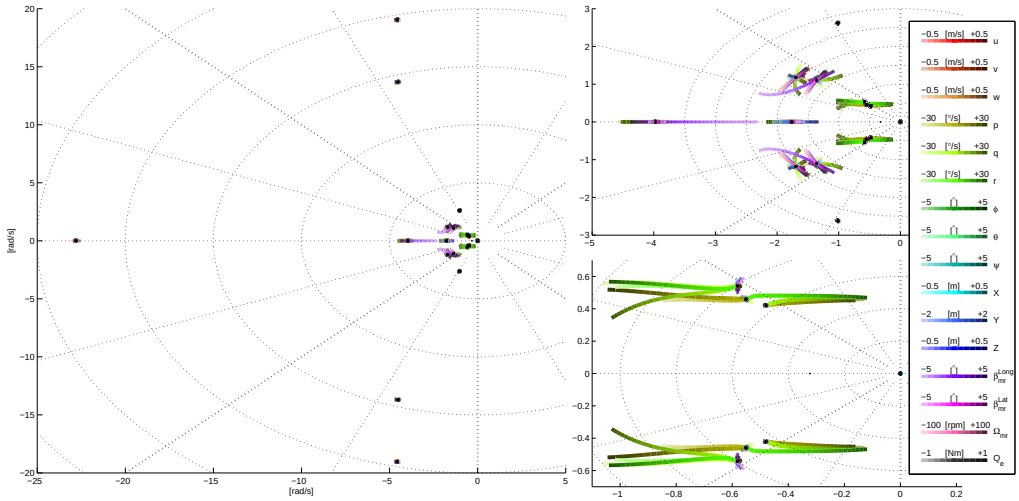


Figure 3.12: Forward Flight Model Sensitivity to Estimate Errors

Blade speed estimate errors cause variations to the translational velocity poles, while the flapping angle estimates have little effect. The largest variation in the pole locations came as a result of errors in the angular rate measurements.

Although the system remains stable the stability margin is reduced considerably by errors in the rate measurements. In practice, these errors come in the form of high frequency sensor noise, while the lower frequency drifts are less. These errors can then easily be reduced by simply filtering the rate estimates before they are used to calculate new gains. If the filter's cutoff frequency is kept in the same order as the bandwidth of the rate poles ($\pm$ 20 [rad/s]), the process will have little or no negative influence on the LQR process.

3.8 Alternative Control Methodology

During the course of this project the clear advantages of using an adaptive controller were highlighted. The most significant advantage was probably its nature to provide a stable control solution with a rapidly changing vehicle model as flight conditions change. Unfortunately one of the disadvantages of using a full state controller with direct state feedback is its problem with feeding back imperfect estimates.

The estimator used had no sensor bias estimation. This, combined with the control structure used, sets the limit that integrators can only be added to the position states to ensure steady hover flight. Should integrators be added to any of the inner states, they could wind up and result in an unstable vehicle. Outer loop integrators could add instability to the system since they are essentially marginally stable poles added to the origin, before the very last feedback is applied. As a result of this single feedback it is normally only possible to produce very slow stable solutions for these integrators.

During the process of comparing the adaptive LQR controller with the non-adaptive CLC controller it became apparent that even with its numerous shortcomings the CLC controller has one big advantage. Due to its layered control approach, where the outer loop's control is fed to the faster inner loops, it allows integrators to be added to any state - even the innermost states. Moving the integrators to the faster inner loops has the advantage that their negative impact on the controller stability can be countered by several loops of control which results in much faster, more responsive integrators.

Both controller structures were evaluated to see if an alternative solution to the direct full state feedback controller could be found; possibly one that incorporates some of the positives of both the adaptive LQR and non-adaptive CLC controllers.

The ideally required CLC control structure for this project was evaluated. To ensure a steady hover controller, CLC requires at least one integrator in one of the control loops. If the integrator is added to the faster states, disturbance rejection properties will be increased dramatically; however, any state biases on the outer layers will add to a greater position error when in hover.

It was decided that the best CLC structure for the cyclic axes will be one where there is an integrator on the fast rate loops and another integrator on velocity. The rate integrator will provide the disturbance rejection, while the velocity integrator will remove any biases on the angle and rate states. The control structure for the longitudinal control loop is illustrated in Figure 3.13.

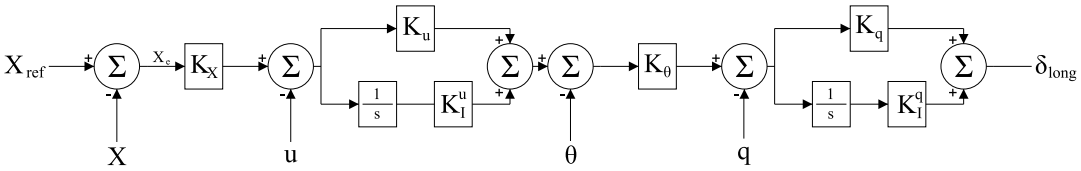


Figure 3.13: Longitudinal CLC Controller Structure

The control equations of the feedback loops are:

$$\begin{aligned}\delta_{long} &= \left(-X_e K_X K_u K_\theta - u K_u K_\theta + \left(-X_e K_X - u\right)\frac{1}{s}K_i^u K_\theta - \theta K_\theta - q\right)\frac{1}{s}K_i^q - q\left(K_q\right) \\ &\quad -\theta\left(K_q K_\theta\right) - u\left(K_q K_\theta K_u\right) - X_e\left(K_q K_\theta K_u K_X\right) + \left(-X_e K_X - u\right)\frac{1}{s}K_i^u K_\theta K_q \end{aligned} \tag{3.8.1}$$

When this system is compared to the current LQR system some similarities can be noted. The current LQR control structure is trajectory based, with an error or small deviation controller[9], but other than that certain parameters of Equation (3.8.1) can be compared directly to some of the state feedback gains of the LQR controller:

$$L_{2,5} = K_q \tag{3.8.2}$$
$$L_{2,8} = K_q K_\theta \tag{3.8.3}$$
$$L_{2,1} = K_q K_\theta K_u \tag{3.8.4}$$
$$L_{2,10} = K_q K_\theta K_u K_X \tag{3.8.5}$$
$$\tag{3.8.6}$$

The only parts that do not form part of the default LQR design are the two integrators. Given their structure, they can be forcefully added to the design by augmenting the vehicle plant model with their dynamics. This will then force LQR to calculate an optimal solution given the integrator positions. A similar augmentation can be made to the lateral plant, with integrators added to the roll rate and lateral velocity. For the heave plant an integrator can be augmented to the vertical velocity state, which will speed up this loop's response time to steady state disturbances. For similar reasons an integrator can be added to the yaw rate state. These

[9]Where the control is calculated from the state errors and not the absolute state values. The control is added to the actuator trajectory to control the plant.

augmentations will result in the following new states:

$$\boldsymbol{u_{int}} = (\boldsymbol{X}K_X + \boldsymbol{u})\frac{1}{s} \tag{3.8.7}$$

$$\boldsymbol{v_{int}} = (\boldsymbol{Y}K_Y + \boldsymbol{v})\frac{1}{s} \tag{3.8.8}$$

$$\boldsymbol{w_{int}} = (\boldsymbol{Z}K_Z + \boldsymbol{w})\frac{1}{s} \tag{3.8.9}$$

$$\boldsymbol{p_{int}} = \left(\boldsymbol{Y}K_YK_vK_\phi + \boldsymbol{v}K_vK_\phi + \boldsymbol{v_{int}}K_i^vK_\phi + \boldsymbol{\phi}K_\phi + \boldsymbol{p}\right)\frac{1}{s} \tag{3.8.10}$$

$$\boldsymbol{q_{int}} = (\boldsymbol{X}K_XK_uK_\theta + \boldsymbol{u}K_uK_\theta + \boldsymbol{u_{int}}K_i^uK_\theta + \boldsymbol{\theta}K_\theta + \boldsymbol{q})\frac{1}{s} \tag{3.8.11}$$

$$\boldsymbol{r_{int}} = \left(\boldsymbol{\psi}K_\psi + \boldsymbol{r}\right)\frac{1}{s} \tag{3.8.12}$$

All of the above parameters can be calculated from the previous time-step optimal control gain $\boldsymbol{L}$:

$$\begin{aligned}
K_X &= \frac{L\left(\delta_{long}, X\right)}{L\left(\delta_{long}, u\right)}, K_Y = \frac{L\left(\delta_{lat}, Y\right)}{L\left(\delta_{lat}, v\right)}, K_Z = \frac{L\left(\delta_{col}, Z\right)}{L\left(\delta_{col}, w\right)} \\
K_u &= \frac{L\left(\delta_{long}, u\right)}{L\left(\delta_{long}, \theta\right)}, K_v = \frac{L\left(\delta_{lat}, v\right)}{L\left(\delta_{lat}, \phi\right)}, K_w = L\left(\delta_{col}, w\right) \\
K_\theta &= \frac{L\left(\delta_{long}, \theta\right)}{L\left(\delta_{long}, q\right)}, K_\phi = \frac{L\left(\delta_{lat}, \phi\right)}{L\left(\delta_{lat}, p\right)}, K_\psi = \frac{L\left(\delta_{rudd}, \psi\right)}{L\left(\delta_{rudd}, r\right)} \\
K_q &= L\left(\delta_{long}, q\right), K_p = L\left(\delta_{lat}, p\right), K_r = L\left(\delta_{rudd}, r\right) \\
K_i^q &= L\left(\delta_{long}, q_{int}\right), K_i^p = L\left(\delta_{lat}, p_{int}\right), K_i^r = L\left(\delta_{rudd}, r_{int}\right) \\
K_i^u &= \frac{L\left(\delta_{long}, u_{int}\right)}{L\left(\delta_{long}, \theta\right)}, K_i^v = \frac{L\left(\delta_{lat}, v_{int}\right)}{L\left(\delta_{lat}, \phi\right)}, K_i^w = L\left(\delta_{col}, w_{int}\right)
\end{aligned}$$

The integrators can be discretised and augmented to the discrete plant model in the same way as Equations (3.3.11) and (3.3.13). A new LQR controller was designed using these augmented plants as the input for the optimal design. These augmentations are continuously updated so that the controller adapts to model changes.

The controller was coded in MATLAB and testing was done to compare the results with the standard LQR controller. This new control structure was expected to improve the vehicle's response by responding more quickly to disturbances. The comparison was focused on a simulation with two common disturbances - wind changes and actuator sensitivity changes.

In the first simulation a lateral actuator offset of 1 [°] was given to the model at time 0 [s] with a constant 5 [m/s] side wind hitting the model at 20 [s]:

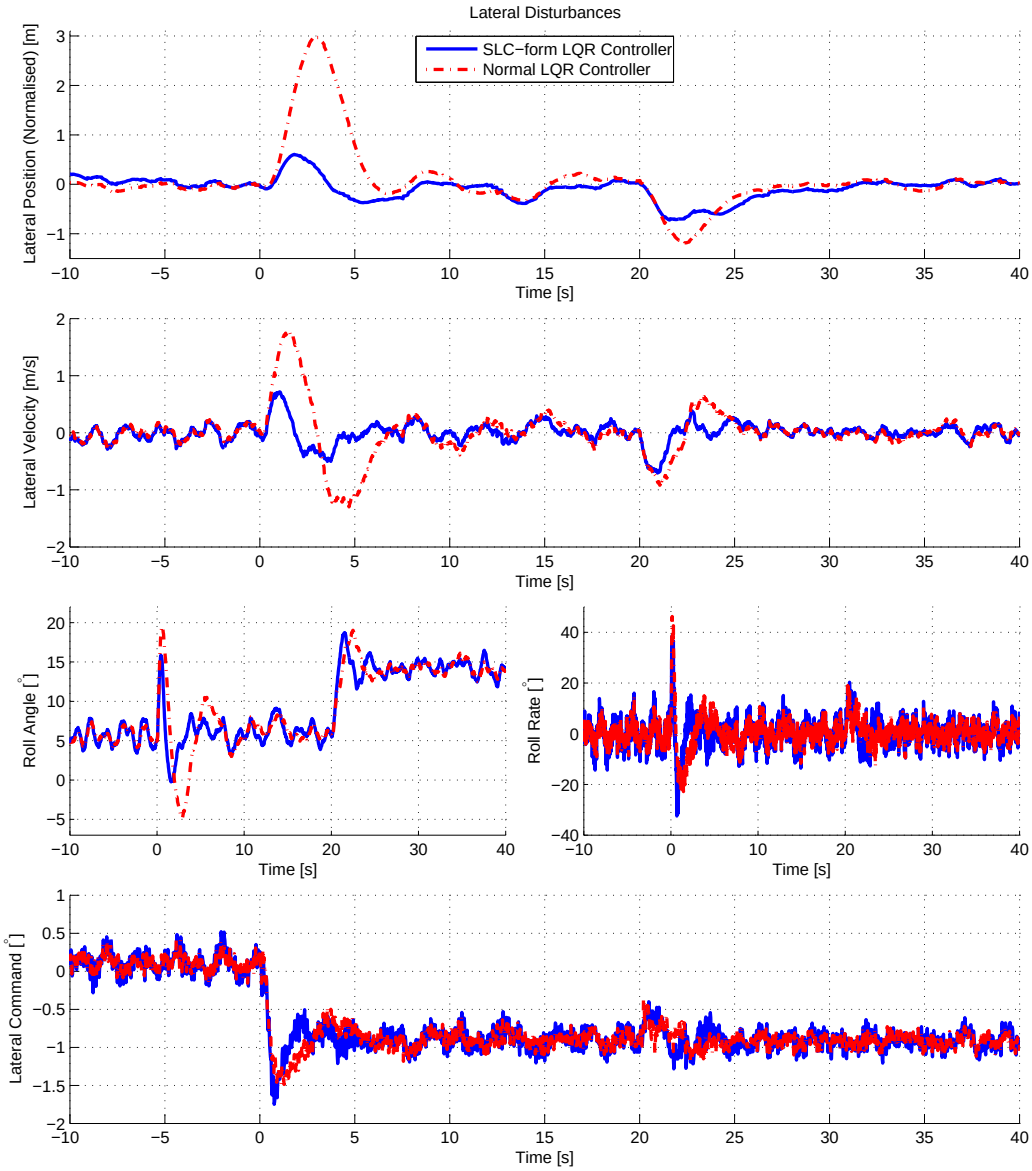


Figure 3.14: Comparative Results of LQR and SLC-form LQR Controllers

It can be seen that the new controller reacts much more quickly to the disturbances. When inspecting the feedback gains that the LQR process produces, it was noted that with a change in weightings, the new controller produces a much wider range of possible gains. This in turn produced a much wider range of possible con-

troller responses, where weighting changes with the old control structure did not always produce the desired change in response.

The reason for this change in behaviour was largely thought to have come from a more stable control structure that allowed for more flexibility in finding an optimal, stable solution. The outer loop position of the integrators in the old control structure impeded stability and thus limited the controller to a narrower range of possible stable responses. This view was confirmed with a closed loop pole plot of the two controllers (see Figure 3.15)[10].

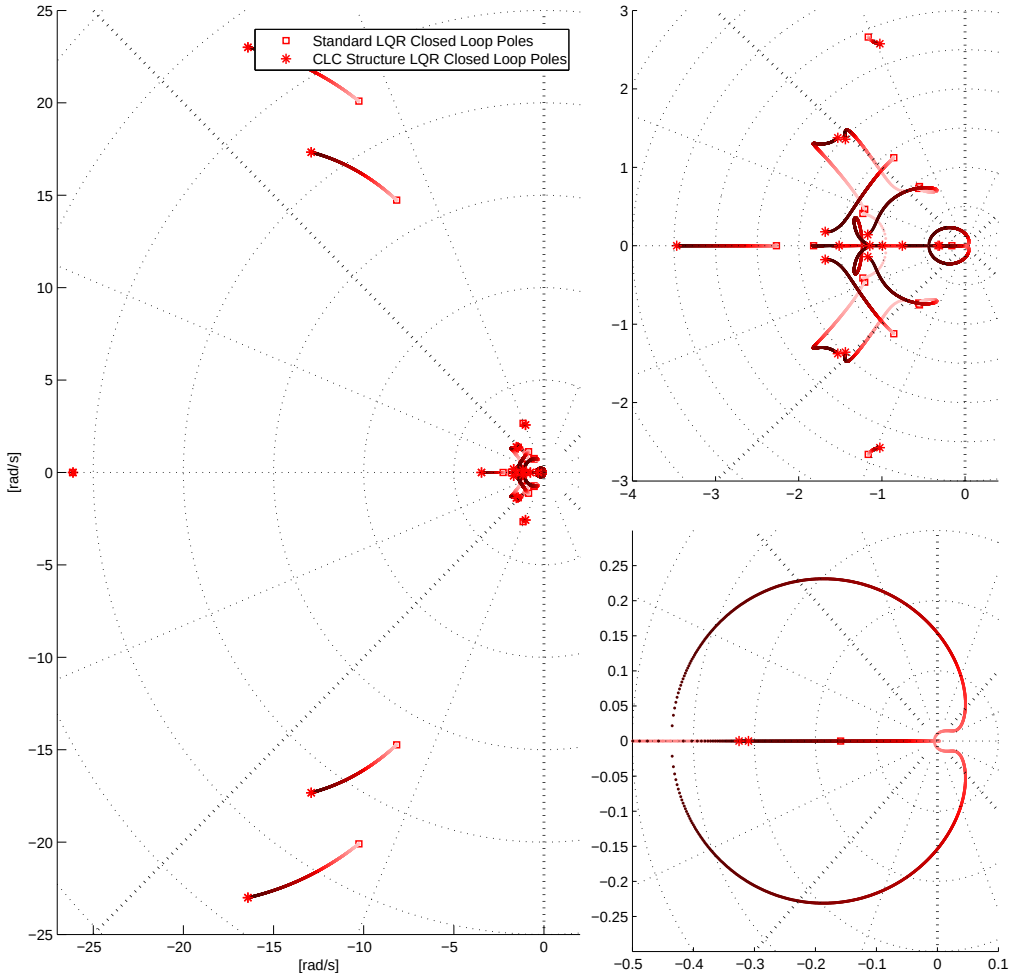


Figure 3.15: Traditional LQR Poles vs Improved Structure LQR Poles

It can be seen in Figure 3.15 that the new control structure generally produces faster and better damped closed loop poles. The integrator poles that reside open loop at the origin are also moved much faster and are more stable in closed loop.

[10]Please note that the loci showing the pole position movement are only used as a visual indication of the faster (darker) poles of the new control structure.

3.9 Summary

This chapter detailed the process followed to design a *model adaptive LQR controller*. The implications of implementing such a controller on hardware were discussed. The controller's robustness to modelling inaccuracies and estimation errors were analysed and finally an alternative, much improved control structure was developed and tested.

Chapter 4

Practical Results

4.1 Stability Analysis and Comparison with a Classical Controller

In an effort to demonstrate the advantages of an adaptive control system it was decided to compare the stability margin of the current controller with a commonly used classical controller.

Consecutive loop closure (CLC) is one of the most popular controllers for this type of vehicle. It offers the advantages that it is relatively easy to design and implement, does not need much computational power to run and offers a very intuitive understanding of the control process. The drawback of the CLC control technique is that it is unfortunately non-adaptive to model changes (as discussed in Section 3.1).

Carstens [3] designed such a controller for the electrical helicopter platform and his design laws will be followed to adapt the controller for the current vehicle. Note that the author did not optimize this controller in any way, as it was not the main controller to be used for this project and only serves as a comparison tool. For details on the design process, refer to Appendix C.2.

The CLC controller was simulated from hover to high speed forward flight. This controller produced similar results at hover than the adaptive LQR controller, but when the speed was increased the adaptive controller showed its advantage. Where the non-adaptive controller became unstable past 6 [m/s] forward flight, the adaptive system easily continued to the flight envelope limit of 18 [m/s]. When the stability margins of both controllers are analysed, it can be seen that the non-adaptive system produces an unstable closed loop system above 7-8 [m/s] forward flight (see Figures 4.1 and 4.2). The longitudinal phugoid mode becomes unstable first, followed by the lateral phugoid mode.

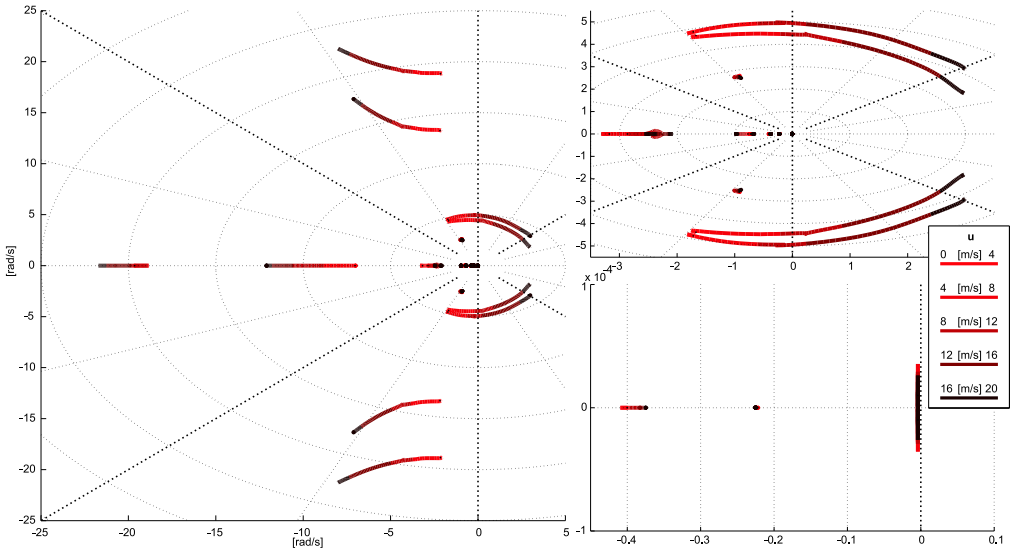


Figure 4.1: CLC Closed Loop Model Pole Movement for Forward Flight

For the adaptive system it can be seen that there are fewer changes in the model at higher speeds. The most notable change came from the lateral and longitudinal velocity/attitude pole pairs that became slower. This is a direct result of the airflow over the vertical and horizontal tail fins during forward flight.

The heave velocity and tail rate modes become faster - this is a direct result of the increased air inflow into the two rotors (main and tail) which increases the rotor's normal sensitivity collective control input.

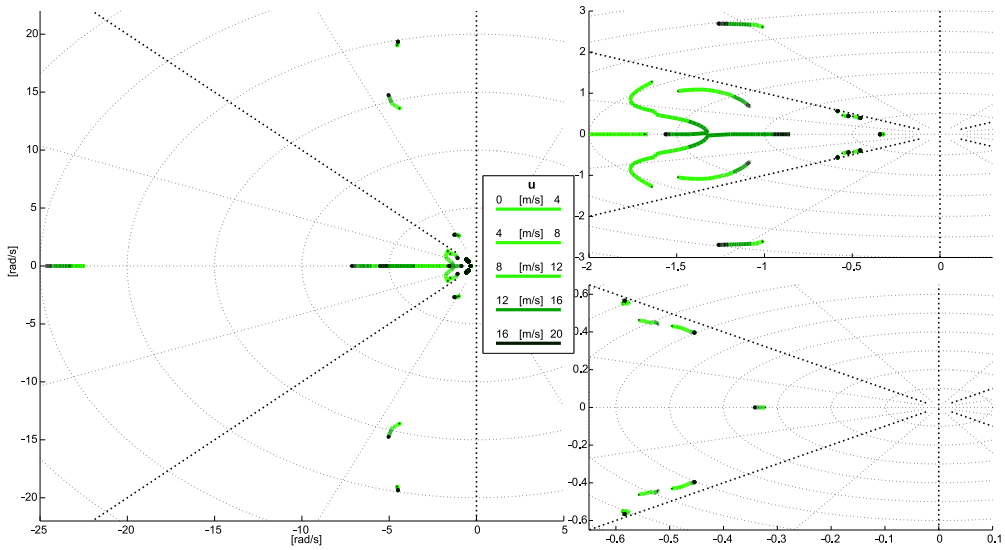


Figure 4.2: LQR Closed Loop Model Pole Movement for Forward Flight

For lateral flight the model changes more rapidly - with the induced destabilizing effect of the vertical stabilizer fin. Here the non-adaptive controller already produces an unstable model at a mere 4 [m/s] lateral velocity. In this instance the lateral phugoid mode reaches instability first, followed by the longitudinal phugoid mode. The adaptive system remains stable, even up to a lateral velocity of 8 [m/s] and beyond.

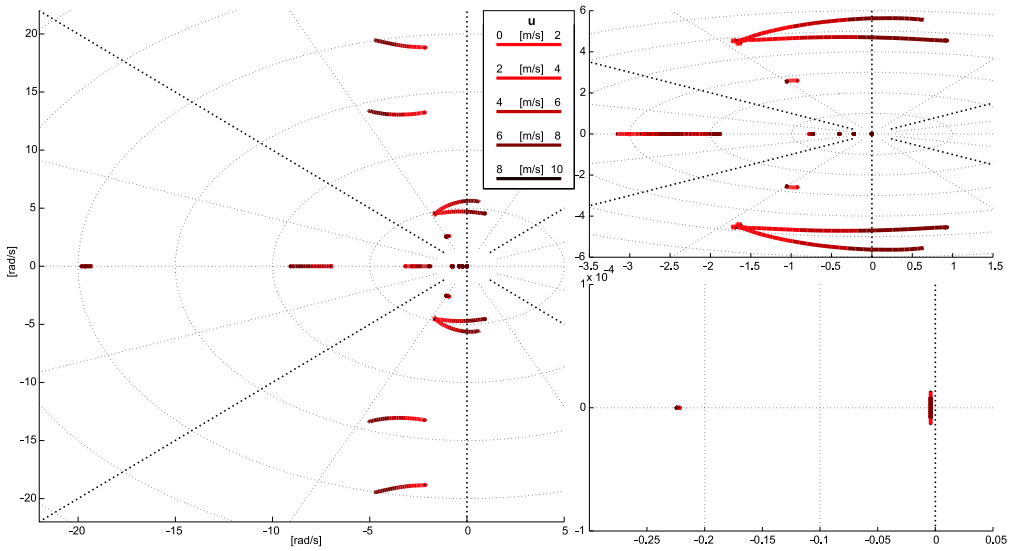


Figure 4.3: CLC Closed Loop Model Pole Movement for Lateral Flight

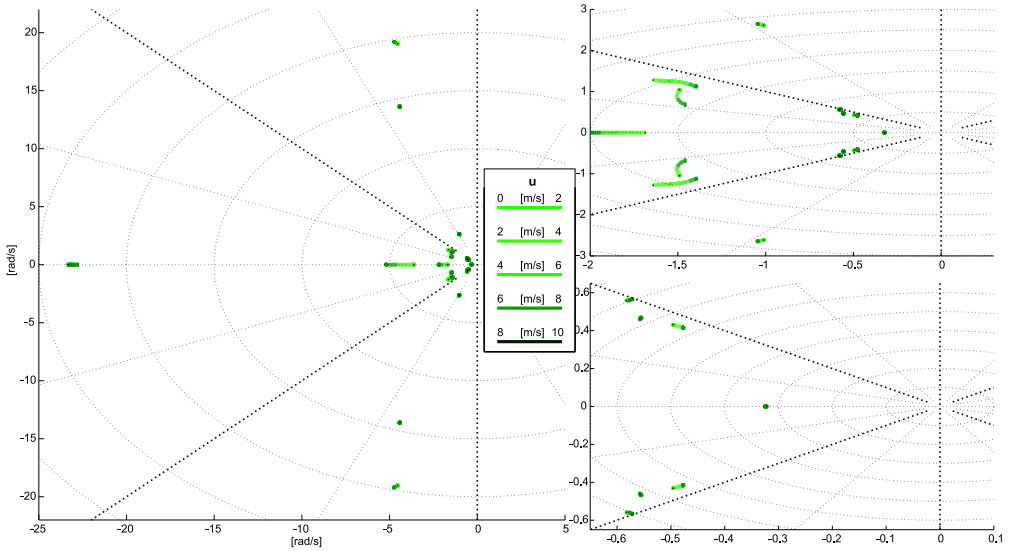


Figure 4.4: LQR Closed Loop Model Pole Movement for Lateral Flight

4.2 Hardware in the Loop

It was important to test the controller as thoroughly as possible in the simulation environment in order to refine the design, so it would be able to handle the variety of possible environmental disturbances and sensor noise.

A HIL simulation was set up, where the full non-linear model was simulated in real time using Matlab's *Simulink*. From this model sensor measurements were generated. The non-linear model was set up in such a way that wind disturbances could be added to the airflow over all the aerodynamic surfaces of the vehicle. Real life equivalent noise was added to the measurements, as well as the effects of time delays, phase delays and quantization[1], as one expects during practical flight testing. These generated noisy sensor measurements are then passed to the on-board computer where the control algorithms can be tested. The calculated autopilot servo commands are then passed back to the Simulink environment in real time, to stimulate the non-linear model and thus complete the loop.

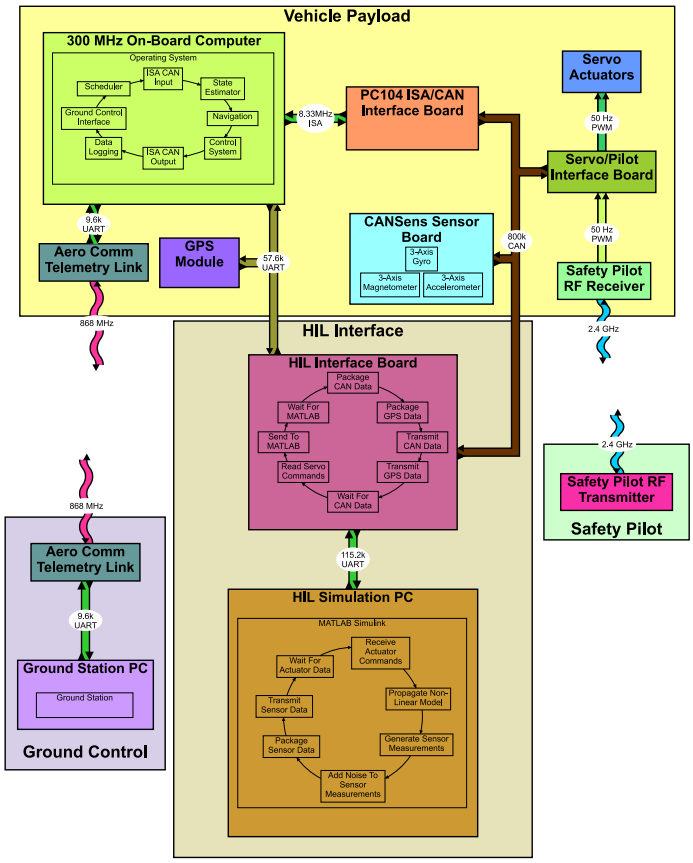


Figure 4.5: System Block Diagram of HIL Setup

[1]When a continuous signal is approximated on a discrete interval - typical result of digitally sampling a signal.

It is important to note that the on-board computer cannot differentiate whether it is part of a HIL simulation or a real flight test - this real-time testing is a very powerful debugging tool and helps to safely eliminate programming errors and system design flaws. See Figure 4.5 for a system block diagram of the HIL setup.

4.2.0.1 Hover Flight

Figure 4.6 shows the results for HIL simulated hover flight (solid lines indicate state references). Initially for the first 12.5 [s] only random wind gusts of up to 3 [m/s] were added to the simulation, thereafter an addition steady state DC wind from the front with a speed of 5 [m/s] was added.

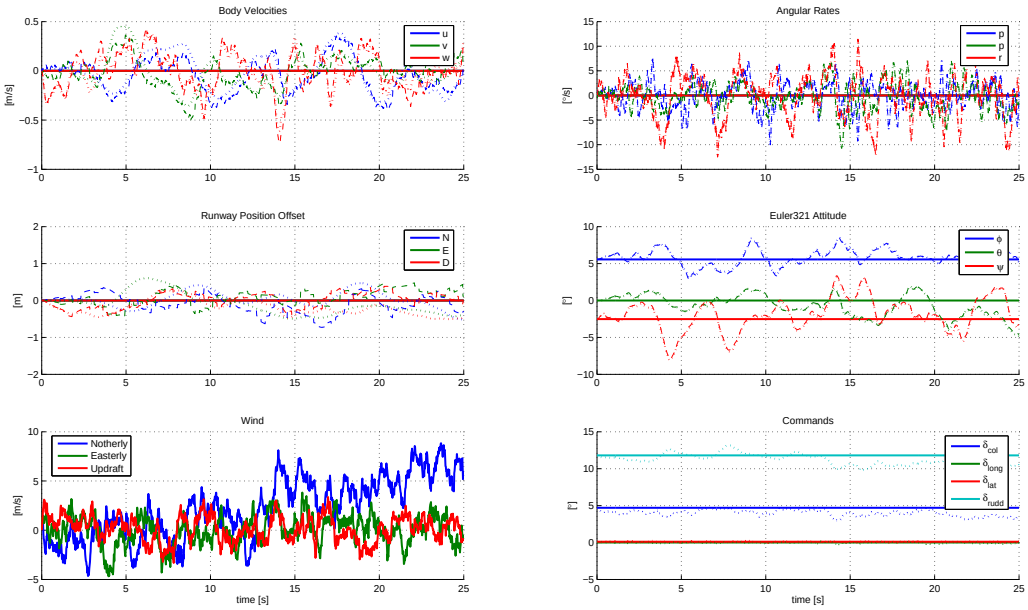


Figure 4.6: Hover Flight Results with 5 [m/s] Northerly Wind and 3 [m/s] Gusts

It can be seen that the controller responds well in windy conditions and reacts quickly to compensate for constant winds, while not responding overly aggressively to wind gusts.

For a full list of hover flight results with winds from every major direction, refer to Appendix C.1.

4.2.0.2 Fast Forward Flight

Figure 4.7 shows the results for HIL simulated fast forward flight (solid lines indicate state references). The starting condition was set to hover, after which the vehicle was accelerated forward at 1 [m/s^2] until a forward speed of 18 [m/s] was reached. This condition is 90 [%] of the maximum forward speed for which the model (derived in Chapter 2) holds.

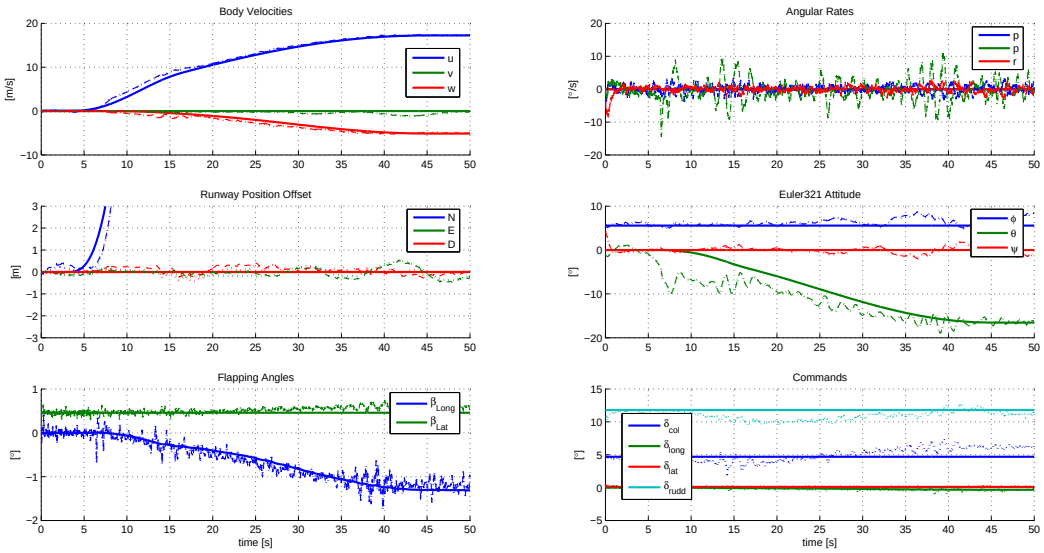


Figure 4.7: 18 [m/s] Fast Forward Flight Results

It can be seen that during the acceleration and deceleration phases there is a noticeable deviation from the trajectory. This deviation was expected, since the steady state trajectory does not include these dynamic effects. Even at this high speed and with sensor noise added to the simulation, the vehicle stays well within 1 [m] of its reference position.

4.2.0.3 Fast Backward Flight

The simulation is started at hover conditions after which the vehicle is accelerated with a maximum acceleration of 1 [m/s^2] backwards, until a backward speed of 5 [m/s] is reached.

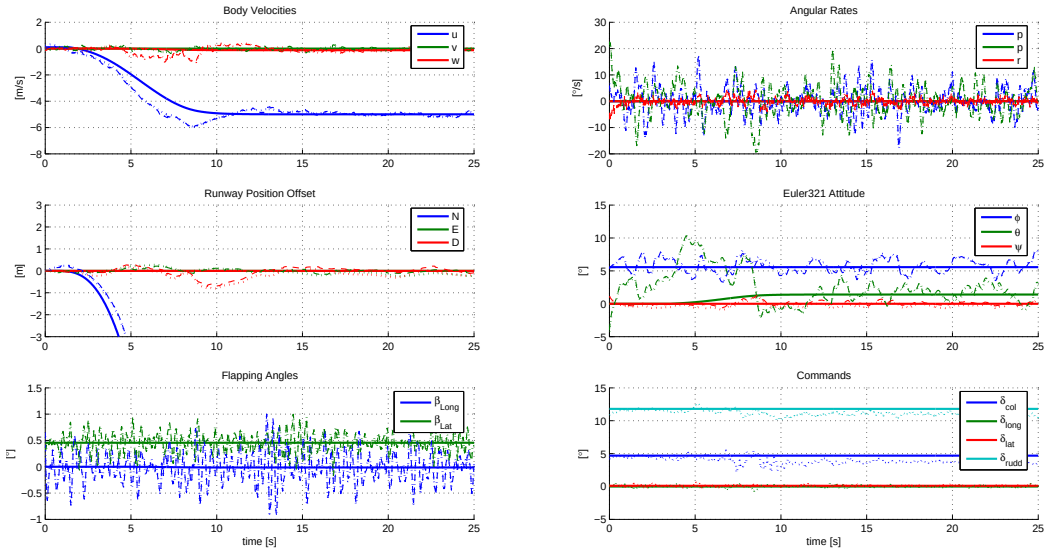


Figure 4.8: 5 [m/s] Fast Backward Flight Results

Again the deviation from the trajectory is noticed during acceleration and deceleration phases, but during steady state the vehicle follows the reference trajectory well. The position errors are much less than 1 [m] throughout the simulation.

For more results on other flight conditions, including lateral, vertical and yaw flight, refer to Appendix C.1.

4.2.0.4 Barrel Roll

The simulation of the barrel roll was started at a horizontal forward velocity of 4 [m/s] (for convenience of plotting, the manoeuvre was executed in the northerly direction).

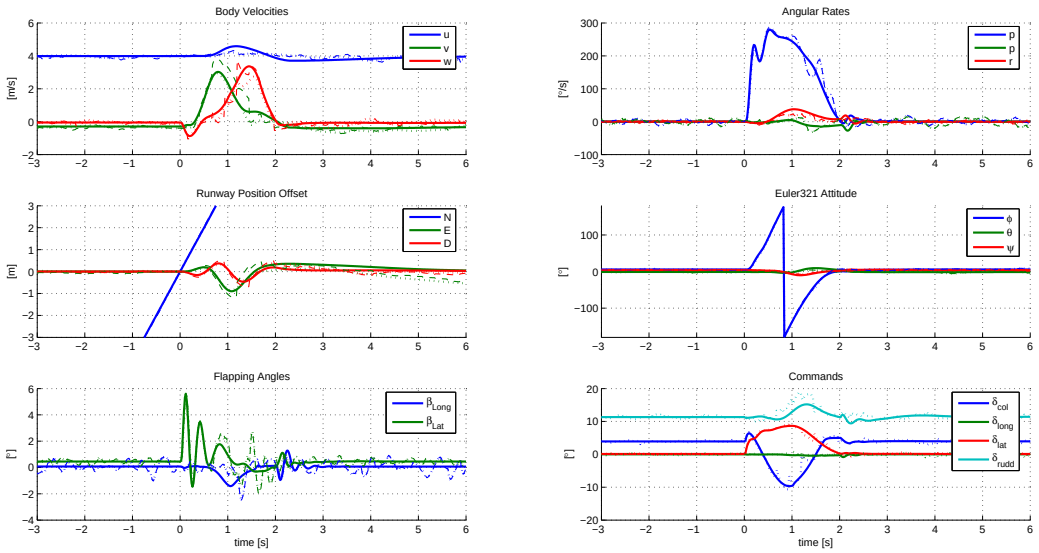


Figure 4.9: Barrel Roll Manoeuvre Results

The results show very good trajectory tracking in the initial phase of the manoeuvre. Larger errors occur in the final phase when the roll rate is lowered back to the hover condition. These errors seem to originate from the incorrect trajectory specification of the lateral blade flapping angle[2]. The position errors are consistently less than 1 [m], with the whole manoeuvre being executed within a radius of less than 1 [m] around the trajectory reference.

Figure 4.10 shows the pole movements of the controlled closed loop system during the barrel roll manoeuvre. The lighter colours signify the initial part of the manoeuvre, while the darker colours signify the end (note the colours indicate the initial hover pole modes of motion).

[2]Flapping angle trajectories are very difficult to predict and are only well defined for steps in the cyclic actuators (as with the initial part of the manoeuvre).

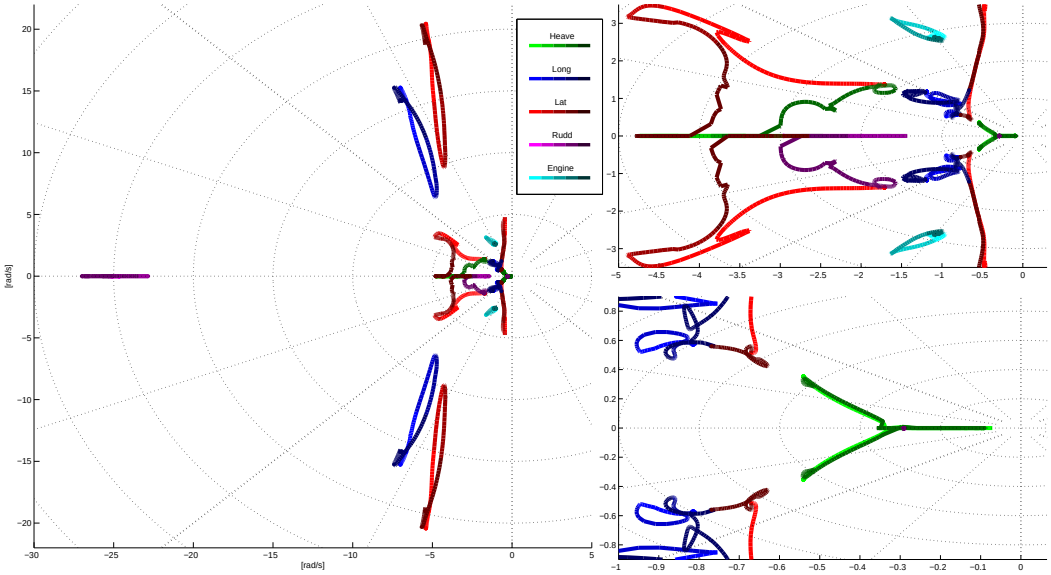


Figure 4.10: Pole Movements During Barrel Roll Manoeuvre

It can be observed that the overall system stability margin decreases in the middle part of the manoeuvre (when the vehicle is in the upside down position) but the system remains stable at all times.

With the exception of the fast (naturally relatively uncontrollable) roll and pitch rate poles, the modes of motion most affected by the manoeuvre is those of the roll and heave. This result is expected since the collective and lateral actuators undergo significant changes during the manoeuvre.

4.2.0.5 180° Stall Turn

The simulation of the stall turn was started at a horizontal forward velocity of 6 [m/s] (for convenience of plotting, the manoeuvre was executed in the northerly direction).

The results show that the trajectory for the heading command was a bit aggressive and that in reality the vehicle might struggle to safely accommodate such a high yaw rate[3]. In spite of this, the results were still adequate, with the vehicle exiting the manoeuvre within 1 [m] of the entry point. Figure 4.12 shows the manoeuvre viewed from the side.

[3]For practical consideration all rates should be kept lower than the maximum rate-sensor measurement capability of 320 [°/s].

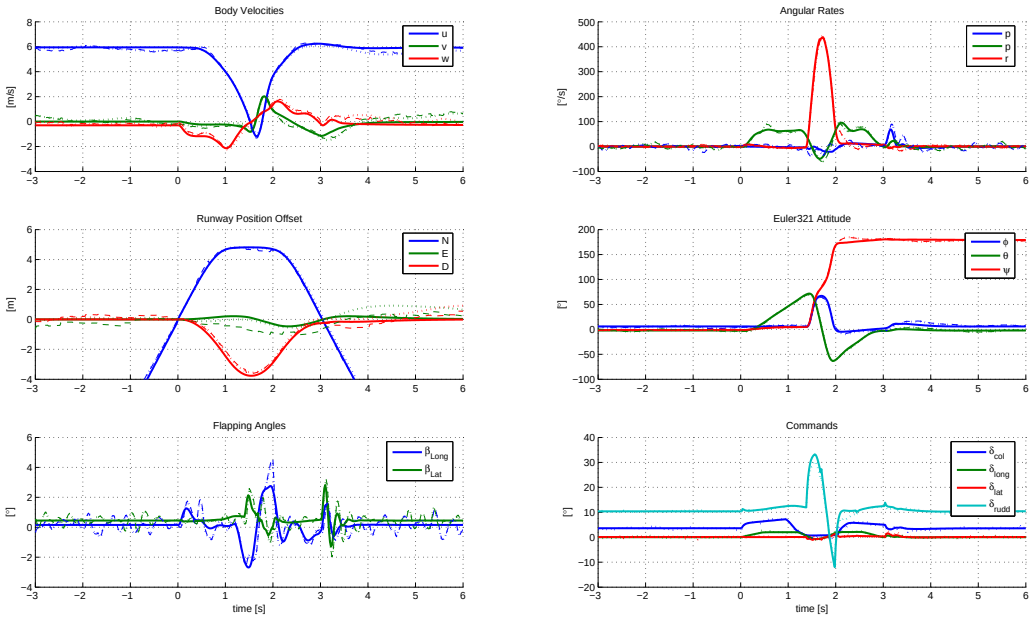


Figure 4.11: 180° Stall Turn Manoeuvre Results

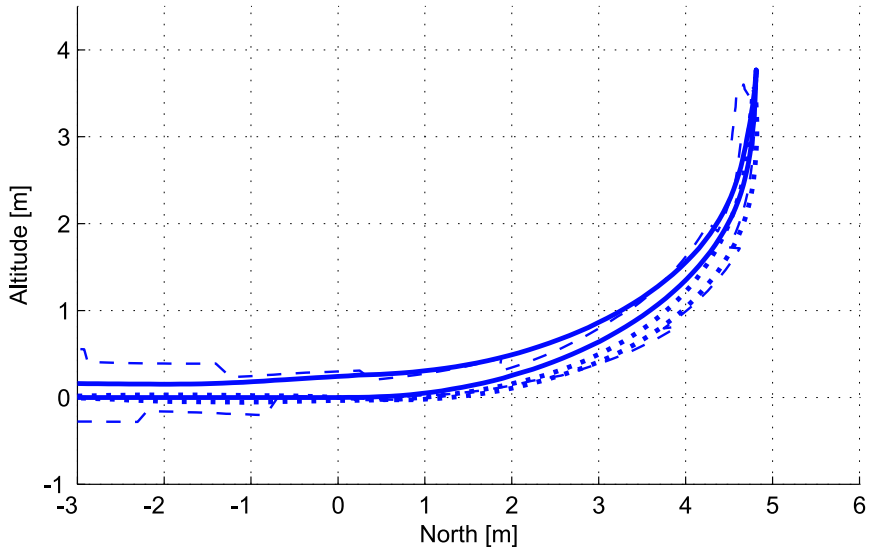


Figure 4.12: 180° Stall Turn Manoeuvre Side View

Figure 4.13 shows the pole movements of the controlled closed loop system during the 180° stall turn manoeuvre. The darker lines indicate the pole locations after the stall while the lighter lines indicate the initial part of the manoeuvre (note the colours indicate the initial hover pole modes of motion).

Although the stability margin decreases substantially toward stall, it can be seen

that the system remains stable at all times.

The most pole movement appears in the heave, longitudinal and rudder modes of motion. This result is expected as these modes are stimulated with very large rudder, longitudinal and collective commands during the manoeuvre.

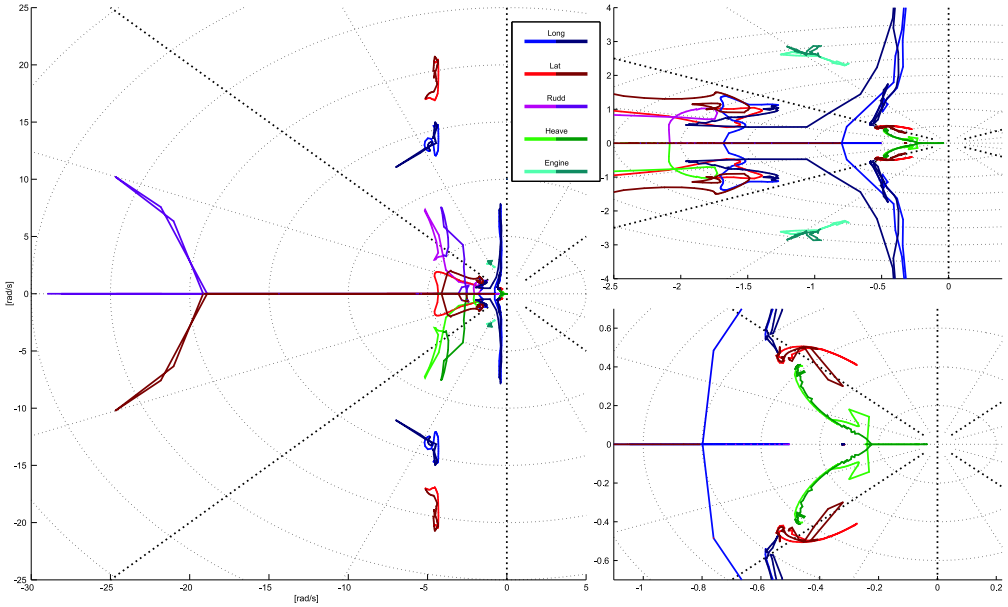


Figure 4.13: Pole Movements During Stall Turn Manoeuvre

4.3 Flight Testing

After extensive HIL simulation the controller and avionics were deemed fit for flight testing. During the simulation time the integrity of the on-board computer and its operating system was evaluated. The servo control electronics, radio control interface and safety pilot control procedures were also comprehensively tested. The system software and specifically the control software were continuously evaluated in order to improve performance, safety and stability.

Because the LQR controller used in this project is a single full state controller, the initial part of flight testing should be approached with caution. To minimise risk, the decision was made not to arm the complete controller during initial flight testing.

The model and control structure were evaluated in order to find a way of separating the controller into smaller units that could be tested independently to reduce flight risk. It was found that by forcefully decoupling the controller into the 4 different control axes (roll, pitch, yaw and collective), the number of model and control errors introduced into the system could be restricted. Although the model, and thus the controller, has some degree of cross coupling between the axes it was

argued that by ignoring this coupling the controller performance will likely be reduced and good results with this decoupled controller will show promise that even better results should be achieved with the complete controller.

During the first flight test the full LQR controller was activated, but only the rudder was controlled by the autopilot - the other axes were under manual safety pilot control. The tail was stable and several heading steps were commanded from the ground station. Figure 4.14 shows the data from such a 30 [°] step. A heading step of similar size was simulated in HIL and the data are shown on the plot:

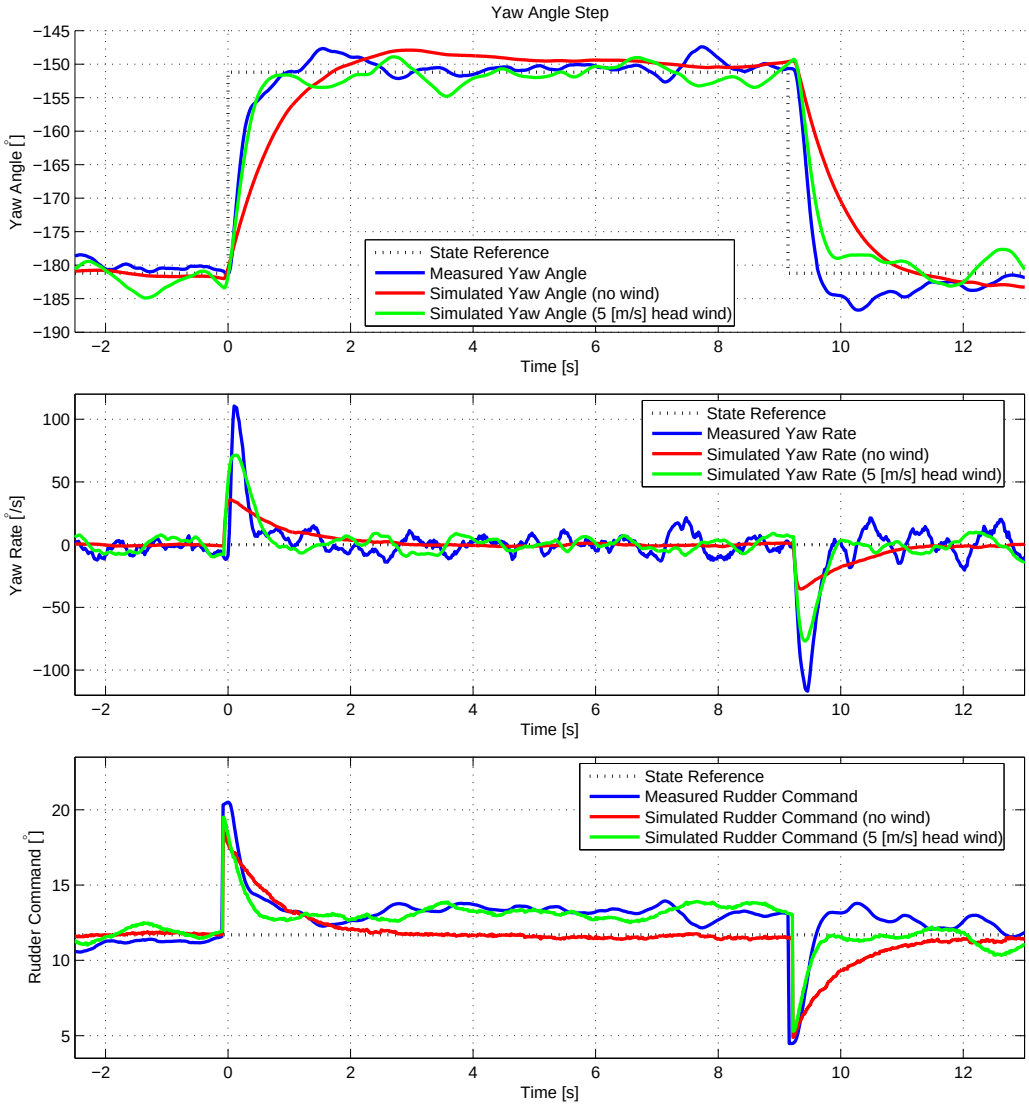


Figure 4.14: Heading Control Flight Test Results, with HIL Comparison

It can be seen that the simulated results deviate in a number of ways from the measured flight test data. Firstly, it can be observed that the flight test response is

quicker than what is expected from simulation. Secondly, the controller used for the flight test required a larger rudder command to keep the heading out of the wind, compared to the into the wind case. When the flight logs for the day were revisited, it was found that during the flight test the wind speed was 10-15 [knots] (or $\pm$5-8 [m/s]). The pilot always armed the controller with the vehicle's nose pointing into the wind. Since the wind can significantly change the model, without the controller adapting for this condition, the control's response could differ greatly from what is expected (see Section 3.7 and Figure 3.12 for the controller's sensitivity to wind).

The faster practical response seen in Figure 4.14 can be a result of the increased sensitivity of the rudder. This increased sensitivity is a direct result of the increased airflow over the tail rotor (additional head wind, plus the portion of the main rotor thrust blown over the tail rotor). In addition, when the yaw angle was stepped to 30 [°], the tail and vertical stabilizer fin were no longer aligned with the wind and thus it would make sense that more rudder be applied to counter the wind on the tail.

When the HIL simulation is taking the 5 [m/s] headwind into consideration, the results compare much better with the practical results. The rudder command offset plus the slightly faster response are now clearly visible (see Figure 4.14).

The collective axis was tested in a similar manner to the yaw axis. Exactly 39.1 [s] after arming and relatively stable flight, an upward altitude step of 2 [m] was commanded from the ground station, followed by a 2 [m] downward command at 56.26 [s]. A similar manoeuvre was simulated using HIL and compared to the practical results.

Again some differences can be noted between the collective responses of the two tests. The first observation of the practical data is that in the time shortly after arming[4], the altitude makes a 1.5 [m] error before stabilising at the reference altitude. This error is expected, since the pilot was climbing at the time he armed the controller. The position integrator quickly brought the vehicle back to its reference altitude.

The second observation is the 1 [m/s] offset in descent rate while in hover as well as the high frequency noise on the velocity and position estimates. On closer observation it was determined that both of these phenomena were a direct result of a bias offset on the z-axis accelerometer measurement. These offsets were a result of the vibration on the airframe that caused the accelerometers to saturate. The z-axis accelerometer was worst affected by this problem, as its default measurement was not zero mean.

By differentiating the slope of the small jumps in the velocity measurement the accelerometer bias was estimated to be in the order of +1 [m/s^2]. This offset, together with the previously discussed 5 [m/s] head wind was added to the HIL simulation. Again the new results produced a much better match to the practical data (see Figure 4.15). The velocity offset, as well as the noisy estimates is now clearly observable - also note how much less control is needed to keep the vehicle in hover.

[4]The controller was armed at time = 0 [s].

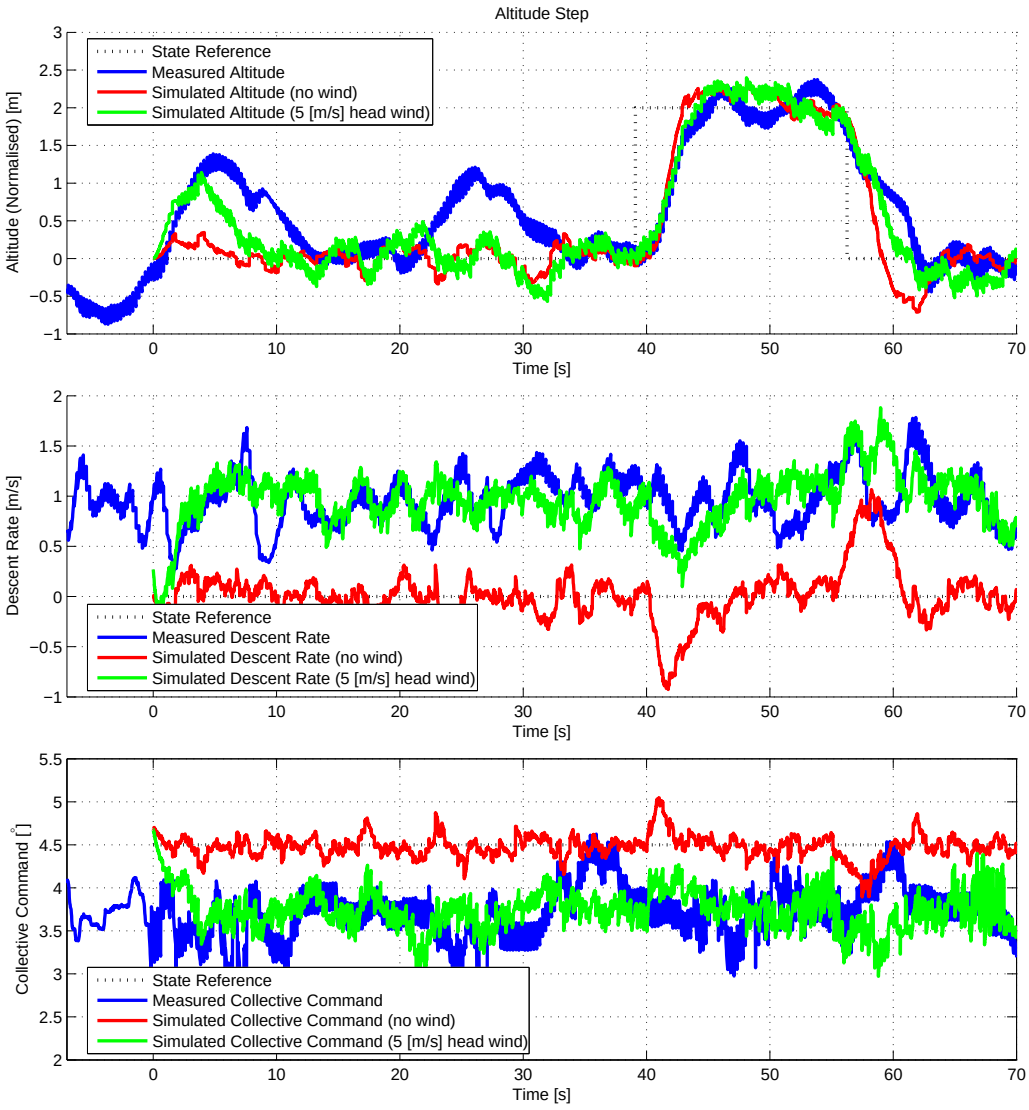


Figure 4.15: Heave Control Flight Test Results, with HIL Comparison

Finally the longitudinal axis was tested. After 39.5 [s] of stable flight, a single 1 [m] forward step was commanded from the ground station. Once again a similar manoeuvre was HIL simulated and compared to the practical results.

Similar phenomena as with the collective axis test were noticed. Again the pilot armed the controller while the vehicle was not completely in a hover state. The vehicle drifted forward while lowering its forward velocity and within 10 [s] it was back at its reference position.

A velocity offset similar to the one noticed during the heave tests can be seen, which also pointed to a bias in the accelerometer measurements. The x-axis accelerometer bias was calculated to be +0.5 [m/s^2] for the time before 20 [s], after which a gradual increase to +1 [m/s^2] was noticed. The exact reason for this in-

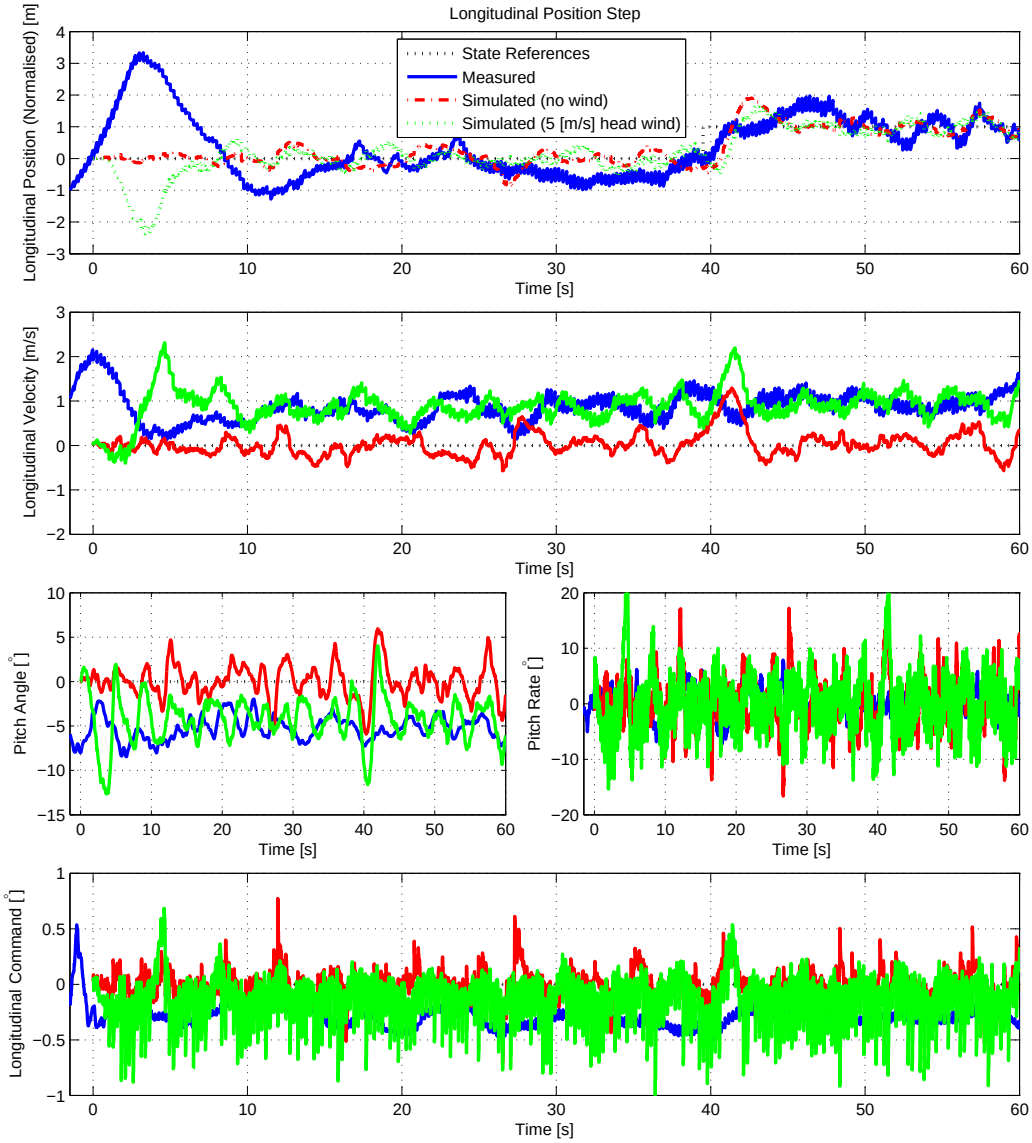


Figure 4.16: Longitudinal Control Flight Test Results, with HIL Comparison

crease is not known, but it is probably as a result of an increase in vibration or a slight error in the attitude estimate as a result of a magnetic disturbance.

The last observation is the offset in the pitch angle - this leaning forward is the controller's way of countering the wind hitting the vehicle from the front. A +1 [m/s^2] offset to the x-axis accelerometer, as well as the previously discussed 5 [m/s] head wind was added to the HIL simulation. The new results matched the practical results much better. The velocity offset, as well as the increase in the forward pitch angle is now clearly observable (see Figure 4.16).

These results of the separately tested axes were extremely positive - especially

for a first flight test. The stable controllers demonstrate that the adapted vehicle model is relatively accurate and that the combined full LQR controller should probably improve these results further. No adjustment to the controller weightings could be identified at this stage and future adjustments would probably only be to alter the response of the vehicle slightly. Unfortunately there were no further opportunities for flight testing, as the vehicle was fitted with new avionics with much reduced processing power.

4.4 Summary

The adaptive LQR controller was compared to a non-adaptive classical controller. Full HIL simulation results were shown, with flight conditions ranging from high speed lateral, longitudinal and vertical flight to aerobatic manoeuvres. Finally practical flight test results were presented.

Chapter 5

Summary and Recommendations

5.1 Summary

This thesis detailed the process of design and practical implementation of a full state controller for a small radio controlled helicopter. Several additions were made to an available simulation model, which was then re-derived in generic form in such a way that it can be applied to any similar helicopter. A full state controller was designed to make use of low cost off the shelf sensors to perform two aerobatic manoeuvres as well as fly an extended part of the conventional flight envelope. The design was validated using extensive HIL simulations. The probable success of flight testing was evaluated through a full sensitivity analysis, where problem parameters and estimator inaccuracies were identified. Finally, a ground station was developed to command the controller during flight testing. The design was partially flight tested, delivering stable results on the very first test.

Several software packages were developed during the design process that will be of great use in similar future and current projects.

- The full non-linear model was integrated into the *Simulink* HIL simulation, together with a runway and wind model.
- The model was also developed in C, which can be transferred through a variety of C-compilers to different processors.
- The non-linear model was time-derived into an adaptable 20 state linearised plant and also developed in C.
- Comprehensive ground station software was developed, which enables real-time monitoring during flight testing. The software also allows the user to change controller weightings in real time and to design and evaluate controller performance.

5.2 Recommendations

There is currently an immense market for helicopter UAVs. The vehicle used in this project has already been earmarked for further research and development in this

field. There are some recommendations that might need consideration for future work on this platform:

- Limited flight testing was conducted on the current system setup. It is recommended that several hours of further flight testing be conducted in order to determine the quality, repeatability and safety of the current setup. It is further suggested that the aerobatic manoeuvres be flight tested in a safe, low risk manner.

- Although this project only focused on two aerobatic manoeuvres, it is recommended that the trajectories of other manoeuvres be designed and tested with the current controller setup.

- The sensors used in this project were found to be noisy and unreliable. This was the result of a combination of low quality, low cost sensors and a vast amount of mechanical vibration on the IMU.

  It is recommended that the accelerometers and magnetometers be replaced by higher quality sensors. In order to obtain the maximum advantage from the better quality sensors, the IMU would have to be adequately vibration damped (with dampers designed to operate efficiently at the nominal blade speed).

  All the IMU sensors were prone to temperature drift, which can be lowered by sampling their individual temperatures[1]. This will result in better temperature calibration. The IMU can also be kept at a constant controlled temperature.

  It is recommended that some kind of Bias-Estimation be applied to the rate gyros and accelerometers, as biases on these sensors result in offsets in the rate, attitude, velocity and position estimates.

  The current GPS sensor proved to be too inaccurate for precision hover flight. Differential updates were streamed over the internet from a third party surveying service and were uploaded in real time to the GPS from the ground station. Although this considerably improved the GPS accuracy, the DGPS protocol was unfortunately upgraded by the third party to an unsupported version[2]. It is suggested that the current GPS be replaced by a newer version that supports the new DGPS protocol, or that a dedicated mobile DGPS surveying station be added to the ground station. The GPS could also be replaced altogether with a more accurate and expensive system.

  Although the ultrasonic sensor produced accurate altitude measurements, its working was intermittent and unreliable as a result of the vibration of the vehicle body, together with an irregular ground reflection surface. It is recommended that this sensor be replaced by a more reliable altitude sensor - the use of a static pressure sensor can be investigated.

[1] The current IMU only supports the temperature measurement of the z-axis gyro substrate.
[2] Upgraded from RTCM version 2.4 to RTCM version 3.0.

In order to increase operating safety it is recommended that the current sensor setup be made redundant by either adding backup sensors, or having additional complementary sensors.

- Although simulations showed that the current controller performs well in windy conditions, the controller was designed to accept wind estimates and adapt accordingly. It is recommended that the current ground station be adapted to measure wind conditions and air pressure, in order for these measurements to be uploaded to the control avionics in real time.

- The current ground station telemetry link was found to be adequate, but the data throughput was almost saturated. It is recommended that the bandwidth on the telemetry link be increased to allow for more real-time telemetry to be sent to the ground station. It might be useful to investigate the use of standard Wi-Fi telemetry links, as this will greatly reduce cost, while increasing bandwidth up to the capability of live video streaming.

- The alternative control methodology that was shown in Section 3.8 was thoroughly tested in simulation with good results. It is recommended that this controller be implemented on the avionics and flight tested.

- The model and controller were developed being mindful of automated take-off and landing. It is recommended that the runway and ground-effect model be used in conjunction with better GPS and altitude sensors to perform this task.

- The current non-linear model can be used to easily simulate autorotation conditions by adding blade speed measurements to the system. Having the capability to make an automated autorotation landing greatly increases the number of possible applications that a similar UAV might be considered for - especially if very valuable and expensive equipment is placed on board. It is recommended that the current control structure be adapted to include autorotation estimation, flight and landing.

Appendices

Appendix A

Blade Element Theory

In Chapter 2 the analysis of blade element theory was briefly discussed, but mostly the results from the analysis were noted. This chapter gives a detailed explanation of how blade element theory is used to calculate a fair share of the vehicle dynamics (thrust, induced velocity, main blade drag and torque, main blade flapping angles and stabilizer bar flapping angles).

It was also noted that the vehicle made use of a stabilizer bar to improve flight dynamics. As this bar is an open loop actuator (with respect to the main blades), it will be the best place to start the analysis.

A.1 Stabilizer Bar Flapping

Figure A.1 illustrates all the terminology that will be used in the stabilizer bar flapping dynamics.

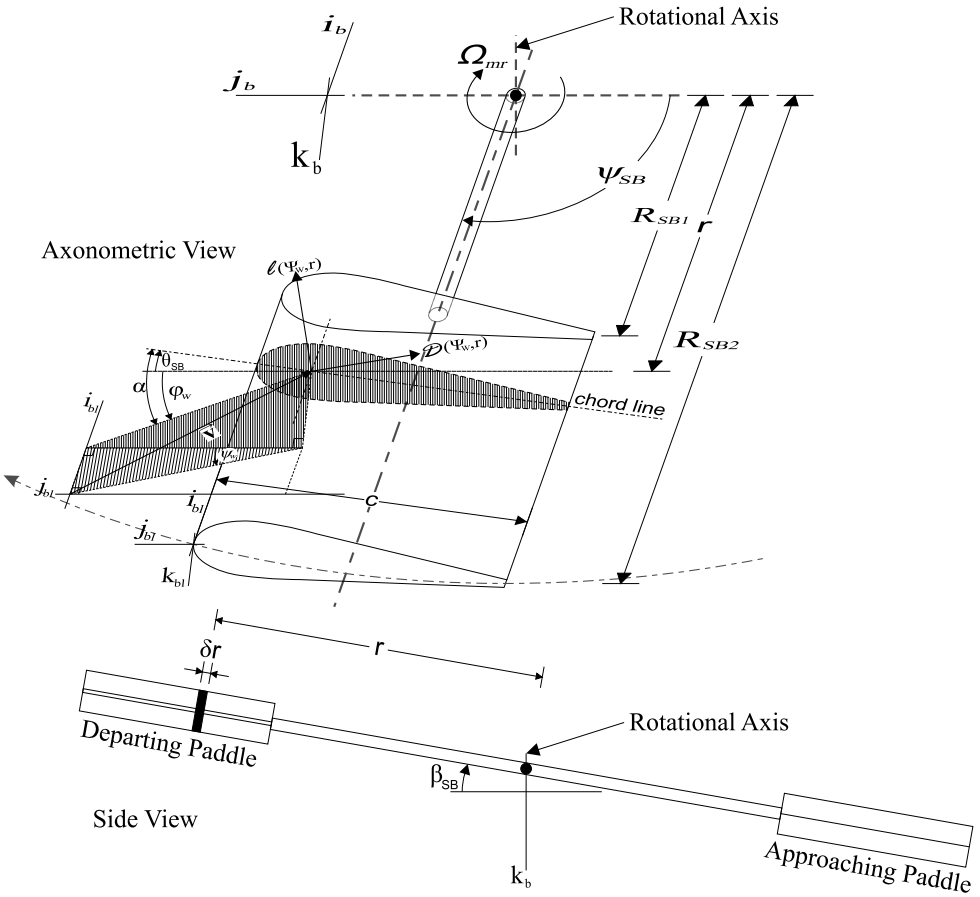


Figure A.1: Stabilizer Bar Blade Element Theory

In Figure A.1 the rotating blade axis system is defined by $\boldsymbol{i}_{bl}$ (always points along the length of the bar, inwards, toward the rotation centre), $\boldsymbol{j}_{bl}$ (always points toward the direction of rotation) and $\boldsymbol{k}_{bl}$ (normal to the rotating plane). The chord length of the aerodynamic part of the bar is c, r is the distance of the blade element from the rotation axis and R_{SB_2} and R_{SB_1} are the outside and inside radiuses of the aerodynamic part of the bar respectively. θ_{SB} denotes the commanded pitch angle of the blade (measured from the rotating plane), while ϕ_w denotes the angle of the incoming airflow's component perpendicular to the rotating plane (in $\boldsymbol{j}_{bl}$ and $\boldsymbol{k}_{bl}$ axes). The angle at which the in plane, non-induced airflow approaches the blade is denoted by ψ_w. Together θ_{SB} and ϕ_w sum to form the *angle of attack*, α, which is the angle from the blade chord to the incoming airflow. Lastly the deflection of the bar from the horizontal body plane is denoted by β_{SB} (positive upwards).

The angle of attack (α) and perpendicular airflow (V) can be rewritten as:

$$V_{\boldsymbol{j}_{bl}} = \Omega_{mr} r + \Omega_{mr} R_{mr} \mu_{mr} \cos(\psi_w) \tag{A.1.1}$$

$$V_{\boldsymbol{k}_{bl}} = w_a - \lambda_{mr} \Omega_{mr} R_{mr} - \dot{\beta}_{SB} r - (p \sin(\psi_{SB}) - q cos(\psi_{SB}))\, r \tag{A.1.2}$$

$$\alpha = \theta_{SB} + \tan^{-1}\left(\frac{V_{\boldsymbol{k}_{bl}}}{V_{\boldsymbol{j}_{bl}}}\right) \tag{A.1.3}$$

, with the commanded pitch angle (as from the swash plate position) being:

$$\theta_{SB} = G_{SB}(\delta_{long} \sin(\psi_{SB}) + \delta_{lat} \cos(\psi_{SB})) \tag{A.1.4}$$

When rotating, a blade element experiences three forces that act upon it in the flapping direction. These are gyroscopic forces, centrifugal forces and a lift force. They can each be written for the blade element as:

$$F^c_{\boldsymbol{k}_{bl}}(r, \psi_{SB}) = -(\delta m(r) \delta r)\, r \Omega^2_{mr} \sin(\beta_{SB}) \tag{A.1.5}$$

$$F^g_{\boldsymbol{k}_{bl}}(r, \psi_{SB}) = -2\,(\delta m(r) \delta r)\, r \Omega_{mr}\, (p \cos(\psi_{SB}) + q \sin(\psi_{SB})) \tag{A.1.6}$$

$$F^l_{\boldsymbol{k}_{bl}}(r, \psi_{SB}) = -\frac{1}{2} \rho V^2_{\boldsymbol{j}_{bl}\boldsymbol{k}_{bl}} a_{sb} \alpha c_{sb} \delta r \cos(\phi_w) \tag{A.1.7}$$

$$\tag{A.1.8}$$

where $\delta m(r)$ is the linear mass density of the blade.

Because the stabilizer bar is a non-flexing stiff structure (i.e. if the one side flaps up, the other side will flap down), the forces on a blade element on the one side of the bar, will also apply a moment on the same element on the opposing side. A differential equation can be written for the moments acting on such a blade element (note that the mass of the opposing element is also included):

$$\begin{aligned} & 2\,(\delta m(r) \delta r)\, r^2 \ddot{\beta}_{SB1} \\ = \; & -2r\,(\delta m(r) \delta r)\, r \Omega_{mr}\, (p \cos(\psi_{SB}) + q \sin(\psi_{SB})) \\ & -2r\,(\delta m(r) \delta r)\, r \Omega_{mr}\, (-p \cos(\psi_{SB}) - q \sin(\psi_{SB})) \\ & -r\,(\delta m(r) \delta r)\, r \Omega^2_{mr} \sin(\beta_{SB}) - r\,(\delta m(r) \delta r)\, r \Omega^2_{mr} \sin(\beta_{SB}) \\ & +r \frac{1}{2} \rho V^2_{\boldsymbol{j}_{bl}\boldsymbol{k}_{bl}} a_{sb} \alpha_{SB1} c_{sb} \delta r \cos(\phi_w) - r \frac{1}{2} \rho V^2_{\boldsymbol{j}_{bl}\boldsymbol{k}_{bl}} a_{sb} \alpha_{SB2} c_{sb} \delta r \cos(\phi_w) \end{aligned} \tag{A.1.9}$$

If it is assumed that the non-aerodynamic section of the stabilizer bar has no weight and creates no lift, the differential Equation in (A.1.9) can be integrated over the length of the bar to calculate the dynamics of the entire bar:

$$\begin{aligned}
&\int_{R_{SB_1}}^{R_{SB_2}} 2\ddot{\beta}_{SB1} r^2 \left(\delta m(r)\delta r\right) \\
= \; &-\int_{R_{SB_1}}^{R_{SB_2}} 4\Omega_{mr} \left(p\cos(\psi_{SB}) + q\sin(\psi_{SB})\right) r^2 \left(\delta m(r)\delta r\right) \\
&-\int_{R_{SB_1}}^{R_{SB_2}} 2\Omega_{mr}^2 \sin(\beta_{SB}) r^2 \left(\delta m(r)\delta r\right) \\
&+\int_{R_{SB_1}}^{R_{SB_2}} \frac{1}{2}\rho a_{sb} c_{sb} r \left(\left(V_{\boldsymbol{j}_{bl}}^{SB1}\right)^2 + \left(V_{\boldsymbol{k}_{bl}}^{SB1}\right)^2\right)\left(\theta_{SB1} + \tan^{-1}\left(\frac{V_{\boldsymbol{k}_{bl}}^{SB1}}{V_{\boldsymbol{j}_{bl}}^{SB1}}\right)\right)\cos(\phi_w)\delta r \\
&-\int_{R_{SB_1}}^{R_{SB_2}} \frac{1}{2}\rho a_{sb} c_{sb} r \left(\left(V_{\boldsymbol{j}_{bl}}^{SB2}\right)^2 + \left(V_{\boldsymbol{k}_{bl}}^{SB2}\right)^2\right)\left(\theta_{SB2} + \tan^{-1}\left(\frac{V_{\boldsymbol{k}_{bl}}^{SB2}}{V_{\boldsymbol{j}_{bl}}^{SB2}}\right)\right)\cos(\phi_w)\delta r
\end{aligned} \tag{A.1.10}$$

The mass integral $\int_{R_{SB_1}}^{R_{SB_2}} r^2 \left(\delta m(r)\delta r\right)$ is actually the formula for the inertia of one stabilizer bar, and would be denoted as I_{SB}. For an advanced ratio of less than 0.15, the horizontal component of airflow is much larger than the vertical component, even for the smallest value of r of R_{SB}. This leads to the approximation that $\left(V_{\boldsymbol{j}_{bl}}^2 + V_{\boldsymbol{k}_{bl}}^2\right) \approx V_{\boldsymbol{j}_{bl}}^2$, $\cos(\phi_w) \approx 1$ and also that $\tan^{-1}\left(\frac{V_{\boldsymbol{k}_{bl}}}{V_{\boldsymbol{k}_{bl}}}\right) \approx \frac{V_{\boldsymbol{k}_{bl}}}{V_{\boldsymbol{k}_{bl}}}$. Also $\theta_{SB1} = -\theta_{SB2}$, so Equation (A.1.10) reduces to:

$$\begin{aligned}
2I_{SB}\ddot{\beta}_{SB1} \;=\; &-4I_{SB}\Omega_{mr}\left(p\cos(\psi_{SB}) + q\sin(\psi_{SB})\right) - 2I_{SB}\Omega_{mr}^2\sin(\beta_{SB}) \\
&+\int_{R_{SB_1}}^{R_{SB_2}} \frac{1}{2}\rho a_{sb} c_{sb} r \left(\theta_{SB1}\left(V_{\boldsymbol{j}_{bl}}^{SB1}\right)^2 + \left(V_{\boldsymbol{j}_{bl}}^{SB1}\right) V_{\boldsymbol{k}_{bl}}^{SB1}\right)\delta r \\
&-\int_{R_{SB_1}}^{R_{SB_2}} \frac{1}{2}\rho a_{sb} c_{sb} r \left(\theta_{SB2}\left(V_{\boldsymbol{j}_{bl}}^{SB2}\right)^2 + \left(V_{\boldsymbol{j}_{bl}}^{SB2}\right) V_{\boldsymbol{k}_{bl}}^{SB2}\right)\delta r
\end{aligned} \tag{A.1.11}$$

Substituting in Equations (A.1.1) and (A.1.2) and applying the integral, results in:

$$\begin{aligned}
& 2I_{SB}\ddot{\beta}_{SB1} \\
= \; & -4I_{SB}\Omega_{mr}\left(p\cos(\psi_{SB})+q\sin(\psi_{SB})\right)-2I_{SB}\Omega_{mr}^2\sin(\beta_{SB}) \\
& +\int_{R_{SB_1}}^{R_{SB_2}}\rho a_{sb}c_{sb}\theta_{SB1}\left(r^3\Omega_{mr}^2+r\left(\Omega_{mr}R_{mr}\mu_{mr}\cos(\psi_w)\right)^2\right)\delta r \\
& +\int_{R_{SB_1}}^{R_{SB_2}}\rho a_{sb}c_{sb}\left(r\Omega_{mr}R_{mr}\mu_{mr}\cos(\psi_w)\left(w_a-\lambda_{mr}\Omega_{mr}R_{mr}\right)\right)\delta r \\
& -\int_{R_{SB_1}}^{R_{SB_2}}\rho a_{sb}c_{sb}r^3\Omega_{mr}\left(\dot{\beta}_{SB1}+p\sin(\psi_{SB})-qcos(\psi_{SB})\right)\delta r \\
= \; & -4I_{SB}\Omega_{mr}\left(p\cos(\psi_{SB})+q\sin(\psi_{SB})\right)-2I_{SB}\Omega_{mr}^2\sin(\beta_{SB}) \\
& +\rho a_{sb}c_{sb}\frac{1}{2}\left(R_{SB_2}^2-R_{SB_1}^2\right)\left(R_{SB_2}^2+R_{SB_1}^2\right)\left(\frac{\Omega_{mr}R_{mr}\mu_{mr}\cos(\psi_w)\left(w_a-\lambda_{mr}\Omega_{mr}R_{mr}\right)}{\left(R_{SB_2}^2+R_{SB_1}^2\right)}\right. \\
& \left.+\frac{1}{2}\Omega_{mr}\left(\theta_{SB1}\Omega_{mr}-\dot{\beta}_{SB1}-p\sin(\psi_{SB})-qcos(\psi_{SB})\right)+\frac{\theta_{SB1}\left(\Omega_{mr}R_{mr}\mu_{mr}\cos(\psi_w)\right)^2}{\left(R_{SB_2}^2+R_{SB_1}^2\right)}\right)
\end{aligned} \tag{A.1.12}$$

If it is assumed that the flapping angle of the stabilizer bar remains small, $\sin(\beta_{SB})$ can be approximated as β_{SB}:

$$\begin{aligned}
& \ddot{\beta}_{SB1}+\Omega_{mr}\frac{\gamma_{SB}}{8}\dot{\beta}_{SB1}+\Omega_{mr}^2\beta_{SB} \\
= \; & -2\Omega_{mr}\left(p\cos(\psi_{SB})+q\sin(\psi_{SB})\right) \\
& +\frac{\gamma_{SB}}{8}\left(\Omega_{mr}\left(\theta_{SB1}\Omega_{mr}-p\sin(\psi_{SB})+qcos(\psi_{SB})\right)\right. \\
& \left.+\frac{2\Omega_{mr}R_{mr}\mu_{mr}\cos(\psi_w)\left(w_a-\lambda_{mr}\Omega_{mr}R_{mr}\right)}{\left(R_{SB_2}^2+R_{SB_1}^2\right)}+\frac{2\theta_{SB1}\left(\Omega_{mr}R_{mr}\mu_{mr}\cos(\psi_w)\right)^2}{\left(R_{SB_2}^2+R_{SB_1}^2\right)}\right)
\end{aligned} \tag{A.1.13}$$

where the stabilizer bar lock number is defined as:

$$\gamma_{SB}=\frac{\rho a_{sb}c_{sb}}{I_{SB}}\left(R_{SB_2}^4-R_{SB_1}^4\right) \tag{A.1.14}$$

Equation (A.1.13) is a second order differential equation for the flapping motion of the stabilizer bar. The flapping angle is very poorly damped, with $\zeta_{SB}=\frac{\gamma_{SB}}{16}\approx 0.05$. The oscillation frequency is however very high and close to the blade speed ($\omega_n^{SB}\approx\Omega_{mr}\approx 180$ [rad/s]). It is impractical to model such fast dynamics, therefore the system can be approximated as being first order, with the same settling time as the second order system.

If it is assumed that the steady state flapping angle takes the form of a first order Fourier series, then the solution can be approximated as:

$$\beta_{SB}(\psi_{SB}) = \left(\frac{1}{(\tau_{SB}s+1)}\right)\left(\beta_{SB}^{long}\sin(\psi_{SB}) - \beta_{SB}^{lat}\cos(\psi_{SB})\right) \quad \text{(A.1.15)}$$

$$\tau_{SB} = \frac{16}{\gamma_{SB}\Omega_{mr}} \quad \text{(A.1.16)}$$

With the steady state derivatives being:

$$\dot{\beta}_{SB1}^{SS}(\psi_{SB}) = \Omega_{mr}\left(\beta_{SB}^{long}\cos(\psi_{SB}) + \beta_{SB}^{lat}\sin(\psi_{SB})\right) \quad \text{(A.1.17)}$$

$$\ddot{\beta}_{SB1}^{SS}(\psi_{SB}) = \Omega_{mr}^{2}\left(-\beta_{SB}^{long}\sin(\psi_{SB}) + \beta_{SB}^{lat}\cos(\psi_{SB})\right) \quad \text{(A.1.18)}$$

The wind axis can be related to the blade axis in the following manner:

$$\psi_{w} = \psi_{SB} - cos^{-1}\left(\frac{u_a}{\Omega_{mr}R_{mr}\mu_{mr}}\right) - \frac{\pi}{2} \quad \text{(A.1.19)}$$

$$\Rightarrow \cos(\psi_{w}) = \sin(\psi_{SB})\left(\frac{u_a}{\Omega_{mr}R_{mr}\mu_{mr}}\right) - \cos(\psi_{SB})\left(\frac{v_a}{\Omega_{mr}R_{mr}\mu_{mr}}\right) \quad \text{(A.1.20)}$$

The steady state solutions of Equations (A.1.15) to (A.1.18), as well as the equation for the blade pitch angle (A.1.4), can now be substituted into the differential Equation (A.1.13):

$$
\begin{aligned}
& \Omega_{mr}^2\left(-\beta_{SB}^{long}\sin(\psi_{SB})+\beta_{SB}^{lat}\cos(\psi_{SB})\right)+\Omega_{mr}^2\left(\beta_{SB}^{long}\sin(\psi_{SB})-\beta_{SB}^{lat}\cos(\psi_{SB})\right) \\
& +\Omega_{mr}\frac{\gamma_{SB}}{8}\Omega_{mr}\left(\beta_{SB}^{long}\cos(\psi_{SB})+\beta_{SB}^{lat}\sin(\psi_{SB})\right) \\
= \; & -2\Omega_{mr}\left(p\cos(\psi_{SB})+q\sin(\psi_{SB})\right) \\
& +\frac{\gamma_{SB}}{8}\Bigg(G_{SB}(\delta_{long}\sin(\psi_{SB})+\delta_{lat}\cos(\psi_{SB}))\Omega_{mr}^2-p\Omega_{mr}\sin(\psi_{SB})+q\Omega_{mr}\cos(\psi_{SB}) \\
& +\frac{2\left(\sin\left(\psi_{SB}\right)u_a-\cos\left(\psi_{SB}\right)v_a\right)\left(w_a-\lambda_{mr}\Omega_{mr}R_{mr}\right)}{\left(R_{SB_2}^2+R_{SB_1}^2\right)} \\
& +\frac{2\left(G_{SB}(\delta_{long}\sin(\psi_{SB})+\delta_{lat}\cos(\psi_{SB}))\right)\left(\sin\left(\psi_{SB}\right)u_a-\cos\left(\psi_{SB}\right)v_a\right)^2}{\left(R_{SB_2}^2+R_{SB_1}^2\right)}\Bigg) \\
& \Omega_{mr}^2\frac{\gamma_{SB}}{8}\left(\beta_{SB}^{long}\cos(\psi_{SB})+\beta_{SB}^{lat}\sin(\psi_{SB})\right) \\
= \; & -2\Omega_{mr}\left(p\cos(\psi_{SB})+q\sin(\psi_{SB})\right) \\
& +\frac{\gamma_{SB}}{8}\left(G_{SB}(\delta_{long}\sin(\psi_{SB})+\delta_{lat}\cos(\psi_{SB}))\Omega_{mr}^2-p\Omega_{mr}\sin(\psi_{SB})+q\Omega_{mr}\cos(\psi_{SB})\right) \\
& +\frac{\gamma_{SB}}{8}\frac{2\left(\sin\left(\psi_{SB}\right)u_a-\cos\left(\psi_{SB}\right)v_a\right)\left(w_a-\lambda_{mr}\Omega_{mr}R_{mr}\right)}{\left(R_{SB_2}^2+R_{SB_1}^2\right)} \\
& +2G_{SB}\delta_{long}\frac{\gamma_{SB}}{8}\frac{\left(\sin^3(\psi_{SB})u_a^2-2\sin^2(\psi_{SB})\cos(\psi_{SB})u_av_a+\sin(\psi_{SB})\cos^2(\psi_{SB})v_a^2\right)}{\left(R_{SB_2}^2+R_{SB_1}^2\right)} \\
& +2G_{SB}\delta_{lat}\frac{\gamma_{SB}}{8}\frac{\left(\cos(\psi_{SB})\sin^2(\psi_{SB})u_a^2-2\sin(\psi_{SB})\cos^2(\psi_{SB})u_av_a+\cos^3(\psi_{SB})v_a^2\right)}{\left(R_{SB_2}^2+R_{SB_1}^2\right)}
\end{aligned}
\tag{A.1.21}
$$

The following trigonometric identities can be applied:

$$\sin^2\left(\psi_{SB}\right) = \frac{1}{2}\left(1-\cos\left(2\psi_{SB}\right)\right) \tag{A.1.22}$$

$$\cos^2\left(\psi_{SB}\right) = \frac{1}{2}\left(1+\cos\left(2\psi_{SB}\right)\right) \tag{A.1.23}$$

$$\sin^2\left(\psi_{SB}\right)\cos\left(\psi_{SB}\right) = \frac{1}{4}\left(\cos\left(\psi_{SB}\right)-\cos\left(3\psi_{SB}\right)\right) \tag{A.1.24}$$

$$\sin\left(\psi_{SB}\right)\cos^2\left(\psi_{SB}\right) = \frac{1}{4}\left(\sin\left(\psi_{SB}\right)+\sin\left(3\psi_{SB}\right)\right) \tag{A.1.25}$$

$$\sin^3\left(\psi_{SB}\right) = \frac{1}{4}\left(3\sin\left(\psi_{SB}\right)-\sin\left(3\psi_{SB}\right)\right) \tag{A.1.26}$$

$$\cos^3\left(\psi_{SB}\right) = \frac{1}{4}\left(3\cos\left(\psi_{SB}\right)+\cos\left(3\psi_{SB}\right)\right) \tag{A.1.27}$$

And then all the first order *sin* terms can be collected to form:

$$\begin{aligned}
&\Omega_{mr}^2 \frac{\gamma_{SB}}{8} \beta_{SB}^{lat} \\
= \; & -2\Omega_{mr} q + \Omega_{mr} \frac{\gamma_{SB}}{8} \left(G_{SB} \delta_{long} \Omega_{mr} - p \right) + \frac{\gamma_{SB}}{8} \frac{2u_a \left(w_a - \lambda_{mr} \Omega_{mr} R_{mr} \right)}{\left(R_{SB_2}^2 + R_{SB_1}^2 \right)} \\
& + \frac{1}{2} G_{SB} \frac{\gamma_{SB}}{8} \frac{\left(3u_a^2 + v_a^2 \right) \delta_{long} - 2u_a v_a \delta_{lat}}{\left(R_{SB_2}^2 + R_{SB_1}^2 \right)}
\end{aligned} \tag{A.1.28}$$

And all the first order *cos* terms can be collected to form:

$$\begin{aligned}
&\Omega_{mr}^2 \frac{\gamma_{SB}}{8} \beta_{SB}^{long} \\
= \; & -2\Omega_{mr} p + \frac{\gamma_{SB}}{8} \left(G_{SB} \delta_{lat} \Omega_{mr}^2 + q \Omega_{mr} \right) - \frac{\gamma_{SB}}{8} \frac{2v_a \left(w_a - \lambda_{mr} \Omega_{mr} R_{mr} \right)}{\left(R_{SB_2}^2 + R_{SB_1}^2 \right)} \\
& + \frac{1}{2} G_{SB} \frac{\gamma_{SB}}{8} \frac{\left(u_a^2 + 3v_a^2 \right) \delta_{lat} - 2u_a v_a \delta_{long}}{\left(R_{SB_2}^2 + R_{SB_1}^2 \right)}
\end{aligned} \tag{A.1.29}$$

From Equations (A.1.28) and (A.1.29) the equations for β_{SB}^{long} ad β_{SB}^{lat} can be written:

$$\begin{aligned}
\beta_{SB}^{long} = \; & -\frac{p}{\Omega_{mr}} \frac{16}{\gamma_{SB}} + \frac{q}{\Omega_{mr}} - \left(w_a - \lambda_{mr} \Omega_{mr} R_{mr} \right) \frac{2v_a}{\Omega_{mr}^2 \left(R_{SB_2}^2 + R_{SB_1}^2 \right)} \\
& + \delta_{lat} G_{SB} \left(1 + \frac{1}{2} \frac{\left(u_a^2 + 3v_a^2 \right)}{\Omega_{mr}^2 \left(R_{SB_2}^2 + R_{SB_1}^2 \right)} \right) - \delta_{long} G_{SB} \frac{u_a v_a}{\Omega_{mr}^2 \left(R_{SB_2}^2 + R_{SB_1}^2 \right)}
\end{aligned} \tag{A.1.30}$$

$$\begin{aligned}
\beta_{SB}^{lat} = \; & -\frac{q}{\Omega_{mr}} \frac{16}{\gamma_{SB}} - \frac{p}{\Omega_{mr}} + \left(w_a - \lambda_{mr} \Omega_{mr} R_{mr} \right) \frac{2u_a}{\Omega_{mr}^2 \left(R_{SB_2}^2 + R_{SB_1}^2 \right)} \\
& + \delta_{long} G_{SB} \left(1 + \frac{1}{2} \frac{\left(3u_a^2 + v_a^2 \right)}{\Omega_{mr}^2 \left(R_{SB_2}^2 + R_{SB_1}^2 \right)} \right) - \delta_{lat} G_{SB} \frac{u_a v_a}{\Omega_{mr}^2 \left(R_{SB_2}^2 + R_{SB_1}^2 \right)}
\end{aligned} \tag{A.1.31}$$

A.2 Main Rotor Flapping

Now that the flapping dynamics has been defined for the stabilizer bar, a similar process can be followed to derive the dynamics of the main rotor blades. The main rotor blades are attached to the hub of the vehicle and they are allowed to flex separately. Rather than to try and model the bending of the blades, it is assumed that the blades do not bend, but rather pivots at the hub. The connection to the hub is then modelled as having a restoring spring moment when the blade pivots from its natural position. This greatly simplifies the blade element theory while still being a fairly accurate representation of the true dynamics. Figure A.2 illustrates this and other terminology used in the analysis.

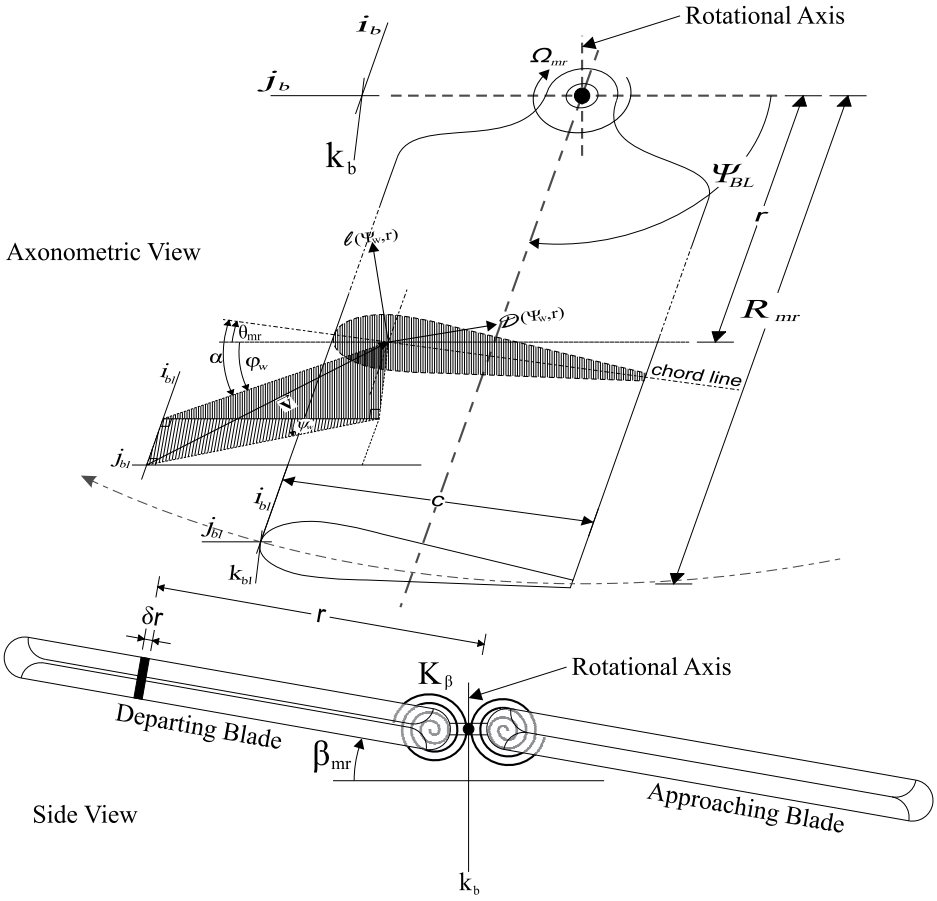


Figure A.2: Main Blade Element Theory

In Figure A.2 the rotating blade axis system is defined by $\boldsymbol{i}_{bl}$ (always points along the length of the blade, inwards, toward the rotation centre), $\boldsymbol{j}_{bl}$ (always point toward the direction of rotation) and $\boldsymbol{k}_{bl}$ (normal to the rotating plane). The chord length of the blade is c, r is the distance of the blade element from the rotation axis and R_{mr} is the total radius of the blade. θ_{mr} denotes the commanded pitch angle

of the blade (measured from the rotating plane), while ϕ_w denotes the angle of the incoming airflow's component perpendicular to the rotating plane (in $\boldsymbol{j}_{bl}$ and $\boldsymbol{k}_{bl}$ axes). The angle at which the in-plane, non-induced airflow approaches the blade is denoted by ψ_w. Together θ_{mr} and ϕ_w sum to form the *angle of attack*, α, which is the angle from the blade chord to the incoming airflow. Lastly the deflection of the blade from the horizontal body plane is denoted by β_{BL} (positive upwards).

The angle of attack (α) and perpendicular airflow (V) can be written as:

$$V_{\boldsymbol{j}_{bl}} = \Omega_{mr} r + \Omega_{mr} R_{mr} \mu_{mr} \cos(\psi_w) \tag{A.2.1}$$

$$V_{\boldsymbol{k}_{bl}} = w_a - \lambda_{mr} \Omega_{mr} R_{mr} - \dot{\beta}_{BL} r - (p \sin(\psi_{BL}) - q cos(\psi_{BL}))\, r \tag{A.2.2}$$

$$\alpha_{mr} = \theta_{mr} + \tan^{-1}\left(\frac{V_{\boldsymbol{k}_{bl}}}{V_{\boldsymbol{j}_{bl}}}\right) \tag{A.2.3}$$

, with the commanded pitch angle (as from the swash plate position and the mechanical linkage to the stabilizer bar) being:

$$\theta_{BL} = \delta_{col} + G_{BL}\delta_{long} \sin(\psi_{BL}) + G_{BL}\delta_{lat} \cos(\psi_{BL}) + K_{SB}\beta_{SB}(\psi_{BL}) \tag{A.2.4}$$

$$\beta_{SB} = \left(\frac{1}{(\tau_{SB}s + 1)}\right)\left(\beta_{SB}^{long} \sin(\psi_{SB}) - \beta_{SB}^{lat} \cos(\psi_{SB})\right) \text{ (From (A.1.15))}$$

$$\psi_{BL} = \psi_{SB} + \frac{\pi}{2} \tag{A.2.5}$$

$$\beta_{SB}(\psi_{BL}) = \left(\frac{1}{(\tau_{SB}s + 1)}\right)\left(\beta_{SB}^{long} \cos(\psi_{BL}) + \beta_{SB}^{lat} \sin(\psi_{BL})\right) \tag{A.2.6}$$

where the stabilizer bar flapping equation has been rewritten with respect to the main blade azimuth angle - Equation (A.2.5) defines the 90 [°] leading of the stabilizer bar in the rotating plane.

When rotating a blade element it experiences three forces that act on it in the flapping direction. These are gyroscopic forces, centrifugal forces and a lift force. They can be written for the blade element as:

$$F^c_{\boldsymbol{k}_{bl}}(r, \psi_{BL}) = -(\delta m(r)\delta r)\, r\Omega_{mr}^2 \sin(\beta_{BL}) \tag{A.2.7}$$

$$F^g_{\boldsymbol{k}_{bl}}(r, \psi_{BL}) = -2\,(\delta m(r)\delta r)\, r\Omega_{mr}\,(p\cos(\psi_{BL}) + q\sin(\psi_{BL})) \tag{A.2.8}$$

$$F^l_{\boldsymbol{k}_{bl}}(r, \psi_{BL}) = -\frac{1}{2}\rho V^2_{\boldsymbol{j}_{bl}\boldsymbol{k}_{bl}} a_{mr} \alpha c_{mr} \delta r \cos(\phi_w) \tag{A.2.9}$$

where $\delta m(r)$ is the linear mass density of the blade.

As a result of the blade stiffness, there is also a restoring moment when the blade pivots from its normal position (in the rotating plane):

$$M^s(\beta_{BL}) = -K_\beta \beta_{BL} \tag{A.2.10}$$

where K_β is the main rotor spring constant. This moment acts on the blade as a whole, so for a blade element, the moment is:

$$M^s_{\boldsymbol{i}_{bl}}(r, \psi_{BL}) = -\frac{\partial K_\beta(r)}{\partial r}\beta_{BL}\delta r \tag{A.2.11}$$

In calculating the blade dynamics, the assumption is made that the transient response of the stabilizer bar flapping is much slower than that of the blades. This will greatly simplify the derivation and will be proved later to be a quite valid assumption. A differential equation can now be written for the moments acting on such a blade element:

$$\begin{aligned}(\delta m(r)\delta r)\, r^2\ddot{\beta}_{BL} &= -\frac{\partial K_\beta(r)}{\partial r}\beta_{BL}\delta r - r2\,(\delta m(r)\delta r)\, r\Omega_{mr}\,(p\cos(\psi_{BL}) + q\sin(\psi_{BL})) \\ &\quad -r\,(\delta m(r)\delta r)\, r\Omega_{mr}^2\sin(\beta_{BL}) \\ &\quad +r\frac{1}{2}\rho V_{\boldsymbol{j}_{bl}\boldsymbol{k}_{bl}}^2 a_{mr}\alpha_{BL}c_{mr}\delta r\cos(\phi_w) \end{aligned} \tag{A.2.12}$$

The differential equation in (A.2.12) can be integrated over the length of the blade to calculate the dynamics of the whole blade:

$$\begin{aligned}&\int_0^{R_{mr}} \ddot{\beta}_{BL} r^2\,(\delta m(r)\delta r) \\ =\ & -\int_0^{R_{mr}} 2\Omega_{mr}\,(p\cos(\psi_{BL}) + q\sin(\psi_{BL}))\, r^2\,(\delta m(r)\delta r) - \int_0^{R_{mr}} \frac{\partial K_\beta(r)}{\partial r}\beta_{BL}\delta r \\ & -\int_0^{R_{mr}} \Omega_{mr}^2\sin(\beta_{BL}) r^2\,(\delta m(r)\delta r) \\ & +\int_0^{R_{mr}} \frac{1}{2}\rho a_{bl}c_{bl} r\left(\left(V_{\boldsymbol{j}_{bl}}^{BL}\right)^2 + \left(V_{\boldsymbol{k}_{bl}}^{BL}\right)^2\right)\left(\theta_{mr} + \tan^{-1}\left(\frac{V_{\boldsymbol{k}_{bl}}^{BL}}{V_{\boldsymbol{j}_{bl}}^{BL}}\right)\right)\cos(\phi_w)\delta r \end{aligned} \tag{A.2.13}$$

The mass integral $\int_0^{R_{mr}} r^2\,(\delta m(r)\delta r)$ is actually the formula for inertia of the one blade, and will be denoted as I_{BL}. For an advance ratio of less than 0.15, the horizontal component of the airflow is much larger than the vertical component, for most values of r. This leads to the approximation that $\left(V_{\boldsymbol{j}_{bl}}^2 + V_{\boldsymbol{k}_{bl}}^2\right) \approx V_{\boldsymbol{j}_{bl}}^2$, $\cos(\phi_w) \approx 1$ and also that $\tan^{-1}\left(\frac{V_{\boldsymbol{k}_{bl}}}{V_{\boldsymbol{j}_{bl}}}\right) \approx \frac{V_{\boldsymbol{k}_{bl}}}{V_{\boldsymbol{j}_{bl}}}$. Equation (A.2.13) reduces to:

$$\begin{aligned} I_{BL}\ddot{\beta}_{BL} &= -K_\beta\beta_{BL} - 2I_{BL}\Omega_{mr}\,(p\cos(\psi_{BL}) + q\sin(\psi_{BL})) - I_{BL}\Omega_{mr}^2\sin(\beta_{BL}) \\ &\quad +\int_0^{R_{mr}} \frac{1}{2}\rho a_{mr}c_{mr} r\left(\theta_{mr}\left(V_{\boldsymbol{j}_{bl}}^{BL}\right)^2 + \left(V_{\boldsymbol{j}_{bl}}^{BL}\right)V_{\boldsymbol{k}_{bl}}^{BL}\right)\delta r \end{aligned} \tag{A.2.14}$$

Substituting in Equations (A.2.1) and (A.2.2) and applying the integral results in:

$$
\begin{aligned}
& I_{BL}\ddot{\beta}_{BL} \\
= \; & -K_\beta\beta_{BL} - 2I_{BL}\Omega_{mr}\left(p\cos(\psi_{BL}) + q\sin(\psi_{BL})\right) - I_{BL}\Omega_{mr}^2\sin(\beta_{BL}) \\
& - \int_0^{R_{mr}} \tfrac{1}{2}\rho a_{mr}c_{mr}\left(\Omega_{mr}r^3 + \Omega_{mr}R_{mr}\mu_{mr}\cos(\psi_w)r^2\right)\left(\dot{\beta}_{BL} + p\sin(\psi_{BL}) - qcos(\psi_{BL})\right)\delta r \\
& + \int_0^{R_{mr}} \tfrac{1}{2}\rho a_{mr}c_{mr}\left(\Omega_{mr}r^2 + \Omega_{mr}R_{mr}\mu_{mr}\cos(\psi_w)r\right)\left(w_a - \lambda_{mr}\Omega_{mr}R_{mr}\right)\delta r \\
& + \int_0^{R_{mr}} \tfrac{1}{2}\rho a_{mr}c_{mr}\theta_{mr}\left(\Omega_{mr}^2r^3 + 2\Omega_{mr}^2R_{mr}\mu_{mr}\cos(\psi_w)r^2 + \Omega_{mr}^2R_{mr}^2\mu_{mr}^2\cos(\psi_w)^2r\right)\delta r \\
\\
= \; & -K_\beta\beta_{BL} - 2I_{BL}\Omega_{mr}\left(p\cos(\psi_{BL}) + q\sin(\psi_{BL})\right) - I_{BL}\Omega_{mr}^2\sin(\beta_{BL}) \\
& - \tfrac{1}{2}\rho a_{mr}c_{mr}\left(\tfrac{1}{4}\Omega_{mr}R_{mr}^4 + \Omega_{mr}R_{mr}\mu_{mr}\cos(\psi_w)\tfrac{1}{3}R_{mr}^3\right)\left(\dot{\beta}_{BL} + p\sin(\psi_{BL}) - qcos(\psi_{BL})\right) \\
& + \tfrac{1}{2}\rho a_{mr}c_{mr}\left(\Omega_{mr}\tfrac{1}{3}R_{mr}^3 + \Omega_{mr}R_{mr}\mu_{mr}\cos(\psi_w)\tfrac{1}{2}R_{mr}^2\right)\left(w_a - \lambda_{mr}\Omega_{mr}R_{mr}\right) \\
& + \tfrac{1}{2}\rho a_{mr}c_{mr}\theta_{mr}\left(\tfrac{1}{4}\Omega_{mr}^2R_{mr}^4 + 2\Omega_{mr}^2R_{mr}\mu_{mr}\cos(\psi_w)\tfrac{1}{3}R_{mr}^3 + \Omega_{mr}^2R_{mr}^2\mu_{mr}^2\cos(\psi_w)^2\tfrac{1}{2}R_{mr}^2\right)
\end{aligned}
\tag{A.2.15}
$$

If it is assumed that the flapping angle of the blade remains small, $\sin(\beta_{BL})$ can be approximated as β_{BL}:

$$
\begin{aligned}
& \ddot{\beta}_{BL} + \frac{\gamma_{mr}}{8}\left(\Omega_{mr} + \Omega_{mr}\mu_{mr}\cos(\psi_w)\frac{4}{3}\right)\dot{\beta}_{BL} + \left(\frac{K_\beta}{I_{BL}} + \Omega_{mr}^2\right)\beta_{BL} \\
= \; & \\
& -2\Omega_{mr}\left(p\cos(\psi_{BL}) + q\sin(\psi_{BL})\right) \\
& - \frac{\gamma_{mr}}{8}\left(\Omega_{mr} + \Omega_{mr}\mu_{mr}\cos(\psi_w)\frac{4}{3}\right)\left(p\sin(\psi_{BL}) - qcos(\psi_{BL})\right) \\
& + \frac{\gamma_{mr}}{8}\left(\Omega_{mr}\frac{4}{3}\frac{1}{R_{mr}} + 2\frac{1}{R_{mr}}\Omega_{mr}\mu_{mr}\cos(\psi_w)\right)\left(w_a - \lambda_{mr}\Omega_{mr}R_{mr}\right) \\
& + \frac{\gamma_{mr}}{8}\theta_{mr}\left(\Omega_{mr}^2 + \frac{8}{3}\Omega_{mr}^2\mu_{mr}\cos(\psi_w) + 2\Omega_{mr}^2\mu_{mr}^2\cos(\psi_w)^2\right)
\end{aligned}
\tag{A.2.16}
$$

where the blade lock number is defined as:

$$
\gamma_{mr} = \frac{\rho a_{mr}c_{mr}}{I_{BL}}R_{mr}^4 \tag{A.2.17}
$$

Equation (A.2.16) is a second order differential equation for the flapping motion of the main rotor blades. The flapping angle is under damped, with $\zeta_{BL} \approx \frac{\gamma_{mr}}{16} \approx 0.23$ and the oscillation frequency is very high and close to the blade speed ($\omega_n^{BL} \approx \Omega_{mr} \approx 180$ [rad/s]). It is impractical to model such fast dynamics, thus the system can be approximated as being first order, with the same settling time as the second order system. It can now be seen that the original assumption to ignore the slow stabilizer bar transient dynamics was a valid one, since the settling time of the blade flapping angle is about five times quicker than that of the stabilizer bar.

If it is assumed that the steady state flapping angle takes on the form of a first order Fourier series, then the solution can be approximated as:

$$\beta_{BL}(\psi_{BL}) = \left(\frac{1}{(\tau_{BL}s+1)}\right)\left(\beta_{BL}^{const} + \beta_{BL}^{long}\cos(\psi_{BL}) + \beta_{BL}^{lat}\sin(\psi_{BL})\right) \quad (A.2.18)$$

$$\tau_{BL} = \frac{16}{\gamma_{mr}\Omega_{mr}} \quad (A.2.19)$$

With the steady state derivatives being:

$$\dot{\beta}_{BL}^{SS}(\psi_{BL}) = \Omega_{mr}\left(-\beta_{BL}^{long}\sin(\psi_{BL}) + \beta_{BL}^{lat}\cos(\psi_{BL})\right) \quad (A.2.20)$$

$$\ddot{\beta}_{BL}^{SS}(\psi_{BL}) = \Omega_{mr}^{2}\left(-\beta_{BL}^{long}\cos(\psi_{BL}) - \beta_{BL}^{lat}\sin(\psi_{BL})\right) \quad (A.2.21)$$

The wind axis can be related to the blade axis in the following manner:

$$\psi_w = \psi_{BL} - cos^{-1}\left(\frac{u_a}{\Omega_{mr}R_{mr}\mu_{mr}}\right) - \frac{\pi}{2} \quad (A.2.22)$$

$$\Rightarrow \cos(\psi_w) = \sin(\psi_{BL})\left(\frac{u_a}{\Omega_{mr}R_{mr}\mu_{mr}}\right) - \cos(\psi_{BL})\left(\frac{v_a}{\Omega_{mr}R_{mr}\mu_{mr}}\right) \quad (A.2.23)$$

The steady state solutions of Equations (A.2.18) to (A.2.21), as well as the Equation for the blade pitch angle (A.2.4), can now be substituted into the differential Equation (A.2.16):

$$
\begin{aligned}
&\Omega_{mr}^2\left(-\beta_{BL}^{long}\cos(\psi_{BL})-\beta_{BL}^{lat}\sin(\psi_{BL})\right)+\left(\frac{K_\beta}{I_{BL}}+\Omega_{mr}^2\right)\left(\beta_{BL}^{const}+\beta_{BL}^{long}\cos(\psi_{BL})+\beta_{BL}^{lat}\sin(\psi_{BL})\right)\\
&+\frac{\gamma_{mr}}{8}\left(\Omega_{mr}+\left(\sin(\psi_{BL})\left(\frac{u_a}{R_{mr}}\right)-\cos(\psi_{BL})\left(\frac{v_a}{R_{mr}}\right)\right)\frac{4}{3}\right)\Omega_{mr}\left(-\beta_{BL}^{long}\sin(\psi_{BL})+\beta_{BL}^{lat}\cos(\psi_{BL})\right)\\
&=\\
&-2\Omega_{mr}(p\cos(\psi_{BL})+q\sin(\psi_{BL}))\\
&-\frac{\gamma_{mr}}{8}\left(\Omega_{mr}+\left(\sin(\psi_{BL})\left(\frac{u_a}{R_{mr}}\right)-\cos(\psi_{BL})\left(\frac{v_a}{R_{mr}}\right)\right)\frac{4}{3}\right)(p\sin(\psi_{BL})-qcos(\psi_{BL}))\\
&+\frac{\gamma_{mr}}{8}\left(\Omega_{mr}\frac{4}{3}\frac{1}{R_{mr}}+2\frac{1}{R_{mr}}\left(\sin(\psi_{BL})\left(\frac{u_a}{R_{mr}}\right)-\cos(\psi_{BL})\left(\frac{v_a}{R_{mr}}\right)\right)\right)(w_a-\lambda_{mr}\Omega_{mr}R_{mr})\\
&+\frac{\gamma_{mr}}{8}\left(\delta_{col}+\left(G_{BL}\delta_{long}+K_{SB}\beta_{SB}^{lat}\right)\sin(\psi_{BL})+\left(G_{BL}\delta_{lat}+K_{SB}\beta_{SB}^{long}\right)\cos(\psi_{BL})\right)\Bigg(\Omega_{mr}^2\\
&+\frac{8}{3}\Omega_{mr}\left(\sin(\psi_{BL})\left(\frac{u_a}{R_{mr}}\right)-\cos(\psi_{BL})\left(\frac{v_a}{R_{mr}}\right)\right)+2\left(\sin(\psi_{BL})\left(\frac{u_a}{R_{mr}}\right)-\cos(\psi_{BL})\left(\frac{v_a}{R_{mr}}\right)\right)^2\Bigg)
\end{aligned}
\tag{A.2.24}
$$

$$
\begin{aligned}
&\Omega_{mr}^2\left(-\beta_{BL}^{long}\cos(\psi_{BL})-\beta_{BL}^{lat}\sin(\psi_{BL})\right)+\left(\frac{K_\beta}{I_{BL}}+\Omega_{mr}^2\right)\left(\beta_{BL}^{const}+\beta_{BL}^{long}\cos(\psi_{BL})+\beta_{BL}^{lat}\sin(\psi_{BL})\right)\\
&+\frac{\gamma_{mr}}{8}\frac{4}{3}\frac{\Omega_{mr}}{R_{mr}}\left(-\sin(\psi_{BL})^2u_a\beta_{BL}^{long}+\sin(\psi_{BL})\cos(\psi_{BL})u_a\beta_{BL}^{lat}\right)\\
&+\frac{\gamma_{mr}}{8}\frac{4}{3}\frac{\Omega_{mr}}{R_{mr}}\left(\sin(\psi_{BL})\cos(\psi_{BL})v_a\beta_{BL}^{long}-\cos(\psi_{BL})^2v_a\beta_{BL}^{lat}\right)\\
&+\frac{\gamma_{mr}}{8}\Omega_{mr}^2\left(-\beta_{BL}^{long}\sin(\psi_{BL})+\beta_{BL}^{lat}\cos(\psi_{BL})\right)\\
&=\\
&-2\Omega_{mr}(p\cos(\psi_{BL})+q\sin(\psi_{BL}))\\
&-\frac{\gamma_{mr}}{8}(\Omega_{mr})(p\sin(\psi_{BL})-qcos(\psi_{BL}))\\
&-\frac{\gamma_{mr}}{8}\frac{1}{R_{mr}}\frac{4}{3}\left(\sin(\psi_{BL})^2u_ap-\sin(\psi_{BL})cos(\psi_{BL})u_aq-\sin(\psi_{BL})\cos(\psi_{BL})v_ap+\cos(\psi_{BL})^2v_aq\right)\\
&+\frac{\gamma_{mr}}{8}\left(\Omega_{mr}\frac{4}{3}\frac{1}{R_{mr}}+2\frac{1}{R_{mr}}\left(\sin(\psi_{BL})\left(\frac{u_a}{R_{mr}}\right)-\cos(\psi_{BL})\left(\frac{v_a}{R_{mr}}\right)\right)\right)(w_a-\lambda_{mr}\Omega_{mr}R_{mr})\\
&+\frac{\gamma_{mr}}{8}\left(\delta_{col}+\left(G_{BL}\delta_{long}+K_{SB}\beta_{SB}^{lat}\right)\sin(\psi_{BL})+\left(G_{BL}\delta_{lat}+K_{SB}\beta_{SB}^{long}\right)\cos(\psi_{BL})\right)\left(\Omega_{mr}^2\right)\\
&+\frac{\gamma_{mr}}{8}\frac{8}{3}\frac{\Omega_{mr}}{R_{mr}}u_a\left(\delta_{col}\sin(\psi_{BL})+\left(G_{BL}\delta_{long}+K_{SB}\beta_{SB}^{lat}\right)\sin(\psi_{BL})^2+\left(G_{BL}\delta_{lat}+K_{SB}\beta_{SB}^{long}\right)\sin(\psi_{BL})\cos(\psi_{BL})\right)\\
&-\frac{\gamma_{mr}}{8}\frac{8}{3}\frac{\Omega_{mr}}{R_{mr}}v_a\left(\delta_{col}\cos(\psi_{BL})+\left(G_{BL}\delta_{long}+K_{SB}\beta_{SB}^{lat}\right)\sin(\psi_{BL})\cos(\psi_{BL})+\left(G_{BL}\delta_{lat}+K_{SB}\beta_{SB}^{long}\right)\cos(\psi_{BL})^2\right)\\
&+\frac{\gamma_{mr}}{8}\frac{2}{R_{mr}^2}u_a^2\left(\delta_{col}\sin(\psi_{BL})^2+\left(G_{BL}\delta_{long}+K_{SB}\beta_{SB}^{lat}\right)\sin(\psi_{BL})^3+\left(G_{BL}\delta_{lat}+K_{SB}\beta_{SB}^{long}\right)\sin(\psi_{BL})^2\cos(\psi_{BL})\right)\\
&-\frac{\gamma_{mr}}{8}\frac{2}{R_{mr}^2}2u_av_a\left(\delta_{col}\sin(\psi_{BL})\cos(\psi_{BL})+\left(G_{BL}\delta_{long}+K_{SB}\beta_{SB}^{lat}\right)\sin(\psi_{BL})^2\cos(\psi_{BL})\right)\\
&-\frac{\gamma_{mr}}{8}\frac{2}{R_{mr}^2}2u_av_a\left(\left(G_{BL}\delta_{lat}+K_{SB}\beta_{SB}^{long}\right)\sin(\psi_{BL})\cos(\psi_{BL})^2\right)\\
&+\frac{\gamma_{mr}}{8}\frac{2}{R_{mr}^2}v_a^2\left(\delta_{col}\cos(\psi_{BL})^2+\left(G_{BL}\delta_{long}+K_{SB}\beta_{SB}^{lat}\right)\sin(\psi_{BL})\cos(\psi_{BL})^2+\left(G_{BL}\delta_{lat}+K_{SB}\beta_{SB}^{long}\right)\cos(\psi_{BL})^3\right)
\end{aligned}
\tag{A.2.25}
$$

The following trigonometric identities can be applied:

$$\sin^2(\psi_{BL}) = \frac{1}{2}(1-\cos(2\psi_{BL})) \tag{A.2.26}$$

$$\cos^2(\psi_{BL}) = \frac{1}{2}(1+\cos(2\psi_{BL})) \tag{A.2.27}$$

$$\sin(\psi_{BL})\cos(\psi_{BL}) = \frac{1}{2}\sin(2\psi_{BL}) \tag{A.2.28}$$

$$\sin^2(\psi_{BL})\cos(\psi_{BL}) = \frac{1}{4}(\cos(\psi_{BL})-\cos(3\psi_{BL})) \tag{A.2.29}$$

$$\sin(\psi_{BL})\cos^2(\psi_{BL}) = \frac{1}{4}(\sin(\psi_{BL})+\sin(3\psi_{BL})) \tag{A.2.30}$$

$$\sin^3(\psi_{BL}) = \frac{1}{4}(3\sin(\psi_{BL})-\sin(3\psi_{BL})) \tag{A.2.31}$$

$$\cos^3(\psi_{BL}) = \frac{1}{4}(3\cos(\psi_{BL})+\cos(3\psi_{BL})) \tag{A.2.32}$$

And then all the zero order terms can be collected to form:

$$\begin{aligned}
&\beta_{BL}^{const}\left(\frac{K_\beta}{I_{BL}}+\Omega_{mr}^2\right)\\
=\;&\frac{\gamma_{mr}}{12}\frac{\Omega_{mr}}{R_{mr}}\left(u_a\beta_{BL}^{long}+v_a\beta_{BL}^{lat}\right)-\frac{\gamma_{mr}}{12}\frac{1}{R_{mr}}(u_a p+v_a q)\\
&+\frac{\gamma_{mr}}{6}\frac{\Omega_{mr}}{R_{mr}}(w_a-\lambda_{mr}\Omega_{mr}R_{mr})+\frac{\gamma_{mr}}{8}\delta_{col}\left(\Omega_{mr}^2+\frac{(u_a^2+v_a^2)}{R_{mr}^2}\right)\\
&+\frac{\gamma_{mr}}{6}\frac{\Omega_{mr}}{R_{mr}}u_a\left(G_{BL}\delta_{long}+K_{SB}\beta_{SB}^{lat}\right)-\frac{\gamma_{mr}}{6}\frac{\Omega_{mr}}{R_{mr}}v_a\left(G_{BL}\delta_{lat}+K_{SB}\beta_{SB}^{long}\right)
\end{aligned} \tag{A.2.33}$$

And all the first order *sin* terms can be collected to form:

$$\begin{aligned}
&\beta_{BL}^{lat}\left(\frac{K_\beta}{I_{BL}}\right)\\
=\;&\beta_{BL}^{long}\frac{\gamma_{mr}}{8}\Omega_{mr}^2-2\Omega_{mr}q-\frac{\gamma_{mr}}{8}\Omega_{mr}p-\frac{\gamma_{mr}}{8}\frac{u_a v_a}{R_{mr}^2}\left(G_{BL}\delta_{lat}+K_{SB}\beta_{SB}^{long}\right)\\
&+\frac{\gamma_{mr}}{4}\frac{u_a}{R_{mr}}\frac{(w_a-\lambda_{mr}\Omega_{mr}R_{mr})}{R_{mr}}+\frac{\gamma_{mr}}{3}\Omega_{mr}\frac{u_a}{R_{mr}}\delta_{col}\\
&+\frac{\gamma_{mr}}{8}\left(\Omega_{mr}^2+\frac{3u_a^2+v_a^2}{2R_{mr}^2}\right)\left(G_{BL}\delta_{long}+K_{SB}\beta_{SB}^{lat}\right)
\end{aligned} \tag{A.2.34}$$

And all the first order *cos* terms can be collected to form:

$$\begin{aligned}
&\beta_{BL}^{long}\left(\frac{K_\beta}{I_{BL}}\right)\\
=\;& -\beta_{BL}^{lat}\frac{\gamma_{mr}}{8}\Omega_{mr}^2 - 2\Omega_{mr}p + \frac{\gamma_{mr}}{8}\Omega_{mr}q\\
&-\frac{\gamma_{mr}}{4}\frac{v_a}{R_{mr}}\frac{(w_a-\lambda_{mr}\Omega_{mr}R_{mr})}{R_{mr}} - \frac{\gamma_{mr}}{3}\Omega_{mr}\frac{v_a}{R_{mr}}\delta_{col}\\
&+\frac{\gamma_{mr}}{8}\left(\Omega_{mr}^2+\frac{u_a^2+3v_a^2}{2R_{mr}^2}\right)\left(G_{BL}\delta_{lat}+K_{SB}\beta_{SB}^{long}\right) - \frac{\gamma_{mr}}{8}\frac{u_a v_a}{R_{mr}^2}\left(G_{BL}\delta_{long}+K_{SB}\beta_{SB}^{lat}\right)
\end{aligned} \tag{A.2.35}$$

The three Equations (A.2.33) to (A.2.35) can be solved simultaneously to acquire the solutions for β_{BL}^{const}, β_{BL}^{long} and β_{BL}^{lat}:

$$\begin{aligned}
\beta_{BL}^{long} =\;& \left(\frac{1}{1+S_\beta^2}\right)\left(\left(\frac{16}{\gamma_{mr}}+S_\beta\right)\frac{q}{\Omega_{mr}} + \left(1-S_\beta\frac{16}{\gamma_{mr}}\right)\frac{p}{\Omega_{mr}}\right.\\
&-2\left(\frac{u_a+S_\beta v_a}{\Omega_{mr}R_{mr}}\right)\frac{(w_a-\lambda_{mr}\Omega_{mr}R_{mr})}{\Omega_{mr}R_{mr}} - \frac{8}{3}\left(\frac{u_a+S_\beta v_a}{\Omega_{mr}R_{mr}}\right)\delta_{col}\\
&-\left(1+\frac{\frac{3}{2}u_a^2+S_\beta u_a v_a+\frac{1}{2}v_a^2}{(\Omega_{mr}R_{mr})^2}\right)\left(G_{BL}\delta_{long}+K_{SB}\beta_{SB}^{lat}\right)\\
&\left.+\left(S_\beta+\frac{\frac{1}{2}S_\beta u_a^2+u_a v_a+\frac{3}{2}S_\beta v_a^2}{(\Omega_{mr}R_{mr})^2}\right)\left(G_{BL}\delta_{lat}+K_{SB}\beta_{SB}^{long}\right)\right)
\end{aligned} \tag{A.2.36}$$

$$\begin{aligned}
\beta_{BL}^{lat} =\;& \left(\frac{1}{1+S_\beta^2}\right)\left(-\left(\frac{16}{\gamma_{mr}}+S_\beta\right)\frac{p}{\Omega_{mr}} + \left(1-S_\beta\frac{16}{\gamma_{mr}}\right)\frac{q}{\Omega_{mr}}\right.\\
&+2\left(\frac{S_\beta u_a-v_a}{\Omega_{mr}R_{mr}}\right)\frac{(w_a-\lambda_{mr}\Omega_{mr}R_{mr})}{\Omega_{mr}R_{mr}} + \frac{8}{3}\left(\frac{S_\beta u_a-v_a}{\Omega_{mr}R_{mr}}\right)\delta_{col}\\
&+\left(1+\frac{\frac{1}{2}u_a^2-S_\beta u_a v_a+\frac{3}{2}v_a^2}{(\Omega_{mr}R_{mr})^2}\right)\left(G_{BL}\delta_{lat}+K_{SB}\beta_{SB}^{long}\right)\\
&\left.+\left(S_\beta+\frac{\frac{3}{2}S_\beta u_a^2-u_a v_a+\frac{1}{2}S_\beta v_a^2}{(\Omega_{mr}R_{mr})^2}\right)\left(G_{BL}\delta_{long}+K_{SB}\beta_{SB}^{lat}\right)\right)
\end{aligned} \tag{A.2.37}$$

$$\begin{aligned}\beta_{BL}^{const} &= \left(\frac{1}{1+\frac{K_\beta}{I_{BL}\Omega_{mr}^2}}\right)\left(\frac{\gamma_{mr}}{12}\left(\frac{u_a}{\Omega_{mr}R_{mr}}\beta_{BL}^{long}+\frac{v_a}{\Omega_{mr}R_{mr}}\beta_{BL}^{lat}\right)\right.\\ &-\frac{\gamma_{mr}}{12}\left(\frac{u_a}{\Omega_{mr}R_{mr}}\frac{p}{\Omega_{mr}}+\frac{v_a}{\Omega_{mr}R_{mr}}\frac{q}{\Omega_{mr}}\right)\\ &+\frac{\gamma_{mr}}{6}\frac{(w_a-\lambda_{mr}\Omega_{mr}R_{mr})}{\Omega_{mr}R_{mr}}+\frac{\gamma_{mr}}{8}\delta_{col}\left(1+\frac{(u_a^2+v_a^2)}{(\Omega_{mr}R_{mr})^2}\right)\\ &\left.+\frac{\gamma_{mr}}{6}\frac{u_a}{\Omega_{mr}R_{mr}}\left(G_{BL}\delta_{long}+K_{SB}\beta_{SB}^{lat}\right)-\frac{\gamma_{mr}}{6}\frac{v_a}{\Omega_{mr}R_{mr}}\left(G_{BL}\delta_{lat}+K_{SB}\beta_{SB}^{long}\right)\right)\end{aligned} \tag{A.2.38}$$

With the blade stiffness factor being:

$$S_\beta = \frac{\frac{K_\beta}{I_{BL}}}{\frac{\gamma_{mr}}{8}\Omega_{mr}^2} \tag{A.2.39}$$

A.3 Main Rotor Drag

Another important part of describing helicopter dynamics is modelling the torque that needs to be produced by the motor during flight. In both fuel and electrically powered vehicles this part of the model will help tremendously in calculating flight efficiency, endurance and range.

We begin the analysis by referring back to Figure A.2 - it can be noted that there are two main forces acting on a blade element in the rotating plane. The first of these is simply the component of lift and the second is the component of profile drag acting in the rotating plane. These can be written as:

$$F_{j_{bl}}^l(r,\psi_{BL}) = \ell(r,\psi_{BL})\sin(\phi_w) \tag{A.3.1}$$

$$F_{j_{bl}}^d(r,\psi_{BL}) = -D(r,\psi_{BL})\cos(\phi_w) \tag{A.3.2}$$

$$\tag{A.3.3}$$

Several literary references, including [15] suggests a suitable formula for the calculation of profile drag of an airfoil:

$$D = \frac{1}{2}\rho V^2 A C_D \tag{A.3.4}$$

where V is the incoming airflow (perpendicular to the span of the wing) and A the area of the wing (approximated by the product of the chord length and the length of the wing). The air density is denoted by ρ, while C_D is the profile drag coefficient.

This formula was applied to the blade element in Figure A.2, together with the lift formula in Equation (A.2.13):

$$F^l_{\boldsymbol{j}_{bl}}(r,\psi_{BL}) = \frac{1}{2}\rho a_{bl}c_{bl}\left(\left(V^{BL}_{\boldsymbol{j}_{bl}}\right)^2+\left(V^{BL}_{\boldsymbol{k}_{bl}}\right)^2\right)\left(\theta_{mr}+\tan^{-1}\left(\frac{V^{BL}_{\boldsymbol{k}_{bl}}}{V^{BL}_{\boldsymbol{j}_{bl}}}\right)\right)\sin(\phi_w)\delta r \tag{A.3.5}$$

$$F^d_{\boldsymbol{j}_{bl}}(r,\psi_{BL}) = -\frac{1}{2}\rho\left(\left(V^{BL}_{\boldsymbol{j}_{bl}}\right)^2+\left(V^{BL}_{\boldsymbol{k}_{bl}}\right)^2\right)c_{mr}\delta r C_{D_{mr}}\cos(\phi_w) \tag{A.3.6}$$

For an advance ratio of less than 0.15, the horizontal component of the airflow is much larger than the vertical component, for most values of r. This leads to the approximation that $\left(V^2_{\boldsymbol{j}_{bl}}+V^2_{\boldsymbol{k}_{bl}}\right)\approx V^2_{\boldsymbol{j}_{bl}}$, $\cos(\phi_w)\approx 1$ and also that $\sin(\phi_w)\approx\phi_w = \tan^{-1}\left(\frac{V_{\boldsymbol{k}_{bl}}}{V_{\boldsymbol{j}_{bl}}}\right)\approx\frac{V_{\boldsymbol{k}_{bl}}}{V_{\boldsymbol{j}_{bl}}}$.

The instantaneous moment that a single blade produces about the rotational axis can now be calculated by integrating the blade element forces and their moment arms along the length of the blade (the above mentioned approximations are also applied):

$$\begin{aligned} Q(\psi_{BL}) &= -\int_0^{R_{mr}}\frac{1}{2}\rho a_{bl}c_{bl}\left(V^{BL}_{\boldsymbol{j}_{bl}}\right)^2\left(\theta_{mr}+\left(\frac{V^{BL}_{\boldsymbol{k}_{bl}}}{V^{BL}_{\boldsymbol{j}_{bl}}}\right)\right)\left(\frac{V^{BL}_{\boldsymbol{k}_{bl}}}{V^{BL}_{\boldsymbol{j}_{bl}}}\right)r\delta r \\ &\quad+\int_0^{R_{mr}}\frac{1}{2}\rho\left(V^{BL}_{\boldsymbol{j}_{bl}}\right)^2 c_{mr}C_{D_{mr}}r\delta r \end{aligned} \tag{A.3.7}$$

Substituting Equations (A.2.1) and (A.2.2) into Equation (A.3.7) results in:

$$\begin{aligned} Q(\psi_{BL}) &= -\int_0^{R_{mr}}\frac{1}{2}\rho a_{bl}c_{bl}\left(\theta_{mr}\left(V^{BL}_{\boldsymbol{j}_{bl}}\right)+\left(V^{BL}_{\boldsymbol{k}_{bl}}\right)\right)\left(V^{BL}_{\boldsymbol{k}_{bl}}\right)r\delta r \\ &\quad+\int_0^{R_{mr}}\frac{1}{2}\rho\left(V^{BL}_{\boldsymbol{j}_{bl}}\right)^2 c_{bl}C_{D_{mr}}r\delta r \\ &= -\frac{1}{2}\rho a_{bl}c_{bl}\int_0^{R_{mr}}\Big((w_a-\lambda_{mr}\Omega_{mr}R_{mr})^2 r\delta r \\ &\quad+\theta_{mr}\left(\Omega_{mr}r+\Omega_{mr}R_{mr}\mu_{mr}\cos(\psi_w)\right)(w_a-\lambda_{mr}\Omega_{mr}R_{mr})\,r\delta r\Big) \\ &\quad+\frac{1}{2}\rho c_{bl}C_{D_{mr}}\int_0^{R_{mr}}\left(\Omega_{mr}r+\Omega_{mr}R_{mr}\mu_{mr}\cos(\psi_w)\right)^2 r\delta r \end{aligned} \tag{A.3.8}$$

Applying the integral results in:

$$\begin{aligned} Q(\psi_{BL}) &= -\frac{1}{2}\rho a_{bl}c_{bl}\Omega^2_{mr}R^4_{mr}\Big(\frac{(w_a-\lambda_{mr}\Omega_{mr}R_{mr})}{\Omega_{mr}R_{mr}}\frac{1}{2} \\ &\quad+\theta_{mr}\left(\frac{1}{3}+\frac{1}{2}\mu_{mr}\cos(\psi_w)\right)\Big)\frac{(w_a-\lambda_{mr}\Omega_{mr}R_{mr})}{\Omega_{mr}R_{mr}} \\ &\quad+\frac{1}{2}\rho c_{bl}C_{D_{mr}}\Omega^2_{mr}R^4_{mr}\left(\frac{1}{4}+2\frac{1}{3}\mu_{mr}\cos(\psi_w)+\frac{1}{2}\mu^2_{mr}\cos^2(\psi_w)\right) \end{aligned} \tag{A.3.9}$$

By assuming that the blade pitch angle is only composed of the commanded collective (stabiliser bar flapping and cyclic commands are small), the torque of the 2 main rotor blades can be averaged over a full rotation:

$$\begin{aligned}
Q_{mr} &= \frac{1}{2\pi}\int_0^{2\pi} 2Q\left(\psi_{BL}\right)\delta\left(\psi_{BL}\right) \\
&= -\frac{1}{2}\rho a_{mr} c_{mr}\Omega_{mr}R_{mr}^3\left(\delta_{col}\left(\frac{1}{3}\right)+\frac{\left(w_a-\lambda_{mr}\Omega_{mr}R_{mr}\right)}{\Omega_{mr}R_{mr}}\frac{1}{2}\right)\left(w_a-\lambda_{mr}\Omega_{mr}R_{mr}\right) \\
&\quad +\frac{1}{8}\rho c_{bl}C_{D_{mr}}\Omega_{mr}^2R_{mr}^4\left(1+\mu_{mr}^2\right) \\
&= \rho(\Omega_{mr}R_{mr})^2\pi R_{mr}^3\Bigg(-\frac{a_{mr}\sigma_{mr}}{2}\left(\delta_{col}\left(\frac{1}{3}\right)+\frac{\left(\mu_{mr}^z-\lambda_{mr}\right)}{2}\right)\left(\mu_{mr}^z-\lambda_{mr}\right) \\
&\quad +\frac{1}{8}C_{D_{mr}}\sigma_{mr}\left(1+\mu_{mr}^2\right)\Bigg) \\
&= \rho(\Omega_{mr}R_{mr})^2\pi R_{mr}^3\left(C_{T_{mr}}\left(\lambda_{mr}-\mu_{mr}^z\right)+\frac{C_{D_{mr}}\sigma_{mr}}{8}\left(1+\mu_{mr}^2\right)\right)
\end{aligned} \tag{A.3.10}$$

Appendix B

Optimal Control

B.1 Optimal Control

Given the discrete MIMO LTI system:

$$\boldsymbol{x}\left(k+1\right) \quad = \quad \boldsymbol{\Phi}\left(k\right)\boldsymbol{x}\left(k\right)+\boldsymbol{\Gamma}\left(k\right)\boldsymbol{u}\left(k\right) \tag{B.1.1}$$

If it is stated that for an interval (k_i, k_f) the total cost of the system can be defined by the cost function:

$$J\left(k_i,k_f\right)=\frac{1}{2}\sum_{k=k_i}^{k_f}\boldsymbol{x}(k)^T\boldsymbol{Q}(k)\boldsymbol{x}(k)+\boldsymbol{x}(k)^T\boldsymbol{M}(k)\boldsymbol{u}(k)+\boldsymbol{u}(k)^T\boldsymbol{M}(k)^T\boldsymbol{x}(k)+\boldsymbol{u}(k)^T\boldsymbol{R}(k)\boldsymbol{u}(k) \tag{B.1.2}$$

, and that the values of $\boldsymbol{Q}\left(k\right)$, $\boldsymbol{R}\left(k\right)$ and $\boldsymbol{M}\left(k\right)$ are known and positive definite.

Then, according to work done by Stengel [20], there must be an optimal solution for $\boldsymbol{x}\left(k\right)$ and $\boldsymbol{u}\left(k\right)$ on the given interval, to minimize the value of J. Let these optimal values be defined as:

$$\boldsymbol{x}\left(k\right) \quad = \quad \boldsymbol{x}\left(k\right)^* \tag{B.1.3}$$

$$\boldsymbol{u}\left(k\right) \quad = \quad \boldsymbol{u}\left(k\right)^* \tag{B.1.4}$$

Several different methods exist to derive these optimal values. As Equation (B.1.2) will be minimized in an effort to obtain an optimal solution for a control algorithm, the most intuitive derivation would be to simply state that the optimal system input will be a linear function of the optimal states:

$$\boldsymbol{u}\left(k\right)^*=-\boldsymbol{C}\left(k\right)\boldsymbol{x}\left(k\right)^* \tag{B.1.5}$$

The principle of optimality can now be applied to the problem.

If the minimum cost of Equation (B.1.2) is rewritten as:

$$J\left(k_i,k_f\right)^* \quad = \quad \frac{1}{2}\sum_{k=k_i}^{k_f}\left[\begin{array}{cc}\boldsymbol{x}\left(k\right)^{*T} & \boldsymbol{u}\left(k\right)^{*T}\end{array}\right]\left[\begin{array}{cc}\boldsymbol{Q}\left(k\right) & \boldsymbol{M}\left(k\right)\\ \boldsymbol{M}\left(k\right)^T & \boldsymbol{R}\left(k\right)\end{array}\right]\left[\begin{array}{c}\boldsymbol{x}\left(k\right)^*\\ \boldsymbol{u}\left(k\right)^*\end{array}\right] \tag{B.1.6}$$

, then the optimal cost can be written in terms of the sum of two optimal cost functions:

$$J\left(k_i,k_f\right)^* = J\left(k_i+1,k_f\right)^* + \frac{1}{2}\left[\begin{array}{cc} \boldsymbol{x}\left(k_i\right)^{*T} & \boldsymbol{u}\left(k_i\right)^{*T} \end{array}\right]\left[\begin{array}{cc} \boldsymbol{Q}\left(k_i\right) & \boldsymbol{M}\left(k_i\right) \\ \boldsymbol{M}\left(k_i\right)^T & \boldsymbol{R}\left(k_i\right) \end{array}\right]\left[\begin{array}{c} \boldsymbol{x}\left(k_i\right)^* \\ \boldsymbol{u}\left(k_i\right)^* \end{array}\right] \tag{B.1.7}$$

Because it was required that the optimal system inputs be a linear function of the optimal states, the optimal cost in Equation (B.1.7) can be written as:

$$J\left(k_i,k_f\right)^* = \frac{1}{2}\boldsymbol{x}\left(k_i\right)^{*T}\boldsymbol{S}\left(k_i,k_f\right)\boldsymbol{x}\left(k_i\right)^* \tag{B.1.8}$$

, with $\boldsymbol{S}\left(k_i\right)$ denoting the current cost weight as a result of $\boldsymbol{x}\left(k_i\right)^*$ and $\boldsymbol{u}\left(k_i\right)^*$, but also all future costs on interval $(k_i+1,\, k_f)$. $\boldsymbol{S}\left(k_i\right)$ will always be positive definite and symmetrical [20].

Using Equation (B.1.8), the optimal cost in equation (B.1.7) can be rewritten in terms of S:

$$\begin{aligned}\frac{1}{2}\boldsymbol{x}\left(k_i\right)^{*T}\boldsymbol{S}\left(k_i\right)\boldsymbol{x}\left(k_i\right)^* &= \frac{1}{2}\boldsymbol{x}\left(k_i+1\right)^{*T}\boldsymbol{S}\left(k_i+1\right)\boldsymbol{x}\left(k_i+1\right)^* \\ &\quad +\frac{1}{2}\left[\begin{array}{cc} \boldsymbol{x}\left(k_i\right)^{*T} & \boldsymbol{u}\left(k_i\right)^{*T} \end{array}\right]\left[\begin{array}{cc} \boldsymbol{Q}\left(k_i\right) & \boldsymbol{M}\left(k_i\right) \\ \boldsymbol{M}\left(k_i\right)^T & \boldsymbol{R}\left(k_i\right) \end{array}\right]\left[\begin{array}{c} \boldsymbol{x}\left(k_i\right)^* \\ \boldsymbol{u}\left(k_i\right)^* \end{array}\right]\end{aligned} \tag{B.1.9}$$

Applying the constraint in Equation (B.1.5) and the system in Equation (B.1.1) to Equation (B.1.9):

$$\begin{aligned}&\boldsymbol{x}(k_i)^{*T}\boldsymbol{S}(k_i)\boldsymbol{x}(k_i)^* \\ &=\left(\boldsymbol{\Phi}(k_i)\boldsymbol{x}(k_i)^*+\boldsymbol{\Gamma}(k_i)\left(-\boldsymbol{C}(k_i)\boldsymbol{x}(k_i)^*\right)\right)^T\boldsymbol{S}(k_i+1)\left(\boldsymbol{\Phi}(k_i)\boldsymbol{x}(k_i)^*+\boldsymbol{\Gamma}(k_i)\left(-\boldsymbol{C}(k_i)\boldsymbol{x}(k_i)^*\right)\right) \\ &+\left[\begin{array}{cc} \boldsymbol{x}(k_i)^{*T} & \left(-\boldsymbol{C}(k_i)\boldsymbol{x}(k_i)^*\right)^T \end{array}\right]\left[\begin{array}{cc} \boldsymbol{Q}(k_i) & \boldsymbol{M}(k_i) \\ \boldsymbol{M}(k_i)^T & \boldsymbol{R}(k_i) \end{array}\right]\left[\begin{array}{c} \boldsymbol{x}(k_i)^* \\ -\boldsymbol{C}(k_i)\boldsymbol{x}(k_i)^* \end{array}\right] \\ &=\boldsymbol{x}(k_i)^{*T}\boldsymbol{\Phi}(k_i)^T\boldsymbol{S}(k_i+1)\boldsymbol{\Phi}(k_i)\boldsymbol{x}(k_i)^*-\boldsymbol{x}(k_i)^{*T}\boldsymbol{\Phi}(k_i)^T\boldsymbol{S}(k_i+1)\boldsymbol{\Gamma}(k_i)\boldsymbol{C}(k_i)\boldsymbol{x}(k_i)^* \\ &-\boldsymbol{x}(k_i)^{*T}\boldsymbol{C}(k_i)^T\boldsymbol{\Gamma}(k_i)^T\boldsymbol{S}(k_i+1)\boldsymbol{\Phi}(k_i)\boldsymbol{x}(k_i)^*+\boldsymbol{x}(k_i)^{*T}\boldsymbol{C}(k_i)^T\boldsymbol{R}(k_i)\boldsymbol{C}(k_i)\boldsymbol{x}(k_i)^* \\ &+\boldsymbol{x}(k_i)^{*T}\boldsymbol{Q}(k_i)\boldsymbol{x}(k_i)^*-\boldsymbol{x}(k_i)^{*T}\boldsymbol{M}(k_i)\boldsymbol{C}(k_i)\boldsymbol{x}(k_i)^*-\boldsymbol{x}(k_i)^{*T}\boldsymbol{C}(k_i)^T\boldsymbol{M}(k_i)^T\boldsymbol{x}(k_i)^* \\ &+\boldsymbol{x}(k_i)^{*T}\boldsymbol{C}(k_i)^T\boldsymbol{\Gamma}(k_i)^T\boldsymbol{S}(k_i+1)\boldsymbol{\Gamma}(k_i)\boldsymbol{C}(k_i)\boldsymbol{x}(k_i)^*\end{aligned} \tag{B.1.10}$$

From Equation (B.1.10) the discrete differential equation for $\boldsymbol{S}\left(k_i\right)$ can be written as:

$$\begin{aligned}\boldsymbol{S}(k_i) &= \boldsymbol{Q}(k_i)+\boldsymbol{\Phi}(k_i)^T\boldsymbol{S}(k_i+1)\boldsymbol{\Phi}(k_i)+\boldsymbol{C}(k_i)^T\left(\boldsymbol{R}(k_i)+\boldsymbol{\Gamma}(k_i)^T\boldsymbol{S}(k_i+1)\boldsymbol{\Gamma}(k_i)\right)\boldsymbol{C}(k_i) \\ &\quad -\left(\boldsymbol{M}(k_i)+\boldsymbol{\Phi}(k_i)^T\boldsymbol{S}(k_i+1)\boldsymbol{\Gamma}(k_i)\right)\boldsymbol{C}(k_i)-\boldsymbol{C}(k_i)^T\left(\boldsymbol{M}(k_i)^T+\boldsymbol{\Gamma}(k_i)^T\boldsymbol{S}(k_i+1)\boldsymbol{\Phi}(k_i)\right)\end{aligned} \tag{B.1.11}$$

Now, in an effort to characterize the relationship between the optimal inputs and optimal states, assume that the states and inputs in Equation (B.1.9) are not

necessarily optimal:

$$\begin{aligned} J\left(k_i, k_f\right) &= \frac{1}{2}\boldsymbol{x}\left(k_i+1\right)^T \boldsymbol{S}\left(k_i+1\right)\boldsymbol{x}\left(k_i+1\right) \\ &\quad +\frac{1}{2}\left[\begin{array}{cc} \boldsymbol{x}\left(k_i\right)^T & \boldsymbol{u}\left(k_i\right)^T \end{array}\right]\left[\begin{array}{cc} \boldsymbol{Q}\left(k_i\right) & \boldsymbol{M}\left(k_i\right) \\ \boldsymbol{M}\left(k_i\right)^T & \boldsymbol{R}\left(k_i\right) \end{array}\right]\left[\begin{array}{c} \boldsymbol{x}\left(k_i\right) \\ \boldsymbol{u}\left(k_i\right) \end{array}\right] \end{aligned} \tag{B.1.12}$$

To ensure this function has the minimum cost for an input $\boldsymbol{u}\left(k_i\right)$, an absolute minimum must be found in the cost function, i.e.:

$$\frac{\partial J\left(k_i, k_f\right)}{\partial \boldsymbol{u}\left(k_i\right)} = 0 \tag{B.1.13}$$

$$\frac{\partial^2 J\left(k_i, k_f\right)}{\partial \boldsymbol{u}\left(k_i\right)^2} > 0 \tag{B.1.14}$$

These derivatives can be calculated by substituting in system Equation (B.1.1):

$$\begin{aligned} \frac{\partial J\left(k_i, k_f\right)}{\partial \boldsymbol{u}\left(k_i\right)} &= \frac{\partial}{\partial \boldsymbol{u}\left(k_i\right)}\frac{1}{2}\Bigg(\left[\begin{array}{cc} \boldsymbol{x}\left(k_i\right)^T & \boldsymbol{u}\left(k_i\right)^T \end{array}\right]\left[\begin{array}{cc} \boldsymbol{Q}\left(k_i\right) & \boldsymbol{M}\left(k_i\right) \\ \boldsymbol{M}\left(k_i\right)^T & \boldsymbol{R}\left(k_i\right) \end{array}\right]\left[\begin{array}{c} \boldsymbol{x}\left(k_i\right) \\ \boldsymbol{u}\left(k_i\right) \end{array}\right] \\ &\quad +\left(\boldsymbol{\Phi}\left(k_i\right)\boldsymbol{x}\left(k_i\right)+\boldsymbol{\Gamma}\left(k_i\right)\boldsymbol{u}\left(k_i\right)\right)^T \boldsymbol{S}\left(k_i+1\right)\left(\boldsymbol{\Phi}\left(k_i\right)\boldsymbol{x}\left(k_i\right)+\boldsymbol{\Gamma}\left(k_i\right)\boldsymbol{u}\left(k_i\right)\right)\Bigg) \\ &= \frac{1}{2}\Bigg(2\left(\boldsymbol{\Phi}\left(k_i\right)\boldsymbol{x}\left(k_i\right)+\boldsymbol{\Gamma}\left(k_i\right)\boldsymbol{u}\left(k_i\right)\right)^T \boldsymbol{S}\left(k_i+1\right)\boldsymbol{\Gamma}\left(k_i\right) \\ &\quad +2\left[\begin{array}{cc} \boldsymbol{x}\left(k_i\right)^T & \boldsymbol{u}\left(k_i\right)^T \end{array}\right]\left[\begin{array}{cc} \boldsymbol{Q}\left(k_i\right) & \boldsymbol{M}\left(k_i\right) \\ \boldsymbol{M}\left(k_i\right)^T & \boldsymbol{R}\left(k_i\right) \end{array}\right]\left[\begin{array}{c} \boldsymbol{0} \\ \boldsymbol{I} \end{array}\right]\Bigg) \\ &= \boldsymbol{x}\left(k_i\right)^T\left(\boldsymbol{\Phi}\left(k_i\right)^T \boldsymbol{S}\left(k_i+1\right)\boldsymbol{\Gamma}\left(k_i\right)+\boldsymbol{M}\left(k_i\right)\right) \\ &\quad +\boldsymbol{u}\left(k_i\right)^T\left(\boldsymbol{\Gamma}\left(k_i\right)^T \boldsymbol{S}\left(k_i+1\right)\boldsymbol{\Gamma}\left(k_i\right)+\boldsymbol{R}\left(k_i\right)\right) \\ &= \left(\left(\boldsymbol{\Gamma}\left(k_i\right)^T \boldsymbol{S}\left(k_i+1\right)^T \boldsymbol{\Phi}\left(k_i\right)+\boldsymbol{M}\left(k_i\right)\right)\boldsymbol{x}\left(k_i\right)\right)^T \\ &\quad +\left(\left(\boldsymbol{\Gamma}\left(k_i\right)^T \boldsymbol{S}\left(k_i+1\right)^T \boldsymbol{\Gamma}\left(k_i\right)+\boldsymbol{R}\left(k_i\right)\right)\boldsymbol{u}\left(k_i\right)\right)^T \\ &= 0 \end{aligned} \tag{B.1.15}$$

$$\begin{aligned} \frac{\partial^2 J\left(k_i, k_f\right)}{\partial \boldsymbol{u}\left(k_i\right)^2} &= \frac{\partial}{\partial \boldsymbol{u}\left(k_i\right)}\Bigg(\left(\left(\boldsymbol{\Gamma}\left(k_i\right)^T \boldsymbol{S}\left(k_i+1\right)^T \boldsymbol{\Phi}\left(k_i\right)+\boldsymbol{M}\left(k_i\right)\right)\boldsymbol{x}\left(k_i\right)\right)^T \\ &\quad +\left(\left(\boldsymbol{\Gamma}\left(k_i\right)^T \boldsymbol{S}\left(k_i+1\right)^T \boldsymbol{\Gamma}\left(k_i\right)+\boldsymbol{R}\left(k_i\right)\right)\boldsymbol{u}\left(k_i\right)\right)^T\Bigg) \\ &= \boldsymbol{\Gamma}\left(k_i\right)^T \boldsymbol{S}\left(k_i+1\right)\boldsymbol{\Gamma}\left(k_i\right)+\boldsymbol{R}\left(k_i\right) \end{aligned} \tag{B.1.16}$$

The second derivative in Equation (B.1.16) is always positive, as $\boldsymbol{S}(k_i+1)$ is positive definite. As a result, the cost in Equation (B.1.12) will be a minimum if Equation (B.1.15) holds:

$$\boldsymbol{u}(k_i) = -\left(\boldsymbol{\Gamma}(k_i)^T\boldsymbol{S}(k_i+1)^T\boldsymbol{\Gamma}(k_i)+\boldsymbol{R}(k_i)\right)^{-1}\left(\boldsymbol{\Gamma}(k_i)^T\boldsymbol{S}(k_i+1)^T\boldsymbol{\Phi}(k_i)+\boldsymbol{M}(k_i)\right)\boldsymbol{x}(k_i) \tag{B.1.17}$$

$$\Rightarrow \boldsymbol{C}(k_i) = \left(\boldsymbol{\Gamma}(k_i)^T\boldsymbol{S}(k_i+1)\boldsymbol{\Gamma}(k_i)+\boldsymbol{R}(k_i)\right)^{-1}\left(\boldsymbol{\Gamma}(k_i)^T\boldsymbol{S}(k_i+1)\boldsymbol{\Phi}(k_i)+\boldsymbol{M}(k_i)\right) \tag{B.1.18}$$

where the solution of $\boldsymbol{C}(k_i)$ comes from the requirement in Equation (B.1.5).

Equation (B.1.11) can now be rewritten using Equation (B.1.18), for any k on the interval (k_i, k_f):

$$\begin{aligned}\boldsymbol{S}(k) &= \boldsymbol{Q}(k)+\boldsymbol{\Phi}(k)^T\boldsymbol{S}(k+1)\boldsymbol{\Phi}(k)\\ &- \left(\boldsymbol{M}(k)+\boldsymbol{\Phi}(k)^T\boldsymbol{S}(k+1)\boldsymbol{\Gamma}(k)\right)\\ &\quad \left(\boldsymbol{\Gamma}(k)^T\boldsymbol{S}(k+1)\boldsymbol{\Gamma}(k)+\boldsymbol{R}(k)\right)^{-1}\left(\boldsymbol{\Gamma}(k)^T\boldsymbol{S}(k+1)\boldsymbol{\Phi}(k)+\boldsymbol{M}(k)\right)\end{aligned} \tag{B.1.19}$$

Appendix C

Results

C.1 Test Results

C.1.1 Wind Simulated Hover Flight

Figures C.1 and C.2 show results of hover simulations, with wind approaching the vehicle from the longitudinal directions. Additional random wind gusts were also added to the simulation. The results are very promising, with little difference when compared to no-wind conditions.

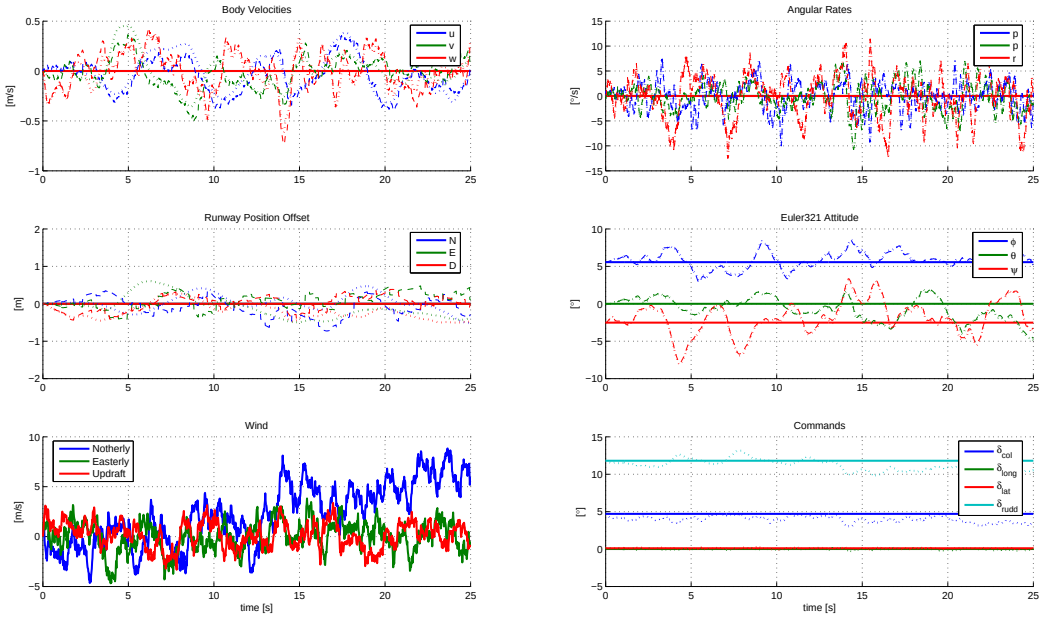


Figure C.1: Hover Flight Results with 5 [m/s] Head Wind and 3 [m/s] Gusts

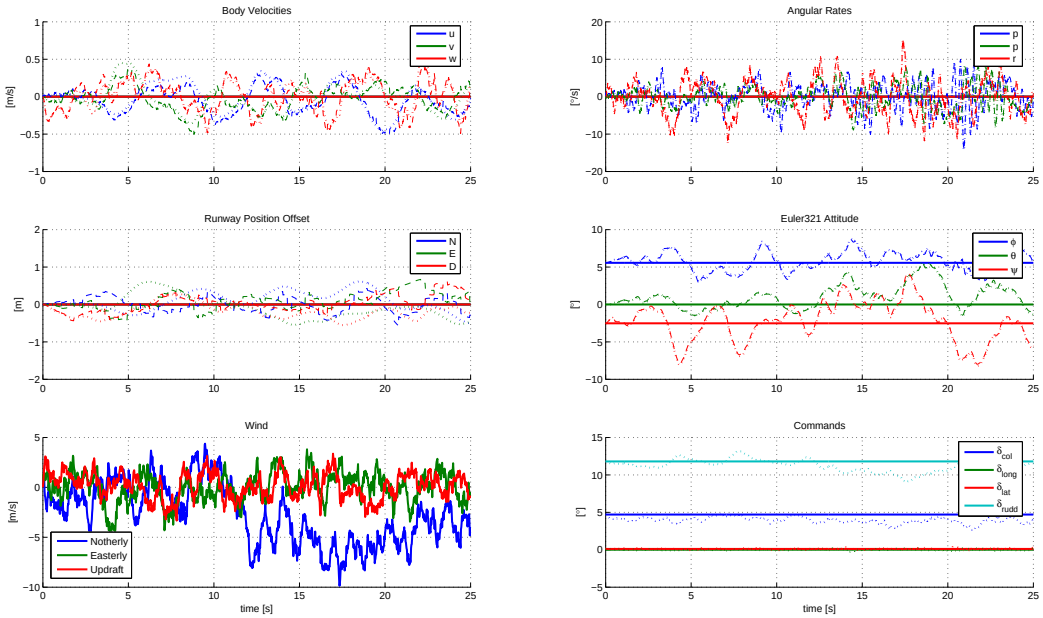


Figure C.2: Hover Flight Results with 5 [m/s] Rear Wind and 3 [m/s] Gusts

Figures C.3 and C.4 show hover simulation results, with wind approaching the vehicle from the lateral directions plus additional gusts from all directions. As expected the results are slightly worse than for wind from the front because of the destabilising effect of the vertical tail fin, but the results are still very good nonetheless. Notice the change in roll angle, as the controller compensates for the additional lateral drag.

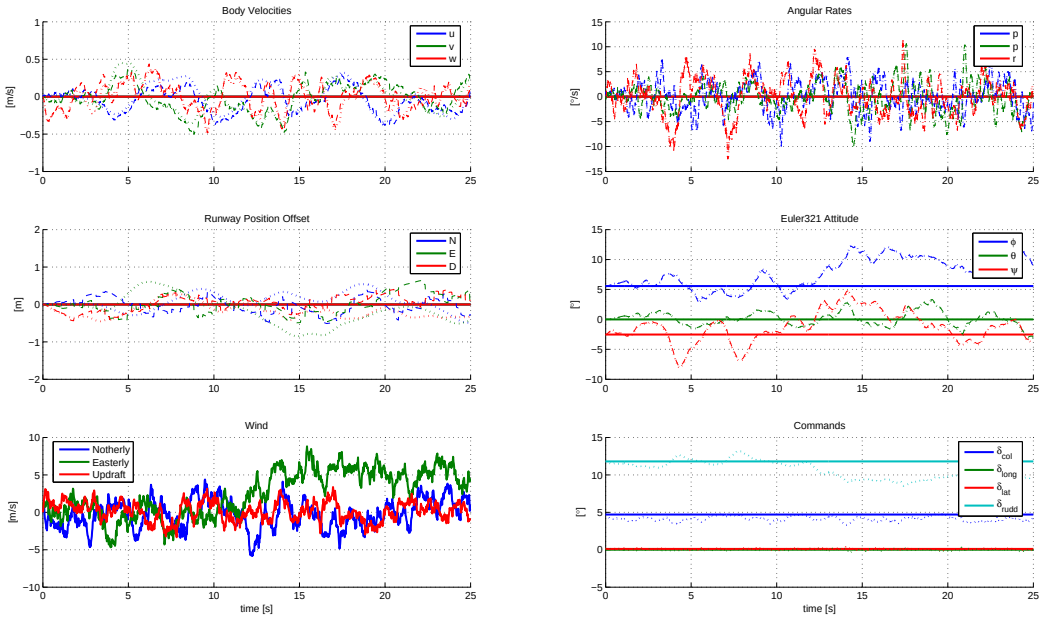


Figure C.3: Hover Flight Results with 5 [m/s] Starboard Side Wind and 3 [m/s] Gusts

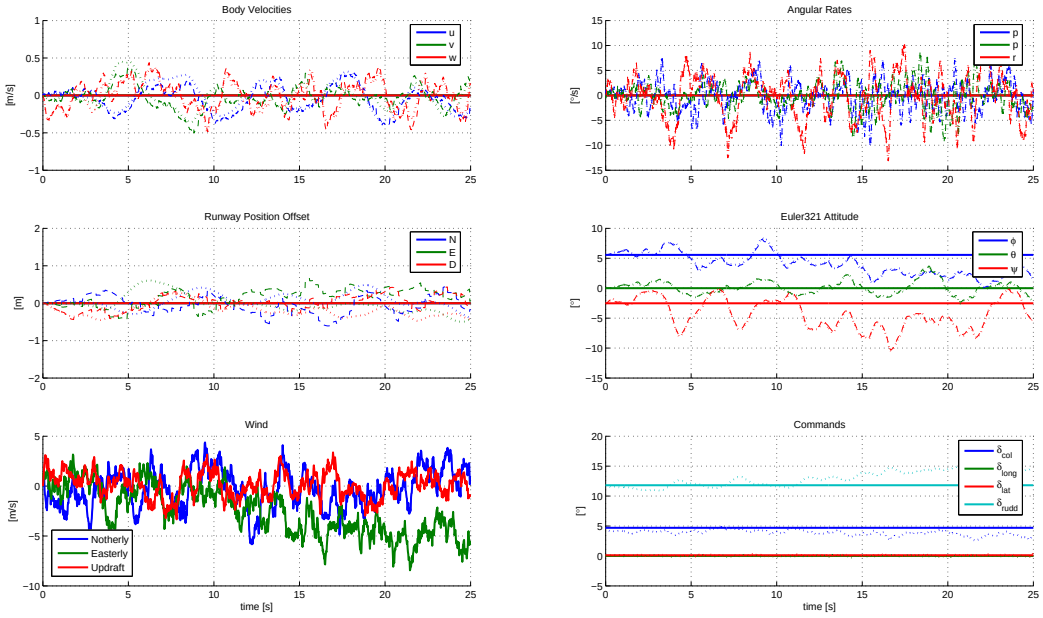


Figure C.4: Hover Flight Results with 5 [m/s] Port Side Wind and 3 [m/s] Gusts

Figures C.5 and C.6 show hover simulation results in up- and downdraft wind conditions. These are especially difficult flying conditions, as the effectiveness of all the vehicle actuators changes significantly with this dramatic increase/decrease

in airflow through the blade plane (unknown to the controller).

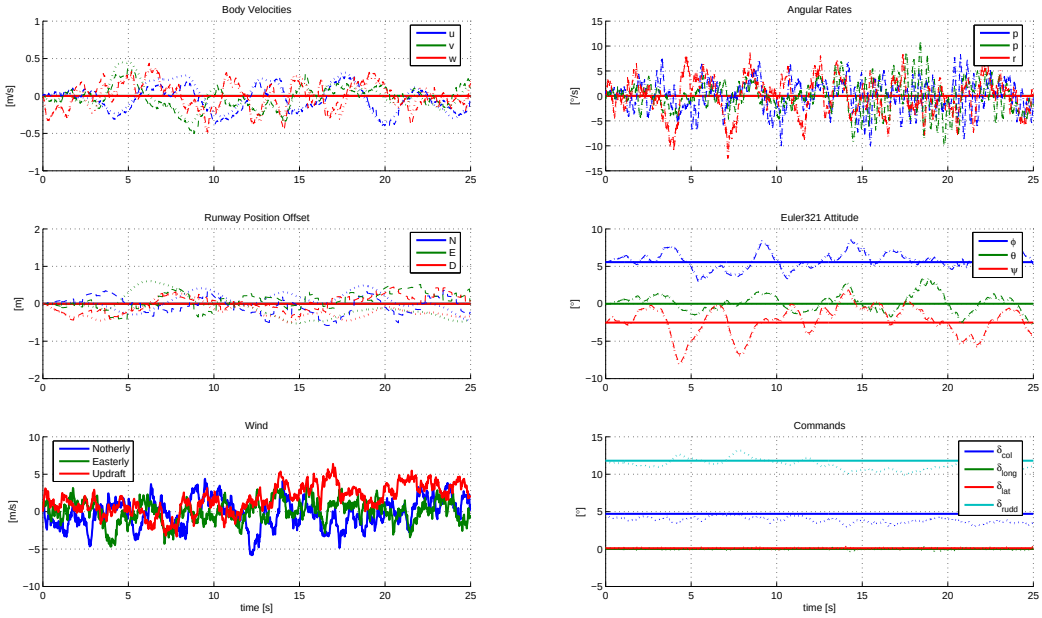


Figure C.5: Hover Flight Results with 3 [m/s] Updraft and 3 [m/s] Wind Gusts

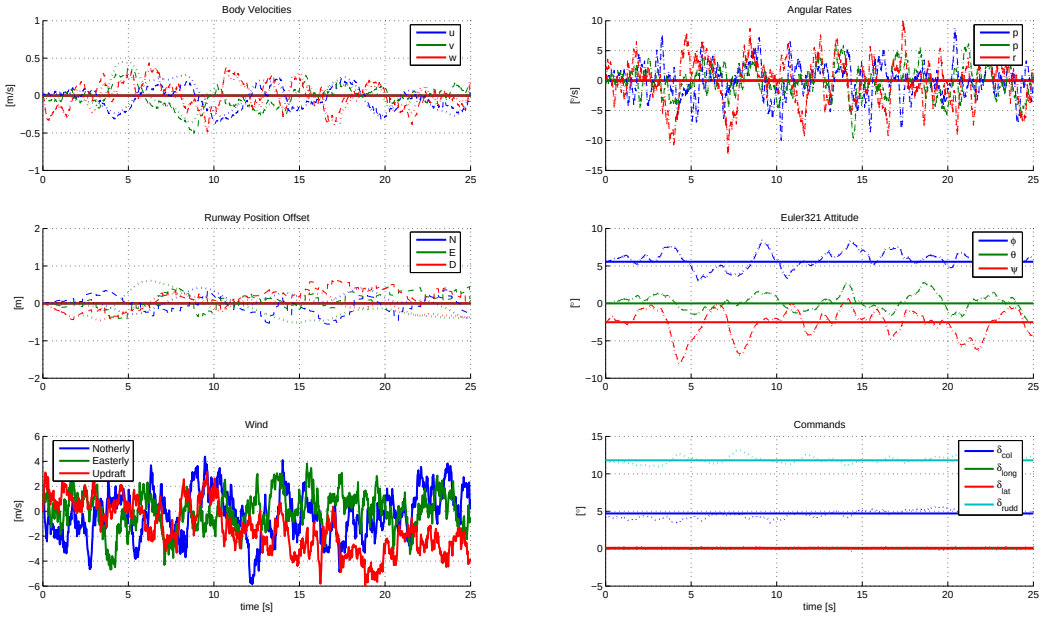


Figure C.6: Hover Flight Results with 3 [m/s] Downdraft and 3 [m/s] Wind Gusts

Figures C.7 and C.8 show simulation results of ascending and descending flight, up to a maximum vertical rate of 4 [m/s].

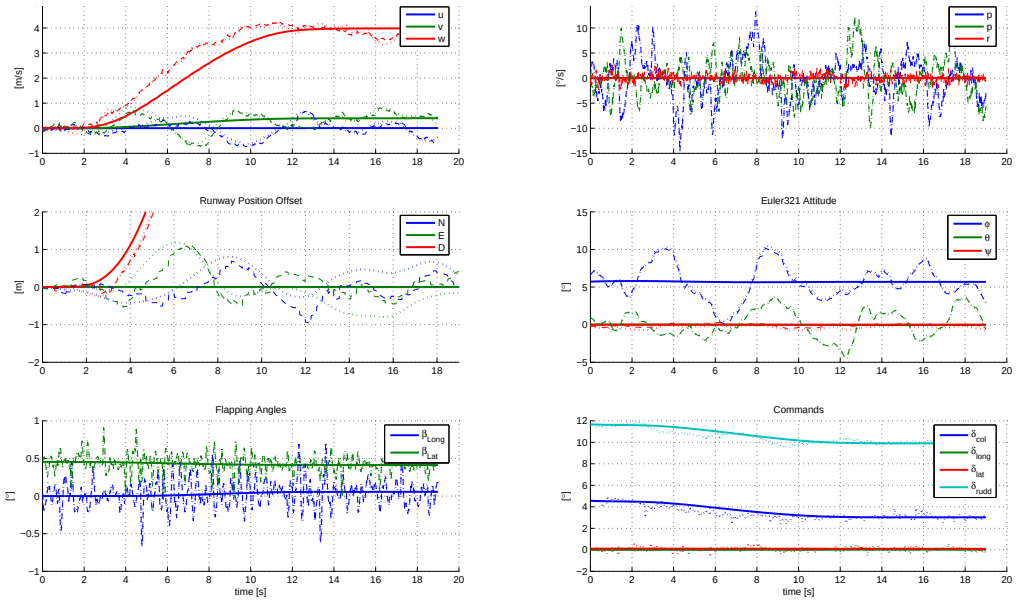


Figure C.7: 4 [m/s] Downwards Flight Results

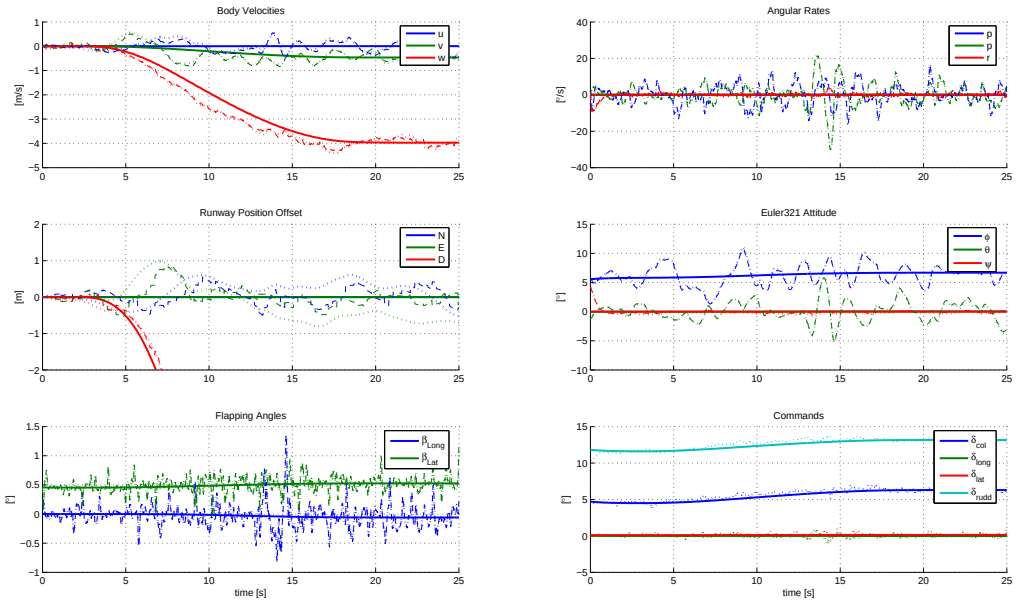


Figure C.8: 4 [m/s] Upwards Flight Results

Figure C.9 shows simulation results of lateral flight up to a maximum speed of 5 [m/s].

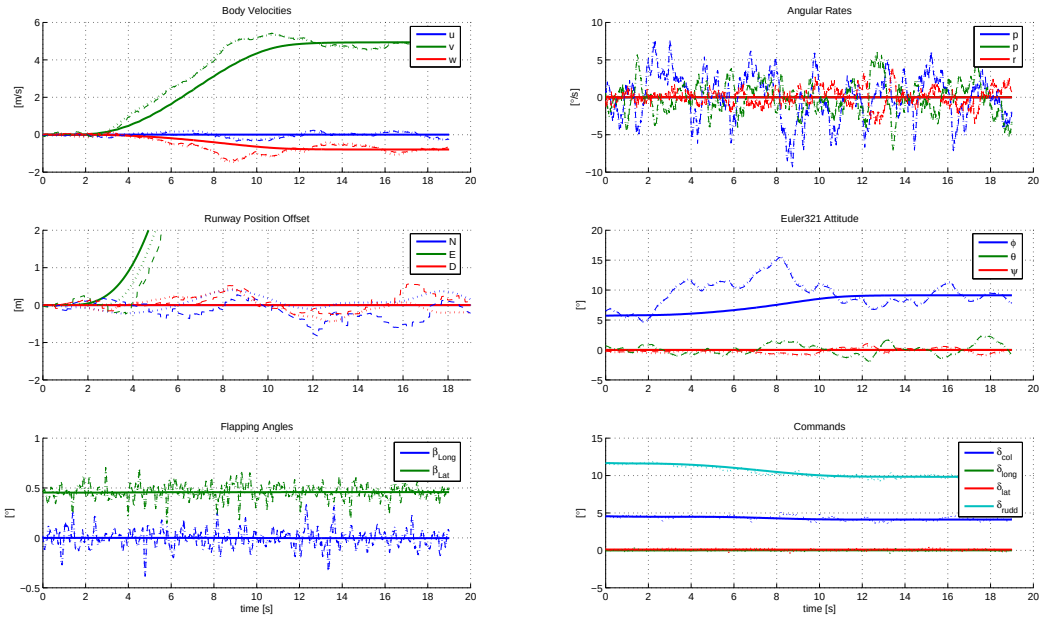


Figure C.9: 5 [m/s] Lateral Flight Results

Figure C.10 shows simulation results of yaw pirouettes up to a maximum rate of positive and negative 60 $[^\circ/s]$.

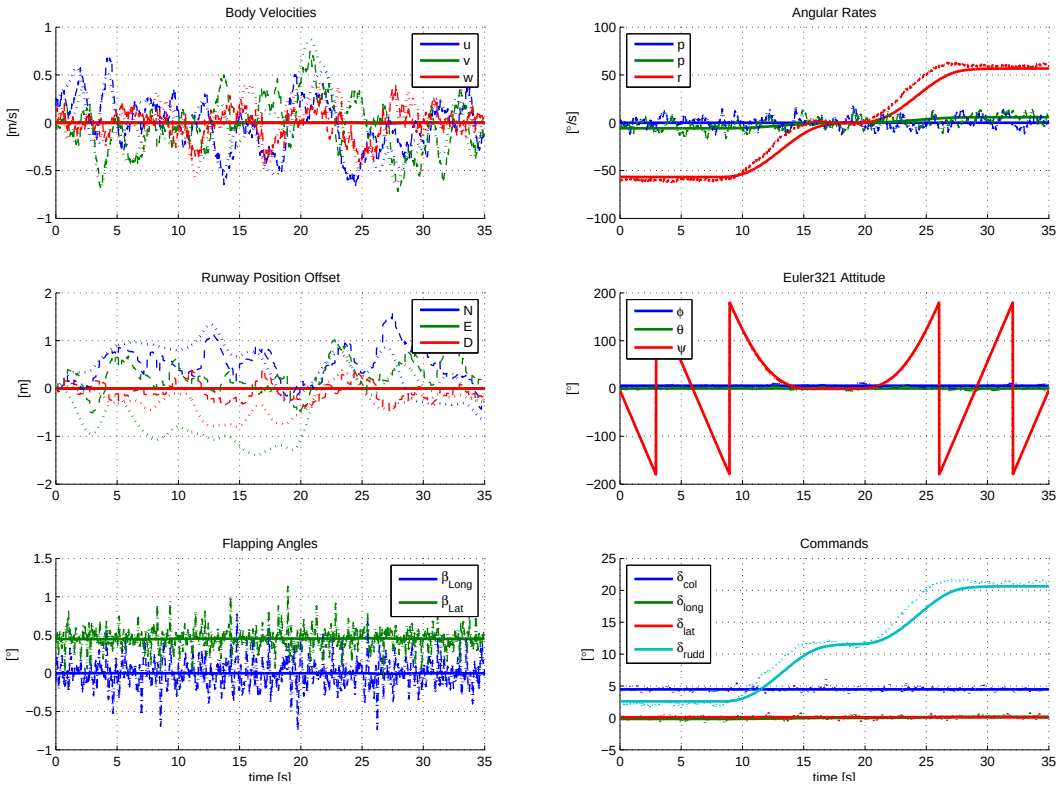


Figure C.10: 60 $[^\circ/s]$ Pirouette Flight Results

C.2 Consecutive Loop Closure Design

The following figures will give a quick overview of the design of the consecutive loop closure controller using root-locus design techniques. The design guidelines were taken from Carstens [3]. Any coupling in the model was forcefully removed, so that a separate model was available for every one of the four system plants.

C.2.1 Longitudinal Loop

Because of the inherent natural rate feedback the stabilizer bar adds to the system, no additional rate feedback was applied. The angle gain was increased from zero until at a gain of 0.12 the system had a phase margin of 45 [°].

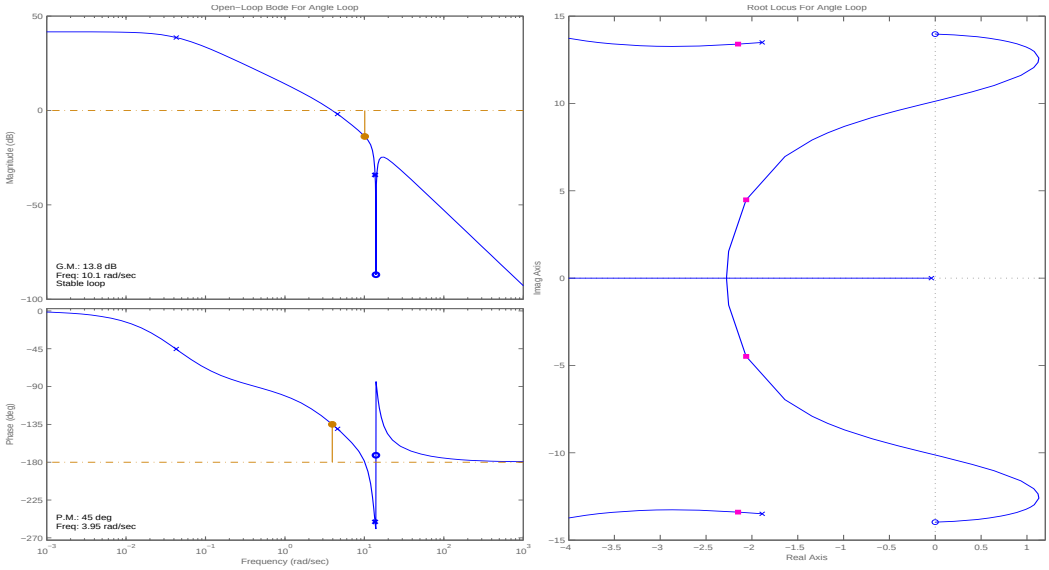


Figure C.11: Consecutive Loop Closure, Longitudinal Angle Loop Root Locus

The velocity feedback was increased until at a gain of 0.5 the system had a gain margin of 30 [dB].

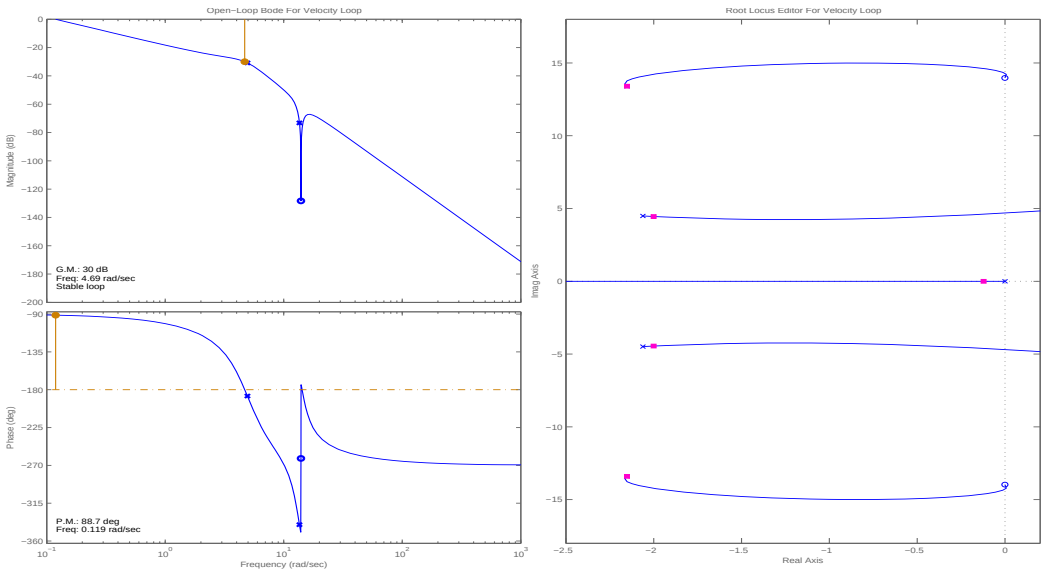


Figure C.12: Consecutive Loop Closure, Longitudinal Velocity Loop Root Locus

Finally an integrator was added to the position loop to compensate for position errors. The speed of the integrator was changed by changing the position of the zero. A settling time of 6 seconds was achieved with a zero position of -4 [mrad/s] and a position gain of 0.65:

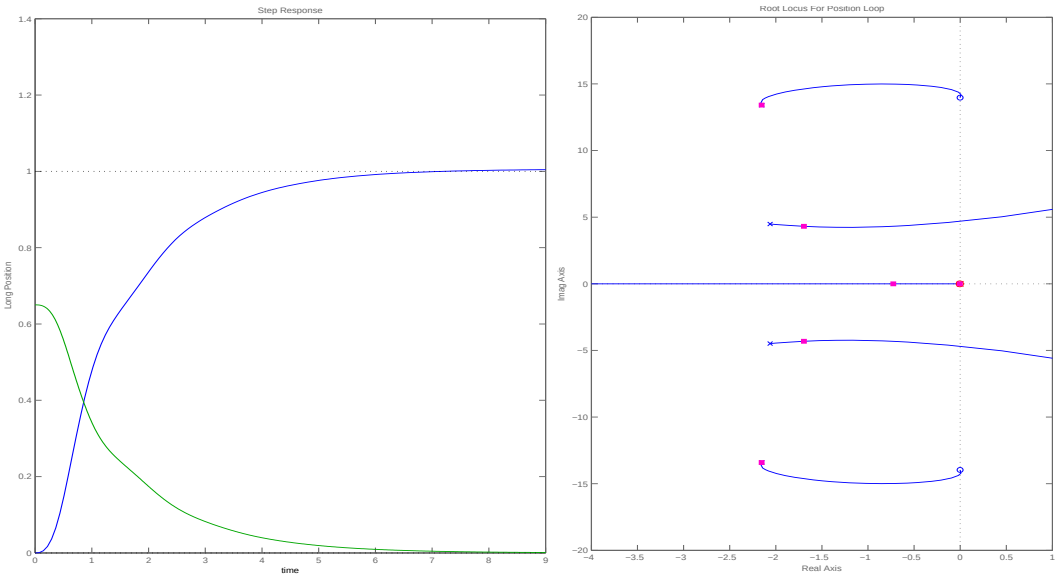


Figure C.13: Consecutive Loop Closure, Longitudinal Position Loop Root Locus

C.2.2 Lateral Loop

No additional rate feedback was added to the natural feedback of the stabilizer bar. The angle gain was increased until at 0.12 the system had a phase margin of 45 [°].

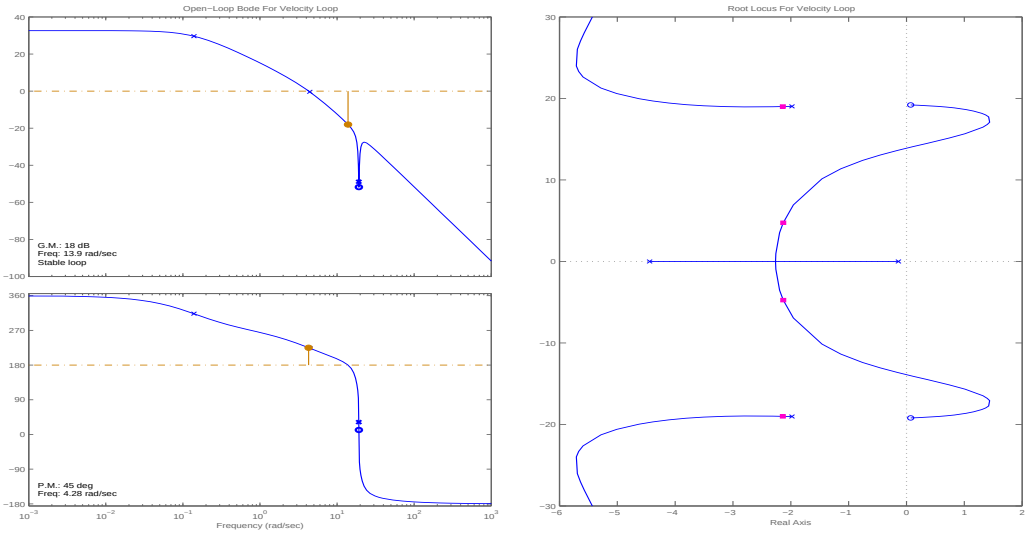


Figure C.14: Consecutive Loop Closure, Lateral Angle Loop Root Locus

The velocity feedback was increased until at a gain of 0.6 the system had a gain margin of 30 [dB]:

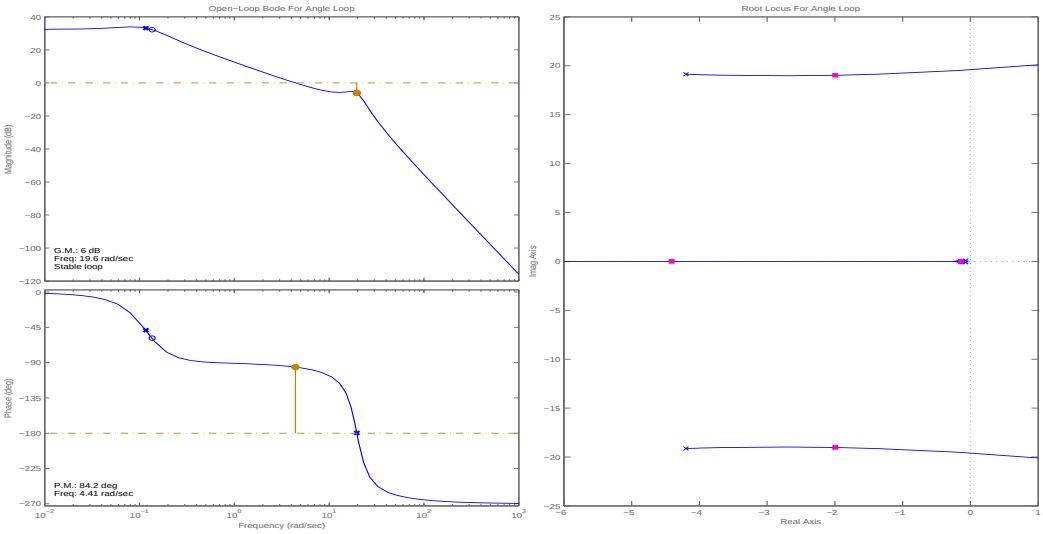


Figure C.15: Consecutive Loop Closure, Lateral Velocity Loop Root Locus

Finally an integrator was added to the position loop to compensate for position errors. The speed of the integrator was changed by changing the position of the PI controller's zero. A settling time of 6 seconds was achieved with a zero position of -4 [mrad/s] and a position gain of 0.7:

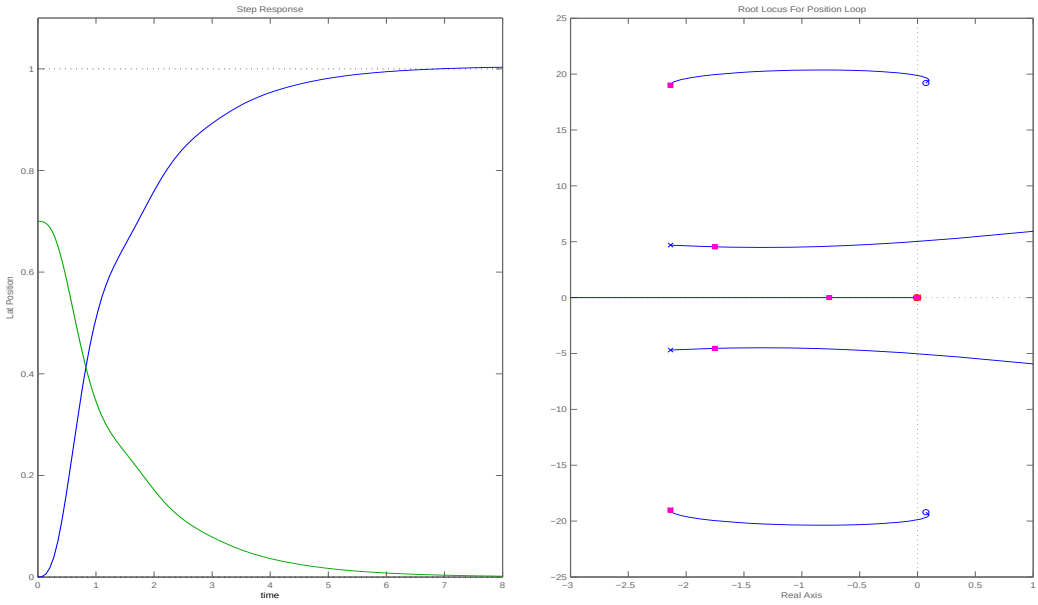


Figure C.16: Consecutive Loop Closure, Lateral Position Loop Root Locus

C.2.3 Heave Loop

The vertical velocity gain was increased until at -0.05 the step response had a settling time of 0.5 [s].

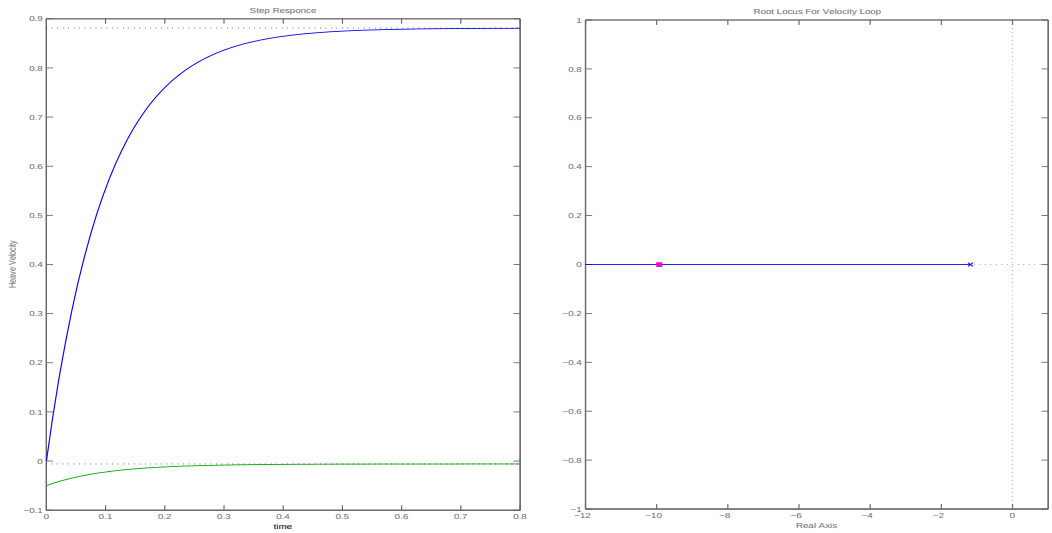


Figure C.17: Consecutive Loop Closure, Heave Velocity Loop Root Locus

Finally an integrator was added to the position loop to compensate for position errors. The speed of the integrator was changed by changing the position of the PI controller's zero. A settling time of 6 seconds was achieved with a zero position of -200 [mrad/s] and a position gain of -2.5:

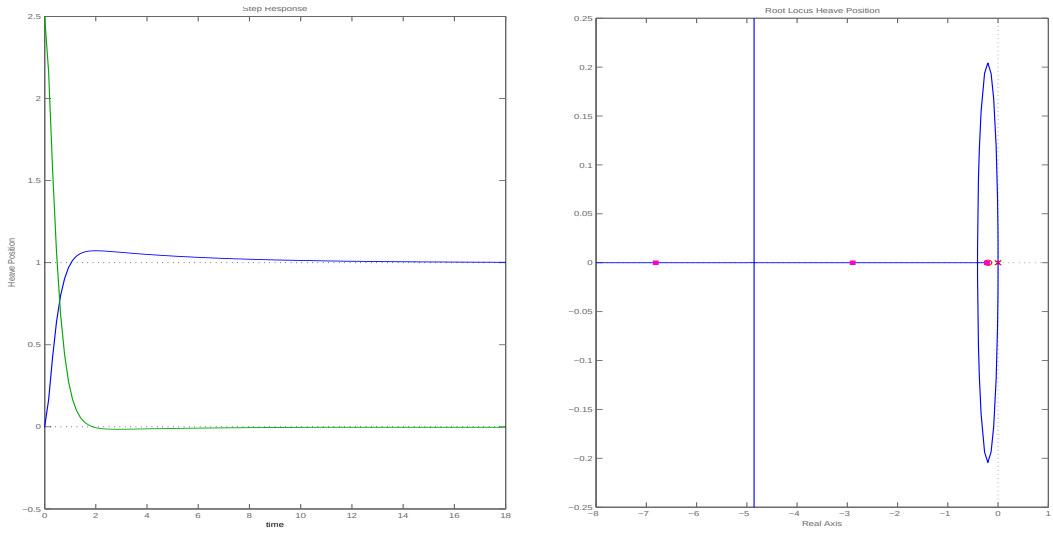


Figure C.18: Consecutive Loop Closure, Heave Position Loop Root Locus

C.2.4 Heading Loop

Finally an integrator was added to the heading loop to compensate for heading errors. The speed of the integrator was changed by changing the position of the PI controller's zero. A settling time of 6 seconds was achieved with a zero position of -350 [mrad/s] and a heading gain of 0.5:

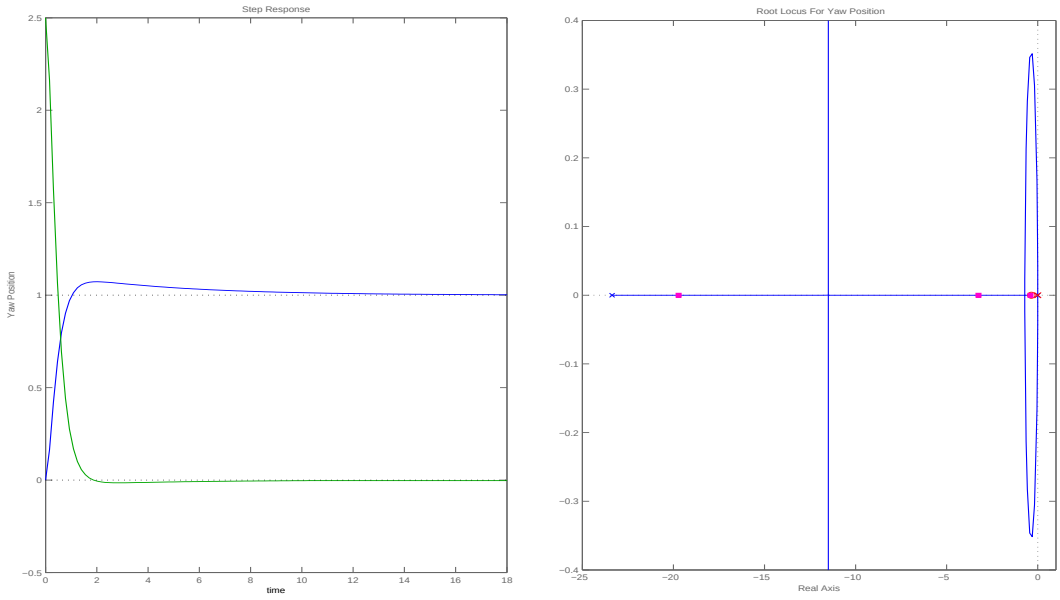


Figure C.19: Consecutive Loop Closure, Heading Position Loop Root Locus

Bibliography

[1] F. Mooren, *"Practical Work Experience Period - Helicopter Platform of the University of Stellenbosch Prestudy"*, Delft University of Technology, April 1996

[2] P.C. Calitz, *"HOPTUS: Elektries-Aangedrewe Helikopter Platform"*, University of Stellenbosch, October 1998

[3] N. Carstens, *"Development of a Low-Cost, Low-Weight Flight Control System for an Electrically Powered Model Helicopter"*, University of Stellenbosch, April 2005

[4] S. Groenewald, *"Development of a Rotary-Wing Test Bed for Autonomous Flight"*, University of Stellenbosch, December 2005

[5] J. Venter, *"Development of an Experimental Tilt-Wing VTOL Unmanned Aerial Vehicle"*, University of Stellenbosch, December 2005

[6] W.J. Hough, *"Autonomous Aerobatic Flight of a Fixed Wing Unmanned Aerial Vehicle"*, University of Stellenbosch, March 2007

[7] L.E. Rossouw, *"Autonomous Flight of an Unmanned Helicopter"*, University of Stellenbosch, April 2008

[8] B. Mettler, M.B. Tischler, & T. Kanade, *"System Identification of Small-Size Unmanned Helicopter Dynamics"* 55th Forum of the American Helicopter Society, 1999.

[9] D.H Shim, *"Hierarchical Flight Control System Syntheses for Rotorcraft-based Unmanned Aerial Vehicles"*, Berkeley, 2000

[10] V. Gavrilets, B. Mettler & E. Feron, *"Dynamic Model for X-Cell 60 Helicopter in Low Advance Ratio Flight"*, MIT Cambridge, December 2002

[11] G.D. Padfield, *"Helicopter Flight Dynamics: The Theory and Application of Flying Qualities and Simulation Modeling"*, AIAA, 1996

[12] A.R.S. Bramwell, *"Helicopter Dynamics"*, Edward Arnolrd, 1976

[13] J. Diebel, *"Representing Attitude: Euler Angles, Unit Quaternions, and Rotation Vectors"*, Stanford University, October 2006

[14] L.N. Hand & J.D. Finch, *"Analytical Mechanics"*, Cambridge University Press, 1998

[15] M.V. Cook, *"Flight Dynamics Principles"*, Oxford, 1997

[16] I.H. Abbott & A.E. von Doenhoff, *"Theory of Wing Sections : Including a Summary of Airfoil Data"*, Dover Publications, 1959

[17] H. Glauert, *"The elements of Aerofoil and Airscrew Theory"*, Cambridge University Press, 1926

[18] I.C. Cheeseman & W.E. Bennett, *"The Effect of the Ground on a Helicopter Rotor in Forward Flight"*, Crown, 1957

[19] M. La Civita, W. Messner, & T. Kanade, *"Modelling of Small-Scale Helicopters with Integrated First-Principles and Integrated System Identification Techniques"*, American Helicopter Society, 2002

[20] R.F. Stengel, *"Optimal Control and Estimation"*, Dover Publications, 1994

[21] A.E. Bryson & Y. Ho, *"Applied Optimal Control : Optimization, Estimation, and Control"*, Waltham, 1969

[22] G.F. Franklin, J.D. Powell & M.L. Workman, *"Digital Control of Dynamic Systems"*, Pearson Education, 2002

[23] J.H. Blakelock, *"Automatic control of aircraft and missiles"*, Wiley, 1991

[24] C.T. Kelley, *"Solving Nonlinear Equations with Newton's Method"*, SAIM, 2003