



UNIVERSITEIT • STELLENBOSCH • UNIVERSITY
jou kennisvenoot • your knowledge partner

Autonomous Flight of an Unmanned Helicopter

by

Louis-Emile Rossouw



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

Master of Science in Engineering

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

Study leader: Prof. Garth Milne

March 2008

Copyright © 2008 University of Stellenbosch
All rights reserved.

Declaration

I, the undersigned, hereby declare that the work contained in this thesis is my own original work and that I have not previously in its entirety or in part submitted it at any university for a degree.

Date:29/02/2009.....

Abstract

The successful development of an autopilot for an unmanned helicopter is presented. The helicopter model is developed from first principles. The model is decoupled to enable the design of four separate controllers. Modal Analysis is done on the helicopter's natural modes of motion. The controllers are designed using a classical successive-loop-closure approach.

The ground station software developed is presented. A guidance controller is developed to enable autonomous waypoint navigation. The practical results obtained during Autonomous flight are shown.

Uittreksel

Die suksesvolle ontwerp van 'n vliegstuuroutomaat vir 'n miniatuur helikopter word voorgedra. Die helikopter model word ontwikkel uit eerste beginsels. Die model word ontkoppel om die ontwerp van vier afsonderlike beheerstelsels toe te laat. Modale Analise word toegepas op die helikopter se natuurlike modusse van beweging. Die beheerstelsels word ontwerp deur van klassieke agtereenvolgende-lus-sluiting gebruik te maak.

Die grondstasie sagteware word voorgedra. 'n Stuurstelsel word ontwikkel sodat outonome roete analise gedoen kan word. Die praktiese resultate wat verkry is gedurende Outonome vlug word voorgedra.

Acknowledgments

This project was the product of years of research in rotary-wing applications at Stellenbosch University. My deepest gratitude goes out to everyone who has contributed to this project. In particular,

- Nicol Carstens, for taking the first few steps into the unknown and laying the foundation for Autonomous unmanned helicopter flight at Stellenbosch University.
- Stephanus Groenewald, for your contribution to the initial development of the X-Cell helicopter platform.
- Mike Davis, for your invaluable advice and hours spent at the flying field.
- Mr. Iain Peddle, for your crucial inputs, friendship and support.
- Prof. Thomas Jones, for your guidance during the writing of this thesis.
- All my friends in the ESL, especially Carlo van Schalkwyk, Steven Kriel and John Wilson. Thanks for blurring the line between doing some serious research and having a great amount of fun.
- All the supporting staff members at the ESL, for allowing me to focus on this project while taking care of the rest.
- My parents, for the continuous supply of rusks and their support throughout this project.
- My girlfriend, Adri Brand, for all your love and support.
- My study leader, the late Prof. Milne, who passed away midway through this project. I will remember him as an excellent academic who always

challenged accepted norms. He was young at heart and for him no challenge was too great. It is my hope that this project would have made him proud.

In memory of Garth Milne

Contents

Declaration	ii
Abstract	iii
Uittreksel	iv
Acknowledgments	v
Contents	viii
Nomenclature	x
List of Figures	xiii
List of Tables	xv
1 Introduction and Overview	1
1.1 Project Background	1
1.2 Project Goals	4
1.3 Achievements	4
1.4 Thesis Outline	5
2 Dynamics of RC Helicopters	7
2.1 Axis System Definition	7
2.2 Development of the Equations of Motion	9
2.3 Attitude Definition	11
2.4 Development of a Helicopter Hover Model	14
2.5 Linearisation of the X-Cell Model	25
2.6 Linear, Decoupled Models	27
2.7 Summary	28

<i>CONTENTS</i>	ix
3 Modal Decomposition	30
3.1 Scaling of Units	30
3.2 Longitudinal Helicopter Model	32
3.3 Lateral Helicopter Model	36
3.4 Heave Helicopter Model	39
3.5 Heading Helicopter Model	40
3.6 Summary	41
4 Control Law Design and Non-linear Simulation	43
4.1 Control Law Design	43
4.2 Non-linear Simulation	55
4.3 Non-linear Controller Simulation	58
4.4 Summary	61
5 Waypoint Navigation and Ground Station	62
5.1 Guidance Controller	62
5.2 Ground Station Software	67
5.3 Summary	72
6 Practical Results	74
6.1 Flight tests	75
6.2 Summary	78
7 Conclusions and Recommendations	79
7.1 Conclusions	79
7.2 Recommendations	80
Appendices	83
A System Architecture	84
A.1 PC104 Stack	85
A.2 IMU CAN Node	89
A.3 Servo Controller CAN Node	90
Bibliography	95

Nomenclature

Greek Letters:

ϕ	Euler roll angle
θ	Euler pitch angle
ψ	Euler yaw angle
ρ	Air density
σ	Solidity ratio
τ_f	Effective rotor time constant
δ_a	Lateral flapping command
δ_b	Longitudinal flapping command
δ_c	Collective blade pitch command
δ_r	Heading command

Small Letters:

a	Acceleration
f_s	Sampling rate frequency
g	Gravitational acceleration
h	Height
k	Controller gain
m	Mass
x, y, z	Body Frame positions
u, v, w	Velocity components in the x,y and z-directions of the body frame
p, q, r	Roll-, pitch-, and yaw rate

NOMENCLATURE

xi

a_1, b_1 First harmonics of the blade flapping angles

Capital Letters:

K Total Controller loop gain from input measurement to output command

T_s Sample Time

J_x, J_y, J_z Moments of inertia

X, Y, Z Forces in the body reference frame

L, M, N Components of moment about the CG in the body reference frame

Ox_B, Oy_B, Oz_B Components of the body reference frame

Ox_E, Oy_E, Oz_E Components of the earth reference frame

Subscripts:

B Body axis system

BH Bell-Hiller Stabilizer

D Down indicator

E Earth axis system

F Fuselage

G Gravity

GY Onboard rate gyro

H Horizontal Stabilizer

M Main Rotor

T Tail Rotor

V Vertical Stabilizer

Acronyms:

6DoF Six degrees of freedom

A/D Analog to digital

ADCs Analogue to Digital Converters

AMR Anisotropic magneto resistive

CAN Controller Area Network

NOMENCLATURE

xii

CG	Center of Gravity
DC	Direct Current
EKF	Extended Kalman Filter
ESL	Electronic Systems Laboratory
FCS	Flight Control System
GPS	Global Positioning System
GUI	Graphical User Interface
HIL	Hardware in the Loop
HILS	Hardware in the Loop Simulator
IMU	Inertial Measurement Unit
LAN	Local Area Network
LQR	Linear Quadratic Regulator
MIT	Massachusetts Institute of Technology
OBC	Onboard Computer
PC104	Embedded Computer Standard
PC	Personal Computer
PCB	Printed Circuit Board
RAM	Random Access Memory
RC	Remote Control
RUAV	Rotary-wing Unmanned Aerial Vehicle
SHI	Simulation Hardware Interface
TPP	Tip path plane
UAV	Unmanned Aerial Vehicle
UCA	User Control Application
USB	Universal Serial Bus
V	Volt

List of Figures

2.1	Aircraft Earth Axis System Definition [12]	8
2.2	Aircraft Body Axis System Definition [9]	9
2.3	Euler 3-2-1 Rotations [5]	12
2.4	Free body diagram of helicopter in the body coordinate system [15].	16
3.1	Simplified Decoupled Longitudinal Helicopter Model	33
3.2	Longitudinal Phugoid Mode Response	35
3.3	Pitch Mode Response	36
3.4	Simplified Decoupled Lateral Helicopter Model	37
3.5	Lateral Phugoid Mode Response	38
3.6	Roll Mode Response	40
3.7	Simplified Decoupled Heave Helicopter Model	40
3.8	Heave Mode Response	41
3.9	Simplified Decoupled Heading Helicopter Model	41
3.10	Heading Mode Response	42
4.1	Successive-loop-closure for the Heading plant.	45
4.2	Root locus of heading plant.	47
4.3	Heading angle step response.	47
4.4	Root locus of the vertical position plant - no velocity feedback. . .	48
4.5	Root locus of heave plant - velocity feedback added.	49
4.6	Heave position step response.	50
4.7	Successive-loop-closure for the Heave plant.	50
4.8	Successive-loop-closure for the Lateral plant.	53
4.9	Successive-loop-closure for the Longitudinal plant.	53
4.10	Root locus of longitudinal plant.	54
4.11	Longitudinal position step response.	54

<i>LIST OF FIGURES</i>	xiv
4.12 HIL Simulator Structure in Simulink	57
4.13 Autopilot model for the HIL Simulator	58
4.14 Graphical Simulator Screenshot - Helicopter on the Runway . . .	59
4.15 Response to a heading step command of 30 degrees	60
4.16 Response to a heave step command of 5 meters	60
4.17 Response to a longitudinal step command of 5 meters	61
4.18 Response to a lateral step command of 5 meters	61
5.1 Flow chart of the Guidance Controller	64
5.2 Screenshot of the Ground Station Tab	68
5.3 Screenshot of the Controller Gains Tab	70
5.4 Screenshot of the Command Post Tab	71
6.1 Practical Step response of the Heading Controller with ± 30 de- gree steps	76
6.2 Practical Step response of the Heave Controller with ± 1 meter steps	76
6.3 Practical Step response of the Longitudinal Controller with a -1 meter step	77
6.4 2D view of the Runway Coordinates during a Fully Autonomous Hover	78
A.1 System Architecture	84
A.2 PC104 Stack	86
A.3 OBC and PC104/CAN Controller Timing diagram [5]	87
A.4 UCA Flow Chart [5]	92
A.5 Avionics system - PC104 Stack bottom right, IMU top left	93
A.6 Servo Controller board [5]	93
A.7 X-Cell during an Autonomous hover	94
A.8 Flight test team from the left: Kyle Davis (safety pilot), Carlo van Schalkwyk and the author.	94

List of Tables

2.1	Eigenvalues of the identified X-Cell Helicopter Hover Model . . .	27
3.1	Approximate Maximum deviations around hover trim	31
5.1	Variables used in Figure 5.1 listed in alphabetical order	66
A.1	Hardware components developed in the ESL	85

Chapter 1

Introduction and Overview

1.1 Project Background

Research on unmanned helicopters in the Electronic Systems Laboratory (ESL) was initiated back in April 1996 with the HOPTUS (HelicOpter PlaTform of the University of Stellenbosch) project. The goal of HOPTUS was to design an electrically powered helicopter platform capable of lifting a payload for extended periods of time [14]. The study was conducted by Mooren [26] under the guidance of Prof. Garth Milne. After a failed coaxial rotor configuration was attempted, Mooren recommended that a four rotor option be investigated. In October 1998 Venter [27] built another coaxial rotor configuration by using two counter rotating electric hand drills. The configuration had no actuators to tilt the tip path plane (TPP)¹ and thus could not be maneuvered.

In January 2003 Nicol Carstens [14] commenced his Masters Degree under the guidance of Prof. Garth Milne with the initial goal of building a helicopter platform, instrumenting the platform with sensors and designing a flight control system (FCS) to automate the aircraft. The biggest stumbling block in reaching his goals was to build the helicopter platform. Initially he tried to improve on the coaxial configuration built by Venter [27] without success. He then built a quad rotor configuration² as suggested by Mooren [26], again without success. After funding became available he was able to

¹See Section 2.4.4.1 for an explanation of the TPP.

²Similar to the Canadian manufactured [28] Dragan Flyer XP-Pro.

purchase an electrically powered radio controlled (RC) helicopter. The reason the platform had been restricted to an electrically powered aircraft was so that it could be utilised for indoor applications. The *JR Voyager E* had the following limitations:

- It was a small helicopter with a rotor diameter of less than a meter and could thus only lift a maximum payload of half a kilogram. This limited lifting capability prevented Carstens from doing processing onboard the helicopter. Instead, processing was done on the ground station computer making the platform dependent on the telemetry link.
- Its small size made it very sensitive to wind disturbances as well as the weight distribution of the payload.
- The platform did not have a governor onboard to regulate the blade speed of the helicopter. Instead, the blade speed changed constantly as the battery voltage dropped. This made it difficult to model the helicopter since the dynamics changed with blade speed.
- To the knowledge of the author, no other research group had published work on the automation of a small-size electrically powered RC helicopter so that it could be used as a guideline.

Even with all these limitations Carstens managed to demonstrate various degrees of freedom under autonomous control. Mechanical failures and time limitations hindered him from ever achieving full autonomous flight. The contributions made by Carstens to the research area of rotary-wing flight control at Stellenbosch University are significant, and can be summarized as follows:

- A comprehensive literature study was compiled.
- A variety of low-cost sensors were evaluated and tested. The sensors proved to work practically for a project of autonomous flight. Filters were evaluated in order to improve sensor measurement quality.
- The controllers were designed using a classical successive-loop-closure approach. The design process was well-documented for use in future autonomous unmanned helicopter projects.

In January 2004 Stephanus Groenewald [9] commenced his Masters Degree under the leadership of Dr. Thomas Jones. Starting with the recommendations made by Carstens, Groenewald set out with the project goal to develop a low-cost avionics system for a more powerful RC helicopter with a greater lifting capability namely the *Miniature Aircraft X-Cell*. The X-Cell helicopter purchased was a methanol powered aircraft with a 70 size *glow*³ engine capable of lifting a maximum payload of 2kg⁴. The X-Cell helicopter platform had the following advantages:

- The increased payload capability allowed avionics to be developed with onboard processing capability in the form of a PC104+ Pentium III compact computer.
- Numerous research groups have published work on methanol powered RC helicopter dynamics with the goal of automating the aircraft. In particular, Gavrilets *et al.* [17] published a comprehensive paper discussing the modeling of the X-Cell helicopter used in their research. The model they obtained has been proved during numerous autonomous flights.
- Methanol powered aircraft are more reliable than electrically powered ones. During the two year duration of Carsten's project he had three engine failures compared to none for the X-Cell.
- The X-Cell had a governor onboard to ensure a constant blade speed.

Groenewald spent a great deal of time contributing to the current avionics system by developing the *CANSens generic sensor node*. *CANSens* is a device with 12 A/D channels used to digitize analogue measurement signals. The significant contributions made by Groenewald to the area of rotary-wing flight control follow:

- Evaluated a number of low-cost sensors to be used for autonomous flight and instrumented them to the helicopter platform.
- Developed the *CANSens generic sensor node* and integrated the avionics with the X-Cell helicopter.

³Glow fuel consists of Methanol (CH_3OH) as base and is usually mixed with Castor/Synthetic Oil and Nitro Methane (CH_3NO_2) [9].

⁴With a set of 680mm rotor diameter blades.

In January 2006 Carlo van Schalkwyk [16] and the author commenced their Masters Degrees under the guidance of Prof. Garth Milne. As the aim was to speed up the progress toward autonomous flight, two students were selected to work on the platform. Even though different project outcomes were stipulated, work on the completion of the platform was shared among them. Most of the platform development was of a mechanical or non-academic type and the process proved to be very time-consuming. Platform development is discussed further in Chapter 6.

1.2 Project Goals

The main goal of this project was to automate the X-Cell RC helicopter for low advance ratio⁵ flight by making use of a classical successive-loop-closure approach. Successive-loop-closure was specified for the following reasons:

- It is a good intuitive approach to controller design.
- It works well with low-cost sensors.
- The method has good robustness properties.
- By closing the loops successively, starting with the inner attitude loops, the controllers can easily be tuned and tested during practical flight tests to verify their correct working.
- Van Schalkwyk was tasked with investigating full-state feedback strategies such as the Linear Quadratic Regulator (LQR) in order to push the envelope of autonomous flight.

1.3 Achievements

Autonomous flight of the X-Cell helicopter was demonstrated successfully. The *secondary project achievements* follow:

- The existing helicopter platform was completed.

⁵The *advance ratio* is the ratio of forward flight speed to the speed of the rotor tip of the helicopter. For the X-Cell flying at 20 m/s at its nominal blade speed of 167.5 rad/sec and main rotor diameter of 0.775 m the advance ratio follows: $\mu = \frac{20}{167.5 \times 0.775} \approx 0.15$

- A non-linear helicopter model was obtained and linearised around hover trim conditions by making use of MATLAB's *linearize* function.
- The linear hover-model was decoupled so that four separate successive-loop-closure controllers could be designed.
- Modal Analysis was performed on the natural modes of motion of the helicopter to enable more effective controller design.
- The designed controllers were simulated in Hardware-in-the-loop⁶ (HIL). The simulations were done in MATLAB Simulink by making use of the full Non-linear helicopter model.
- A guidance controller was designed to enable the helicopter to fly a given set of waypoints autonomously in compliance with a pre-programmed set of rules.
- The ground station software used to manage the autopilot was extended in order to meet the project requirements.
- Autonomous flight test results were presented. These results verified the controllers as well as the non-linear models used to design them.

1.4 Thesis Outline

In **Chapter 2** the basic definitions used throughout this project are given. A general non-linear RC helicopter model is developed. The model is linearised and decoupled.

In order to control specific modes of motion most effectively it is necessary to feed back the most dominant helicopter states that influence those modes. The natural modes of motion of the helicopter are analysed in **Chapter 3** through Modal Analysis.

The design of the control laws are discussed in **Chapter 4**. A brief overview of the Hardware-in-the-loop-simulator (HILS) is given. Non-linear HIL simulation results are shown.

⁶See Section 4.2.1 for a description of HIL.

CHAPTER 1. INTRODUCTION AND OVERVIEW

6

The guidance controller is discussed in **Chapter 5**. A flow diagram of the guidance controller shows the logic behind autonomous waypoint navigation. A brief description of the project specific ground station software is given.

The practical results obtained during autonomous flight tests are shown in **Chapter 6**.

Chapter 2

Dynamics of RC Helicopters

2.1 Axis System Definition

2.1.1 Earth Reference Frame

An earth reference frame can be defined as a frame in which the law of inertia holds. Simply stated, an object will not change its velocity spontaneously unless an external resultant force is applied to it. The position of the aircraft in relation to the earth reference frame can be accurately calculated by applying Newton's laws of translational and rotational motion. As the angular velocities induced on the airframe will be much greater than the earth's angular rotation it can be assumed that the earth is non-rotating for calculations in the earth frame.

The origin of the reference frame is defined to be at the starting position of the aircraft on the runway. The reference frame is said to be earth-fixed. The advantage of an earth fixed reference frame with its origin at the starting position of the aircraft is that the position offsets in each axis now have added significance: It is an indication of the distance the aircraft has moved from its original starting position on the surface of the earth. These position offsets lose their significance the greater they become since the earth axis does not factor in the curvature of the earth's surface. For example, say the aircraft lifts off one meter from the earth's surface and travels a great distance forward parallel to the reference frame. Relative to the frame the height position offset will indicate that the aircraft is still one meter above the earth's surface when in fact the altitude of the aircraft is much higher due to the cur-

vature of the earth. That being said and considering the relatively small intended distance that the aircraft would travel away from the origin the effect of the earth's curvature can be ignored when interpreting the position offsets.

The axis components are defined as follows: The OX_E -axis points to the North pole and is tangential to earth's surface. The OY_E -axis points to the East, also tangential to the earth's surface. The OZ_E -axis is perpendicular to both the OX_E and OY_E axes and points down toward the center of the earth. The position offset components are defined as NED. (North, East and Down displacement)

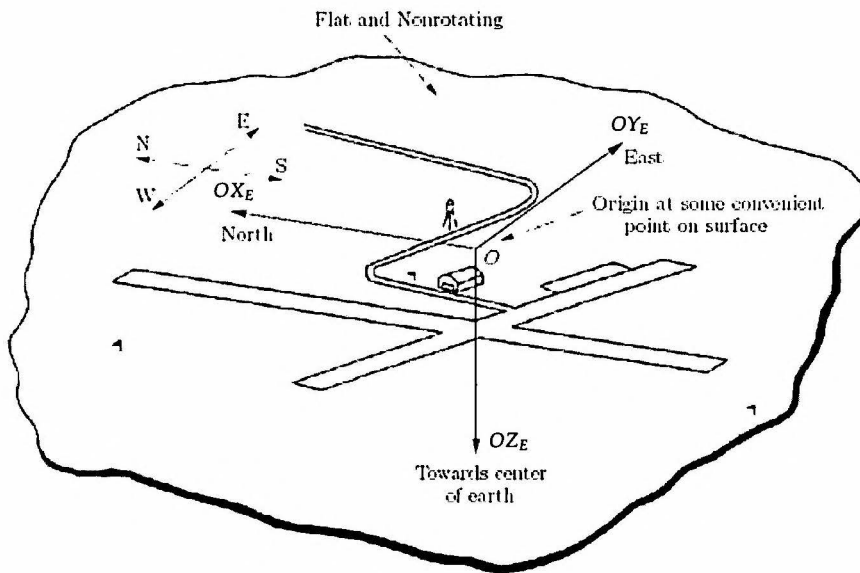


Figure 2.1: Aircraft Earth Axis System Definition [12]

2.1.2 Body Reference Frame

A body reference frame is a right handed reference frame that can be defined as having its origin at the center of gravity of the aircraft. Since this frame is fixed to the aircraft it translates and rotates along with it. The aircraft's spatial behavior can be described with three translatory components and three

rotary components[8]. The Body Reference Frame is important since all inertial measurements are made in this axis system.

The axis components are defined as follows: The OX_B -axis (longitudinal) points forward towards the nose of the aircraft. The OY_B -axis (lateral) is perpendicular to the OX_B -axis and points to the right of the aircraft. The OZ_B -axis (vertical) is perpendicular to the $X_B Y_B$ -plane and points downward. The aircraft is said to *roll* about the OX_B -axis, *pitch* about the OY_B -axis and *yaw* around the OZ_B -axis.

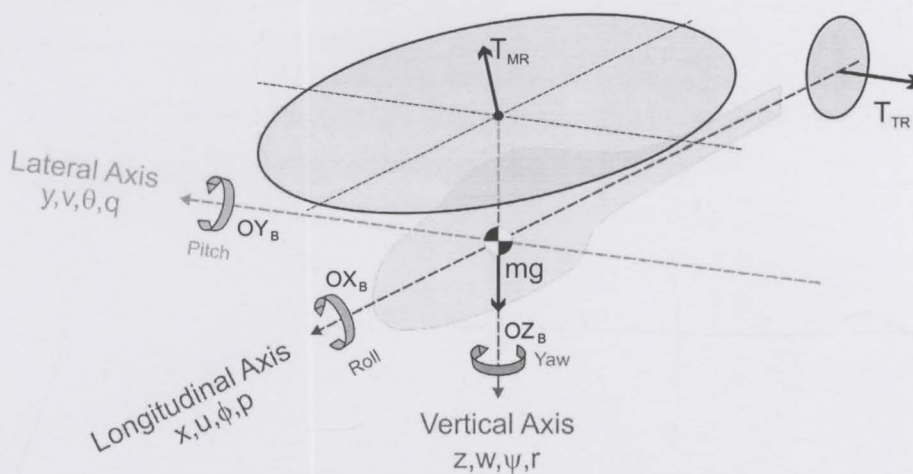


Figure 2.2: Aircraft Body Axis System Definition [9]

2.2 Development of the Equations of Motion

Now that the axis systems and the notations used to describe them have been defined the equations of motion can be defined by making use of Newton's second law of motion. The linear and angular velocity vectors as well as the

force and moment vectors follow,

$$\mathbf{v}_B = u\mathbf{i}_B + v\mathbf{j}_B + w\mathbf{k}_B \quad (2.2.1)$$

$$\boldsymbol{\omega}_B = p\mathbf{i}_B + q\mathbf{j}_B + r\mathbf{k}_B \quad (2.2.2)$$

$$\mathbf{F}_B = X\mathbf{i}_B + Y\mathbf{j}_B + Z\mathbf{k}_B \quad (2.2.3)$$

$$\mathbf{M}_B = L\mathbf{i}_B + M\mathbf{j}_B + N\mathbf{k}_B \quad (2.2.4)$$

In Equations (2.2.1) to (2.2.4), \mathbf{i}_B , \mathbf{j}_B and \mathbf{k}_B are unit vectors defined along the OX_B , OY_B and OZ_B axes respectively.

The generalized equations of motion in body coordinates for an aircraft with six degrees of freedom follow [3]:

$$X = m[\dot{u} - vr + wq - x(q^2 + r^2) + y(pq - \dot{r}) + z(pr + \dot{q})]$$

$$Y = m[\dot{v} - wp + ur + x(pq + \dot{r}) - y(p^2 + r^2) + z(qr - \dot{p})]$$

$$Z = m[\dot{w} - uq + vp + x(pr - \dot{q}) + y(qr + \dot{p}) - z(p^2 + q^2)]$$

$$L + Yz - Zy = \dot{p}J_x + qr(J_z - J_y) + J_{xy}(pr - \dot{q}) - J_{xz}(pq + \dot{r}) + J_{yz}(r^2 - q^2)$$

$$M + Zx - Xz = \dot{q}J_y + pr(J_x - J_z) + J_{yz}(pq - \dot{r}) + J_{xz}(p^2 - r^2) - J_{xy}(qr + \dot{p})$$

$$N + Xy - Yx = \dot{r}J_z + pq(J_y - J_x) - J_{yz}(pr + \dot{q}) + J_{xz}(qr - \dot{p}) + J_{xy}(q^2 + p^2)$$

where

$[x, y, z]^T = \text{distance between the body center and body axis center}$

$[X, Y, Z]^T = \text{applied forces}$

$[L, M, N]^T = \text{applied moments}$

$$\begin{bmatrix} J_x & J_{xy} & J_{xz} \\ J_{yx} & J_y & J_{yz} \\ J_{zx} & J_{zy} & J_z \end{bmatrix} = \text{body inertia tensor}$$

If it is assumed that the body measurements are taken at the center of gravity and that the origin of the body axis is defined here then

$$x = y = z = 0$$

Furthermore, if it is assumed that the body of the aircraft is symmetric about the OX_BOZ_B plane and that the mass is uniformly distributed then

$$J_{xy} = J_{yx} = J_{yz} = J_{zy} = 0$$

Finally, if it is assumed that the aircraft body axes are aligned to the principle inertia axes¹ then

$$J_{xz} = J_{zx} = 0$$

Shown below are the simplified non-linear, coupled differential equations describing the aircraft's motion in six degrees of freedom [5],

$$X = m(\dot{u} - vr + wq) \quad (2.2.5)$$

$$Y = m(\dot{v} - wp + ur) \quad (2.2.6)$$

$$Z = m(\dot{w} - uq + vp) \quad (2.2.7)$$

$$L = \dot{p}J_x + qr(J_z - J_y) \quad (2.2.8)$$

$$M = \dot{q}J_y + pr(J_x - J_z) \quad (2.2.9)$$

$$N = \dot{r}J_z + pq(J_y - J_x) \quad (2.2.10)$$

where m is the mass of the aircraft and J_x , J_y and J_z are the moments of inertia about the OX_B , OY_B and OZ_B axes respectively. The derivation of these equations from first principles can be found in [7].

2.3 Attitude Definition

As described in Section 2.1.1 it is essential to know the aircraft's position relative to the earth frame in order to guide the aircraft along a specific course. Since the body frame has been defined to coincide with the aircraft it is important to know how the body frame is orientated relative to the earth frame so that the aircraft's position in the earth frame can be determined. Making use of Euler angles is the most common technique to describe the aircraft's attitude. Euler angles represent the spatial orientation of any frame relative to a reference frame as a composition of rotations. The order of these rotations can differ and will be discussed below.

¹"This simplification is not often used owing to the difficulty of precisely determining the principle inertia axes. However, the symmetry of the aeroplane determines that J_{xz} is generally very much smaller than J_x , J_y and J_z and can often be neglected[3]"

2.3.1 Euler Angles

Euler angles consist of a set of three angles and a specific order of rotation. This project made use of the Euler 3-2-1² order of rotation.

In Figure 2.3, assume the earth and body axes are initially aligned. All rotations are clockwise. Rotate the body reference frame about the OZ_E -axis through the yaw angle, ψ . Next, rotate about the OY_1 -axis through the pitch angle, θ . Lastly, rotate about the OX_2 -axis through the roll angle, ϕ . A complete definition of the aircraft's attitude is now achieved by knowing the order of rotation (Euler 3-2-1) and the magnitude of the three Euler angles.

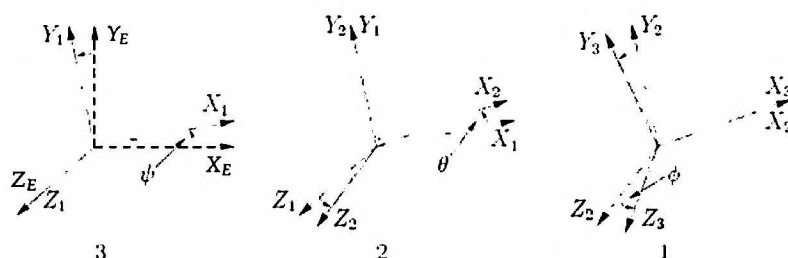


Figure 2.3: Euler 3-2-1 Rotations [5]

2.3.2 Transformation Matrix

The attitude of the aircraft has been defined. The next step is to coordinate vectors in the body axis system into the earth axis system and vice versa. This is achieved by means of the Direction Cosine Matrix (DCM). For a complete derivation of the DCM see [4]. A vector coordinated in the body axes by means of the DCM can be represented as,

$$\mathbf{V}_B = \mathbf{T}(\phi, \theta, \psi) \mathbf{V}_E \quad (2.3.1)$$

²Euler angles are numbered from 1 to 3 representing *roll*, *pitch* and *yaw*. Thus, Euler 3-2-1 executes in the following sequence: First *yaw*, then *pitch* and lastly *roll* the aircraft.

where V_B is the vector in body axes, V_E is the vector in earth axes and T is the DCM. Similarly, by making use of the inverse DCM it follows that

$$V_E = T^T(\phi, \theta, \psi)V_B \quad (2.3.2)$$

The Euler 3-2-1 DCM is defined as,

$$T(\phi, \theta, \psi) = \quad (2.3.3)$$

$$\begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\cos\theta \\ \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\cos\theta \end{bmatrix}$$

2.3.3 Euler Angle Differential Equations

As the aircraft can now be coordinated into the earth axis, the attitude of the aircraft needs to be propagated by utilizing the angular velocity measurements p, q and r ,

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (2.3.4)$$

as given by [13]. As can be seen from Equation (2.3.4) there are singularities at pitch angles of $\pm 90^\circ$ due to the $\tan\theta$ and $\sec\theta$ terms. Since the main objective of this project was near-hover flight the pitch angle singularities were not considered a problem³.

³When working on a project where pitching through $\pm 90^\circ$ is required quaternions could be used. For more information about quaternions see [5]. Another approach could be to switch from one set of euler angle definitions to another when approaching points of singularity.

2.4 Development of a Helicopter Hover Model

The goal of this section is to give the reader a basic understanding of the relevant aircraft specific forces and moments that act on a miniature RC helicopter. A similar approach to that of Shim [15] was followed, but with specific focus on obtaining a model around hover trim conditions⁴.

2.4.1 Helicopter Specific Forces and Moments

Up until this point all the equations that have been developed show the kinematic-dynamic relationship that holds true for any rigid body motion. The helicopter specific forces and moments, represented by F_{heli} and M_{heli} respectively, will now be developed. The Newton-Euler equations for a miniature RC helicopter in translational and rotational motion follow,

$$\dot{V}^B = \frac{1}{m}F_{heli} - \omega^B \times V^B \quad (2.4.1)$$

$$\dot{\omega}^B = \frac{1}{J_B}(M_{heli} - \omega^B \times J_B \omega^B) \quad (2.4.2)$$

where

$$V^B = \begin{bmatrix} u & v & w \end{bmatrix}^T$$

$$\omega^B = \begin{bmatrix} p & q & r \end{bmatrix}^T$$

$$J_B = \begin{bmatrix} J_x & J_{xy} & J_{xz} \\ J_{yx} & J_y & J_{yz} \\ J_{zx} & J_{zy} & J_z \end{bmatrix}$$

The two variables in Equations (2.4.1) and (2.4.2) that make these equations aircraft specific are F_{heli} and M_{heli} , as their subscript implies. They are defined as a set of unique forces and moments that are induced on the rigid body due to the physical and aerodynamic properties of the helicopter. An approach

⁴For a complete Aerodynamic Analysis of the helicopter components that induce forces and moments on the airframe see Shim [15] or Gavrillets *et al.* [17].

was adopted whereby the forces and moments induced due to each significant aircraft component were analysed from first principles. The following components were identified for the helicopter:

1. Main Rotor
2. Tail Rotor
3. Fuselage
4. Horizontal stabilizer
5. Vertical stabilizer
6. Bell-Hiller stabilizer bar
7. Onboard active yaw rate damping system⁵

The Bell-Hiller stabilizer bar and the onboard active yaw rate damping system are considered *complementary* systems to an RC helicopter that add to its stability but do not form part of fundamental helicopter dynamics. In order to keep the initial development of the helicopter model as simple as possible they will only be introduced once the core helicopter dynamics (first five components listed) have been discussed. Figure 2.4 illustrates how the forces and moments work in on the rigid body due to these identified components⁶. Forces in the OX_B , OY_B and OZ_B -axis are denoted by X , Y and Z respectively. Moments around these axes are denoted by L , M and N respectively. Subscripts M , T , F , H , V , G , BH and GY denote the main rotor, tail rotor, fuselage, horizontal stabilizer, vertical stabilizer, gravity, Bell-Hiller stabilizer and rate gyro respectively.

⁵Commonly known as a Gyro in RC hobby circles.

⁶The forces and moments printed in red are the most significant when deriving a model for hover.

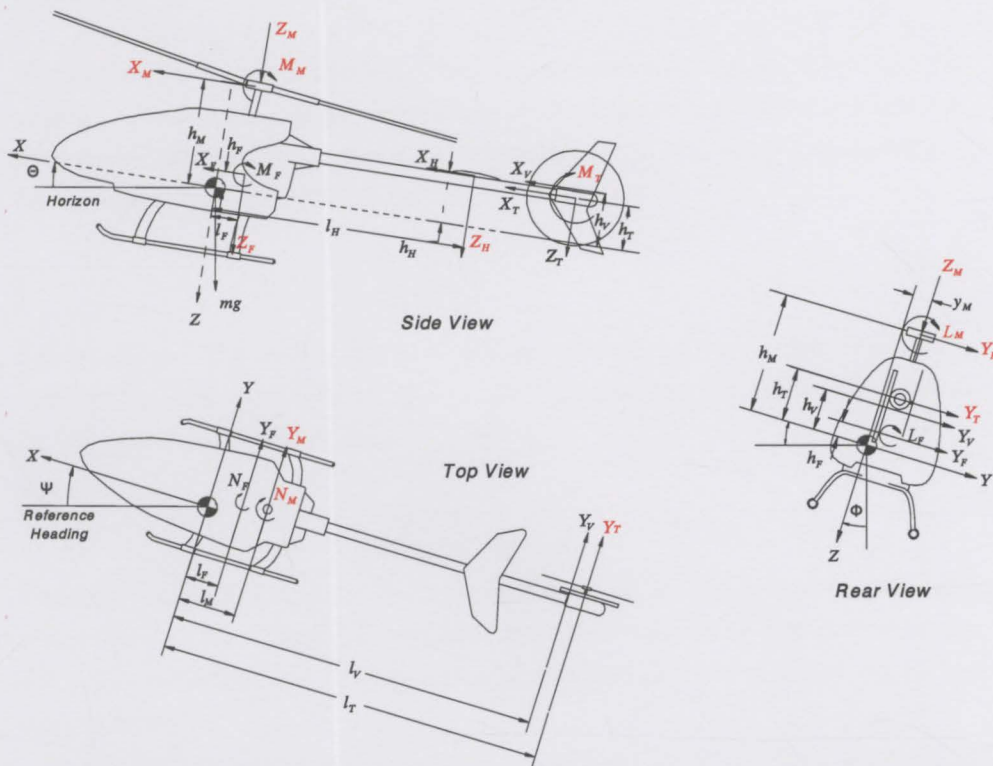


Figure 2.4: Free body diagram of helicopter in the body coordinate system [15].

2.4.2 Simplifications for Hover

The following simplifications are made around hover trim conditions:

Negligible Forces The main and tail rotors will be the dominant forces acting on the rigid body. The constant down-wash being created by the main rotors has a significant aerodynamic effect in the OZ_B -axis on the helicopter fuselage and horizontal stabilizer. The rest of the forces are assumed negligible, thus

$$\begin{aligned}
 (X, Y)_F &= 0 \\
 (X, Y)_H &= 0 \\
 (X, Y, Z)_V &= 0 \\
 (X, Z)_T &= 0
 \end{aligned}
 \tag{2.4.3}$$

Negligible Moments The tail rotor only generates a moment about its own centre about the OY_B or pitch-axis due to the drag of the tail rotor blades as they rotate, known as an anti-torque moment. The anti-torque moment can be assumed negligible [17], thus

$$(L, N, M)_T = 0 \quad (2.4.4)$$

CG position The helicopter's CG is aligned with the main rotor shaft of the helicopter. This was achieved practically by balancing the helicopter around its CG with lead weights, thus

$$(y, l)_m = 0 \quad (2.4.5)$$

Centrifugal Forces and Gyroscopic terms Due to the low velocities and small angular rates during hover, the centrifugal forces and gyroscopic terms are assumed negligible. In Equations (2.4.1) and (2.4.2) the following terms are neglected,

$$\begin{aligned} -w^B \times V^B &\cong 0 \\ \frac{1}{J_B}(-w^B \times J_B w^B) &\cong 0 \end{aligned} \quad (2.4.6)$$

Small angle approximations The following small angle simplifications for roll and pitch angles are made:

$$\begin{aligned} \sin x &\approx x \\ \cos x &\approx 1 \end{aligned} \quad (2.4.7)$$

The DCM simplifies to:

$$\begin{aligned} T(\phi, \theta, \psi)_{small\ angles} &\approx \\ &\begin{bmatrix} \cos\psi & \sin\psi & -\theta \\ \phi\theta\cos\psi - \sin\psi & \phi\theta\sin\psi + \cos\psi & \phi \\ \theta\cos\psi + \phi\sin\psi & \theta\sin\psi - \phi\cos\psi & 1 \end{bmatrix} \end{aligned} \quad (2.4.8)$$

2.4.3 Gravity

The force acting on the aircraft due to gravitational acceleration is considered. The magnitude of this force depends on the mass of the aircraft and is always

aligned with the Z_E -axis pointing vertically downward toward the center of the earth. This force is coordinated into the body axes as follows,

$$\begin{aligned} \begin{bmatrix} X_G \\ Y_G \\ Z_G \end{bmatrix} &= T(\phi, \theta, \psi)_{small\ angles} \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \\ &= m \begin{bmatrix} -g\theta \\ g\phi \\ g \end{bmatrix} \end{aligned} \quad (2.4.9)$$

2.4.4 Complementary Stability Systems

Now that the helicopter's core forces and moments have been analysed the two *complementary* stability systems namely the Bell-Hiller stabilizer bar and the active yaw rate damping system will be investigated.

2.4.4.1 Bell-Hiller stabilizer bar

The Bell-Hiller stabilizer bar is common to all small-scale helicopters [17]. The stabilizer bar is composed of two weighted paddles at its tips. These weighted paddles have two distinct mechanisms for providing stability to the helicopter dynamics, namely gyroscopic and aerodynamic effects. The weights at the tips of the bar provide the gyroscopic effect in that they increase the inertia in the plane of rotation. The Bell-stabilizer bar, utilising this effect, was widely used in the Bell UH-1 H helicopters, hence the name. The stabilizer will tend to remain in the same plane of rotation by resisting external torque [15]. Since the main rotor pitch levers are connected to the stabilizer bar through linkages the gyroscopic effect will act as a source of mechanical feedback for roll and pitch rates. The Hiller-stabilizer makes use of the aerodynamic forces exerted on the helicopter. The paddles have a symmetrical airfoil shape. The main rotor blade pitch is controlled through the teetering motion of the stabilizer bar. The response of the blade is said to be aerodynamically damped. The Bell-Hiller stabilizer bar makes use of both these mechanisms to provide the maximum amount of mechanical damping.

Before stating the differential equations that characterize the Bell-Hiller stabilizer, a brief description of flapping angles, that are used in these equations, will be given. First of all the tip-path plane (TPP) needs to be defined. The TPP is an imaginary plane that runs tangential from the one outside tip of the helicopter blades to the other at any instant in time. The longitudinal and lateral flapping angles, a_1 and b_1 respectively⁷, indicate the angles between this imaginary plane and some arbitrary plane defined as being perpendicular to the main shaft. The longitudinal and lateral flapping angles are measured between these two planes in line with the OX_B and OY_B -axes respectively. a_1 is defined as positive when it flaps up along the positive OX_B -axis (nose) and down along the negative OX_B -axis (tail). b_1 is defined as positive when it flaps up along the negative OY_B -axis (left) and down along the positive OY_B -axis (right). By commanding flapping angles, moments are induced about the CG of the helicopter which causes it to move in any direction. Mettler *et al.* [18] proposed the following first order model describing the flapping of the rotor blades (a_1, b_1) which is dominated by the response of the Bell-Hiller stabilizer:

$$\dot{b}_1 = -\frac{b_1}{\tau_f} - p + \frac{B_{\delta_a}}{\tau_f} \delta_a \quad (2.4.10)$$

$$\dot{a}_1 = -\frac{a_1}{\tau_f} - q + \frac{A_{\delta_b}}{\tau_f} \delta_b \quad (2.4.11)$$

where τ_f is the effective time constant of the servo rotor response to swash plate tilt angles and with the cross-coupling derivatives being ignored⁸. Gavrilets *et al.* [17] calculated τ_f to be approximately 0.12 seconds for their X-Cell. B_{δ_a} and A_{δ_b} are the effective steady-state lateral and longitudinal gains from the cyclic inputs to the main rotor flap angles. These two gains are the same for the X-Cell and have a value of 4.2 as given by Gavrilets *et al.*[17]. δ_a and δ_b are the lateral and longitudinal control inputs respectively as given by the pilot stick or control system commands.

⁷The subscript used when defining the flapping angles indicates that they are the first order harmonic coefficient approximations of the Fourier series that represent the angle between a plane perpendicular to the main shaft and the TPP.

⁸Gavrilets *et al.* [17] argues that the cross-coupling derivatives are approximately an order of magnitude smaller than the direct derivatives during low advance ratio flight and can thus be ignored.

2.4.4.2 Active yaw rate stabilizer

The Gyro can be operated in two modes namely *heading hold* and *normal* mode. In *heading hold* mode the Gyro makes use of an integrator on the yaw rate to compensate for heading drift. In *normal* mode the Gyro simply feeds back the yaw rate. For this project the Gyro was operated in its *normal* mode since it allowed more freedom in designing the heading controller. There were two main reasons for using the Gyro:

1. The safety-pilot was accustomed to flying with the Gyro. The Gyro made the helicopter easier to fly during normal test procedures or in the event of an emergency take-over where the human pilot had to be in full control of the aircraft.
2. No additional yaw rate feedback would be required once the device was configured correctly.

The active yaw rate damping system during its *normal* mode of operation was characterized by measuring the output of the Gyro given certain yaw rate disturbances. The gyro feedback gain was identified as $-1/6.6$. This Gyro's effect was modeled by subtracting $r/6.6$ from the tail rotor collective pitch, δ_r .

2.4.5 Forces and Moments Summary

By substituting Equations (2.4.3) to (2.4.9) into Equations (2.4.1) and (2.4.2) the differential equations simplify as follows:

$$\dot{V}^B \approx \frac{1}{m} \begin{bmatrix} X_M + X_G \\ Y_M + Y_T + Y_G \\ Z_M + Z_H + Z_F + Z_G \end{bmatrix} \quad (2.4.12)$$

$$\dot{w}^B \approx \frac{1}{J_B} \begin{bmatrix} L_M + Y_M h_M + Y_T h_T \\ M_M - X_M h_M + Z_H l_H \\ N_M - Y_T l_T + N_{GY} \end{bmatrix} \quad (2.4.13)$$

2.4.6 Forces and Moments Decomposition

Now that all the relevant forces and moments have been identified the relevant *aerodynamic equations* must be substituted into the force and moment variables of Equations (2.4.12) and (2.4.13). During hover, the following dominant forces and moments were identified:

Main Rotor Forces The Force produced by the main rotor is known as Thrust. For the helicopter to stay in the air this force needs to be vectored upwards against the pull of the gravitational force defined in Equation (2.4.9). The Thrust Force generated due to the main rotor can be represented as follows:

$$\begin{bmatrix} X_M \\ Y_M \\ Z_M \end{bmatrix} = T(b_1, a_1, \psi)_{small\ angles} \begin{bmatrix} 0 \\ 0 \\ -T_M \end{bmatrix} \quad (2.4.14)$$

for small flapping angles⁹. $T(b_1, a_1, \psi)_{small\ angles}$ coordinates the Thrust Force acting in the *TPP axes*¹⁰ into the *Body axes*.

Tail Rotor Forces The tail rotor provides the lateral force required to cancel out the yaw-moment N_M induced by the main rotor. The tail rotor force will prevent the helicopter from spinning around its CG, thus

$$Y_T = -T_T \quad (2.4.15)$$

where T_T is the lateral thrust produced by the tail rotor.

⁹Similar to the approximation made by Padfield [19] during low advance ratio flight, small flapping angles are assumed throughout this project.

¹⁰The Thrust acts perpendicular to the TPP.

Aerodynamic Drag Forces Vertical drag on the fuselage and horizontal stabilizer: The variables Z_F and Z_H can be obtained by means of general aerodynamic equations involving the cross-sectional area of these components and the amount of down-wash present during hover.

Main Rotor Moments The Moments induced about the main rotor hub follow:

$$\begin{bmatrix} L_M \\ M_M \\ N_M \end{bmatrix} = \begin{bmatrix} K_\beta b_1 \\ K_\beta a_1 \\ -Q_e \end{bmatrix} \quad (2.4.16)$$

$$(2.4.17)$$

where Q_e is the main rotor torque¹¹ and K_β is known as rotor stiffness term. Gavrillets *et al.* [17] refers to K_β as *the restraint in the blade attachment to the rotor head*.

Padfield [19] uses the example where the rotor stiffness term is assumed zero. Consider a change in the shaft alignment. The main rotor blades will continue rotating in their current plane of rotation since there is no moment that will try and align the blades perpendicularly with the shaft again. Gavrillets *et al.* [17] identified a torsional stiffness term K_β of $54Nm/rad$ for their X-Cell helicopter.

Rate Gyro Moments Moments induced due to the active yaw rate damping system: The effect of the Gyro can be modeled by subtracting $r/6.6$ from δ_r . The Gyro feedback gain can be modeled by adding the feedback term to the \dot{r} expression through the variable N_{GY} in Equation (2.4.13). The Moments induced due to the active yaw rate damping system follow:

$$\begin{bmatrix} L_{GY} \\ M_{GY} \\ N_{GY} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ J_z(-\frac{r}{6.6}) \end{bmatrix} \quad (2.4.18)$$

where the moment J_z will be canceled out in Equation (2.4.13), leaving the desired term $-r/6.6$ in the \dot{r} equation.

¹¹ $(-Q_e)$ can be interpreted as the anti-torque moment induced due to the drag of the rotating blades through the air.

2.4.7 Non-Linear Hover Model

The helicopter specific and external forces and moments were identified and analysed from first principles. A basic non-linear hover model for low advance ratio flight was developed. The following ten state non-linear model is defined:

$$\dot{x} = F(x, u) \quad (2.4.19)$$

with

$$x = \left[u \ v \ p \ q \ \phi \ \theta \ a_1 \ b_1 \ w \ r \right]^T \quad (2.4.20)$$

$$u = \left[\delta_a \ \delta_b \ \delta_c \ \delta_r \right]^T \quad (2.4.21)$$

where δ_a and δ_b are the lateral and longitudinal cyclic pitch angles and δ_c and δ_r are the main and tail rotor collective pitch angles. These inputs control all the degrees of freedom of the helicopter.

$$\begin{aligned} \dot{x} &= F(x, u) \\ &= \begin{bmatrix} -\frac{1}{m}T_M a_1 - g\theta \\ -\frac{1}{m}(T_M b_1 + T_T) + g\phi \\ \frac{1}{J_x}((K_\beta - T_M h_M)b_1 - T_T h_T) \\ \frac{1}{J_y}((K_\beta + T_M h_M)a_1 - Z_H l_H) \\ p \\ q \\ -\frac{1}{\tau_f}a_1 - q + \frac{A_{\delta_b}}{\tau_f}\delta_b \\ -\frac{1}{\tau_f}b_1 - p + \frac{B_{\delta_a}}{\tau_f}\delta_a \\ \frac{1}{m}(-T_M + Z_H + Z_F) + g \\ \frac{1}{J_z}(-Q_e + T_T l_T - J_z \frac{r}{\delta}) \end{bmatrix} \end{aligned} \quad (2.4.22)$$

2.4.8 Linear Hover Model

Before the model is linearised a quick summary of which parameters are a function of which force and moment terms is given:

- Main Rotor Thrust T_M : u, v, w and δ_c
- Tail Rotor Thrust T_T : v, r and δ_r
- Main Rotor Torque Q_e : δ_c

The Jacobian matrices¹² are found by linearising the system equation $F(x, u)$ as follows,

$$\left[\frac{\partial F_j}{\partial x_i} \right] = \begin{bmatrix} \frac{\partial F_u}{\partial u} & 0 & 0 & 0 & 0 & -g & \frac{\partial F_{a_1}}{\partial a_1} & 0 & 0 & 0 \\ 0 & \frac{\partial F_v}{\partial v} & 0 & 0 & g & 0 & 0 & \frac{\partial F_v}{\partial b_1} & 0 & 0 \\ 0 & \frac{\partial F_p}{\partial v} & 0 & 0 & 0 & 0 & 0 & \frac{\partial F_p}{\partial b_1} & 0 & 0 \\ \frac{\partial F_q}{\partial u} & 0 & 0 & 0 & 0 & 0 & \frac{\partial F_q}{\partial a_1} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & \frac{\partial F_{a_1}}{\partial a_1} & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & \frac{\partial F_{b_1}}{\partial b_1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{\partial F_w}{\partial w} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{\partial F_r}{\partial r} \end{bmatrix}$$

$$\left[\frac{\partial F_j}{\partial u_i} \right] = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{\partial F_u}{\partial \theta_M} & \frac{\partial F_u}{\partial \theta_T} \\ 0 & 0 & \frac{\partial F_p}{\partial \theta_M} & \frac{\partial F_p}{\partial \theta_T} \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \frac{\partial F_{a_1}}{\partial \delta_b} & 0 & 0 \\ \frac{\partial F_{b_1}}{\partial \delta_a} & 0 & 0 & 0 \\ 0 & 0 & \frac{\partial F_w}{\partial \delta_c} & 0 \\ 0 & 0 & 0 & \frac{\partial F_r}{\partial \delta_r} \end{bmatrix} \quad (2.4.23)$$

where the terms with a negligible contribution during hover have been omitted.

¹²Often referred to as the *stability derivatives* in the Aerospace community.

The linearised *Hover Model* is defined as

$$\dot{x} = Ax + Bu \quad (2.4.24)$$

where

$$A = \begin{bmatrix} X_u & 0 & 0 & 0 & 0 & -g & X_{a_1} & 0 & 0 & 0 \\ 0 & Y_v & 0 & 0 & g & 0 & 0 & Y_{b_1} & 0 & 0 \\ 0 & R_v & 0 & 0 & 0 & 0 & 0 & R_{b_1} & 0 & 0 \\ M_u & 0 & 0 & 0 & 0 & 0 & M_{a_1} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & -\frac{1}{\tau_f} & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & -\frac{1}{\tau_f} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & Z_w & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & N_r \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & Y_{\theta_M} & Y_{\theta_T} \\ 0 & 0 & R_{\theta_M} & R_{\theta_M} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \frac{A_{\delta_b}}{\tau_f} & 0 & 0 \\ \frac{B_{\delta_a}}{\tau_f} & 0 & 0 & 0 \\ 0 & 0 & Z_{\delta_c} & 0 \\ 0 & 0 & 0 & N_{\delta_r} \end{bmatrix} \quad (2.4.25)$$

2.5 Linearisation of the X-Cell Model

The full non-linear Equations for an X-Cell helicopter as developed by Gavrillets *et al.* in [17] were used by Medellín-Colombia [22] to create a Simulink model in MATLAB. The model parameters were tailored to this project's helicopter specifications as summarized in [9]. To obtain numeric values for the hover model MATLAB's *linearize* function was used to linearise the non-linear Simulink model around hover trim conditions. The advantage of linearising the full

non-linear model as opposed to the simplified hover model developed in Section 2.4 was that the accuracy of the model was increased. The linearised A and B matrices follow,

$$A = \begin{bmatrix} -0.036 & 0 & 0 & 0 & 0 & -g & -9.55 & 0 & 0 & 0 \\ 0 & -0.13 & 0 & 0 & g & 0 & 0 & 9.55 & 0 & 0 \\ 0 & -0.16 & 0 & 0 & 0 & 0 & 0 & 383.6 & 0 & 0 \\ -0.001 & 0 & 0 & 0 & 0 & 0 & 203.1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.002 & 0 & 0 & -1 & 0 & 0 & -8.39 & 0 & 0 & 0 \\ 0 & 0.002 & -1 & 0 & 0 & 0 & 0 & -8.39 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1.11 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -23.37 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1.28 & -6.77 \\ 0 & 0 & 11.23 & -20.15 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 35.31 & 0 & 0 \\ 35.31 & 0 & 0 & 0 \\ 0 & 0 & -164.5 & 0 \\ 0 & 0 & 0 & 147.3 \end{bmatrix} \quad (2.5.1)$$

Stability derivatives with a very small magnitude were replaced by zero's. The approximations made by Mettler *et al.* [18] for a helicopter performing near-hover flight maneuvers were used as a guideline to which terms may be neglected. The system poles are summarized in Table 2.1.

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 35.31 & 0 & 0 \\ 0 & 0 & 1.28 & -6.77 \\ 0 & 0 & 11.23 & -20.15 \\ 0 & 0 & 0 & 0 \\ 35.31 & 0 & 0 & 0 \\ 0 & 0 & -164.5 & 0 \\ 0 & 0 & 0 & 147.3 \end{bmatrix} \quad \text{cross-coupling terms}$$

Notice the four cross-coupling terms in the B-matrix on the collective and rudder commands (-1.28, 11.23, -6.77, -20.15). These terms arose due to the coupling between the tail and main rotors since they were connected via a gear¹³. The cross-coupling terms were much smaller than terms -164.5 and 147.3 that arose due to changes in the main and tail rotor collective blade pitch angles. By neglecting the four cross-coupling terms the full helicopter model could be decoupled into four separate models for which separate feedback controllers could be designed. Gavrillets *et al.* [23] made the following statement concerning decoupling:

"Based on flight experiments, longitudinal-vertical and lateral-directional dynamics of the X-Cell in low advance ratio flight (up to $\mu = 0.15$) are sufficiently decoupled to design separate feedback controllers"

2.7 Summary

The relevant axis systems used in this project were defined. A simple non-linear helicopter model was developed from first principles by identifying the relevant forces and moments and illustrating how they work in on the airframe. A full non-linear X-Cell Helicopter Model, created in Simulink, was linearised using MATLAB. The linearised model was decoupled so that separate feedback controllers could be designed.

¹³The gear ratio from the tail rotor to the main rotor is 4.66 [18].

Although controllers were designed using the linearised decoupled model, they were simulated using the full non-linear Simulink model where the assumptions made during model development could be tested.

Chapter 3

Modal Decomposition

The goal of this chapter is to present a detailed analysis of the linearised helicopter model that was developed in Chapter 2. Modal decomposition uses an eigenvector transformation matrix to convert the separate system models to their modal forms. The modal form is useful since it highlights the individual modes of the system and shows how the inputs and outputs couple into these modes. The eigenvector transformation matrix also indicates the magnitude and phase contributions of each element of the original state vector to the specific mode of motion. Modes can be controlled more effectively by feeding back the most dominant states that influence them. This project followed a similar approach to that of Peddle [4] in his analysis of the modes of motion of a model RC airplane.

3.1 Scaling of Units

The elements of the state vector were scaled so that one unit of each variable had approximately the same physical effect on the aircraft motion. A *maximum deviation* approach was adopted [2]. The maximum deviations of each motion and control variable from their trim conditions were estimated and compared. The units of each variable were then scaled so that the identified maximum deviations were of a similar magnitude. Scaling of the units resulted in a more fair reflection of the influence of each individual state on a specific mode of motion. The approximate maximum deviation of the ten states and four control inputs are shown in Table 3.1. The values in Table 3.1 should be in the correct order of magnitude to ensure effective scaling. It

Table 3.1: Approximate Maximum deviations around hover trim

Variable	Max	Units
u	5	m/s
q	60	deg/s
θ	8	deg
a_1	6	deg
v	5	m/s
p	60	deg/s
ϕ	8	deg
b_1	6	deg
w	5	m/s
r	40	deg/s
δ_a	3	deg
δ_b	3	deg
δ_c	4	deg
δ_r	3	deg

was shown in Table 3.1 that the *maximum deviations* were all of a similar magnitude except for the rates (p, q) and r which were in the order of ten times greater. Thus, the rates were the only variables that needed to be scaled relative to the other variables. The rate units were changed to tens of degrees per second (10deg/s). A brief description of scaling follows:

$$\begin{aligned}\dot{x}_o &= Ax_o + Bu \\ y &= Cx_o\end{aligned}\tag{3.1.1}$$

where x_o is the original state vector and u is the system input. Let

$$x_o = Tx_s\tag{3.1.2}$$

where x_s is the scaled state vector and T is the scaling matrix. By substituting Equation (3.1.2) into Equation (3.1.1) it follows that

$$\begin{aligned}\dot{x}_s &= A_s x_s + B_s u \\ y &= C_s x_s\end{aligned}\tag{3.1.3}$$

where

$$\begin{aligned} A_s &= T^{-1}AT \\ B_s &= T^{-1}B \\ C_s &= CT \end{aligned} \quad (3.1.4)$$

The scaling matrix for the helicopter model follows:

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10(\frac{\pi}{180}) & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & (\frac{\pi}{180}) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & (\frac{\pi}{180}) & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10(\frac{\pi}{180}) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & (\frac{\pi}{180}) & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & (\frac{\pi}{180}) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10(\frac{\pi}{180}) \end{bmatrix}$$

with state order

$$x = [u \ q \ \theta \ a1 \ v \ p \ \phi \ b1 \ w \ r]^T$$

The model was converted from radians to degrees, hence the $(\frac{\pi}{180})$ terms in the scaling matrix.

3.2 Longitudinal Helicopter Model

The eigenvector transformation matrix that was used to convert the longitudinal plant to its modal form is shown below¹.

$$\begin{bmatrix} u \\ q \\ \theta \\ a1 \end{bmatrix} = \begin{bmatrix} -0.0012 & -0.0034 & 1.0000 & 0.0000 \\ 1.0000 & 0.0000 & 0.0115 & -0.0001 \\ -0.2065 & -0.4760 & -0.1075 & -0.6284 \\ -0.2065 & 0.4760 & 0.0003 & 0.0001 \end{bmatrix} \begin{bmatrix} z1 \\ z2 \\ z3 \\ z4 \end{bmatrix} \quad (3.2.1)$$

¹For a complete derivation of the eigenvector transformation matrix see Bryson [2] or Peddle [4].

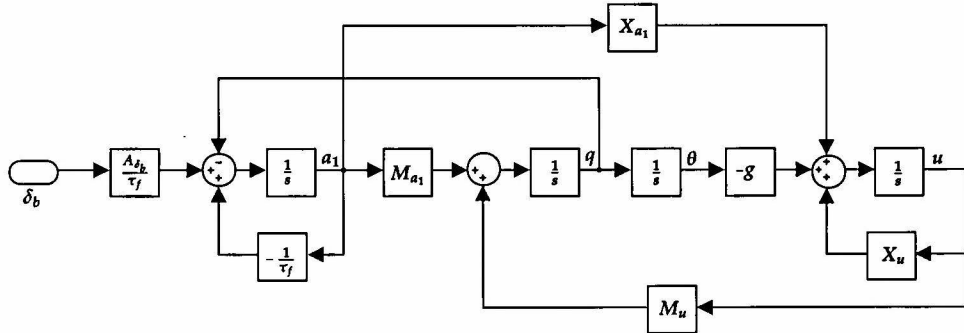


Figure 3.1: Simplified Decoupled Longitudinal Helicopter Model

where $z_{(1 \rightarrow 4)}$ is the modal state vector. The scaled longitudinal helicopter model is written in modal form as

$$\dot{z}_{long} = F_{long} z_{long} + G_{long} u_{long} \quad (3.2.2)$$

where

$$F_{long} = \begin{bmatrix} -4.2 & 13.62 & 0 & 0 \\ -13.62 & -4.2 & 0 & 0 \\ 0 & 0 & -0.018 & 0.14 \\ 0 & 0 & -0.14 & -0.018 \end{bmatrix} \quad (3.2.3)$$

$$G_{long} = \begin{bmatrix} -0.3 \\ 4249.7 \\ 14.3 \\ -3221.7 \end{bmatrix} \quad (3.2.4)$$

Two sets of complex poles are visible in F_{long} . The higher frequency pole set is known as the Pitch Mode of motion while the lower frequency set is known as the Phugoid mode of motion. These modes of motion will be analysed with specific focus on the influence of the respective states on each mode of motion.

3.2.1 Phugoid mode

The Longitudinal Phugoid mode poles are summarized below,

$$\lambda = -0.018 \pm j0.14 \quad (3.2.5)$$

$$\zeta = 0.13 \quad (3.2.6)$$

$$w_n = 0.14 \quad (3.2.7)$$

The Phugoid mode eigenvector is displayed in phasor form below,

$$\begin{bmatrix} u \\ q \\ \theta \\ a_1 \end{bmatrix} = \begin{bmatrix} 1.0000 \angle 0.00 \\ 0.0115 \angle -0.27 \\ 0.6375 \angle 80.30 \\ 0.0003 \angle 13.45 \end{bmatrix} \quad (3.2.8)$$

where the phasor angles are shown in degrees. The phasor form of the eigenvector is an indication of the individual state contributions on the mode of motion being investigated. The Phugoid mode can be described as a sinusoidal exchange of the helicopter's kinetic and potential energy as it descends and climbs.

Figure 3.2 shows a time history response of the Phugoid mode of motion. This was obtained by initialising the scaled system state vector (x_s) with the real parts of the Phugoid mode eigenvector [2]. The following observations are made from the plot:

- The Phugoid mode of motion is well observed in both the forward velocity u and the pitch angle θ .
- The pitch angle θ leads the forward velocity u by approximately 80° . When the pitch angle θ reaches its maximum/minimum the forward velocity u is always close to zero. Alternatively, the forward speed u is at a maximum/minimum when the pitch angle θ is close to zero. This system can be seen as an exchange of kinetic and potential energy. At maximum/minimum pitch angles the *potential energy* is greatest while at zero pitch angles the *kinetic energy* is greatest. This exchange of energy will continue in a sinusoidal fashion with an exponential decay.

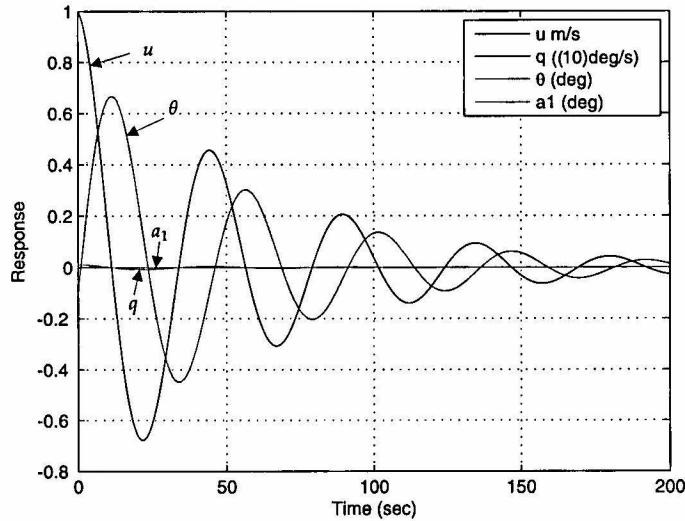


Figure 3.2: Longitudinal Phugoid Mode Response

3.2.2 Pitch mode

The Pitch mode poles are summarized below,

$$\lambda = -4.2 \pm j13.62 \quad (3.2.9)$$

$$\zeta = 0.29 \quad (3.2.10)$$

$$w_n = 14.25 \quad (3.2.11)$$

The Pitch mode eigenvector is displayed in phasor form below,

$$\begin{bmatrix} u \\ q \\ \theta \\ a_1 \end{bmatrix} = \begin{bmatrix} 0.0036 \angle 69.97 \\ 1.0000 \angle 0.00 \\ 0.5189 \angle 66.55 \\ 0.5189 \angle -66.55 \end{bmatrix} \quad (3.2.12)$$

Figure 3.3 shows a time history response of the Pitch mode of motion. The following observations are made from the plot:

- The Pitch mode of motion is well observed in the pitch rate q , pitch angle θ and flapping angle a_1 .
- The Flapping angle a_1 has a significant effect on the Pitch mode of mo-

tion.

- The forward speed u has an insignificant effect on the Pitch mode of motion.

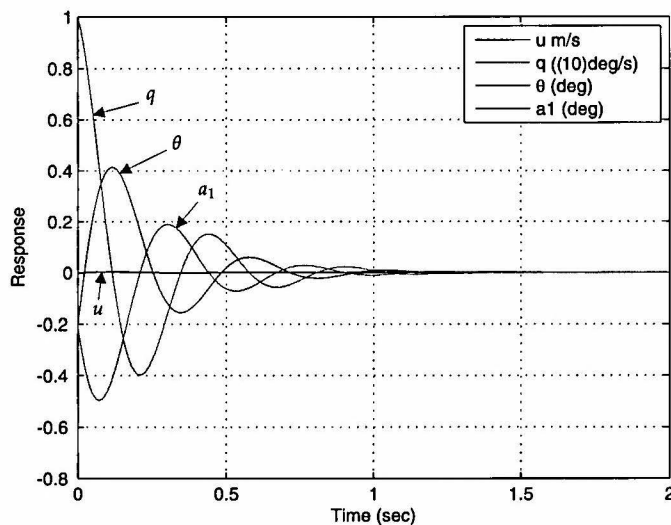


Figure 3.3: Pitch Mode Response

3.3 Lateral Helicopter Model

The eigenvector transformation matrix is shown below,

$$\begin{bmatrix} v \\ p \\ \phi \\ b_1 \end{bmatrix} = \begin{bmatrix} 0.0003 & 0.0015 & 1.0000 & 0.0000 \\ 1.0000 & 0.0000 & -0.0078 & -0.0002 \\ -0.1093 & -0.4044 & 0.3640 & 0.4680 \\ -0.1093 & 0.4044 & 0.0234 & 0.0000 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} \quad (3.3.1)$$

The scaled lateral helicopter model can be written in modal form as

$$\dot{z}_{lat} = F_{lat} z_{lat} + G_{lat} u_{lat} \quad (3.3.2)$$

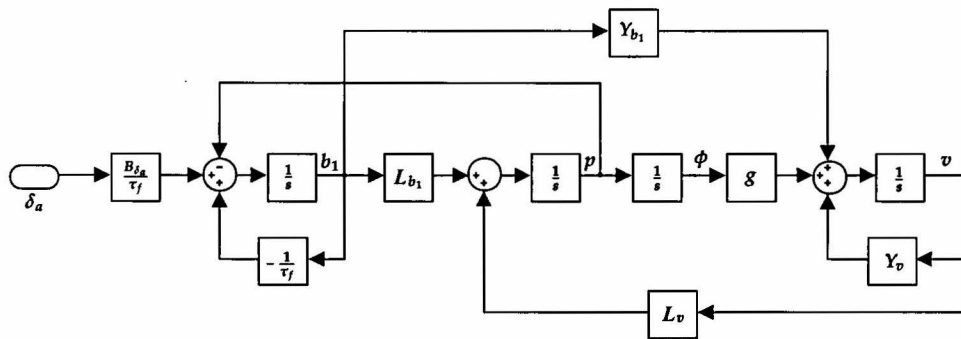


Figure 3.4: Simplified Decoupled Lateral Helicopter Model

where

$$F_{lat} = \begin{bmatrix} -4.2 & 19.13 & 0 & 0 \\ -19.13 & -4.2 & 0 & 0 \\ 0 & 0 & -0.066 & 0.096 \\ 0 & 0 & -0.096 & -0.066 \end{bmatrix} \quad (3.3.3)$$

$$G_{lat} = \begin{bmatrix} 0.7 \\ 5003.4 \\ -7.6 \\ 4329.6 \end{bmatrix} \quad (3.3.4)$$

Two sets of complex poles are visible in F_{lat} . The higher frequency pole set is known as the Roll Mode of motion while the lower frequency set is known as the Phugoid mode of motion.

3.3.1 Phugoid mode

The Lateral Phugoid mode poles are summarized below,

$$\lambda = -0.066 \pm j0.96 \quad (3.3.5)$$

$$\zeta = 0.57 \quad (3.3.6)$$

$$w_n = 0.12 \quad (3.3.7)$$

The Phugoid mode eigenvector is displayed in phasor form below,

$$\begin{bmatrix} v \\ p \\ \phi \\ b_1 \end{bmatrix} = \begin{bmatrix} 1.0000 \angle 0.00 \\ 0.0078 \angle 1.27 \\ 0.5928 \angle 52.12 \\ 0.0234 \angle -0.04 \end{bmatrix} \quad (3.3.8)$$

Figure 3.5 shows a time history response of the Phugoid mode of motion. The following observations are made from the plot:

- The Phugoid mode of motion is well observed in the lateral velocity v and the roll angle ϕ .
- Similar to the Phugoid mode analysed in the Longitudinal Helicopter model there exists an exchange of potential and kinetic energy in this mode of motion, hence the roll angle ϕ leads the lateral velocity v by approximately 50° .

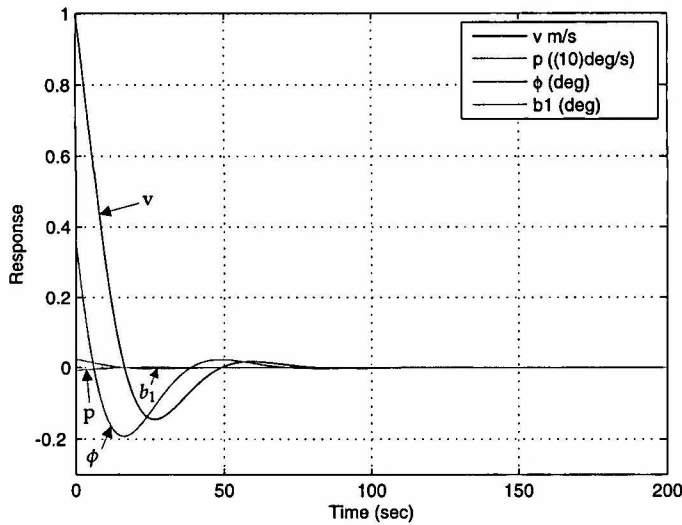


Figure 3.5: Lateral Phugoid Mode Response

3.3.2 Roll mode

The Roll mode poles are summarized below,

$$\lambda = -4.2 \pm j19.13 \quad (3.3.9)$$

$$\zeta = 0.21 \quad (3.3.10)$$

$$w_n = 19.6 \quad (3.3.11)$$

The Roll mode eigenvector is displayed in phasor form below,

$$\begin{bmatrix} v \\ p \\ \phi \\ b_1 \end{bmatrix} = \begin{bmatrix} 0.0015 \angle 78.78 \\ 1.0000 \angle 0.00 \\ 0.4189 \angle 74.87 \\ 0.4189 \angle -74.87 \end{bmatrix} \quad (3.3.12)$$

Figure 3.6 shows a time history response of the Roll mode of motion. The following observations are made from the plot:

- The Roll mode of motion is well observed in the roll rate r , roll angle ϕ and flapping angle b_1 .
- The Flapping angle b_1 has a significant effect on the Roll mode of motion.
- The lateral speed v has an insignificant effect on the Roll mode of motion.

3.4 Heave Helicopter Model

The Heave mode pole is summarized below,

$$\lambda = -1.11 \quad (3.4.1)$$

$$\zeta = 1 \quad (3.4.2)$$

$$w_n = 1.11 \quad (3.4.3)$$

Figure 3.8 shows a time history response of the Heave mode of motion.

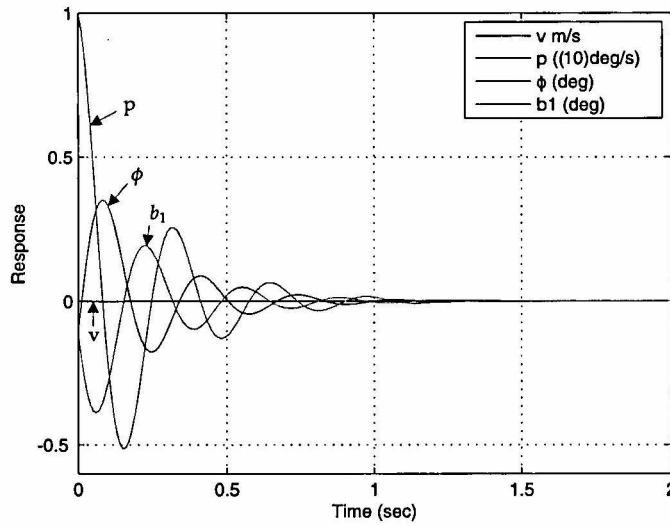


Figure 3.6: Roll Mode Response

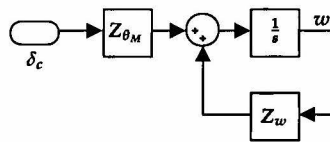


Figure 3.7: Simplified Decoupled Heave Helicopter Model

3.5 Heading Helicopter Model

The Heading mode pole is summarized below,

$$\lambda = -23.37 \tag{3.5.1}$$

$$\zeta = 1 \tag{3.5.2}$$

$$\omega_n = 23.37 \tag{3.5.3}$$

Figure 3.10 shows a time history response of the Heading mode of motion.

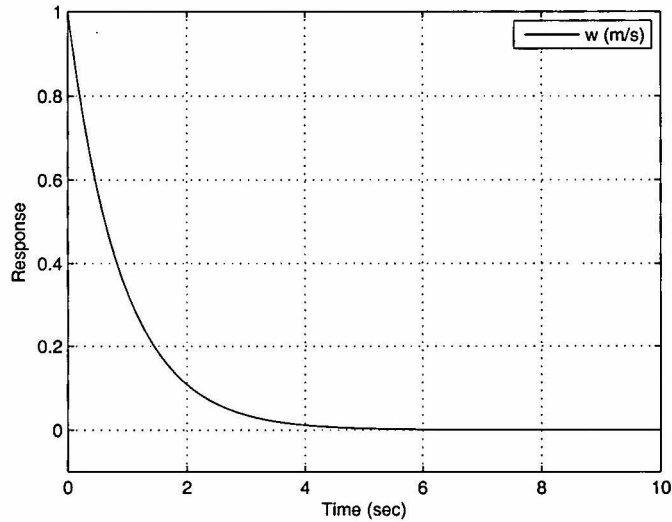


Figure 3.8: Heave Mode Response

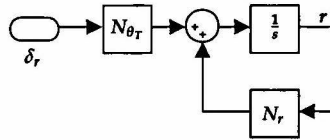


Figure 3.9: Simplified Decoupled Heading Helicopter Model

3.6 Summary

In this chapter a basic analysis of the modes of motion of a small RC helicopter was presented. The magnitude and phase contributions of each state on the modes of motion were shown so that controllers could be designed more effectively.

The decoupled models were illustrated with block diagrams. These block diagrams showed which states were affected by which system derivatives.

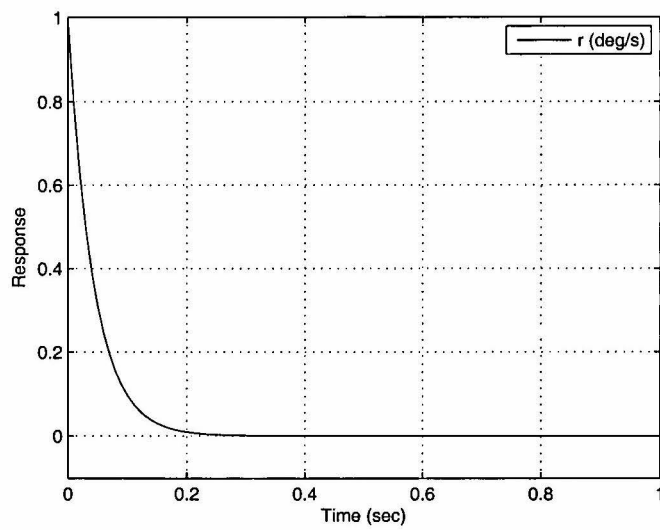


Figure 3.10: Heading Mode Response

Chapter 4

Control Law Design and Non-linear Simulation

4.1 Control Law Design

An RC helicopter is a highly non-linear, multi-input multi-output (MIMO) system. It has been demonstrated that RC helicopters can be controlled successfully using multiple classic single-input single-output (SISO) decoupled controllers [14].

The Control Law Design was simplified to the design of four separate, linearised, decoupled, near-hover models. These models were the Heading, Heave, Longitudinal and Lateral models. The success of this design methodology depended on the fact that each decoupled model was predominantly affected by one of the four control inputs.

Cross-coupling was ignored during the design phase of the controllers. The full non-linear model was used during simulation to demonstrate that the designed controllers could effectively control the helicopter around hover trim conditions. The system Model was augmented to include three position states and a heading state.

Successive-loop-closure has been widely used to automate unmanned helicopters [14, 15, 24] and was also adopted in this project. This method first controls the fast modes of motion of the system by closing control loops with

the dominant states that influence these modes. Once the fast dynamics are under control the slower dynamics can be controlled with ease. The advantage of successive-loop-closure is that the system can be tested practically by closing the loops successively, starting with the fast inner loops, and verifying that the system responds as expected before closing the slower outer loops.

4.1.1 Heading

The equations that define the heading model, with the addition of the heading angle ψ , follow,

$$\begin{aligned} \dot{r} &= N_r r + N_{\theta_T} \delta_r \\ \dot{\psi} &= r \end{aligned} \quad (4.1.1)$$

which leads to the following transfer function:

$$\frac{\psi}{\delta_r} = \frac{N_{\theta_T}}{s(s - N_r)} \quad (4.1.2)$$

The Gyro is operated in *normal* mode. This means that the Gyro damps the yaw rate of the helicopter by simple proportional feedback of the yaw rate. Thus, unlike similar projects where the rate Gyro was operated in *heading hold*¹ mode the plant did not have to be augmented with an additional r_{fb} (rate feedback) state. The effect of the proportional feedback term due to the Gyro was captured during modeling as was shown in Section 2.4.6. The Gyro's proportional feedback can be set up by adjusting a gain on the Gyro device. Once it was set up correctly no additional yaw rate feedback was required from the heading controller.

With the inner yaw rate loop closed, the open loop transfer function from heading angle to input cyclic follows,

$$\frac{\psi}{\delta_r} = \frac{147.3}{s(s + 23.37)} \quad (4.1.3)$$

The specifications for the heading angle controller are listed below,

¹In *heading hold* mode PI control is implemented on the yaw rate.

- Rise time of under 3s
- Overshoot of less than 20%
- Zero steady state errors

Since zero steady state errors were required Proportional-Integral (PI) control was implemented on the heading angle. The control law is described by the following equation:

$$\delta_r = -K_\psi e_\psi - K_{I\psi} \int e_\psi dt \quad (4.1.4)$$

where

$$K_\psi = k_\psi \quad (4.1.5)$$

$$K_{I\psi} = k_{I\psi}$$

$$e_\psi = \text{heading angle error}$$

The successive-loop-closure strategy is illustrated in Figure 4.1. Suitable val-

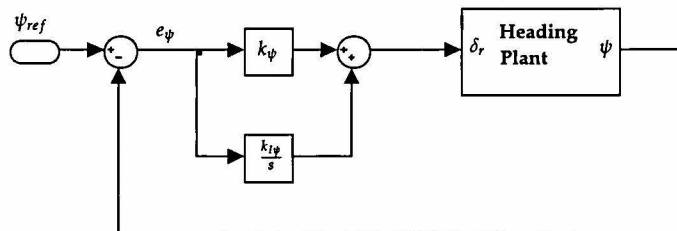


Figure 4.1: Successive-loop-closure for the Heading plant.

ues for gains k_ψ and $k_{I\psi}$ that would satisfy the specifications now had to be found. A brief description on how these values were obtained follows:

PI control introduces a pole and a zero into the system as follows:

$$D(s) = K_p + \frac{K_I}{s} = \frac{K_p(s + \frac{K_I}{K_p})}{s} \quad (4.1.6)$$

In Equation (4.1.6) it can be seen that a free integrator is introduced at the origin. The zero position is defined by the ratio of the integrator and pro-

portional gains ($-\frac{K_I}{K_p}$). A more intuitive approach to PI controller design was adopted whereby there was designed for the zero position first and then for the total controller gain. Thus, the following two steps were followed in the design of the PI controller:

1. The free integrator was introduced at the origin. The zero was then placed (designed for the ratio $\frac{K_I}{K_p}$). The closer the zero was placed to the origin in the s-plane, the more it canceled out the effect of the free integrator at the origin. The further away from the origin the zero was placed the more overshoot was introduced. The zero was placed as far away as possible from the origin while still adhering to the overshoot specification. The plant was Augmented as follows,

$$\begin{bmatrix} \dot{x} \\ \dot{x}_c \end{bmatrix} = \begin{bmatrix} A & B(\frac{K_I}{K_p}) \\ \bar{0} & 0 \end{bmatrix} + \begin{bmatrix} B \\ 1 \end{bmatrix} r \quad (4.1.7)$$

where x , x_c and r denote the system state vector, control state vector and reference input to the system respectively.

2. Designed for the feedback gain K_p by making use of the Augmented plant of Equation (4.1.7) and the root locus plot with ψ feedback to δ_r .

The following feedback gains were designed: $k_\psi = 0.2$ and $k_{I\psi} = 0.06$. The root locus with feedback from ψ to δ_r , and yaw angle step response are shown in Figures 4.2 and 4.3.

4.1.2 Heave

The equations that define the heave model, with the addition of the position state z , follow,

$$\begin{aligned} \dot{w} &= Z_w w + Z_{\theta_M} \delta_c \\ \dot{z} &= w \end{aligned} \quad (4.1.8)$$

which results in the following transfer function:

$$\frac{z}{\delta_c} = \frac{Z_{\theta_M}}{s(s - Z_w)} \quad (4.1.9)$$

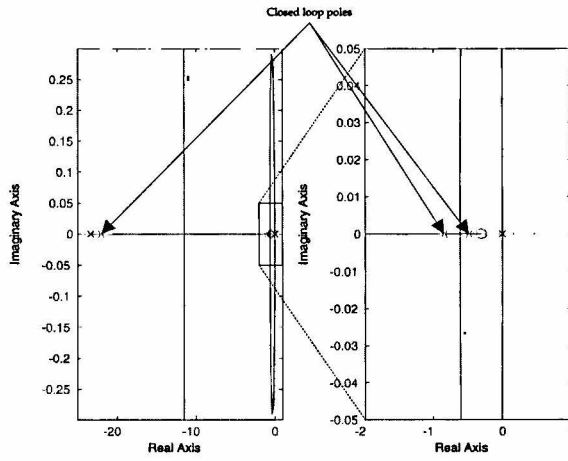


Figure 4.2: Root locus of heading plant.

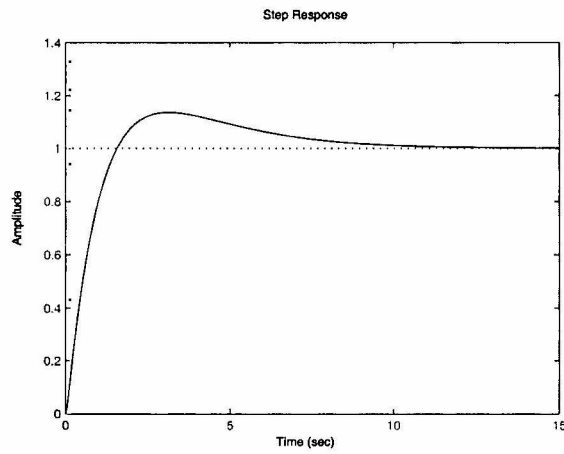


Figure 4.3: Heading angle step response.

The heave plant has a similar structure to that of the heading plant from Section 4.1.1 but its dynamics are approximately twenty times slower. From Equation (4.1.9) the transfer function from vertical position to input cyclic follows,

$$\frac{z}{\delta_c} = -\frac{164.5}{s(s + 1.11)} \quad (4.1.10)$$

The specifications for the vertical position controller are listed below,

- Rise time of under 3s
- Overshoot of less than 20%
- Zero steady state errors

Since zero steady state errors were required PI control was used on the vertical position z . Consider the case where only the vertical position z state is used as feedback. There will now be three poles close to the origin: The heave mode pole at -1.11 rad/s , the vertical position and PI controller poles, both at the origin. A zero will also be introduced due to the PI controller. Now, unless this zero is placed very close to the origin, thereby largely canceling the effect of the free integrator introduced by the PI controller, the system will be marginally stable as can be seen in Figure 4.4. By feeding back the ver-

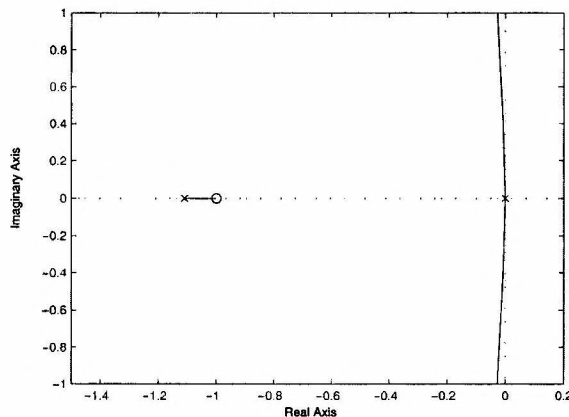


Figure 4.4: Root locus of the vertical position plant - no velocity feedback.

tical velocity w state the heave mode pole at -1.11rad/s was moved further left on the real axis of the s -plane, thereby making the plant more stable. The root locus with feedback from z to δ_c , and vertical position step response are shown in Figures 4.5 and 4.6. The control law is described by the following equation:

$$\delta_c = -K_w w - K_z e_z - K_{Iz} \int e_z dt \quad (4.1.11)$$

where

$$K_w = k_w \quad (4.1.12)$$

$$K_z = k_w k_z$$

$$K_{Iz} = k_w k_{Iz}$$

$$e_z = \text{vertical position error}$$

The successive-loop-closure strategy is illustrated in Figure 4.7. The following feedback gains were designed: $k_w = -0.03$, $k_z = 0.7$ and $k_{Iz} = 0.07$.

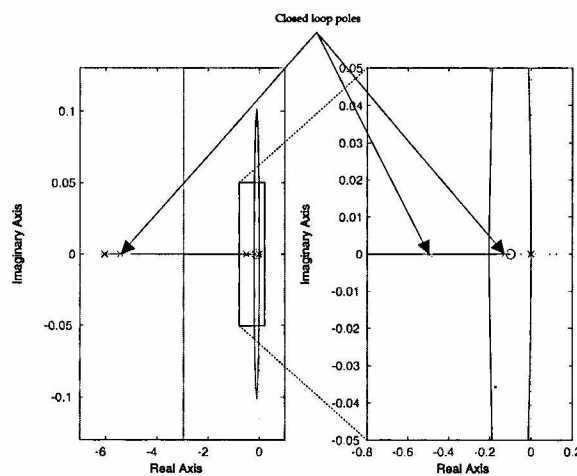


Figure 4.5: Root locus of heave plant - velocity feedback added.

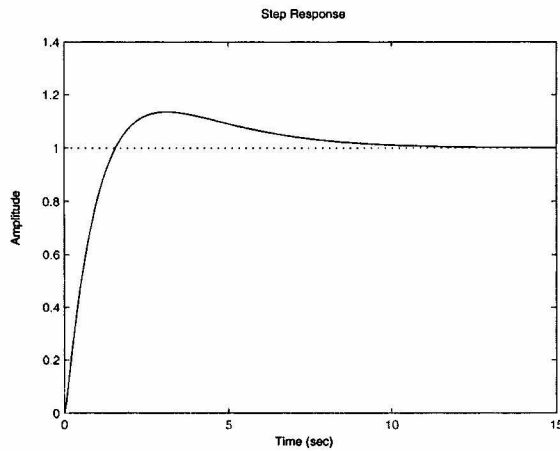


Figure 4.6: Heave position step response.

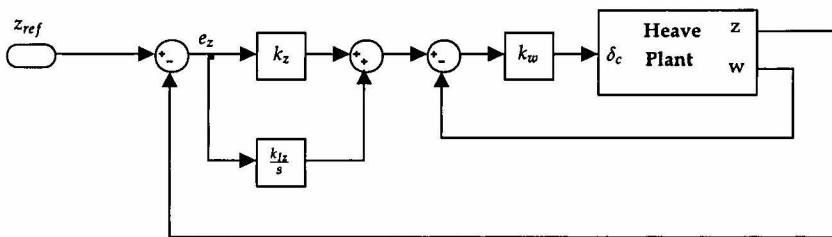


Figure 4.7: Successive-loop-closure for the Heave plant.

4.1.3 Lateral and Longitudinal

4.1.3.1 Roll and Pitch

The equations that define the roll and pitch models follow,

$$\begin{aligned} \dot{b}_1 &= -p - \frac{1}{\tau_f} b_1 + \frac{B\delta_a}{\tau_f} \delta_a & (4.1.13) \\ \dot{a}_1 &= -q - \frac{1}{\tau_f} a_1 + \frac{A\delta_b}{\tau_f} \delta_b \\ \dot{p} &= L_v v + L_{b_1} b_1 \\ \dot{q} &= M_u u + M_{a_1} a_1 \end{aligned}$$

Gavrilets *et al.* [17] went on to approximate the transfer functions from cyclic inputs to rates for slow forward flight with the following second order transfer functions,

$$\begin{aligned}\frac{q}{\delta_b} &= \frac{\frac{A_{\delta_b}}{\tau_f} w_{nq}^2}{s^2 + s/\tau_f + w_{nq}^2} \\ \frac{p}{\delta_a} &= \frac{\frac{B_{\delta_a}}{\tau_f} w_{np}^2}{s^2 + s/\tau_f + w_{np}^2}\end{aligned}\quad (4.1.14)$$

where

$$\begin{aligned}w_{nq}^2 &= M_{a_1} \\ w_{np}^2 &= R_{b_1}\end{aligned}$$

4.1.3.2 Velocity Dynamics

The horizontal velocity dynamics follow,

$$\begin{aligned}\dot{u} &= X_u u - g\theta + X_{a_1} a_1 \\ \dot{v} &= Y_v v + g\phi + Y_{b_1} b_1\end{aligned}\quad (4.1.15)$$

For near-hover flight the helicopter frame and blades were approximated as a rigid body [14] so that Equation (4.1.15) simplified to

$$\begin{aligned}\dot{u} &\approx -g\theta \\ \dot{v} &\approx g\phi\end{aligned}\quad (4.1.16)$$

for small angles θ and ϕ .

4.1.3.3 Horizontal Controllers

Equations (4.1.14) and (4.1.16) were included for the sake of completeness but were not used to design the horizontal controllers. Rather, the full decoupled models derived in Section 2.6 and represented by Equations (4.1.13) and (4.1.15) were used. A root locus plot was used where each state was fed back individually, starting with the fast dynamics, and closing the loops successively until the outer loop specifications were satisfied. A root locus plot shows how all the system poles and zeros are effected while feeding back any

specific state.

The specifications for the horizontal position controllers are listed below,

- Rise time of under 5s
- Overshoot of less than 10%
- Zero steady state errors

The control laws are described by the following equations:

$$\delta_a = -K_\phi\phi - K_vv - K_yy - K_{Iy} \int e_y dt \quad (4.1.17)$$

$$\delta_b = -K_\theta\theta - K_uu - K_x x - K_{Ix} \int e_x dt \quad (4.1.18)$$

where

$$K_\phi = k_\phi \quad (4.1.19)$$

$$K_\theta = k_\theta$$

$$K_v = k_\phi k_v$$

$$K_u = k_\theta k_u$$

$$K_y = k_\phi k_v k_y$$

$$K_x = k_\theta k_u k_x$$

$$K_{Iy} = k_\phi k_v k_{Iy}$$

$$K_{Ix} = k_\theta k_u k_{Ix}$$

$$e_y = \text{lateral position error}$$

$$e_x = \text{longitudinal position error}$$

The successive-loop-closure strategies are illustrated in Figures 4.8 and 4.9. Equations (4.1.17) and (4.1.18) did not include pitch or roll rate feedback terms. This has become standard practice in RC helicopter projects [9, 14, 15, 25] due to the mechanical *lagged rate feedback system* introduced through the Bell-Hiller stabilizer bar. Mettler *et al.* [24] acknowledged that additional rate feedback would help to reduce the gust response of the helicopter but found that it was not essential.

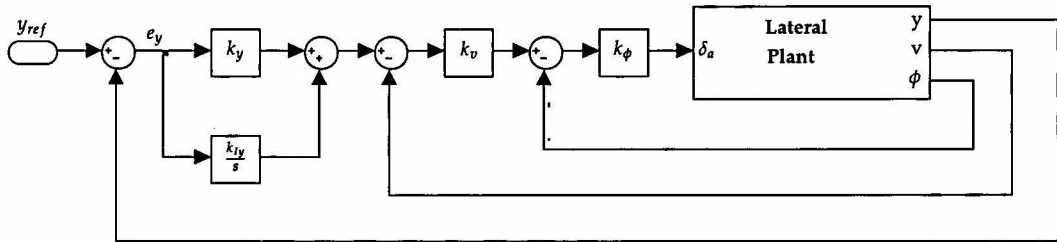


Figure 4.8: Successive-loop-closure for the Lateral plant.

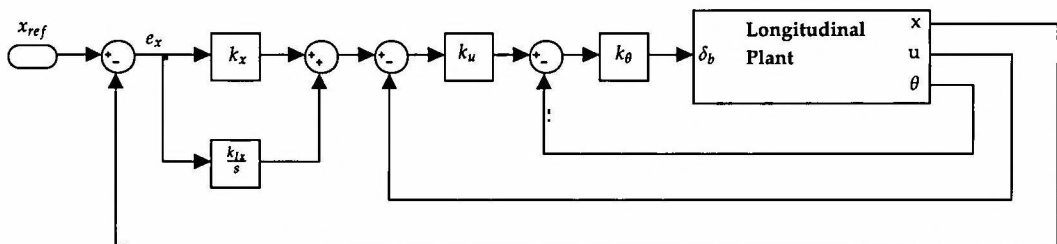


Figure 4.9: Successive-loop-closure for the Longitudinal plant.

The loops were closed successively, starting with the fast dynamics on the inner loops and ending with the slower dynamics on the outer loops. The gains were selected using the following guidelines:

- Maximise the gains without stimulating unmodeled effects.
- Maintain an adequate gain and phase margin throughout.

The fundamental task in controlling an aircraft is to control its attitude [4]. By having control of the attitude of an aircraft its acceleration is controlled indirectly. Since acceleration integrates down into velocity and position over time it will only be necessary for the outer control loops to compensate for reference velocity and position commands. Without adequate attitude regulation on the inner loops there would have to be more reliance on the outer loops to stabilise the aircraft i.e. velocity and position errors have to be incurred first for the outer loop controllers to react and correct for them. With adequate attitude regulation the aircraft is stabilised with the fast inner loop before errors can be made in velocity and position.

The following feedback gains were designed: *Lateral plant*: $k_\phi = 0.2$, $k_v = 0.25$, $k_y = 0.4$ and $k_{Iy} = 0.001$. *Longitudinal plant*: $k_\theta = 0.2$, $k_u = -0.25$, $k_x = 0.4$ and $k_{Ix} = 0.001$. The root locus with feedback from x to δ_b , and longitudinal position step response are shown in Figures 4.10 and 4.11.

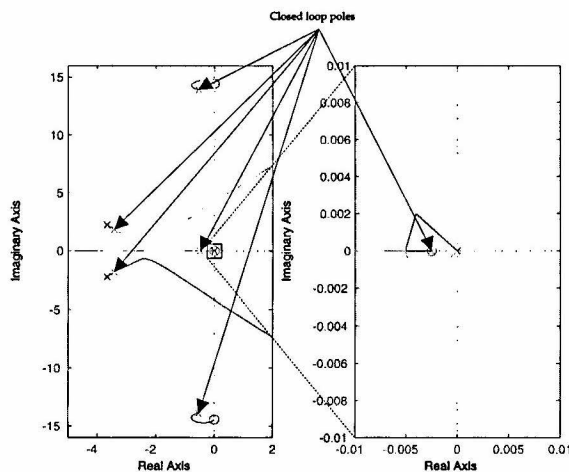


Figure 4.10: Root locus of longitudinal plant.

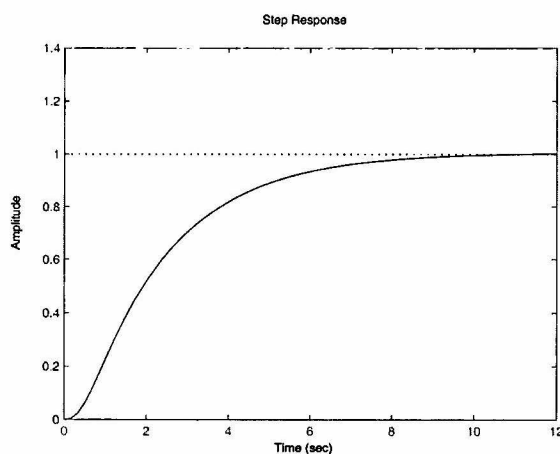


Figure 4.11: Longitudinal position step response.

4.1.4 Controllers overview

A brief description of the controller designs was given. Since the project goal was low advance autonomous flight there were designed for low bandwidth outer loop controllers. Anti-windup was implemented throughout. Saturation blocks were used to limit the position and heading errors. The saturation blocks prevented the actuators from saturating.

4.2 Non-linear Simulation

Simulations were done using the full non-linear helicopter model in order to put the simplifications made during decoupling to the test. Simulation was done with MATLAB's Simulink package. The controllers were developed in three phases:

1. Controllers were implemented using Simulink's user friendly block-diagrams.
2. Controllers were coded in C-code and used in Matlab's S-function block-diagrams during Simulations.
3. The C-code used in the S-function block-diagrams was integrated onto the OnBoard Computer (OBC).

During the last phase Simulink was used to simulate the working of the controllers but with the important difference that the controllers were now running real-time onboard the OBC. This real-time simulation onboard the OBC was made possible by a Hardware-In-The-Loop (HIL) board that will be discussed in more detail shortly. A graphical display that was developed at Stellenbosch University for the UAV group was modified and used during all Simulation phases. The display provided valuable visual feedback on the working of the controllers during real-life situations.

4.2.1 Hardware in the Loop Simulator (HILS)

4.2.1.1 HIL Simulator overview

Hardware in the loop simulation involves replacing parts of a purely software simulation environment by some of the hardware components to be used during practical flight tests. This dramatically reduces the risk of failure on

flight test day since the correct working of the hardware can be tested in a safe environment first before exposing the aircraft to the dangers of a real-life situation. The HIL simulator consists of three main parts:

1. Simulink models of the X-Cell helicopter, its runway and the environment. These models are used to generate forces and moments which are passed on to a six degrees of freedom (6DoF) block. The outputs of the 6DoF block are used to generate a sensor model.
2. A *Simulation Hardware Interface* (SHI) for the hardware to communicate with these models.
3. A graphical display output.

The SHI to the avionics makes it is possible to fly the helicopter with the RC transmitter while viewing the response of the helicopter on the *graphical display*. The autopilot can be armed (or disarmed) from the RC transmitter during simulations allowing the control algorithms to run real-time onboard the avionics. During simulation, commands can be issued and telemetry data displayed on the ground station via the telemetry link. Testing crucial hardware at the test bench has many advantages:

- It instills a high degree of trust in the designed control algorithms.
- It ensures that the system is thoroughly tested before attempting a flight test.
- It familiarizes the operator of the ground station with the necessary procedures so that he/she can be well prepared for any situation that might occur on flight test day.

The Simulink structure of the HIL simulator can be seen in Figure 4.12. The *X-Cell Helicopter model* makes use of the control vector (actuator commands) and the current state of the aircraft to propagate the data vector (aircraft states) through to the next Simulink time step. The data vector serves as an input to the *Autopilot Model* where it is converted to equivalent sensor measurements. These sensor measurements are then passed on to the control algorithms which calculate the new control vector. The *Wings Control Panel* switches the HILS on or off so that the *Autopilot Model* uses either the control algorithms implemented onboard the avionics or the algorithms simulated

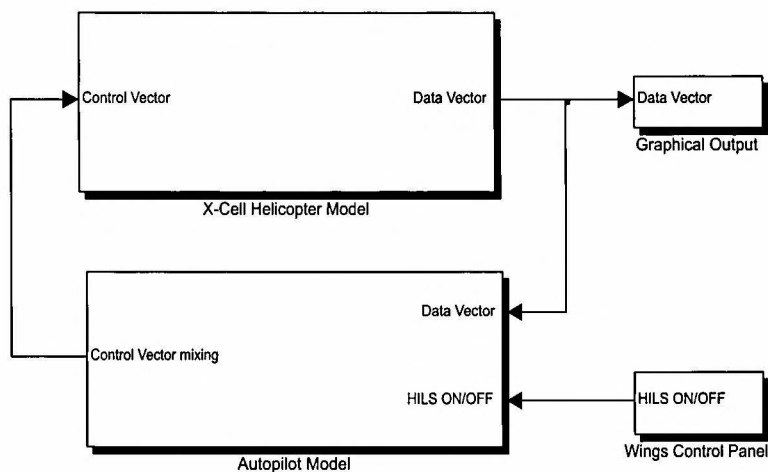


Figure 4.12: HIL Simulator Structure in Simulink

in Simulink. The *Graphical Output* block uses the attitude and position information of the data vector and outputs it to an OpenGL graphics engine so that the helicopter can then be displayed in a three dimensional (3D) environment.

4.2.1.2 X-Cell Helicopter Model

The *X-Cell Helicopter model* calculates a new data vector every time step by making use of the control vector and the current aircraft states. This is accomplished by putting all the aircraft specific and environmental forces and moments through a 6DoF block².

4.2.1.3 Autopilot Model

Figure 4.13 shows an expansion of the *Autopilot Model* from Figure 4.12. The *Sensor Model* converts the data vector to equivalent sensor measurements and adds noise to the measurements. The sensor measurements are used by the control algorithms to calculate the actuator commands. The *Simulation Hardware Interface (SHI)* is used when HILS is enabled on the *Wings Control Panel* in Figure 4.12, otherwise the *Simulated Control Algorithms* are used. The SHI block packages the sensor measurement data and writes it out via the serial

²A 6DoF block contains the six degrees of freedom dynamic equations.

RS232 port of the computer to the HIL board. The HIL board decodes the data and writes it out to the avionics package. The data is read in by the avionics package and passed on to the control algorithms running onboard the OBC. New actuator commands are calculated. These actuator commands are written back to the HIL board where they are packaged and sent to Simulink via the RS232 port completing the loop.

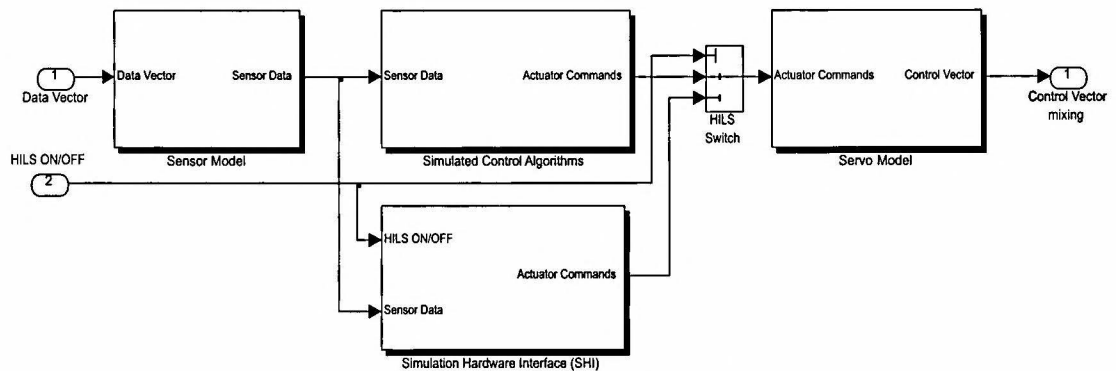


Figure 4.13: Autopilot model for the HIL Simulator

4.2.1.4 Graphical Display

The *graphical display* simulator is written in C-code and makes use of the OpenGL graphics library to display the aircraft's motion through a virtual 3D surrounding. A screenshot of the *graphical display* is shown in Figure 4.14. The major advantage of the *graphical display* simulator is that the controllers can be seen at work as they would operate in a real-life situation.

4.3 Non-linear Controller Simulation

The controllers designed in Section 4.1 were simulated in HIL before attempting any flight tests. The simulated responses of each controller to commanded references in HIL were then compared to those obtained during the design phase. Disturbances such as wind and sensor noise were simulated in order to test the controllers thoroughly. The step response of the controllers is shown in Figures 4.15 to 4.18.



Figure 4.14: Graphical Simulator Screenshot - Helicopter on the Runway

From the figures it can be seen that the HIL responses follow the linear designed responses well. The following observations are made:

- The heading steady state response is accurate to within two degrees.
- The heave steady state response stays within one meter of the linear designed response.
- The Longitudinal and Lateral steady state responses are accurate to within one meter.

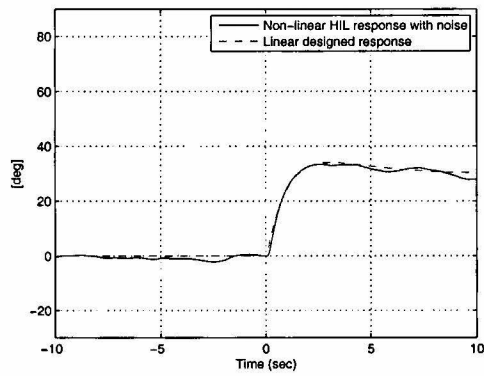


Figure 4.15: Response to a heading step command of 30 degrees

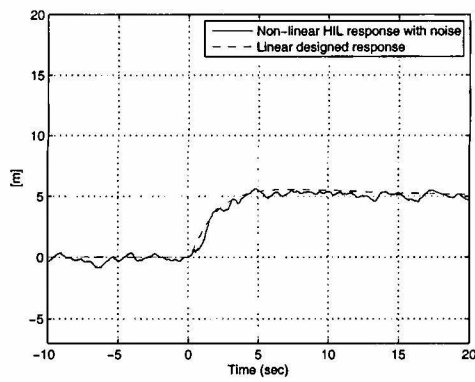


Figure 4.16: Response to a heave step command of 5 meters

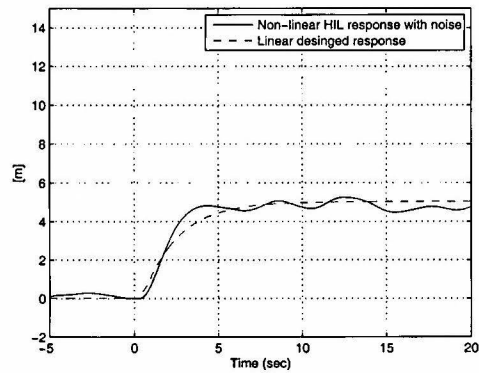


Figure 4.17: Response to a longitudinal step command of 5 meters

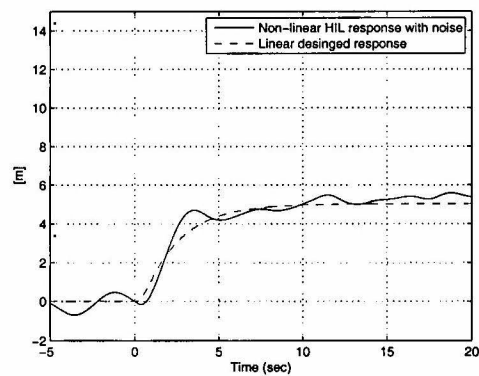


Figure 4.18: Response to a lateral step command of 5 meters

4.4 Summary

This chapter outlined the process followed in the design of the decoupled controllers. The *Hardware in the loop Simulator (HILS)* was presented. Simulation results were obtained by simulating the designed controllers in HIL. With the controllers simulated and the system tested thoroughly in HIL, the X-Cell was now ready for practical testing.

Chapter 5

Waypoint Navigation and Ground Station

The X-Cell helicopter can be enabled to fly certain waypoints by making use of a *Guidance Controller*. A *Guidance Controller* is a top decision making layer that guides the helicopter to navigate itself through a given set of waypoints by following a preprogrammed set of rules.

The *Ground Station* consists of a laptop running the ground station software, an RC transmitter used by the safety pilot and an RF transducer module providing a telemetry link to the avionics of the aircraft.

5.1 Guidance Controller

A great advantage of a helicopter is its ability to fly in any direction without necessarily having to change its heading. However, due to aerodynamic drag effects the helicopter flies most efficiently in the positive Ox_B -axis. Also, coupling between the heading and the lateral modes of motion will be stimulated when flying laterally. Thus, for the helicopter to fly toward a waypoint in the most efficient way the heading of the helicopter should be changed toward the waypoint before flying there. The main goal of the *Guidance Controller* was to change the heading and fly toward a new waypoint according to a basic set of rules:

- Only change heading if the next waypoint is further than a user-defined radius away from the helicopter's current position.

- First update the heading reference for a user-defined period of time and then only update the position reference. Pointing the helicopter in the direction of the next waypoint first will ensure that only the longitudinal controller is stepped when the position reference is uploaded and will prevent the situation where both the longitudinal and lateral controllers are stepped simultaneously. This simultaneous stepping of the controllers could stimulate unmodelled cross-coupling effects.
- Update the heading reference every time step while flying toward a waypoint until the helicopter has entered a user-defined radius around the waypoint.
- Hover at the waypoint within the user-defined radius for a user-defined period of time before the next waypoint is considered and the process repeats itself.

The new heading was calculated with the following simple equation,

$$\psi_{new} = \text{atan2}(\Delta y, \Delta x) \quad (5.1.1)$$

where Δy and Δx denote the position offsets in the lateral and longitudinal body axes respectively. $\text{atan2}(y, x)$ returns the four quadrant inverse tangent (arctangent) of the real parts of y and x and is defined over the interval $[-\pi, \pi]$ which covers the entire heading range of the helicopter. It differs from $\text{atan}(y, x)$ whose results are limited to the interval $[-\frac{\pi}{2}, \frac{\pi}{2}]$. A flow chart of the *Guidance Controller* is shown in Figure 5.1. Descriptions of the variables used in Figure 5.1 are given in Table 5.1.

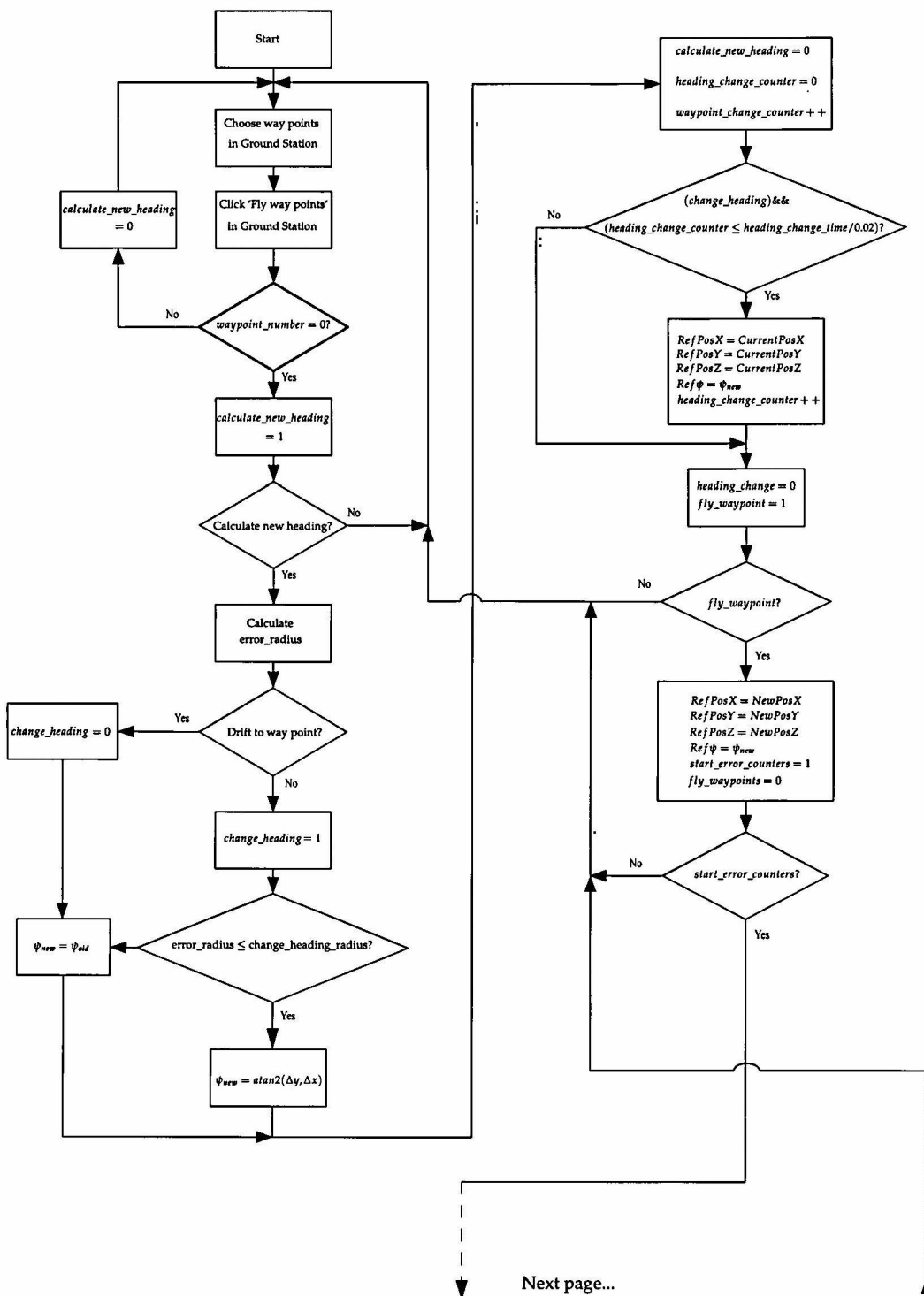


Figure 5.1: Flow chart of the Guidance Controller

...Previous page

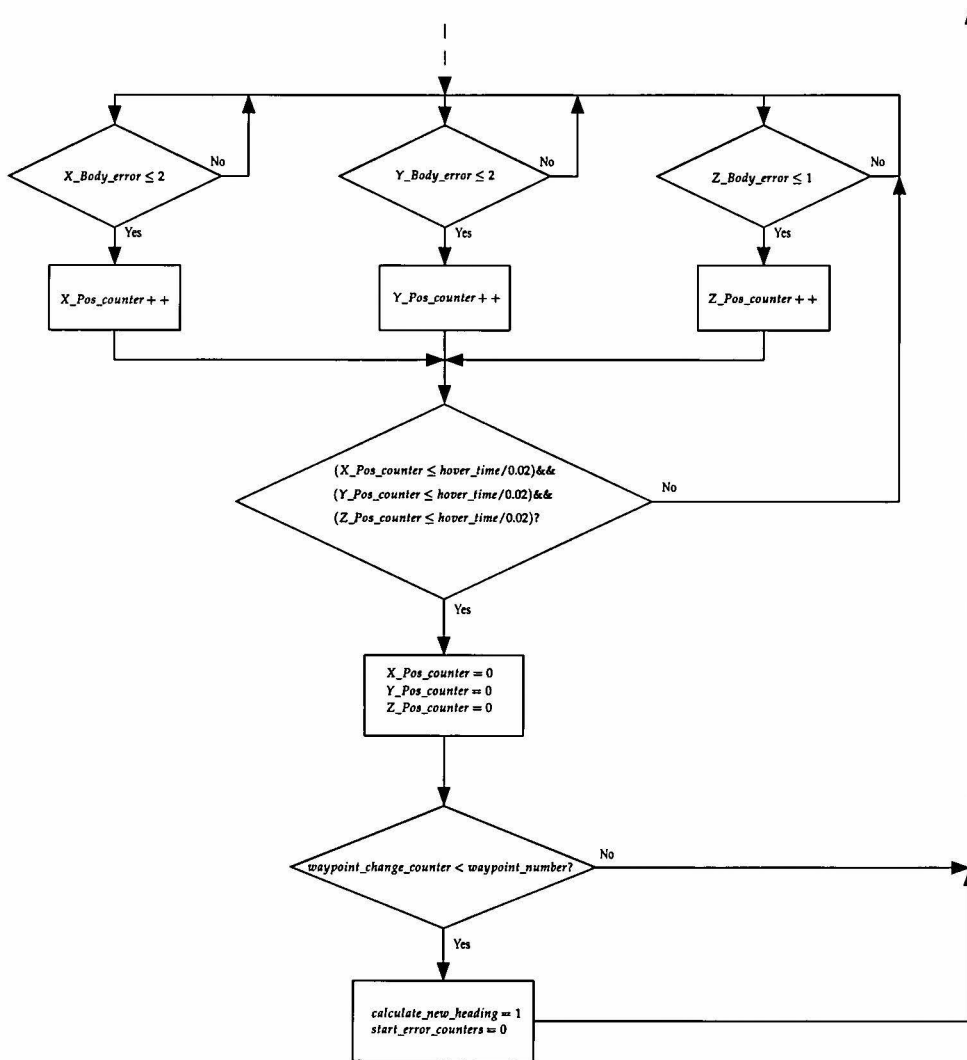


Table 5.1: Variables used in Figure 5.1 listed in alphabetical order

Variable	Description
calculate_new_heading	Determines if a new heading reference should be calculated
change_heading	Determines if the new calculated heading should be uploaded to the heading reference or if the helicopter should drift toward the new waypoint without changing its heading
change_heading_radius	If the next waypoint falls outside of this radius the heading of the helicopter should be changed
error_radius	This radius is defined as the shortest distance between the current and next waypoints.
fly_waypoint	Determines if the next waypoint should be uploaded to the position references
heading_change_counter	Increments every time step while the heading is being changed
heading_change_time	User-defined time in which the heading can be changed without uploading the next waypoint to the position references
hover_time	User-defined time during which the helicopter must hover within specified boundaries close to the waypoint before a new waypoint can be uploaded
start_error_counters	Determines when it must be checked if the helicopter is hovering within the specified boundaries close to the waypoint
waypoint_change_counter	Variable that increments after a new waypoint has been uploaded
waypoint_number	Variable that indicates the total number of waypoints to be flown during a navigation mission

5.2 Ground Station Software

The Ground Station Software developed in [5] was extended to meet the requirements of this project. The Ground Station Software was developed using Borland C++ builder and allows an operator to manage the avionics with a user-friendly Graphical User Interface (GUI). The Ground Station Software allows the user to:

- Manage the Autopilot.
- Evaluate the performance of the sensor and estimator data to be used by the controllers.
- Verify the correct working of the controllers.
- Operate the Guidance Controller.
- Change estimator variances and controller gains.
- Give step commands in order to evaluate the controllers.

The different tabs of the Ground Station Software will be discussed briefly in the following section.

5.2.1 Ground Station Tab

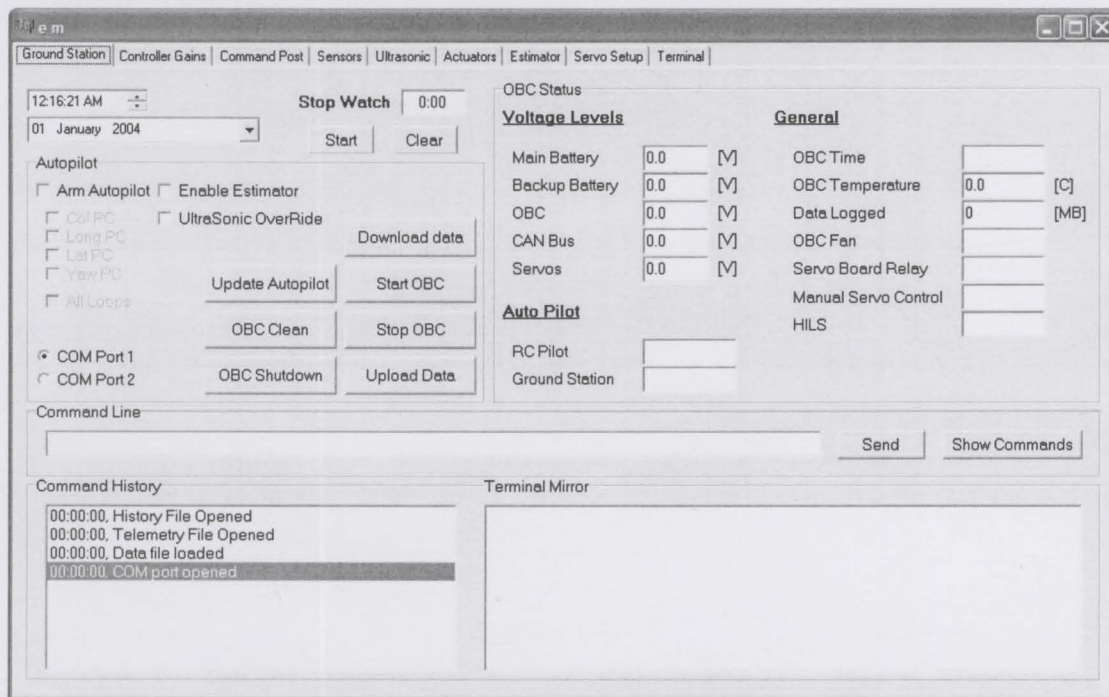


Figure 5.2: Screenshot of the Ground Station Tab

Autopilot Group Box

- *Download Data* downloads the *aircraft.dat* file containing various constants like estimator variances, controller gains and sensor calibration values to the OBC.
- *Upload Data* uploads Data logged onboard the OBC to a PC via the LAN connection.
- *Ultrasonic OverRide* uses the Ultrasonic sensor measurement instead of the GPS height measurement to estimate height in the EKF.
- The four designed controllers can be armed partially or all at the same time with the check boxes shown.

OBC Status Group Box

- Important avionics Status information such as the battery voltages and OBC temperature is displayed.
- The status of the Autopilot (*Armed* or *Disarmed*) is displayed.

Other Group Boxes

The *Command History* logs all mission activities. *Terminal Mirror* is used to monitor the OBC status. The *Command line* allows predefined commands to be issued to the OBC.

5.2.2 Controller Gains Tab

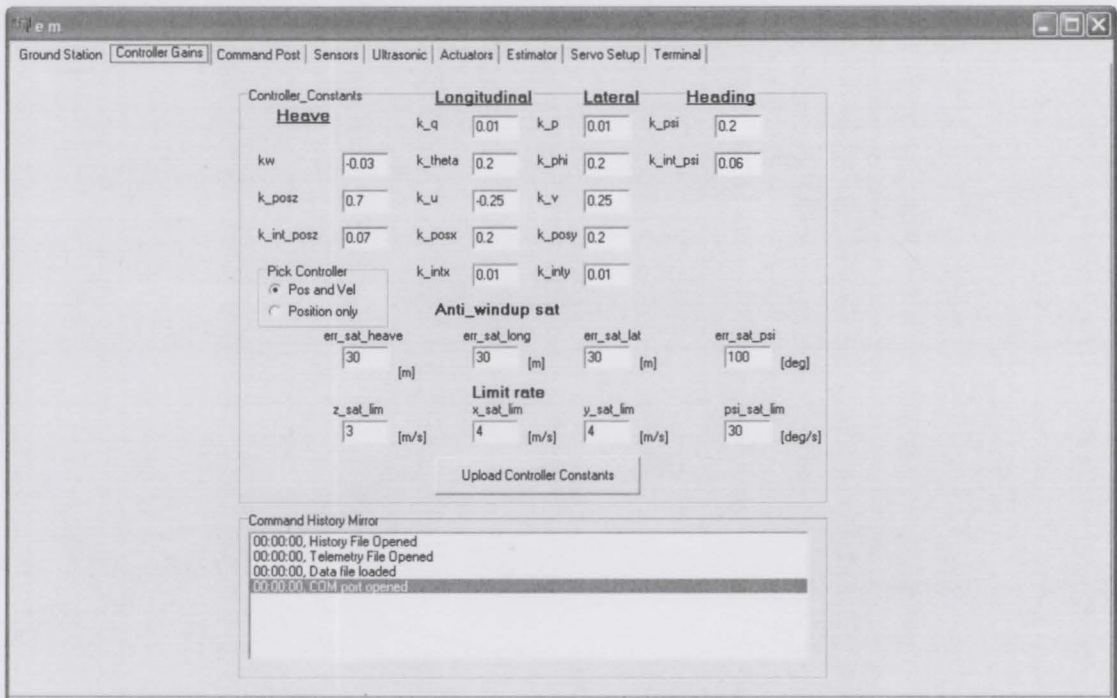


Figure 5.3: Screenshot of the Controller Gains Tab

Controller Constants Group Box

The following controller parameters can be modified:

- Controller gains.
- Anti-Windup saturation limits.
- The maximum rate at which the helicopter can move in any direction.

5.2.3 Command Post Tab

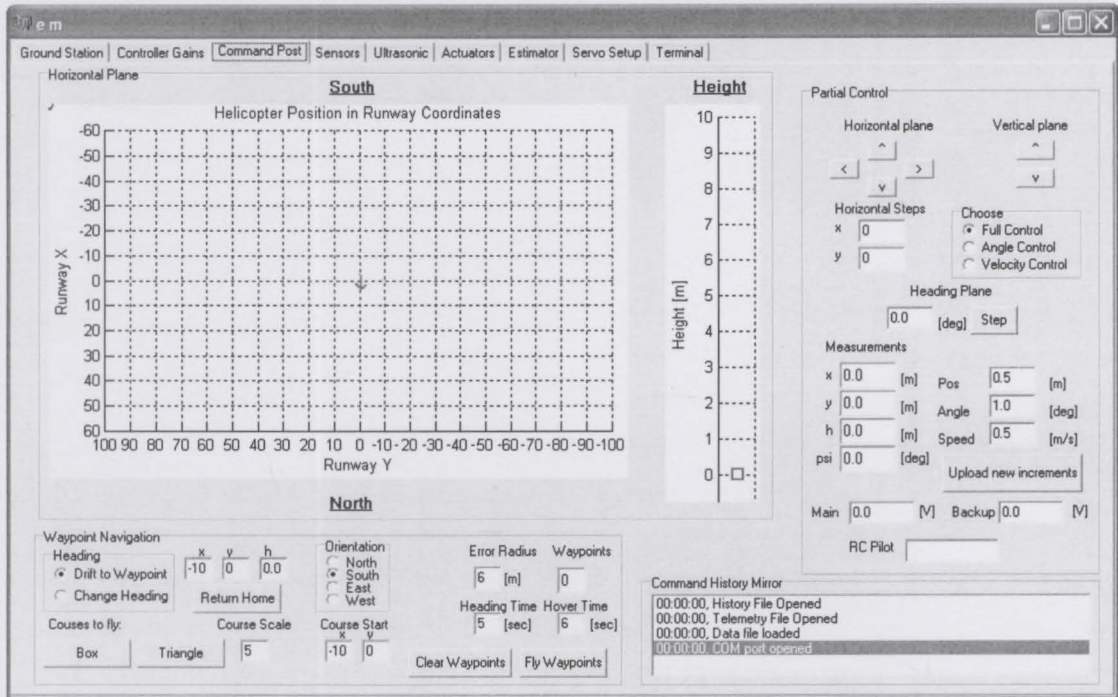


Figure 5.4: Screenshot of the Command Post Tab

Waypoint Navigation Group Box

- Choose whether the helicopter must change its heading toward the next waypoint or drift there while maintaining its current heading in the *Heading* group box.
- The *Courses to fly* offers two preprogrammed courses that the helicopter could fly namely a square and a triangle. The *Orientation* group box determines in which direction the preprogrammed courses will be orientated and the *Course Scale* adjusts the distance each waypoint will be separated from the next one. The *Course Start* text box allows the user to enter the Runway Coordinates where the start of the preprogrammed course should be.

- The *Return Home* text box allows the operator to enter the Runway Coordinates of a safe place for the helicopter to land in the event of an emergency. This can be useful if the communication link to the helicopter malfunctions. Numerous safe landing zones can be identified along any specific mission course.
- The *Error Radius*, *Waypoints*, *Heading Time* and *Hover Time* text boxes are linked with variables *error_radius*, *waypoint_number*, *heading_change_time* and *hover_time* respectively. See Table 5.1.

Partial Control Group Box

- The *Choose* group box allows the operator to choose which controller loops to close.
- Step commands can be given to the controllers by using the arrows shown. The increments of the Runway position step commands can be changed with *Upload new increments*.
- The Runway position and heading estimates are displayed in order to evaluate the helicopter's response to step commands.

Horizontal Plane Group Box

A two dimensional map around the origin of the Runway is displayed where the helicopter position and orientation are represented by an arrow and the waypoints are displayed as dots. The mouse cursor becomes a cross-hair when dragged over the map and allows the user to click any number of waypoints on the map. These waypoints will be flown in the order in which they were clicked by the operator. A corresponding height map is displayed to the right of the group box to aid the user further with visual feedback.

5.2.4 Other Tabs

The rest of the Tabs are similar to those discussed in [5].

5.3 Summary

The need for a Guidance Controller during autonomous flight was discussed. The Guidance Controller logic was illustrated by means of a detailed flow

CHAPTER 5. WAYPOINT NAVIGATION AND GROUND STATION

73

chart. The added features of the Ground Station Software were discussed. The Ground Station Software was shown to be user-friendly, interactive and easy to use.

Chapter 6

Practical Results

Similar to Carstens [14], most of the practical testing in this project involved platform development. It was found that the 70 size glow engine was running too hot. This increased the risk of engine failure which could severely hamper the project's progress. It was decided to upgrade the engine to a 90 size glow engine and to decrease the weight of the payload where possible. The weight of the payload was decreased by more than half a kilogram by redesigning the extended skids and the avionics box. The original heavy vibration dampers were removed. Vibration damping was added to the Inertial Measurement Unit (IMU) by encapsulating the IMU stack with sponge.

Groenewald [9] never flew with the *servo*board powered up as part of his practical testing. It was found that the *servo*board introduced RF noise into the system which interfered with the RC transmitter frequency used by the safety pilot. This interference nearly led to a crash when the safety pilot flew the helicopter out of range. Luckily the pilot could regain control of the helicopter and prevent a crash. Shortly after this incident RF noise interference caused a crash in Roos's [6] project which also made use of a similar *servo*board and RC transmitter frequency band¹. Since the *RF noise interference* issue affected all the projects in the UAV group it was decided to dedicate time to try and solve the problem. Better results were obtained by improving the grounding of the avionics. However, the results were still not good enough. A solution was found by acquiring a 2.4GHz RC Transmitter for the helicopter.

¹RC transmitter frequency band at 35MHz.

Vibration proved to be one of the greatest obstacles in this project. It was found that the extended skids developed by Groenewald [9] were not stiff enough and induced a great deal of vibration on the rest of the platform, particularly the Ultrasonic Sensor used to determine the vehicle's height and the crucial IMU on which important sensors like the rate gyros, accelerometers and the magnetometer were located. A lot of time was devoted to developing algorithms to try and improve the Ultrasonic sensor measurement's quality. The best results were obtained when the sensor was moved from its original position on the nose of the helicopter to the bottom of the avionics box where the vibration was much less.

Eventually after all the platform issues had been sorted out success was achieved promptly with Fully Autonomous flight. The four separately designed controllers were tested one at a time until all degrees of freedom were being controlled autonomously.

6.1 Flight tests

The relatively simple heading and heave loops were closed first. Lastly the longitudinal and lateral loops were closed. The latter controllers were tested by closing the loops successively, starting with the fast inner loops of roll and pitch angles, then the speed and finally the position loops. The practical results obtained during full autonomous flight will be shown in the following section.

6.1.1 Heading

From Figure 6.1 it can be seen that the heading dynamics were slightly faster than predicted in the *simulation model*. However, the response adheres to the required design specifications and the heading gains were left unchanged. The heading angle could be controlled to within five degrees.

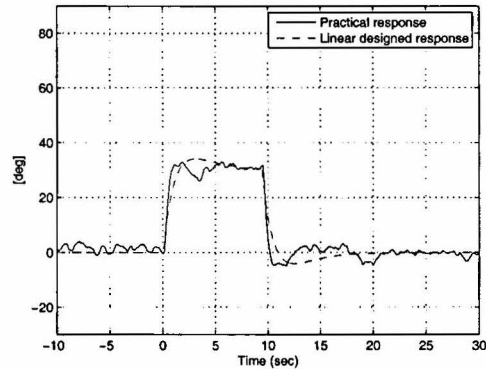


Figure 6.1: Practical Step response of the Heading Controller with ± 30 degree steps

6.1.2 Heave

The Heave controller step response can be seen in Figure 6.2. Due to GPS drift and windy conditions the helicopter slowly moved around during autonomous hover. The cross-coupling introduced due to this movement was mainly responsible for the constant one meter oscillation around the commanded altitude since the X-Cell lost and gained altitude during roll and pitch maneuvers. However, the mean value of the heave response was around the commanded reference altitude.

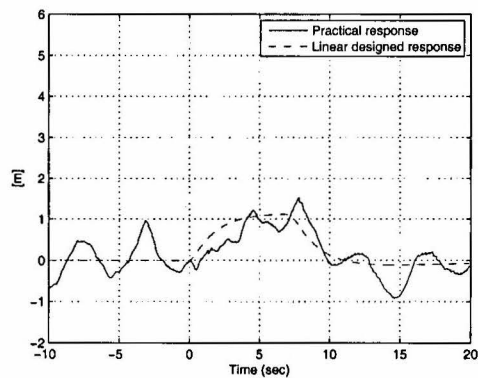


Figure 6.2: Practical Step response of the Heave Controller with ± 1 meter steps

6.1.3 Longitudinal/Lateral

The last control loops to be closed were those of the longitudinal/lateral controllers. Once the attitude loops of roll and pitch were successfully closed the X-Cell was able to hover autonomously with its position drifting slowly over time. Once the speed and position loops were closed the X-Cell hovered autonomously within a four² meter radius for over four minutes until it's fuel started running out. A one meter step response of the longitudinal controller is shown in Figure 6.3. Here too a one meter oscillation around the commanded reference is noticed. One of the reasons for this oscillation is that instead of hovering in one place the X-Cell is moving around slowly due to GPS drift and windy conditions. This slow movement stimulates cross-coupling effects between the longitudinal and lateral dynamics which contribute to the position oscillation. Also, low bandwidth longitudinal/lateral controllers were designed as a first iteration. The magnitude of these oscillations can be decreased if the controller gains are increased, specifically the inner angle loop gains. A two dimensional view of the *x* and *y Runway Coordinates* during an autonomous hover can be seen in Figure 6.4. The numbers indicated on the plot are spaced in 20 second intervals. *Number 1* is in the center of the plot and indicates the *xy runway* position reference.

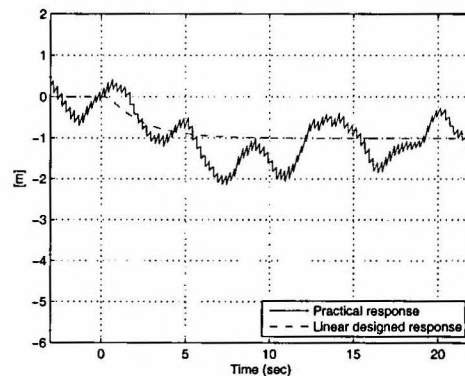


Figure 6.3: Practical Step response of the Longitudinal Controller with a -1 meter step

²The four meter radius is around the measured GPS position. Due to GPS drift of approximately four meters the actual radius within which the X-Cell hovered practically was around eight meters.

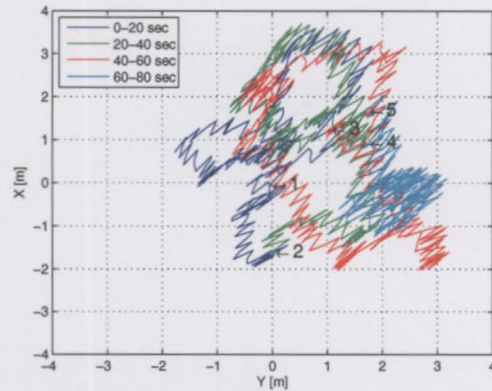


Figure 6.4: 2D view of the Runway Coordinates during a Fully Autonomous Hover

6.2 Summary

The practical results obtained were presented and discussed. Platform development was shown to be a big part of this project. Once the platform was established autonomous flight was demonstrated successfully. The controller step responses were compared to those designed for in Chapter 4.1 and were found to be adequate.

Chapter 7

Conclusions and Recommendations

7.1 Conclusions

This thesis reported two years of work with the goal to achieve full autonomous flight of an unmanned helicopter. It was the first project at Stellenbosch University to achieve this goal. This project made use of a variety of integrated hardware developed within the UAV group:

- The avionics package. Also known as the PC104 stack consisting of the PC/104CAN controller board, OBC, RF and GPS adapter board, uBlox GPS and Aerocomm RF modules.
- The IMU stack consisting of sensors such as the rate gyros, accelerometers and magnetometer.
- The servoboard. Also known as the pilot interface. It passes either the control system or safety pilot commands to the servos.
- The HIL board. This was used during non-linear simulations and greatly reduced the risk of failure during practical testing.

For a more detailed analysis of the system architecture see Appendix A. The ESL's UAV group has an *object* system in place whereby software building blocks which are common to UAV projects are created. The goal of these building blocks is to speed up the progress of projects by providing researchers with functional software of which the inputs and outputs are well-defined.

This allows the researchers to use software without having to write the code, thereby minimising possible errors and saving valuable development time. Although the *object* system was still in the early stages of development at the start of this project the following *object* software was successfully integrated and used:

- The ground station software. Although significant changes were made to the ground station software in order to meet the requirements of this project, a great deal of it remained unchanged.
- The OBC software. The skeleton of the OBC software as developed by Hough [5] was used. A useful part of the OBC software was the general Extended Kalman Filter (EKF). The EKF provided this project with the necessary estimates for use in successive-loop-closure.
- MATLAB m-files used to create the *aircraft.dat* file. The *aircraft.dat* file contains a number of constants read in by both the ground station and OBC software. The creation of the *aircraft.dat* file ensured that the important constants used during non-linear simulations were automatically updated to the ground station and OBC software where it was used practically during flight tests or HIL simulations.
- MATLAB Simulink functional blocks. Except for the aircraft specific *X-Cell Helicopter Model* and designed *Simulated Control Algorithms* discussed in Section 4.2, all the functional blocks of the non-linear simulation environment as used by previous UAV projects in the ESL could be used.

A functional set of hardware and *object* code was crucial to the success of this project. It allowed this project to focus on the research, modeling, controller development, ground station development and non-linear simulation of the X-Cell helicopter.

7.2 Recommendations

Recommendations on how the current system can be extended and improved follow:

General

- The current system can be extended by enabling autonomous take-off and landing of the X-Cell helicopter. This should not be too difficult seeing that the X-Cell can already hover autonomously. However, testing the X-Cell close to the ground greatly increases the risk of a crash seeing that the safety pilot will have little or no time to take back control in the event of an emergency. This phase will have to be simulated thoroughly in HIL and test procedures similar to those in Roos's [6] project¹ will have to be followed whereby take-off and landing are tested on a virtual ground at a safe altitude first.
- The controllers were designed using successive-loop-closure with PI control on the outer position loops. This method was in step with most autonomous unmanned helicopter projects investigated [9, 14, 15, 24]. However, controllers can be designed and evaluated by moving PI control to the inner loops of attitude and velocity for the longitudinal/lateral and heading/heave controllers respectively. This will minimize the errors on the inner loops which should theoretically lead to a more stable system.
- The GPS antenna should be mounted on the *horizontal stabilizer* located on the end of the tail boom. Currently it is mounted on top of the noise housing of the servo interface. Here it is close to electronics and underneath the rotating blades. This would be an attempt to improve the *lock time* i.e. the time the GPS module takes to obtain a 3D Fix.
- It is strongly recommended for the researcher working on the automation of an unmanned helicopter to get into the sport of flying hobby RC helicopters. This will facilitate more intuitive controller design choices. The potential risks and limitations of the platform will be better understood. Flight tests can be done without having to call on the services of an external safety pilot, saving time and money.

Hardware

- Vibration damping is the main area of work that still needs attention on the platform. The previous project [9] relied on four vibration dampers

¹Roos achieved autonomous take-off and landing with a model RC airplane in his project.

to isolate the avionics box from the helicopter airframe. However, this did not isolate the vibration well enough leading to noisy sensor measurements. In this project a different approach was followed whereby the IMU stack was encapsulated with sponge. Although this improved the IMU sensor measurements, connections on the PC104 Stack vibrated loose on several occasions. All the connections were securely fastened with a glue gun but a long-term solution must be found.

- The PC104 Stack was originally designed for use in autonomous RC airplane projects where vibration is less of an issue. A more compact design in the form of a single PCB having the same functionality as the PC104 Stack is required on which the impact of vibration will be much less. At the time of writing such a system was being designed in the ESL.
- A more accurate GPS module would improve the estimator data and decrease the drift experienced by the X-Cell. Such a module would enable the X-Cell to hover within a small confined space, increasing the number of applications for which the platform could be utilised.
- Alternative height sensors should be investigated. The ultrasonic sensor measurement proved to deteriorate rapidly in the presence of vibration.
- A USB Bluetooth dongle could be added to the OBC. This will enable the wireless transfer of data between the OBC and the ground station laptop. At present a LAN cable is used for the purpose of data transfer. The current Linux kernel fully supports the Bluetooth standard. This was recommended by the previous project [9] but was never implemented due to time limitations.

Appendices

Appendix A

System Architecture

The hardware used in this project was developed by a number of different students for use by the Computer and Control group at Stellenbosch University and is summarized in Table A.1. The avionics system onboard the X-Cell was integrated and tested by Groenewald [9]. The system architecture is shown in Figure A.1. The avionics architecture developed has the

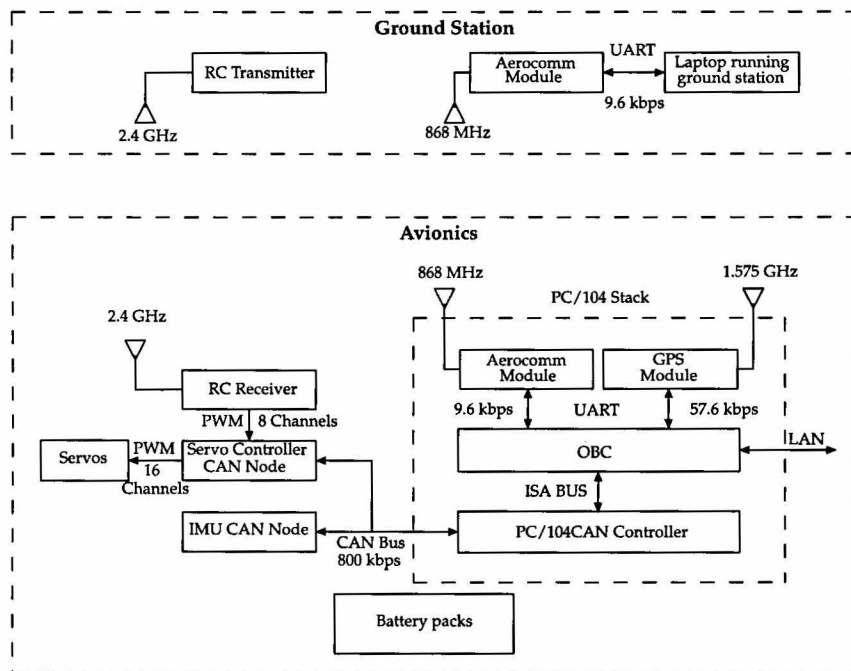


Figure A.1: System Architecture

Table A.1: Hardware components developed in the ESL

Hardware Component	Developed by
PC104/CAN Controller board	J. Venter [10]
GPS and RF link daughter board	J. Venter
Servo Controller CAN Node	J. Venter
IMU Generic sensor base board	S. Groenewald [9]
IMU sensor specific CAN Node	J. Bijker [11]

following characteristics:

- Both the *PC104/CAN Controller* and the *GPS and RF link daughter* boards were designed with the same dimensions as the PC104 computer so that the three boards could fit compactly on top of each other in a configuration known as the *PC104 stack*.
- The platform provides large amounts of processing power through the PC104+ 300 MHz Intel Pentium III processor.
- The CAN bus was used to create a modular design whereby the architecture could be easily extended. This property was exploited during HIL simulations where the HIL board was connected to the CAN bus and simulated sensor data was placed on the bus for use by the OBC.

The various components of the avionics will be discussed briefly in the following section.

A.1 PC104 Stack

A picture of the *PC104 Stack* is shown in Figure A.2. From the bottom, the three boards are listed:

- The Kontron MOPSIcd7 PC104+ OBC.
- The PC104/CAN Controller board.
- GPS and RF link daughter board.

The boards were separated by lightweight plastic standoffs. The external fans were configured to maintain the nominal operating temperature of the OBC at 40 degrees celsius.

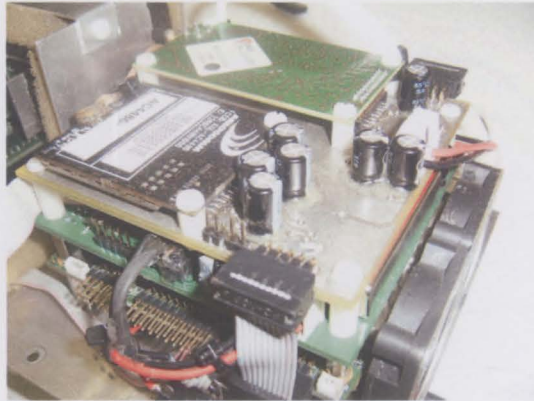


Figure A.2: PC104 Stack

A.1.1 OBC

The Kontron MOPSIcd7 PC104+ computer used in this project has the following specifications:

- 300 MHz Intel Pentium III processor
- Upgradeable 64 Mb RAM
- Upgradeable 32 Mb solid state IDE drive
- PCI bus
- ISA bus
- 2 x RS232 ports
- 1 Parallel port
- Ethernet
- 2 x USB ports
- Keyboard, mouse and VGA connectors

The main advantage of using the Intel processor is that it can perform floating point calculations efficiently. The OBC ran a reduced Linux kernel with the advantage that the operating system could be changed at any level. The entire operating system only occupied about 1.2 MB of the system's disk space.

A.1.2 PC104/CAN Controller board

The PC104/CAN Controller board has two main functions:

- Provides a data path between the OBC and the CAN bus. Data from the OBC is transferred to the *PC104/CAN Controller* board via the *Industry Standard Architecture (ISA) bus* of the OBC. This data is then packaged according to the CAN specification and placed onto the CAN bus. Similarly, when data is transferred from the *PC104/CAN Controller* board to the OBC, data from the CAN bus is packaged according to the ISA specification and placed onto the ISA bus of the OBC.
- Dictates the timing of the system. See Figure A.3 which illustrates the interaction between the *UCA* onboard the OBC and the *PC104/CAN Controller* board.

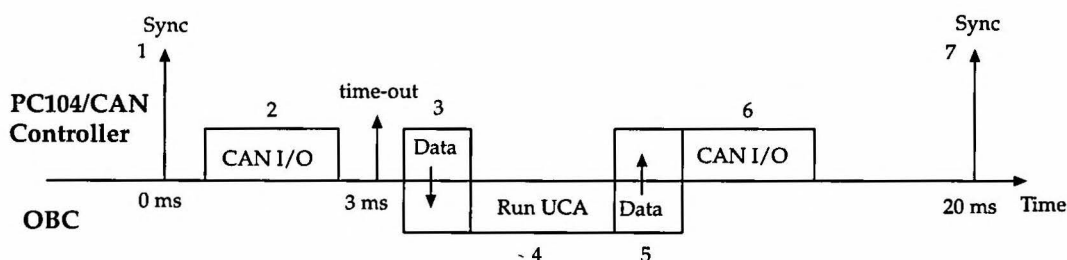


Figure A.3: OBC and PC104/CAN Controller Timing diagram [5]

The timing of the avionics package shown in Figure A.3 can be explained in seven steps,

1. The *PC104/CAN Controller* issues a *global time synchronisation packet* on the CAN bus. This triggers the following series of events:
 - The *sensor nodes* place their most recent measurements on the CAN bus.
 - The *servo controller* updates all actuator nodes on the CAN bus to the latest actuation levels calculated during the previous time-step.

- The safety pilot's commanded actuation levels are placed on the CAN bus for use by the OBC during partial arming of the controllers or where the safety pilot needs to have full control of the aircraft.
2. The CAN data is received by the *PC104/CAN Controller* board for a period of 3 ms after the *global time synchronisation packet* has been issued. A 3 ms time-out is imposed where after all CAN activity on the bus will be disregarded.
 3. The relevant CAN data is packaged and transferred to the OBC via the ISA bus.
 4. With the latest CAN data at its disposal, the User Control Application (UCA) gets invoked. See Figure A.4 which gives a brief summary of the UCA as well as the relevant C-code files associated with each step.
 5. The actuation levels calculated by the control algorithms along with any additional CAN packets are enqueued to the *PC104/CAN Controller* board.
 6. The data is transferred to the *PC104/CAN Controller* board via the ISA bus. The data is packaged and placed on the CAN bus. Another time-out is issued so that there is no bus activity in anticipation of the time synchronisation command.
 7. The *global time synchronisation packet* is issued and the process repeats itself.

A.1.3 GPS and RF Link Daughter board

The GPS and RF Link Daughter board has two main functions:

1. Provides interfaces for the u-Blox GPS receiver and the AeroComm RF transducer.
2. Provides a data path for both these devices to connect directly with the OBC via the RS232 ports.

The GPS receiver used was a u-Blox RCB-LJ OEM Receiver. An active antenna with 25 dB gain was used to receive information from the satellites. Although the u-Blox receiver and antenna configuration used in this project only provided us with around 4 meter accuracy, it was low-cost and adequate for use in demonstrating autonomous flight.

The RF transducer used was the AeroComm AC4486 RF Modem. This device operates at a frequency of 868 MHz and enables wireless RS232 communication.

A.2 IMU CAN Node

The IMU Node consists of two Printed Circuit Boards (PCBs) stacked on top of each other. A picture of the IMU is shown in Figure A.5. From the bottom, the two boards are listed:

- The generic sensor platform base board known as *CANSens*. This board digitizes the analog sensor measurements. The analog signals pass through the following steps before it is placed onto the CAN bus¹:
 1. Second order Butterworth filters suppress the high frequency noise.
 2. The filtered analog signals are sampled by 16-bit Analog to Digital Converters (ADCs) at 1kHz.
 3. Finally the signals are digitally low pass filtered down to a bandwidth of 8 Hz.
- The IMU sensor specific board. This board contains the following sensors:
 - 3 x ADXRS150 Single-Axis Rate Gyroscopes from Analog Devices
 - 2 x ADXL210E Dual-Axis Accelerometers from Analog Devices
 - 1 x HMC2003 Honeywell Magnetometer

The *rate gyroscopes* are capable of measuring angular rates of up to $\pm 150^\circ/s$. The *accelerometers* are capable of measuring accelerations up to $\pm 10g$. The *magnetometer* is an anisotropic magneto resistive (AMR)

¹For an in-depth analysis of the *CANSens* board see [9].

sensor which provides a vector measurement of the earth's magnetic field. The *magnetometer* is capable of measuring magnetic fields of up to ± 2 Gauss. However, since the maximum magnetic field in the vicinity of Stellenbosch is 0.26 Gauss, the sensor output was magnified by a factor four. The original IMU sensor specific board as used in [9] made use of *accelerometers* capable of only measuring $\pm 2g$. Although it was this project's aim to fly low bandwidth maneuvers, the new *high performance* accelerometers were chosen for two reasons:

1. They will be able to measure aggressive maneuvers for use in future projects.
2. The high noise levels during hover will not saturate the sensor measurements.

A.3 Servo Controller CAN Node

The *servo controller* board is shown in Figure A.6. The *servo controller* board's main functions are listed below:

- It allows either the autopilot or the safety pilot to command the helicopter's actuators.
- It provides critical safety features:
 - Backup battery logic: The *servo controller* board is powered off the main battery. However, in the event of main battery failure the board will automatically switch over to using the backup battery situated in the servo controller housing box.
 - Backup if CAN bus fails: In the event that the OBC shuts down for some arbitrary reason the *servo controller* board will automatically map through the actuator commands given by the safety pilot to the actuators.
 - Smooth transition algorithms: To prevent spikes in the commanded actuator levels the *servo controller* board uses smooth transition algorithms to allow a more gradual transition when switching the autopilot on or off. Smooth transition prevents the helicopter from making any sudden movements during the switch-over process which could potentially damage the actuators.

APPENDIX A. SYSTEM ARCHITECTURE

91

For a more in-depth analysis of the *servo controller* board see [5].

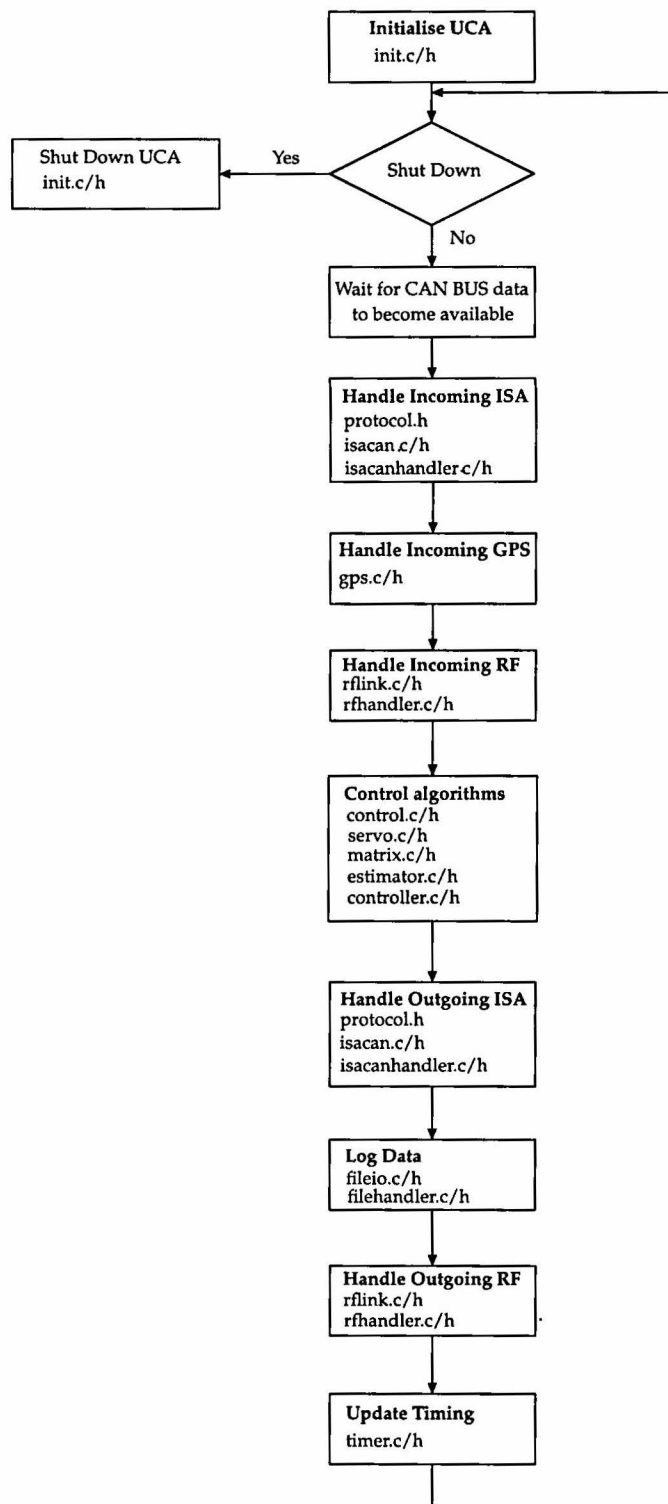


Figure A.4: UCA Flow Chart [5]



Figure A.5: Avionics system - PC104 Stack bottom right, IMU top left

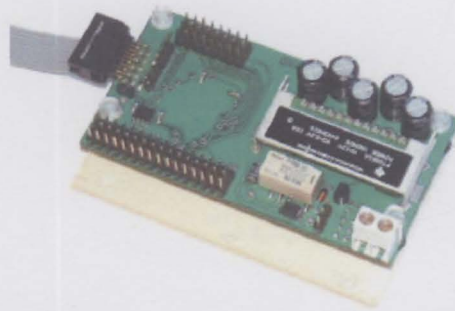


Figure A.6: Servo Controller board [5]



Figure A.7: X-Cell during an Autonomous hover



Figure A.8: Flight test team from the left: Kyle Davis (safety pilot), Carlo van Schalkwyk and the author.

Bibliography

- [1] G.F Franklin, J.D. Powell, M. Workman. *Digital Control of Dynamic Systems*. Addison Wesley Longman, 1998.
- [2] A. E. Bryson, Jr. *Control of Spacecraft and Aircraft*. Princeton University Press, 1994.
- [3] Cook MV *Flight Dynamics Principles*. Butterworth Heinemann, 1997.
- [4] I.K. Peddle. *Autonomous Flight of a Model Aircraft*. Masters dissertation, Stellenbosch University, 2005.
- [5] W.J. Hough. *Autonomous Aerobatic Flight of a Fixed Wing Unmanned Aerial Vehicle*. Masters dissertation, Stellenbosch University, 2006.
- [6] J.C. Roos. *Autonomous Take-Off and Landing of an Unmanned Aerial Vehicle*. Masters dissertation, Stellenbosch University, 2006.
- [7] J.H. Blacklock. *Automatic Control of Aircraft and Missiles, 2nd ed.* Wiley-Interscience, 1991.
- [8] Peter Luethi, Thomas Moser
<http://www.trash.net/~luethi/study/ins/ins.html>
- [9] S. Groenewald *Development of a Rotary-Wing Test Bed for Autonomous Flight*. Masters dissertation, Stellenbosch University, 2005.

- [10] J. Venter *Development of an experimental tilt-wing VTOL Unmanned Aerial Vehicle*. Masters dissertation, Stellenbosch University, 2006.
- [11] J. Bijker *Development of an attitude heading Reference System for an Air Ship*. Masters dissertation, Stellenbosch University, 2006.
- [12] NASA History Web page. <http://history.nasa.gov/SP-367/appendc.htm>. 2006.
- [13] P.H. Zipfel. *Modelling and Simulation of Aerospace Vehicle Dynamics*. AIAA Education Series, 2000.
- [14] N. Carstens. *Development of a Low-Cost, Low-Weight Flight Control System for an Electrically Powered Model Helicopter*. Masters dissertation, Stellenbosch University, 2005.
- [15] D.H. Shim *Hierarchical Flight Control System Synthesis for Rotorcraft-based Unmanned Aerial Vehicles*. PhD thesis, California University, Berkeley, March 2000.
- [16] C. Van Schalkwyk *Full state control of a Fury X-Cell Unmanned Helicopter*. Masters dissertation, Stellenbosch University, 2007.
- [17] V. Gavrilets, B. Mettler, E. Feron *Dynamic Model for X-Cell 60 Helicopter in Low Advance Ratio Flight* Laboratory of Information and Decision Systems Technical Report P-2543.
- [18] B. Mettler, M.B. Tischler, T. Kanade *System Identification of Small-Size Unmanned Helicopter Dynamics*. 55th Forum of the American Helicopter Society, Montreal, Canada, May 1999.
- [19] G.D. Padfield *Helicopter Flight Dynamics: The Theory and Application of Flying Qualities and Simulation Modeling*. AIAA Education Series, 1996.
- [20] B. Mettler, C. Dever, E. Feron *Scaling Effects and Dynamic Characteristics of Miniature Rotorcraft* Journal of Guidance, Control, and Dynamics. Vol. 27, No. 3, May-June 2004.

- [21] B. Mettler, C. Dever, E. Feron *Identification Modeling, Flying Qualities and Dynamic Scaling of Miniature Rotorcraft* Presented at the NATO SCI-120 Symposium on "Challenges in Dynamics, System Identification, Control and Handling Qualities For Land, Air, Sea and Space Vehicles. Berlin, Germany, May 2002.
- [22] Website: <http://www.control-systems.net>
- [23] V. Gavrillets, B. Mettler, E. Feron *Control Logic for Automated Aerobatic Flight of a Miniature Helicopter*. AIAA Guidance, Navigation, and Control Conference, Monterey, August 2002.
- [24] B. Mettler, T. Kanade, M.B. Tischler and W. Messner *Attitude Control Optimization for a Small-Scale Unmanned Helicopter* AIAA Guidance, Navigation and Control Conference, Denver, Colorado, 2000.
- [25] H.J. Kim, D.H. Shim, S. Sastry *A Flight Control System for Aerial Robots: Algorithms and Experiments*. Control Engineering Practice, April 2003.
- [26] F. Mooren *Practical Work Experience Period - HOPTUS Prestudy*. Delft University of Technology, 1996.
- [27] D.W. Venter *HOPTUS: A case study*. Universit of Stellenbosch, 1996.
- [28] Z. Dragan *RC Toys: Dragon Flyers*. <http://www.rctoys.com>