CRANFIELD UNIVERSITY

EMMANUEL BIJU JOSEPH

DIFFERENTIAL FLATNESS BASED FLIGHT CONTROL OF TILT-ROTOR VTOL AIRCRAFT

SCHOOL OF AEROSPACE, TRANSPORT AND MANUFACTURING
Master by Research in Aerospace

MSc
Academic Year: 2020 - 2021

Supervisor: Dr James F Whidborne
Associate Supervisor: Dr Lejun Chen
November 2021

CRANFIELD UNIVERSITY

SCHOOL OF AEROSPACE, TRANSPORT AND MANUFACTURING
Master by Research in Aerospace

MSc

Academic Year 2020 - 2021

EMMANUEL BIJU JOSEPH

Differential Flatness Based Flight Control of Tilt-Rotor VTOL Aircraft

Supervisor: Dr James F Whidborne
Associate Supervisor: Dr Lejun Chen
November 2021

© Cranfield University 2021. All rights reserved. No part of this publication may be reproduced without the written permission of the copyright owner.

ABSTRACT

Urban Air Mobility (UAM) is identified as the future of mobility, and several airframes are being trialled to fit air transport within modern cities. Speed and efficiency are the primary requirements deliberated for UAM Vertical Take-off and Landing (VTOL) aircraft, and configurations incorporating tilt-rotors and tilt-wings have been identified as viable solutions. For quadrotors and robotics, control techniques leveraging the differentially flat property of system dynamics have achieved significantly improved tracking performance and motivated the application of this control technique for VTOL aircraft. This study proposes a novel control architecture based on the differential flatness property of aircraft kinematics, to improve controller performance for autonomous flight. A conceptual validation of the proposed technique was implemented on a Planar VTOL (PVTOL) model as a precursor to the research. Consequently, an architecture for implementing the concept in tilt-rotor VTOL aircraft (Aston Martin Vision Volante) was devised, implemented, and tested. The control architecture achieved the speculated performance improvements and lays foundation for further research and development into the proposed control technique.

Keywords:

UAM, Control Allocation, Control Architecture

ACKNOWLEDGEMENTS

I would like to thank my supervisor, Dr James F Whidborne, for the excellent guidance, constant encouragement, and relentless support that he provided throughout my time as his student. I have been extremely lucky to have a supervisor who cared so much about my work, aspirations and wellbeing. I owe it to him for making this research a memorable one.

I thank Dr Lejun Chen for his supervision and guidance at times of struggle and for his patient assistance. I would also like to mention Dr Julien Enconniere and Namuk Kang for rendering their helping hands during the course of my research.

My deep and sincere gratitude to my family for their continuous and unparalleled love, help and support. I am forever indebted to my parents for giving me the opportunities and experiences that have made me who I am. I express my indebtedness to all my friends for helping me keep things in perspective and providing me the necessary escape from my studies.

Finally, I would like to thank Cranfield University, not only for providing the opportunity to undertake this research, but also for giving me the opportunity to meet so many interesting people and develop as a flight control engineer.

TABLE OF CONTENTS

ABSTRACT i
ACKNOWLEDGEMENTS ii
LIST OF FIGURES v
LIST OF TABLES vii
LIST OF NOTATIONS viii
LIST OF ABBREVIATIONS ix
1 INTRODUCTION 1
1.1 Overview 1
1.2 Objectives 3
1.3 Scope and Contributions 3
1.4 Thesis Outline 4
2 LITERATURE REVIEW 5
2.1 Differential Flatness 5
2.2 Control Allocation 10
3 CONCEPT VALIDATION 15
3.1 PVTOL Aircraft 15
3.2 PVTOL Differential Flatness Control 18
3.2.1 Differentially Flat Equations 18
3.2.2 Controller Architecture 20
3.2.3 Feedback Controller 21
3.3 Non-linear Simulation 26
3.4 Conclusion 29
4 TILT-ROTOR FLIGHT DYNAMICS MODELLING 30
4.1 Aston Martin Vision Volante 31
4.2 Flight Dynamics Model 32
4.2.1 Coordinate Reference Frames 32
4.2.2 Architecture 34
4.2.3 Aerodynamics 37
4.2.4 Propeller Dynamics 44
4.2.5 Integration 47
4.3 Model Validation 47
4.4 Conclusion 51
5 DIFFERENTIAL FLATNESS CONTROLLER DESIGN 52
5.1 Differentially Flat Equations 52
5.2 Control Architecture 55
5.3 Control Allocation 58
5.3.1 Linear Control Allocation 60
5.3.2 Non-linear Control Allocation 66
5.3.3 Tilt-rotor angle Scheduling 68
5.4 Dynamics Linearisation 69

5.5 Conclusion 72
6 DISTURBANCE CONTROLLER DEVELOPMENT 74
6.1 Controller Structure 75
6.2 Controller Development 77
6.2.1 Requirements 77
6.2.2 Hover Controller 78
6.2.3 Cruise Controller 80
6.3 Controller Evaluation 86
6.3.1 Module Integration 86
6.3.2 Non-linear Simulation 89
6.4 Conclusion 100
7 CONCLUSION 102
7.1 Future Work 103
7.2 Potential Impact 105
REFERENCES 106
APPENDICES 111
Appendix A Conceptual Validation 111
Appendix B Control Allocation 112
Appendix C Controller Implementation 113

LIST OF FIGURES

Figure 2-1 Trajectory Generator and Trajectory Tracking Controller for fixed wing aircraft [10] 8
Figure 3-1 Planar VTOL Model 16
Figure 3-2 PVTOL with Tilting Thrust 17
Figure 3-3 Differential Flatness Control Architecture for PVTOL 21
Figure 3-4 Feedback Linear Controller Structure 22
Figure 3-5 Step Response – PVTOL Longitudinal Closed Loop Dynamics 23
Figure 3-6 Step Response – PVTOL Closed Loop Linear Velocity Dynamics .. 25
Figure 3-7 Bode Plot - Gain and Phase margins of Velocity Controller 26
Figure 3-8 Non-linear Simulation - PVTOL Differential Flatness Controller 27
Figure 3-9 Non-Linear Simulation - PVTOL Controller (Disturbance Rejection) 28
Figure 4-1 Aston Martin Vision Volante [29] 31
Figure 4-2 Body and Propeller Reference Frame 34
Figure 4-3 Flight Dynamics Model Architecture (Top Level) 35
Figure 4-4 Flight Dynamics Model Architecture 37
Figure 4-5 Post-Stall Characteristics [35] 41
Figure 4-6 Lift Coefficient Post-Stall Approximation 41
Figure 4-7 Hover Phase Trim 48
Figure 4-8 Cruise Phase Trim 50
Figure 5-1 Airframe Forces Representation 54
Figure 5-2 Differential Flatness Flight Control Architecture 56
Figure 5-3 Tilt rotor angle (Γ) Scheduling 69
Figure 5-4 Trim Model Structure 71
Figure 6-1 Linear Disturbance Controllers – Structure 75
Figure 6-2 Tuning Requirement - Pitch Rate Step Response 77
Figure 6-3 Tuning Requirement - Velocities (*u,w*) Step Response 78
Figure 6-4 Step Response - Hover Controller (Linear Dynamics) 79
Figure 6-5 Pole Zero Map - Hover Controller (Linear Dynamics) 80

Figure 6-6 Parametrised Cruise Controller Gains 82
Figure 6-7 Step Response - Cruise Controllers 83
Figure 6-8 Pole Zero Map - Cruise Controller 84
Figure 6-9 Bode Plot - Cruise Controllers..... 85
Figure 6-10 Control Input Dynamic Saturation Limits 87
Figure 6-11 Control Allocator Implementation..... 88
Figure 6-12 Hover Test Control Demand..... 90
Figure 6-13 Hover Flight Test 91
Figure 6-14 Hover Flight Test without Differential Flatness 92
Figure 6-15 Cruise Test Input Demands..... 94
Figure 6-16 Cruise Flight Test..... 95
Figure 6-17 Cruise Flight Test without Differential Flatness 96
Figure 6-18 Transition Flight Test..... 98
Figure 6-19 Transition Flight Test without Differential Flatness..... 99

LIST OF TABLES

Table 4-1 Airframe Operational and Geometric Characteristics[31]; [32]..... 32

LIST OF NOTATIONS

$\boldsymbol{u}$	Control input
$\boldsymbol{v}$	Virtual Control
$U\ V\ W$	Body frame velocities along x, y, z axes
$\alpha\ \beta$	Aerodynamics angles, i.e., Angle of Attack and Sideslip Angle
$\phi\ \theta\ \psi$	Body frame attitude angles – Roll, pitch and yaw angles
Γ	Tilt-rotor actuation angle w.r.t horizontal
g	Gravitational acceleration
m	Mass
δ_{surf}	Aerodynamic control surface deflection input ('surf' replaced with any control surface name)
ρ	Air density
V_T	Freestream Velocity
$S_{ref}\ b_{ref}\ c_{ref}$	Aerodynamic reference area and lengths

LIST OF ABBREVIATIONS

UAM	Urban Air Mobility
VTOL	Vertical Take-off Landing
PVTOL	Planar Vertical Take-off Landing
UAV	Unmanned Aerial Vehicle
INDI	Incremental Nonlinear Dynamic Inversion
PID	Proportional Integral Derivative
PI	Proportional Integral
FMS	Flight Management System
FCS	Flight Control System
CA	Control Allocation
AVL	Athena Vortex Lattice
CG	Centre of Gravity
AOA	Angle of Attack
CFD	Computational Fluid Dynamics
OBDM	Onboard Dynamics Model

1 INTRODUCTION

1.1 Overview

Air travel has enabled humanity to travel large distances in a short period and today the world economy has grown to be deeply reliant on travel. A country's transportation infrastructure is seen as a measure of its development and sign of eminence. While various technological advancements were made in road and rail transportation, there have not been many disruptive technologies in aerospace that have redefined and rethought air travel.

For a long time, air travel has been regarded as a means of intercontinental and regional travel, confining both airframe and propulsion technologies to enable high-density passenger travel across long distances. But, today Urban Air Mobility (UAM) has gathered large interest, posing great capacity towards addressing the growing transportation demands of the future generations, unlocking and tapping into a completely different transport market. The first step to actualise UAM is to develop an aircraft that fits within its requirements and is potentially a disruptive technology.

Vertical Take-off Landing (VTOL) aircraft have been studied and built since the advent of flight. However, technological restrictions, complex dynamics and inefficiency were some of the few limitations of VTOL aircraft (excluding rotorcrafts), which regarded them to be impractical and uncommon. With the emergence of electric propulsion, advanced computer-based development methods, and better material technology, VTOL aircraft are now considered viable and efficient, all while meeting the requirements of UAM.

UAM VTOL aircraft are mainly classified into three types; Multi-copter, Lift+Cruise and Thrust Vectored. Multi-copters generate lift with fixed rotors (similar to hovercraft) and achieve linear or lateral velocities by varying aircraft attitudes. Multi-copters are considered viable for short distances but have limitations in terms of cruise speed and high-speed efficiency. The Lift+Cruise configuration operates with separate thrust sources for inducing lift and achieving cruise. While this configuration is relatively efficient and dynamically better, the configuration is deemed to be structurally heavier with redundant thrust sources. Thrust vectored

configurations are primarily of two types: tilt-rotors and tilt-wings. In this configuration, vectored thrust sources enable achieving vertical lift like a rotorcraft, after which it transitions to a fixed-wing aircraft maintaining flight with aerodynamic lift and linearly acting thrust. This configuration is regarded as the most efficient and viable, despite its technological complexity [1].

In the perspective of flight dynamics and control, thrust vectored airframe configurations comprise several complications due to its convoluted dynamics, especially when transitioning between hover and cruise flight phases. The transition manoeuvre with largely varying nonlinear dynamics require control techniques different from conventional aircraft. Additionally, thrust vectoring creates an over-actuated system that requires proper utilisation of the different control effectors based on its effectiveness.

Over-actuation, complex dynamics and multiple flight phases are some of the several challenges in developing, implementing and certifying flight control systems for these aircraft. While conventional linear controllers have been well established in the domain of flight control, the control problem of these aircraft pose limitations within these standard techniques. With advancements in computation and development methods, there are several control concepts being studied that would eventually be established as a standard for such complex airframes.

This research focuses on designing, developing and validating a feasible flight controller for a tilt-rotor aircraft. The study is construed on the well-understood property of system dynamics, named differential flatness. This property has been very successfully utilised in the domain of robotic and multirotor control, and this study aims to devise a control scheme that can implement the same in the context of a certifiable flight controller. The research intends to devise a feasible architecture, and implement on a model-based simulation environment to assess and test the practicality of the proposed architecture. This study anticipates laying the foundation for further work in differential flatness based flight controls.

1.2 Objectives

The research project aims to conceptualise and test a differential flatness based flight control system for a tilt-rotor VTOL aircraft. A viable controller architecture and its implementation would be demonstrated and tested through simulation tests in SIMULINK. The objectives of the project are,

- Conceptual validation of the differential flatness-based control technique
- Tilt-rotor flight dynamics modelling
- Design and development of differential flatness-based longitudinal controller
- Implementation and test of differential flatness-based longitudinal controller

1.3 Scope and Contributions

Tilt-rotor aircraft are understood to have very different dynamics and control requirements when compared to fixed-wing aircraft. Linear controllers or various other control techniques can be employed to achieve reliable and certifiable tilt-rotor flight controllers. This research intends to devise, develop and test flight controllers based on differential-flatness, which is expected to achieve better tracking performance.

The first phase of the project identifies the feasibility of the control technique by implementing the control concept based on a preliminary architecture for a Planar Vertical Take-off Landing (PVTOL) dynamics model. The PVTOL implementation of the conceptualised technique is intended to develop critical understanding of the feasibility of implementing such a controller to improve performance and operational efficiency in VTOL aircraft.

Subsequently, a flight dynamics model is developed as a test bed for controller development and implementation. The chosen airframe, Aston Martin Vision Volante tilt-rotor luxury VTOL aircraft was conceptualised by Aston Martin in partnership with Rolls Royce, Cranfield University and Cranfield Aerospace Solutions. The aircraft is a three passenger aircraft consisting of two forward coaxial tilt-rotors and one coaxial aft rotor. The dynamics model is obtained and

modified with several assumptions to simplify the research problem, all while ensuring the critical elements of flight physics are included.

Following dynamics modelling, an implementable differential flatness-based flight controller architecture is formulated and the required modules to facilitate its implementation are studied. The dynamics model fidelity is limited to the propeller and aerodynamic assumptions and data, which could represent a deviation from real flight dynamics. Control allocation schemes and control strategies are reviewed, assessed and developed. The final stage involves the integration of all the modules within the architecture for simulation based non-linear testing to assess feasibility and performance of the proposed concept in tilt-rotor VTOL flight control.

The study intends to contribute towards advancing and introducing a novel control approach for UAM. Differential flatness has not been implemented in tilt-rotor flight control, and this paves way for understanding the proposed control technique, providing insights to direct the further work that should follow. The project scope is limited to assessing the feasibility and performance of the proposed architecture in simulation. Thus, hardware implementation and flight tests are excluded in this study.

1.4 Thesis Outline

The thesis comprises of 7 chapters. Chapter 1 introduces the project with the research problem, project objective and the scope of research. Chapter 2 elucidates briefly the literature study carried out by the author to understand the control concepts involved. Chapter 3 describes conceptual validation of the proposed control technique for a PVTOL dynamics model. Chapter 4 details the flight dynamics modelling of the Vision Volante, based on which the controller development was performed. Chapter 5 explains the formulation of the differential flatness-based control architecture and the control allocation schemes utilised. Chapter 6 details the development and implementation of the disturbance controllers, and simulation-based performance evaluations in SIMULINK. Chapter 7 finally concludes the thesis with final remarks, future work and potential impact.

2 LITERATURE REVIEW

Differential flatness is a system dynamics concept that has been implemented in the control of various systems, including flight guidance. Upon review, differential flatness has not been considered for UAM flight control against the various scheduled and unified control approaches like linear parameter varying, dynamic inversion or model predictive control techniques [2]. The devising of a novel architecture that leverages the concept of differential flatness in flight control was recognised as an opportunity to inspire a new direction in the field of flight control. The author reviewed several documents that illustrate the various concepts and implementations of differential flatness. With the devised control architecture, control allocation was yet another aspect the author studied to meet the project objectives. This chapter documents the key material that inspired and contributed to formulating the problem statement and the designed controller architecture.

2.1 Differential Flatness

Essentially, control theory is based on algorithms to achieve control over a system state by governing its inputs. Some of the modern control techniques involve system dynamic inversion to compute control inputs based on the modelled dynamic characteristics. However, the process of inverting entire dynamics is computationally expensive and is subject to trade-offs [3].

Fliess *et al* [4] introduced the concept of differential flatness using the formalism of differential algebra. According to the concept, a system is said to be flat if a set of variables (flat outputs) can be obtained, such that the system is algebraic over the differential field generated by the set of flat outputs, i.e., a system states and inputs can be determined from a specific set of outputs without integration[3]; [5]; [6]. In [6], the author explains the two types of flatness: explicit and implicit flatness where it differs based on whether the causal relationships are introduced analytically or through implicit functions respectively. In flight control, flatness enables reformulating the trajectory tracking problem as a state tracking problem, making it possible to consider higher-order derivatives of a reference trajectory in the form of states and inputs, which is of relevance in autonomous flight control

[7]. Differential flat systems can be proven beneficial in situations where explicit trajectory generation is possible, and with a planned trajectory in place, the appropriate control inputs can be computed [8].

More precisely, if a system has states $x \in \mathbb{R}^n$, and inputs $u \in \mathbb{R}^m$ then the system is flat if output $y \in \mathbb{R}^m$ is of the form [3]; [5]; [6]

$$y = h\left(x, u, \dot{u}, \dots, u^{(r)}\right)$$

such that,

$$x = f(y, \dot{y}, \dots, y^{(q)})$$

$$u = g(y, \dot{y}, \dots, y^{(q)})$$

Many classes of systems commonly used in nonlinear control are flat systems and all controllable linear systems can also be shown to be flat [3]. Also, systems that can be transformed into a linear system by coordinate changes or feedback transformations (static or dynamic) can also be termed flat [3]; [8]. According to [3], many systems with strong nonlinear control techniques can be expressed as flat systems. A flat system does not imply transformation of the system with feedback or conversion of a nonlinear system to linear, rather is an implication of the nonlinear system dynamics being well characterised [3]. This proper characterisation of the nonlinear nature of a system, when exploited to compute system inputs for control, becomes a powerful technique to implement amongst control systems, motion planning, trajectory generation and stabilisation [3]; [8].

Through various examples, the authors of [3] provide a generalised proof that the flat output can serve as a first virtual control in a backstepping process. This generalisation arises from the proof that the flat dynamics can be identified as endogenous feedback acting on the flat output. This implies that the knowledge of flat output is useful for both trajectory generation and tracking. In [3], the authors provide several systems that portray the flatness property, this includes PVTOL vehicles which is an important source of inspiration for the author's work.

Richard Murray in [8] elucidates on the concept of using feedforward compensators in control system design, and leverages differential flatness for generating feasible trajectories. He outlines the two degree of freedom controller design approach that focuses on feedback and feedforward compensator design. In this technique, the feedforward compensator computes the nominal inputs required for reference tracking, while the feedback compensator corrects the errors [8]. Murray outlines the control problem into trajectory generation and trajectory tracking, where based on the nonlinear dynamic representation of the system, a reference trajectory is defined and the system with its gain scheduled linear error controllers are expected to track the trajectory. An example of its implementation was depicted in steering control of an automobile and introduces the concept of generating flat outputs based on the desired trajectory, as per the system dynamics. While this technique is quite relevant for pre-planned trajectory tracking, it was observed to be difficult in flight control. Nevertheless, Murray in [8] provides a flat output representation of a vectored thrust aircraft which along with the controller design approach was instrumental in this study.

Whidborne and Cowling's implementation of a trajectory generator in [9] and [10], based on differential flatness, coupled with linear controllers, operate to precisely track quadrotors between two points. The implementation comprised of a trajectory planner to produce trajectories and a trajectory following controller (feedback controller) to compensate for deviations in the presence of uncertainties and unmodeled dynamics. The trajectory generator when input with mission demands, computed optimal trajectories utilising the differential flat dynamics, which then computed reference states and inputs that were feedforwarded alongside the feedback controller input demands. The reference inputs and states were observed to be accurate due to the resemblance of flat dynamics and full state model, enabling improved trajectory tracking [9]; [10]. A representation of the inverted differentially flat equations of PVTOL and fixed wing aircraft are provided in [10], where a controller architecture using the similar UAV control structure for a fixed wing aircraft (see Figure 2-1) is proposed. The given implementation of differential flatness was fundamental to the author's controller architecture development.

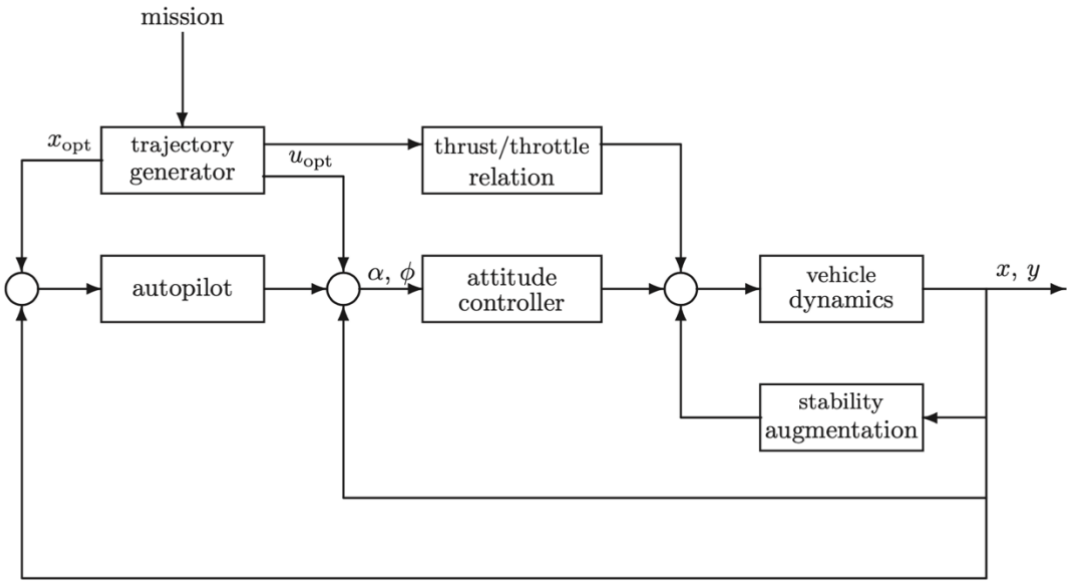


Figure 2-1 Trajectory Generator and Trajectory Tracking Controller for fixed wing aircraft [10]

Differential flatness is a control concept that is of keen interest to the drone racing community due to its performance improvements in tracking when paired with an appropriate control technique. An interesting study by Ezra and Sertac [7] implements quadrotor control with improved tracking performance using differential flatness and Incremental Nonlinear Dynamic Inversion (INDI). In their implementation, the differential flatness unit generated feedforward control inputs, which along with the INDI controller enables accurate tracking even without any modelled aerodynamic drag or external disturbance, indicating robust performance characteristics in practical implementations. Reformulating the trajectory tracking problem as a state problem [7] using flat outputs and its derivatives enables control of the higher-order derivatives; jerk and snap tracking. Another study at University of Cincinnati [11] implemented differential flatness within a similar feedforward architecture for a tilting rotor quadrotor. The trajectory input demands were assumed to be differentiable, and the linear PD controllers were feedforwarded with the second-derivative position inputs from the differential flatness equations. The rotor tilt control was also designed to be influenced by the differential flatness inputs with a tuned gain, synchronising the rotor-tilting based on the demanded angular speeds. The proposed architecture

was implemented and visible performance improvements were documented in [11]. Similar to mentioned research, differential flatness based control techniques in quadrotors have been widely studied, and most implementations were proven to achieve tracking performance improvements [5]; [7]; [11]; [12]. All these studies propose control architectures which utilise a feedback controller to be informed with feedforward or additional loop based on differentially flat dynamics estimating the desired trajectory control inputs. However, it is important to mention that the quadrotor dynamics in itself is differentially flat, making the flat outputs a good enough method to predict control inputs, providing the significant performance improvements, especially when coupled with robust feedback controllers to deal with the uncertainties.

For fixed-wing aircraft, in [6], the differential flatness property of flight dynamics is utilised for flight guidance. The study assumes a traditional passenger aircraft with an inner loop flight control system. The architecture intended the Flight Management System (FMS) to compute the desired trajectory, for which the differential flatness controller would compute control inputs to the inner FCS. The guidance controller used a trained multi-layer neural network to solve for the control input demands. The proposed controller was implemented in simulation, and was seen to be capable of accurate trajectory tracking. This study clearly expressed the flat output representation of fixed wing flight dynamics, where the inputs are a function of state differentials. While several studies indicate the differential flatness property of flight dynamics, [6] illustrated the potential application of the property in modern control techniques and mildly suggested its viability in fixed wing flight control.

It is important to mention that there hasn’t been any significant research carried out in differential flatness based flight control for VTOL aircraft, however, [13]; [14]; [15] explores control of the non-minimum phase characteristics of PVTOL dynamics utilising the flatness property. In [13] , the tracking problem was split across a state tracker and trajectory generator, where the state tracker computed inputs for desired states based on the differential trajectory states from the bounded state trajectory generator. The control scheme was compared with

conventional linearised dynamic controllers, and was proven to perform better as it does not approximate the nonminimum phase system by minimum phase system [13]. For a flat system, defining the state tracker is simple, and controlling the input states would simplify the control problem to good reference trajectory generation, which is an open-loop problem (see [3]; [8]; [10]). [15] describes another control technique that achieves non-minimum phase control by exploiting the differential parametrisation of system states, i.e., outputs and control inputs in terms of flat outputs which obtains the required control inputs for equilibrium. While [13]; [14]; [15] explored PVTOL control based on differential flatness, these studies were focussed on exploiting the PVTOL flat dynamics to propose control over non-minimum phase dynamics, and barely explored its viability in VTOL flight control.

This review of differential flatness is aimed at building an understanding of the existing implementations of control techniques utilising the concept in order to devise a controller scheme that is viable for VTOL flight control. [5]; [7]; [9]; [10]; [11]; [12]; [16]; [17] discussed various approaches of implementing the control concept for quadrotor control, and all claim of tracking performance improvements with incorporating differential flatness. However, a concern for the given study was that unlike quadrotor, the tilt-rotor VTOL dynamics is not differentially flat. [6] and [10] give insights into flat output representation of fixed wing flight dynamics, which suggested the possibility of incorporating the technique within tilt-rotor cruise phase. This review also highlighted that there has not been any relevant tilt-rotor VTOL flight controller development that exploits differential flatness, making this study a novel approach to tilt-rotor flight control. Realising the potential of differential flatness in improving tracking performance, along with quadrotor and fixed wing dynamics being differentially flat, this study lays foundation for new areas of inquiry for flight control development.

2.2 Control Allocation

With modern aircraft being over actuated for the purpose of improving controllability and performance characteristics, Control Allocation (CA) has been a keen topic of interest for control engineers [18]. A methodology to conveniently

actuate the redundant effectors optimally for performance is a challenge in itself, and control allocation has been studied intensively over the years [18]. The control allocation problem can be defined as the resolution of the control effector inputs for desired virtual control demands based on the control effectiveness matrix that characterises the authority of each effector [19], i.e. $B\boldsymbol{u} = \boldsymbol{v}$ where B is effectiveness matrix, $\boldsymbol{v}$ is desired demands and $\boldsymbol{u}$ control effector inputs. According to [19], CA can be broadly classified into two categories: Online and offline control allocation.

Online solutions are those calculated in real time, whereby the control effectiveness matrix ***B*** and the effector limits are based on instantaneous flight conditions. The online computation eliminates ambiguity of control effectiveness variations with varying flight condition, which usually is a great deal of concern for aircraft as it traverses through different operating points within its flight envelope. Additionally, it provisions for dealing with effector failure and reconfiguration. With the benefits, it comes with added overheads of optimisation algorithms and onboard dynamic models [19] which require high computational capability to process in real time flight. In contrast, offline solutions are simple and computationally inexpensive methods that involve precomputed dataset or algorithms, but often require foreknowledge of at least the control effectiveness matrix ***B*** relevant across the envelope. The author reviewed different implementation techniques of control allocation in VTOL flight control to conclude on a technique that is viable for the proposed control architecture.

A practical study of control allocation for a tilting quadrotor fixed-wing aircraft in [20] utilises an offline control allocation technique to allot the control surface deflections, tilt-rotor angles and rotor speeds. Due to computational limitations of the Pixhawk open-source autopilot [20], the study was confined to an offline technique and further simplified the allocation problem by ganging, i.e., merging common control effectors for reduced control effector inputs (usually in lateral dynamics). The authors opted for a daisy chaining control allocation approach [19], where the inputs are computed in a stepped manner. First, the aerodynamic surface deflections are computed and checked whether the desired demands can

be met. If saturated, the residual torques and thrusts are then attained by tilt mechanism and rotor speeds. The order of the daisy chaining approach assumed that the control surface actuations are more efficient than electric motors. The ineffectiveness of the aerosurfaces at low speeds (hover) were then resolved by multiplying deflection with a weighting function of dynamic pressure. While the allocation technique functions satisfactorily and is simple in implementation, it is highly dependent on the linearised equations that resolve for the control deflections and tilting angles requiring the flight dynamics to be within the represented limits.

While daisy chaining involves a cascaded resolution of control inputs by saturation, pseudo inverse method is a linear optimal unconstrained solution that is widely implemented in VTOLs and UAVs. Optimal control allocation involves minimising a cost function, and conventionally a quadratic norm is used [21]. The analytic approach of pseudoinverse enables fast computation and supports response shaping using weights. As described in [21], a typical approach to use the 2-norm gives the cost function,

$$\min_{u} \frac{1}{2}\boldsymbol{u}^T\boldsymbol{u} \qquad \text{subject to} \qquad \boldsymbol{v}_c - B\boldsymbol{u} = \mathbf{0}$$

and its analytical solution is given by

$$\boldsymbol{u} = B^T(BB^T)^{-1}\boldsymbol{v}$$

The author in [21] implemented a control system for a dual axis tilt rotorcraft UAV exploiting the weighted pseudo inverse technique. The study carried out extensive modelling of the rotorcraft dynamics, which formed a strong basis for defining the UAV's control effectiveness matrix. The weights of the pseudo inverse algorithm were tuned to alter the relation between motor actuation and gyroscopic dynamics, in order to meet the expected pitching response [21]. The weight tuning was carried out using a full nonlinear model to completely analyse the response variations. A conventional PD controller that generates forces and moment demands were implemented, and the system indicated improvements in performance and robustness [21].

The X35B implemented CA [22] based on a cascaded pseudo inverse algorithm which was exceptional. First, an onboard dynamic model computed control effectiveness, saturation limits, rate limits and weightings which was then used by an unconstrained algorithm to solve for the desired acceleration demands. In the occurrence of effector saturation, the saturated effector is removed from the active effector set, and the loop recomputes for effector inputs [22]. This is carried out iteratively until the demand is met. While this method cannot be considered optimal, it was efficient given its throughput and storage constraints. The key to the algorithm was however its high-fidelity onboard dynamics model of aerodynamic and propulsive characteristics, that informed the algorithm of the true nature of the control effectiveness across the flight envelope.

Hybrid implementations of pseudo inverse CA aided with an independent algorithm to aid the computation of a specific set of control inputs has been proven viable in [23]; [24]. In [23], the tilting hexacopter control allocation was designed based on pseudo inverse alongside a search strategy that would optimize the shared tilting angle. The structure required the pseudo inverse to compute the control inputs, after which the remaining control demands were met using the search algorithm. The key advantage was the ability to set actuator limits, whereby the optimisation algorithm ensured that the tilting angle was computed such that the pseudo inverse demands would be within the attainable demand region. Whereas in [24], the tilt-rotor aircraft control allocation is implemented with a pseudo inverse algorithm that computes for rotor speeds alongside a yaw moment algorithm that computes the tilting angle increments. While the author indicates compromise in yaw authority in transition, the short transition durations avoided practical issues with experimental tests, indicating the validity of the approach.

This review on control allocation primarily focussed on the pseudo inverse control allocation technique, due to its viability in practical implementation and flexibility to operate within different architectures as shown in [18]; [21]; [22]; [24]. As suggested commonly across literature, a key requisite for implementation of the technique is deriving the control effectiveness matrix, and a straightforward

approach has been detailed in [21]; [24]; [25]; [26]. While pseudo inverse assumes linearity and excludes saturation constraints, it has benefits of being analytical, optimal, reconfigurable and tuneable [19]; [21].

3 CONCEPT VALIDATION

Differential flatness conceptually establishes the fact that for a flat system, system inputs can be obtained with known state differentials and outputs. A system is said to be differentially flat if the system outputs can be represented with a smooth function of the state and inputs, where such outputs are called flat outputs [3]; [5]. These flat outputs enable obtaining system input commands from desired states and outputs.

Although differential flatness control techniques have been implemented for several applications including quadrotors, such a control method has not been implemented for tilt-rotor VTOL aircraft. The project was initiated with the conceptual understanding of differential flatness, and the feasibility of achieving tilt-rotor flight control had to be validated with a rudimentary outline of the control problem. Consequently, the concept was validated on a Planar Vertical Takeoff Landing (PVTOL) model which establishes the fundamental VTOL kinematic characteristics upon which a simple differential flatness controller was implemented to prove the functionality of the proposed control concept which would then be scaled for a tilt-rotor VTOL airframe.

3.1 PVTOL Aircraft

Tilt-rotor VTOL aircraft are dynamically complex, with non-linear aerodynamics and propeller dynamics requiring extensive modelling efforts to model its flight characteristics. The beginning of the research had to first establish the viability of the proposed control technique, and this required a flight dynamics model that could closely represent the tilt-rotor VTOL flight dynamics without considering its aerodynamics or propeller dynamics.

PVTOL is a widely used dynamics model used to represent aircraft that are capable of vertical take-off, like rotorcrafts and special thrust vectored aircraft[10]; [14]; [27]. The mathematical model represents the dynamics of an aircraft constrained within the vertical plane, i.e., with respect to the x-z plane. It has three degrees of freedom corresponding to its position and orientation in the plane.

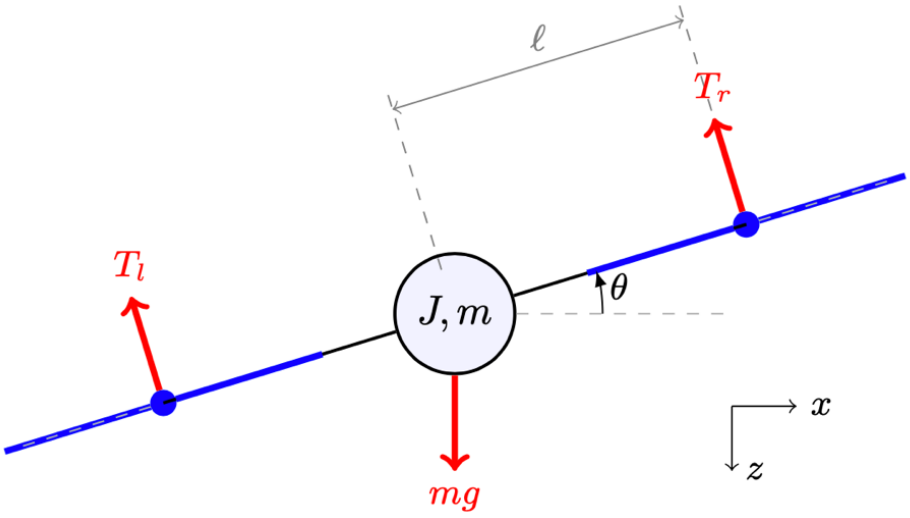


Figure 3-1 Planar VTOL Model

Accordingly, the dynamical model of the PVTOL aircraft can be defined based on the Lagrangian approach or Newton's laws. The equations of motion in the inertial frame of reference can be given by,

$$m\ddot{x} = -\sin\theta\, U_1$$
$$m\ddot{z} = -\cos\theta\, U_1 + mg$$
$$J\ddot{\theta} = -U_2 l$$

where x is the horizontal displacement, z is the vertical displacement, θ is the angle the PVTOL makes with the horizontal line, m is the airframe mass, l is the moment arm length from thrust to centre, and J is the inertia.

Generally, PVTOL models assume independent thrusters that produce the required force and moment during flight as shown in Figure 3-1. The thrust inputs are classified into two – collective and differential. The collective thrust is responsible for producing forces in the forward and upward direction, whereas the differential thrust produces the torque required to vary the pitch attitude. The control inputs are given by

$$U_1 = T_l + T_r \qquad\qquad U_2 = T_l - T_r$$

where T_l and T_r are left and right thrust forces respectively

Since most rate measurements are obtained in the body frame, and because controller development is more convenient in the aircraft body frame, the body frame equations of motion were obtained as,

$$\dot{U} = -g\sin\theta - qW \qquad (3.1)$$

$$\dot{W} = g\cos\theta - \frac{U_1}{m} + qU \qquad (3.2)$$

$$\dot{q} = \frac{U_2 l}{J} \qquad (3.3)$$

$$\dot{x} = U\cos\theta + W\sin\theta \qquad (3.4)$$

$$\dot{z} = W\cos\theta - U\sin\theta \qquad (3.5)$$

$$\dot{\theta} = q \qquad (3.6)$$

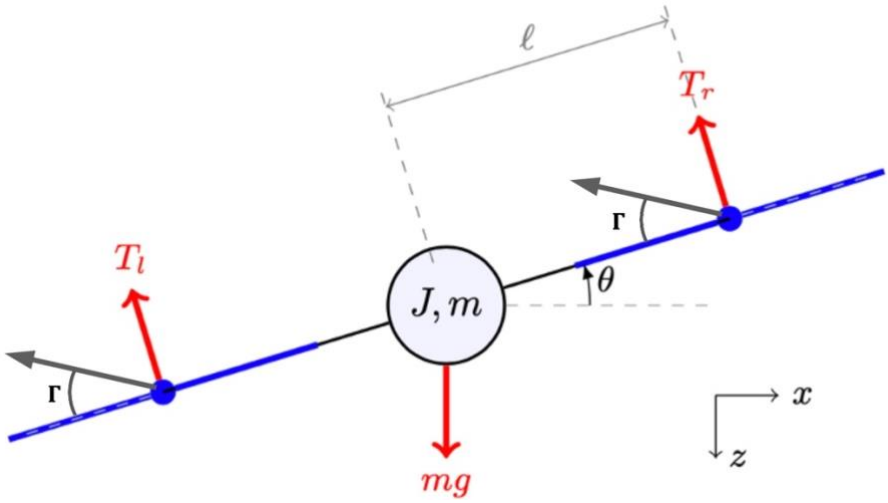


Figure 3-2 PVTOL with Tilting Thrust

Since this study was aimed at the control of tilt-rotor VTOL aircraft, the same PVTOL thrust inputs were assumed a collective tilting angle (Γ), to represent the tilting rotors (thrust vectoring). The tilting angle was defined with respect to the airframe body horizontal as shown in Figure 3-2. The tilting action results in vectored thrust which can act in both x and z axes, and this alters the body frame PVTOL system equations as,

$$\dot{u} = -g\sin\theta - qw + \frac{U_1\cos\Gamma}{m} \quad (3.7)$$

$$\dot{w} = g\cos\theta - \frac{U_1\sin\Gamma}{m} + qu \quad (3.8)$$

$$\dot{q} = \frac{U_2\sin\Gamma}{m} \times \frac{l}{J} \quad (3.9)$$

$$\dot{x} = u\cos\theta + w\sin\theta \quad (3.10)$$

$$\dot{z} = w\cos\theta - u\sin\theta \quad (3.11)$$

$$\dot{\theta} = q \quad (3.12)$$

Finally, the three input, three degree of freedom PVTOL model was then modelled in SIMULINK (see Appendix 7.2A.1) to simulate and develop associated controllers to validate the proposed control concept. The modelled airframe parameters were chosen for convenience of simulation and intuitiveness for manual validation as the airframe is a theoretical object for evaluating the proposed concept having some practical significance. The airframe parameters chosen to model the PVTOL dynamics are given below

Mass (m)	4 kg
Inertia (I_{yy})	0.25 kg m^2
Moment arm (l)	1 m

3.2 PVTOL Differential Flatness Control

This section describes the differential flatness-based control approach developed for the modelled PVTOL dynamics. The controller elucidated in this section skeletons the fundamental control approach for the tilt-rotor VTOL aircraft.

3.2.1 Differentially Flat Equations

Roughly speaking, differential flatness control technique is based on an inverted dynamic of flat systems, which enables the representation of the control inputs as a function of its state differentials. A flat system indicates that the nonlinear structure of the system is characterised well and this characteristic is exploited to design control techniques to achieve trajectory tracking and stabilization [3].

For a given non-linear continuous system dynamics, when system inputs can be represented by a smooth function of the state differentials and outputs, the system is termed differentially flat. Mathematically, this can be represented as

$$\dot{x} = f(x, u) \qquad\qquad y = h(x)$$

$$u = g(x, \dot{x}, \ddot{x})$$

where f, h and g are smooth vector fields of x

With the given understanding of what differential flatness is and with the previously defined PVTOL dynamics given in equations (3.7) – (3.12), the inverted PVTOL dynamics is obtained (see Equation 3.13). This inversion and representation of the control inputs as a function of its state differentials is valid because the PVTOL is a flat system [8]; [10].

The state differentials of the PVTOL dynamics are given as,

$$x = \begin{bmatrix} U \\ W \\ \theta \end{bmatrix} \qquad \dot{x} = \begin{bmatrix} \dot{U} \\ \dot{W} \\ \dot{\theta} \end{bmatrix} \qquad \ddot{x} = \begin{bmatrix} \ddot{U} \\ \ddot{W} \\ \ddot{\theta} \end{bmatrix}$$

Then, the inverted dynamics of the flat system dynamics should have the control inputs as a function of the above state differentials, i.e.

$$u = \begin{bmatrix} U_1 \\ U_2 \\ \Gamma \end{bmatrix} = g(x, \dot{x}, \ddot{x})$$

Solving for the inverted dynamics from equations (3.7) – (3.12)

$$\begin{bmatrix} U_1 \cos\Gamma \\ U_1 \sin\Gamma \\ U_2 \sin\Gamma \end{bmatrix} = \begin{bmatrix} -\left(\dot{U} + g\sin\theta + qW\right) \times m \\ \left(-\dot{W} + g\cos\theta + qU\right) \times m \\ -\dfrac{\dot{q}J}{l} \end{bmatrix} = \begin{bmatrix} -\left(\dot{U} + g\sin\theta + \dot{\theta}\,W\right) \times m \\ \left(-\dot{W} + g\cos\theta + \dot{\theta}U\right) \times m \\ -\dfrac{\ddot{\theta}J}{l} \times m \end{bmatrix} \qquad (3.13)$$

$$\Gamma = \tan^{-1}\left(\frac{U_1 \sin\Gamma}{U_1 \cos\Gamma}\right) = \tan^{-1}\left(\frac{-\left(\dot{U} + g\sin\theta + \dot{\theta}\, W\right) \times m}{\left(-\dot{W} + g\cos\theta + \dot{\theta} U\right) \times m}\right) \tag{3.14}$$

From the above equations, it can be inferred that for the differentially flat PVTOL dynamics, control inputs can be computed from the state differentials obtained from the desired trajectory. The control inputs to be fed into the system based on the inverted system dynamics enable precise and accurate control, achieving better trajectory tracking performance. This theoretical advantage can prove significant in autonomous and pre-planned flight, which is currently a keen area of research interest and the key motivation of the author's work.

3.2.2 Controller Architecture

An appropriate control architecture is vital to achieve optimal control by translating the desired trajectory inputs into control inputs that drive the system to perform as required. The control architecture lays foundation of an implementable controller, and it is key to ensure that it curates the right abstraction of different units that could realise the proper functioning of the control system.

The PVTOL dynamics and its control has been formulated to conceptually validate the feasibility of controlling a practical system, hence, the control architecture had to be devised which would account for uncertainties and disturbances such as gust and turbulences. While the inverse dynamics is based on an idealistic assumption, a feedback control would enable the system to suitably respond to errors, disturbances, and changing operating conditions, consequently rectifying the limitations of the assumption that the inverse dynamics hold. This necessitates the need to collectively utilise the control inputs generated by the inverted dynamics and the feedback controllers. This architecture is inspired from the two degree of freedom design technique in control theory that separates controllers into feedforward compensators for nominal inputs and feedback compensators to correct for plant disturbances [8]. Subsequently, the controller architecture for differential flatness control of PVTOL was devised as shown in Figure 3-3.

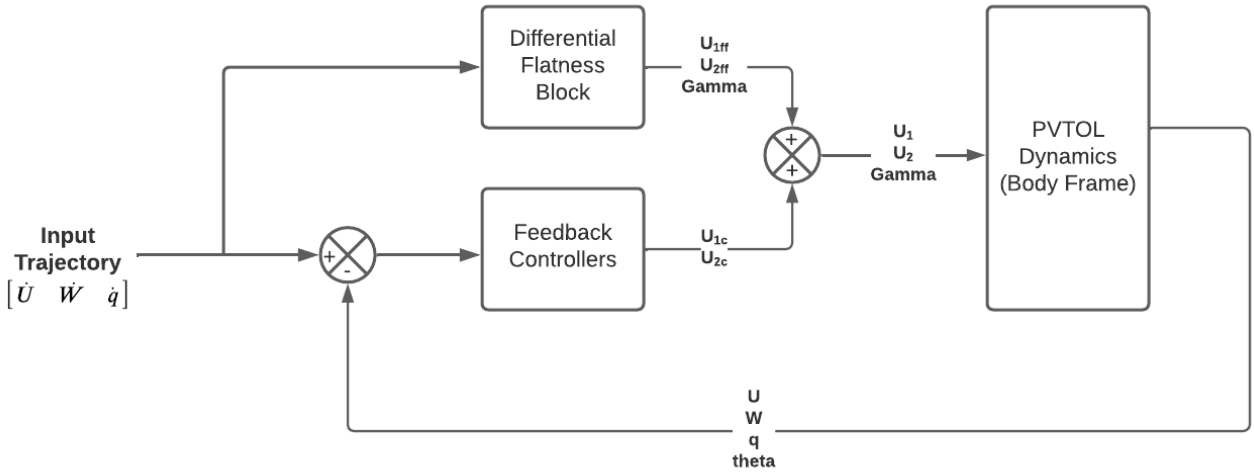


Figure 3-3 Differential Flatness Control Architecture for PVTOL

The input trajectory is fed into the differential flatness unit and linear controller unit which comprises of the inverted dynamics and feedback controllers respectively. Since the inverted dynamics compute control inputs based on state differentials (see equation 3.13) the trajectory inputs should be in the form of accelerations rates $(\dot{U}, \dot{W})$ and rate of pitch rate $(\ddot{\theta})$. However, the feedback controllers function based on the integral of trajectory demands (U, W, θ) and state feedback. Collectively both the units work in tandem to achieve the desired trajectory, with the differential flatness unit feedforwarding the appropriate control inputs and linear controllers correcting the input commands in the event of system disturbances, uncertainties or errors.

3.2.3 Feedback Controller

The feedback controllers were designed to ensure that the system performed with the desired characteristics in the presence of plant uncertainties and external disturbances. To follow the desired trajectory, the controllers should ensure that it achieves the inputs U, W and θ by commanding the collective thrust, differential thrust and tilting angle inputs. Since linear velocity (U) is dependent on the aircraft attitude, thrust tilt angle and total thrust, the feedback control complexity had to be reduced by setting thrust tilt angle based on the differential flatness unit. This constrained the feedback control to two inputs, enabling the cascading

of the linear velocity and attitude controllers to operate seamlessly. The feedback controller structure implemented with the PVTOL dynamics model is given below in Figure 3-4.

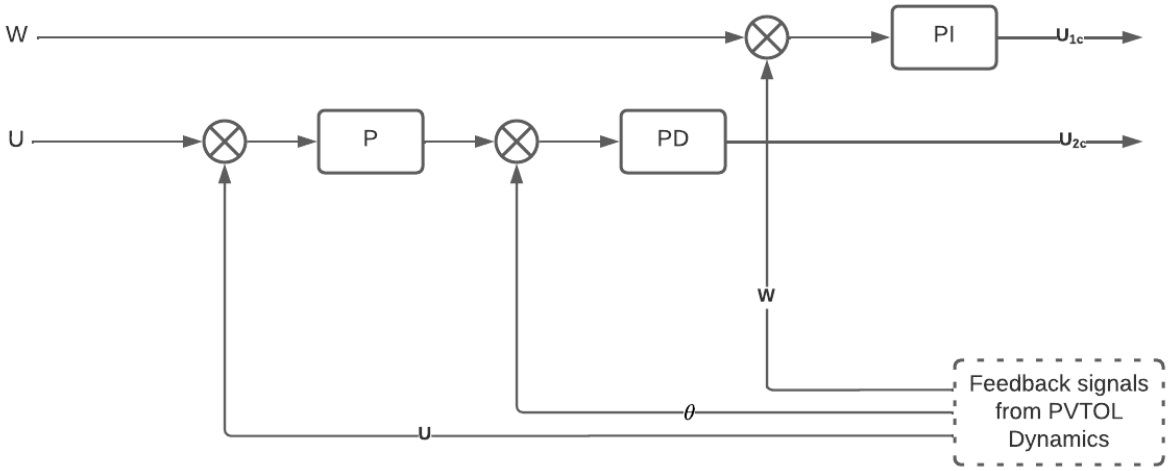


Figure 3-4 Feedback Linear Controller Structure

The longitudinal controller employs a Proportional-Integral (PI) controller to achieve a smooth response without steady state error, and the controller computes collective thrust inputs (U_{1c}) to account for the uncertainties and disturbances which the differential flatness unit would oversee. From the linearized dynamics, the transfer function of longitudinal velocity and collective thrust (see equation 3.15) indicates that the system is marginally stable with an integrator pole. The negative steady state gain arises from the directional notation of z-axis being downwards (NED Coordinate Reference) and collective thrust acting upwards. Thus, a negative stabilising gain should shift the poles to obtain a stable response, and the integral term would avoid steady state-errors.

$$\frac{W}{U_1} = -\frac{0.25}{s} \tag{3.15}$$

The closed loop transfer function of the longitudinal dynamics with the tuned PI controller is,

$$\frac{W}{W_c} = \frac{2.089\ (s + 0.03656)}{(s + 0.03722)(s + 2.052)} \tag{3.16}$$

Observing transfer function 3.16, it can be perceived that the tuned PI controller has moved the poles of the system to a negative value, stabilising the system dynamics. Also, a pole-zero pair in the dynamics negates their effect in the system, generating a damped first-order response as observed in the step response (Figure 3-5). The step response also indicates the acceptable steady-state error and rise time characteristics of the system.

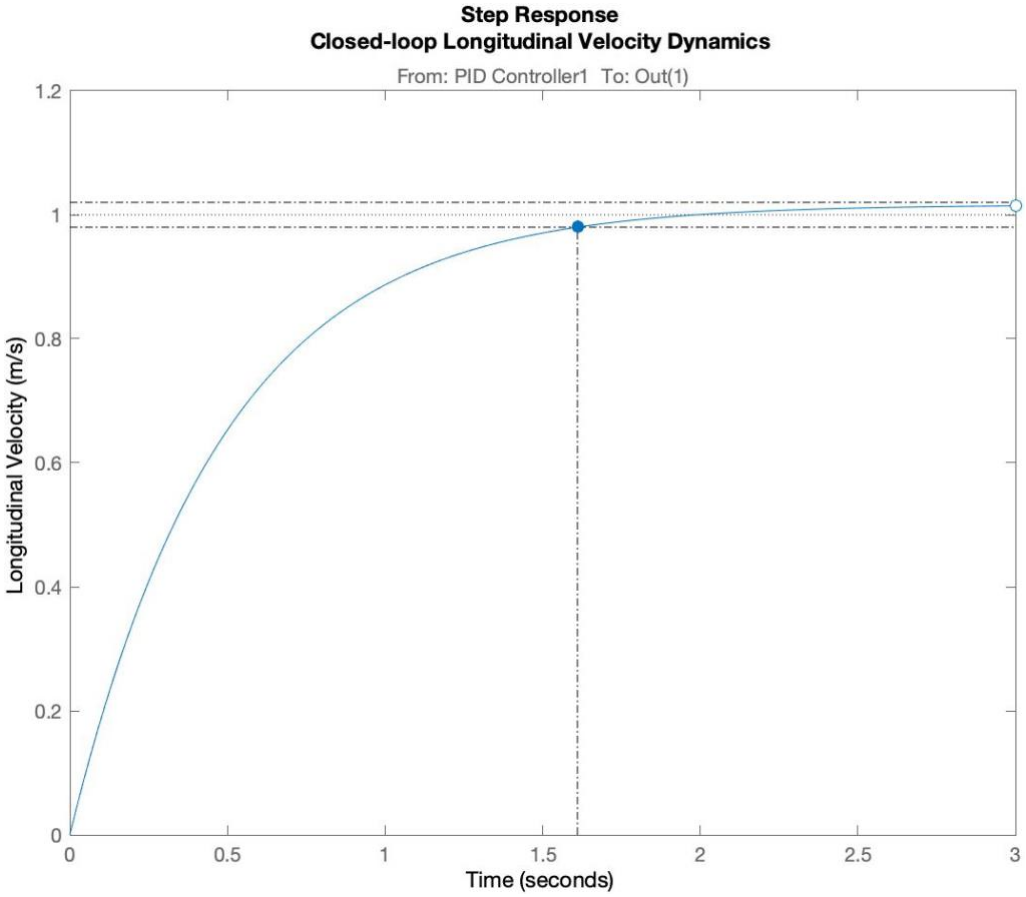


Figure 3-5 Step Response – PVTOL Longitudinal Closed Loop Dynamics

Similarly, a cascaded Proportional-Derivative (PD) and Proportional (P) controllers are implemented to compute the corrective differential thrust input (U_{2c}) based on the attitude (θ) and linear velocity (U) feedback. The inner loop computes the control input to realise the desired attitude, which is computed by the outer loop controller. The PD controller was selected to eliminate the requirement of cascading another inner loop controller for the pitch rate dynamics. The transfer function of the pitch attitude dynamics to differential thrust

input (see equation 3.17) indicates the marginally stable pitch and pitch rate dynamics.

$$\frac{\theta}{U_2} = \frac{4}{s^2} \tag{3.17}$$

With the stabilizing PD controller in place, the closed loop pitch attitude dynamics is given as,

$$\frac{\theta}{\theta_c} = \frac{1.3853\ (\mathrm{s} + 0.002243)}{(\mathrm{s} + 0.002247)(\mathrm{s} + 1.383)} \tag{3.18}$$

Assessing the closed-loop pitch dynamics, the tuned PD controller improves the system dynamics by moving the pitch and pitch rate poles towards the negative region, rendering pitch stability. Again, the zero-pole pair nullifies its effect in the system, resulting in a first-order response characteristic.

Observing the transfer function of the linear velocity to the closed loop pitch input (see equation 3.19), the system poles indicate innate stability. The high negative steady state gain can be associated with the inverted pitch-velocity relationship and the high thrust input at linearization. The high frequency pole is positioned far off to be deemed dominant. Finally, the zero pole indicates that the system has integration dynamics, eliminating the requirement of an integral term in the outer loop controller.

$$\frac{U}{\theta_c} = \frac{-1771.6(s + 0.002243)}{s(s + 0.00247)(s + 1.398)(s + 129)} \tag{3.19}$$

The closed loop dynamics of the linear velocity dynamics with the outer loop proportional controller is given below,

$$\frac{U}{U_c} = \frac{130.38}{(s + 129)(s^2 + 1.39s + 1.011)} \tag{3.20}$$

From the above closed-loop dynamics, the system can be observed to be stabilised with an imaginary pole pair which results in second-order response characteristics. The remaining poles are non-dominant, considering one being at high frequency and the other being negated by an equivalent zero. The response

of the system was chosen to be slow considering it being an outer loop, thus achieving overall acceptable system response. The step response plot (see Figure 3-6) indicates the closed-loop system response to be second order with acceptable overshoot and rise time just above 4 seconds, which was made intentionally slow to avoid high pitch change commands. The Bode plots show the system has satisfactory stability margins of 45 dB and 64.6° gain and phase margins respectively (see Figure 3-7).

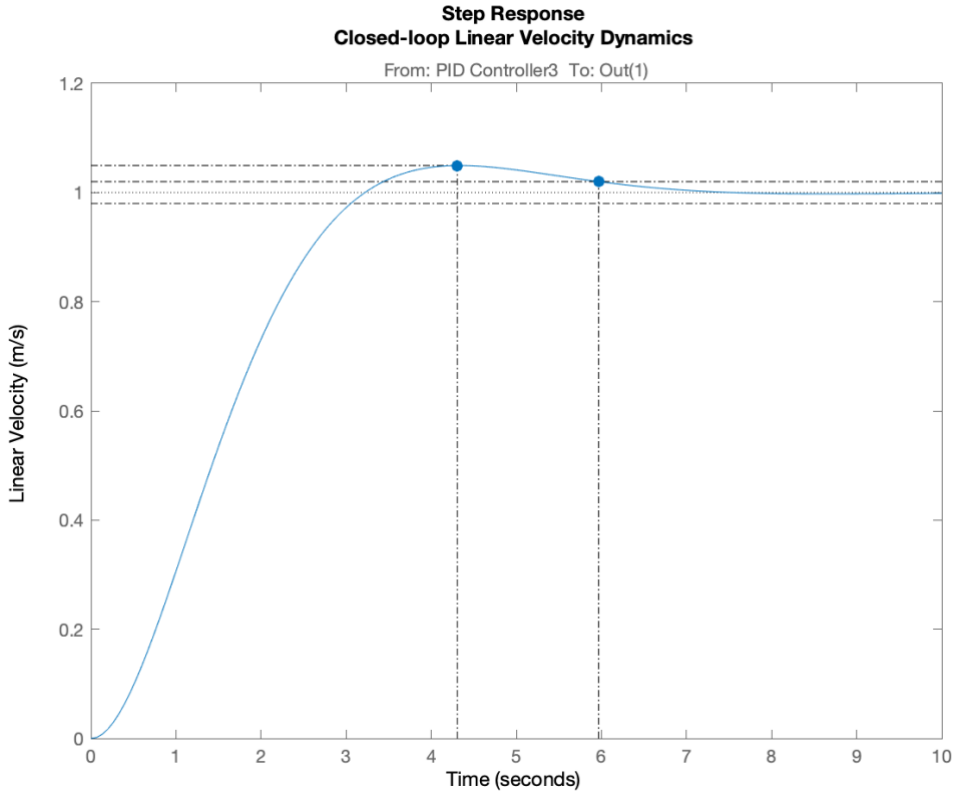


Figure 3-6 Step Response – PVTOL Closed Loop Linear Velocity Dynamics

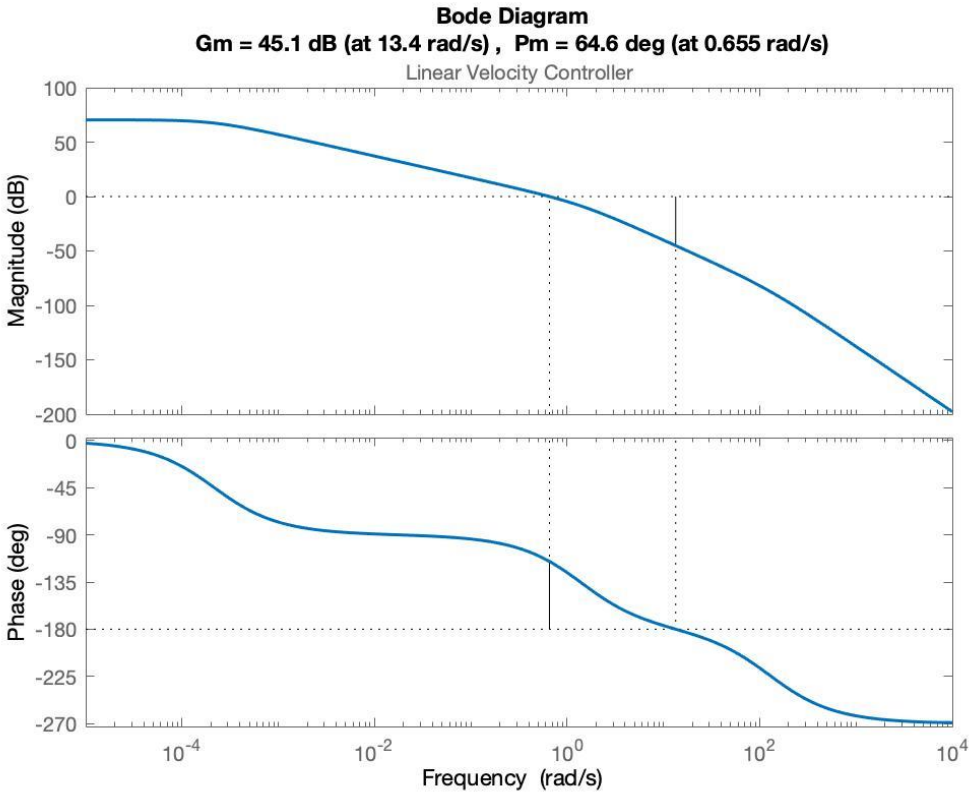


Figure 3-7 Bode Plot - Gain and Phase margins of Velocity Controller

3.3 Non-linear Simulation

Upon development of both differential flatness unit and feedback controllers, the units were integrated with the PVTOL dynamics model in SIMULINK to simulate and test the controller's functionality. The control units were integrated as per the designed controller architecture discussed in section 3.2.2.

The input trajectories were fed into the system as $\dot{U}$ and $\dot{W}$ with no pitch attitude inputs since the feedback controllers stabilize pitch based on the linear velocity. This is justified considering the objective of the controller development to be only a conceptual validation of the differential flatness controllers for VTOL aircraft.

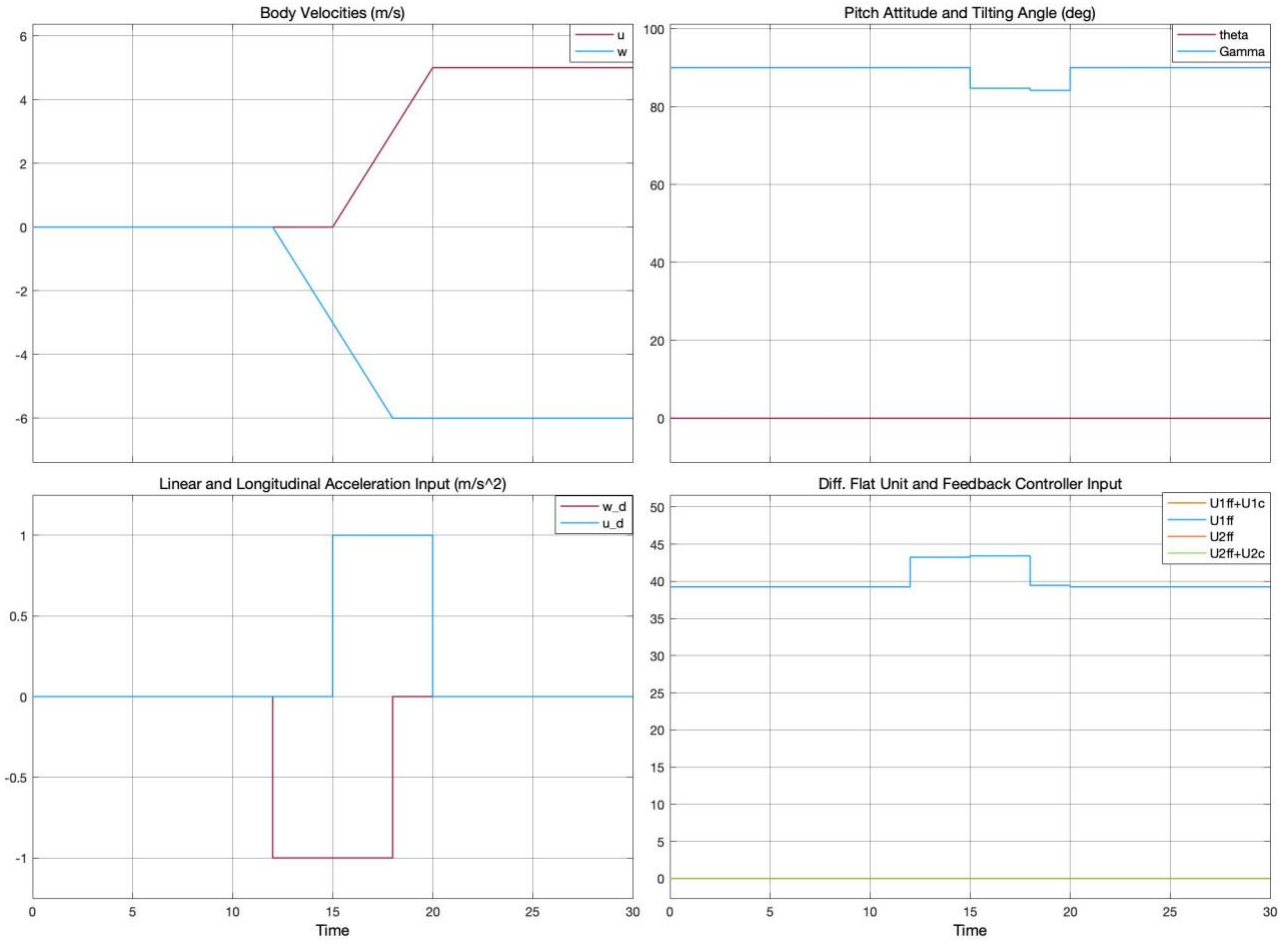


Figure 3-8 Non-linear Simulation - PVTOL Differential Flatness Controller

From the above non-linear simulation for linear and longitudinal acceleration demands, the differential flatness controllers can be seen to function satisfactorily. The controller achieves the desired performance with the differential flatness unit computing precise control inputs, reducing feedback controller effort which can be observed with the control input plots in Figure 3-9. With the varying velocity demands, the inverted dynamics also precisely computes the required tilting angles to maintain the desired trajectory. While the idealistic approach of dynamics and system responses cannot be expected in practical systems, this can sufficiently support the feasibility of the proposed control approach.

The significance of feedforwarding the differential flatness unit is to eliminate the feedback controller effort which primarily functions based on the error signals. However, the feedback controllers perform the critical role of shunting the adverse effects of disturbances and uncertainties. This function is evident in Figure 3-9 where a step plant disturbance input in the longitudinal dynamics is immediately countered by the feedback controller whereas the differential flatness unit failed to respond to the same. This characteristic of the feedback controller is critical to ensure stability and control, especially in practical systems.

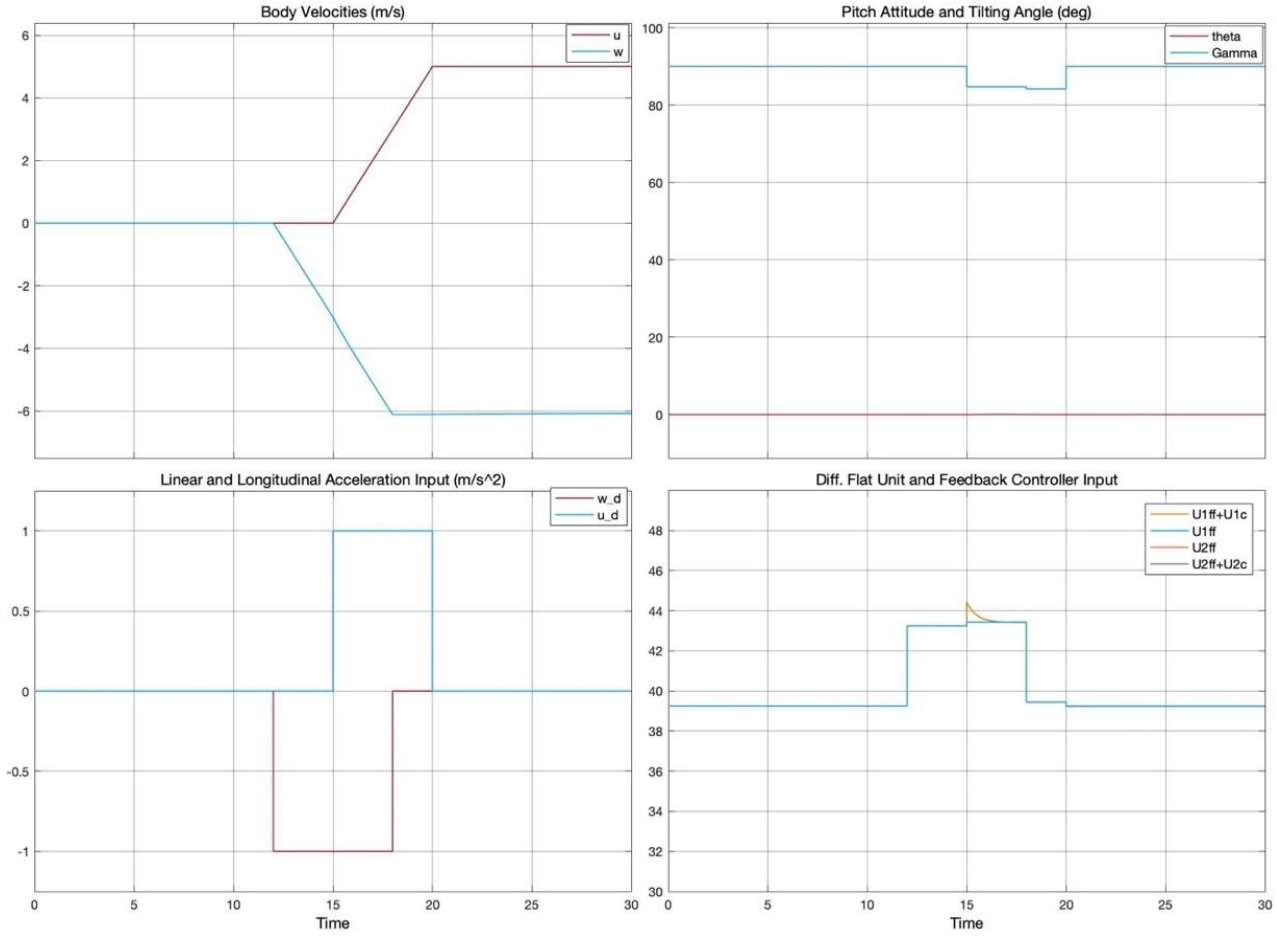


Figure 3-9 Non-Linear Simulation - PVTOL Controller (Disturbance Rejection)

3.4 Conclusion

The conceptual study was intended to identify and characterise the feasibility of the proposed control technique for tilt-rotor VTOL aircraft. The PVTOL dynamics model was specifically selected because of its comparable dynamics, and was adapted to incorporate thrust vectoring to closely relate to tilt-rotor VTOL longitudinal dynamics. Finally, a control architecture based on differential flatness was conceptualised and implemented in Simulink.

It is important to state that the study was strictly idealistic, and several practical concerns in flight control have been overlooked. Ruling out lateral dynamics, actuation dynamics, aerodynamics and propeller dynamics quite significantly affects the fidelity of the studied model, however these factors were implied to be insignificant for conceptual validation of the control architecture.

To conclude, the conceptual validation study established and quite successfully proved the feasibility of the proposed control concept based on differential flatness, and formed an important foundation for the further proceedings of the author’s study.

4 TILT-ROTOR FLIGHT DYNAMICS MODELLING

A flight dynamics model is a mathematical representation of the flying characteristics of an aircraft. The dynamics model illustrates the behaviour of an aircraft at different operating conditions throughout its flight envelope, and it enables the engineer to understand and develop specific flight systems. In practice, the modelling process is often iterative [28], where the flight dynamics model will be subjected to multiple iterations to improve its fidelity in representing flight physics that is dependent on extensive understanding and data of the aircraft. The flight dynamics model is built based on the airframe structure by integrating the models of various components that influence flight such as atmosphere, aerodynamics, thrust or propeller dynamics.

In aircraft development, flight dynamics modelling is considered an invaluable part because of its ability to test and validate flight controllers and various other systems in the absence a real aircraft which can be exorbitantly expensive [28]. When concerning flight control development, the flight dynamics model enables to closely assess an aircraft's flying characteristics and build controllers that optimally work for the specific airframe, and even simulate and virtually flight test. It enables engineers to decide what control laws would be appropriate, what would make the aircraft efficient amongst the various other aspects that concern flight control development.

The author's study to develop a differential flatness flight controller for a tilt-rotor VTOL aircraft evidently required a flight dynamics model to implement and validate the conceptual control technique, and to do so, the Aston Martin Vision Volante Aircraft was chosen. This chapter expands on the specifics of the airframe and the associated flight dynamics model developed.

4.1 Aston Martin Vision Volante

A special project commissioned by Aston Martin, the Vision Volante was conceptualised and designed to preview the future of urban and inter-city air travel with an advanced and autonomous hybrid-electric flight. The concept was intended to take full advantage of the latest aerospace and electrification technologies, and to be benchmarked as the pinnacle of UAM [29].

The aircraft was conceptualised to be a three-passenger aircraft, propelled by three coaxial open propellers. The delta wing configuration and canard-like forward wings were designed to generate sufficient lift at cruise, while keeping its compact form factor for the urban spaces. The large aft propeller design that melds into the airframe behind the passenger cabin was critical to generate sufficient vertical thrust during hover, while generating minimal drag during cruise. The tilt-rotor setup of the forward rotor enables the aircraft to hover and transition to cruise by vectoring propeller thrust [30].

The hybrid-electric aircraft was designed to be capable of flying a distance of 805 km at speeds of 402 kmph[29]; [30]. Autonomous flight and stepped levels of pilot interactions were some of the key pilot features thought in the design. With regards to control surfaces, the aircraft had inner and outer aileron on the main delta wing, twin vertical rudders and a moving canard.

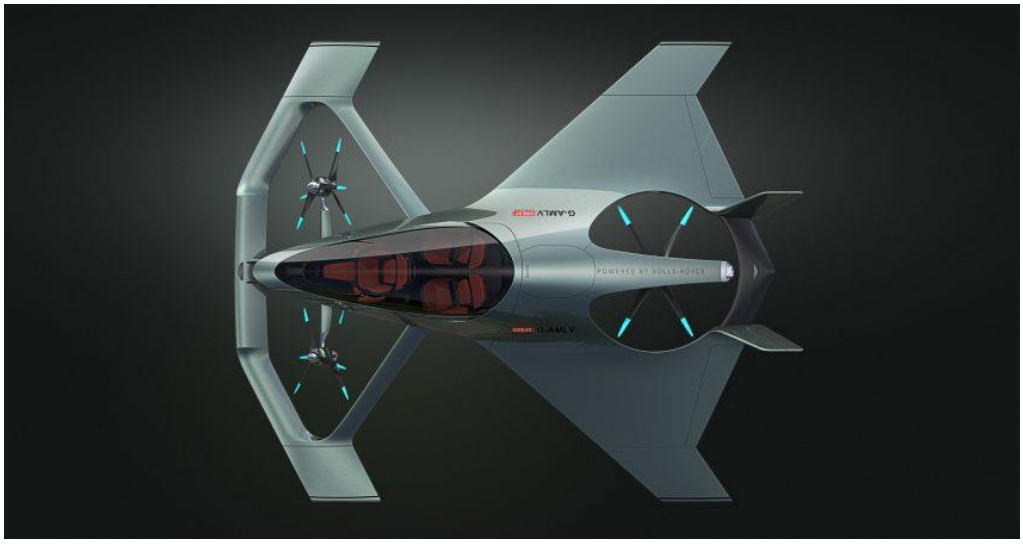


Figure 4-1 Aston Martin Vision Volante [29]

Operational Characteristics		Geometric Characteristics	
MTOW	2000 Kg	Span	10 m
CG Positions	6m (aft-CG)	Wing Span Area	15.8431 m^2
V_{MAX}	402 kmph	Canard Span Area	5.0162 m^2
Max. Altitude	20000 m	Airfoils (Wing/Canard)	NACA 5412/3412

Table 4-1 Airframe Operational and Geometric Characteristics[31]; [32]

4.2 Flight Dynamics Model

Building a well-structured and elaborate flight dynamics model is a very critical step in model-based flight control development. A structured model would enable easy troubleshooting and iterative improvement of the model as the modelling data volume and understanding of the flight physics improve at various phases of development.

4.2.1 Coordinate Reference Frames

Essentially, the model of flight dynamics is the computation of the flight parameters based on the different forces and moments acting on an airframe. The different types of force vectors acting on an airframe can be modelled with respect to different reference frames, which then must be translated into net forces and moments acting on the airframe body. A proper definition of the reference frames used in modelling and data acquisition is key to ensure a coherent representation of its flight dynamics.

BODY FRAME OF REFERENCE

The body frame of reference is centred on the airframe centre, and this is the primary reference frame to which all the other dynamic forces and moments are computed and transformed into. The major reason for fixing the body frame as the primary reference frame is due to the non-linear equations of motions being

based on the body frame forces and moments. Fixed to the aircraft centre of gravity (CG), the coordinate system has the x-axis running towards the nose parallel to the fuselage reference line, the z-axis pointing downwards in the plane of symmetry and the y-axis extending through the right aircraft wings perpendicular to the x-z plane.

WIND FRAME OF REFERENCE

Aerodynamics on an aircraft body are forces and moments induced due to the relative motion of air flowing across the aerodynamic surfaces of the aircraft, hence, they are dependent on the aircraft orientation against the airflow. The aerodynamics forces are obtained in the reference frame of the airflow, and Angle of Attack (α) and Sideslip Angle (β) are used to specify airflow vectors with respect to the body axis. A Direction Cosine Matrix facilitates (see Equation 4.1) the transformation of the aerodynamic forces and moments from the wind frame to body frame.

$$DCM_{wb} = \begin{bmatrix} \cos\alpha\, cos\beta & \sin\beta & \sin\alpha\cos\beta \\ -\cos\alpha\sin\beta & \cos\beta & -\sin\alpha\sin\beta \\ -\sin\alpha & 0 & \cos\alpha \end{bmatrix} \quad (4.1)$$

PROPELLER FRAME OF REFERENCE

The propeller induced forces and moments are computed using the blade momentum theory (see section 4.2.4) in the reference frame centred at the axis of rotation of the propeller as shown in Figure 4-2 (Left propeller omitted for clarity). The x-axis extends perpendicularly outwards the propeller disc plane, while y-axis and z-axis lie perpendicular to each other on the propeller disc plane. Unlike conventional aircraft, the tilting feature of the forward propellers results in a varying transformation matrix, as a function of the tilt-rotor angle (Γ). The transformation matrix that resolves propeller forces and moments in the propeller frame to body frame is given by,

$$DCM_{bp}(\Gamma)^{1} = \begin{bmatrix} \cos\Gamma & 0 & \sin\Gamma \\ 0 & 1 & 0 \\ -\sin\Gamma & 0 & \cos\Gamma \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}_{\Gamma_{aft}=90°} \quad (4.2)$$

where Γ is the tilting angle of the propeller with respect to the body frame x-axis.

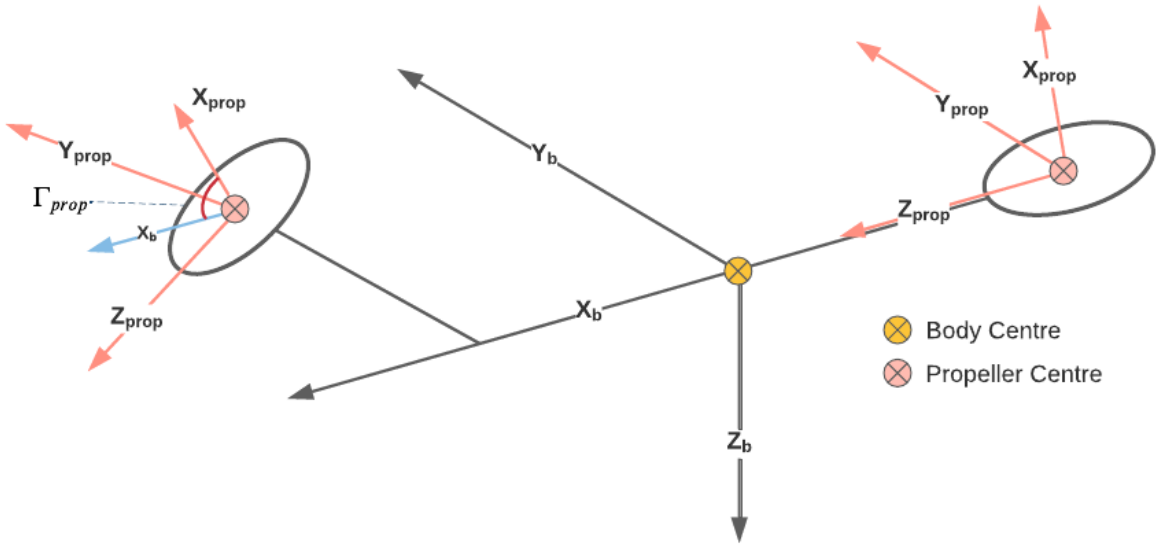


Figure 4-2 Body and Propeller Reference Frame[2]

4.2.2 Architecture

As discussed earlier, the flight dynamics model computes the aircraft states based on the net forces and moments acting on the airframe, and is driven by the different modules functioning in coherence to compute the data for driving the equations of motion. A top-level architectural representation of the developed flight dynamics model is illustrated in Figure 4-3.

The control input commands block consists of the actuation dynamics of the control surfaces and propellers, and is responsible for relaying the control input information to compute the current forces and moments due to the control inputs.

---

[1] The aft propeller does not tilt, and hence has a fixed transformation matrix given by $\Gamma_{aft} = 90°$

[2] Left propeller omitted for clarity

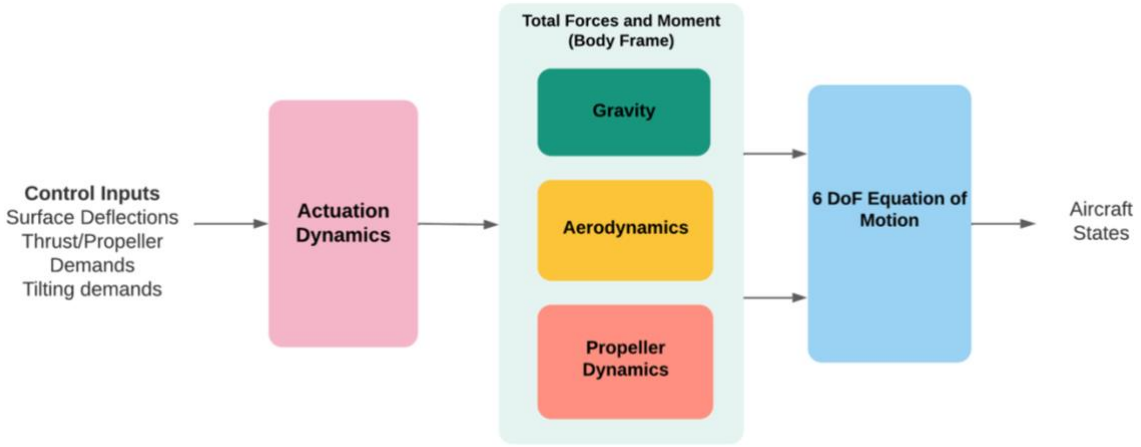


Figure 4-3 Flight Dynamics Model Architecture (Top Level)

The Gravity block computes the varying gravitational forces acting on the aircraft based on the orientation of the aircraft with reference to earth. Since the gravitational vector is always oriented to the earth centre, a DCM matrix rotates the gravitational force vector from earth to body frame.

The Aerodynamics block is responsible for computing the aerodynamics forces and moments acting on the aircraft. It consists of the aerodynamic coefficient database, which characterises the aerodynamics of the aircraft at various operating conditions. Generally, dynamics model fidelity is often improved by iteratively improvising the quality of the coefficient database, which is derived from various methods like Computational Fluid Dynamics (CFD), Wind Tunnel Testing and Flight Testing. Based on the flight states, relevant aerodynamic coefficients and control surface deflection inputs, the unit computes the net aerodynamics forces and moments acting in the body frame.

The propeller dynamics block computes the forces and moments generated by the individual propellers based on the propeller geometry, propeller aerodynamics data and flight parameters. This modular approach of the propeller dynamics allows for a unified model for all flight phases of the VTOL aircraft. Once the propeller forces and moments are computed in the propeller reference frame, it is then transformed to the body frame (see Equation 4.2).

Finally, the 6 Degree of Freedom Equations of Motion (6DoF EoM) block comprise of the differential equations that calculates the different flight states as a result of the forces and moments acting on the body frame at a given time instant. The EoM block assumes the body to be rigid, eliminating the need to consider the forces acting between individual elements of mass [33]. All forces and moments are totalled and input into the EoM block, which computes the velocities $(U\ V\ W)$, attitude $(\phi\ \theta\ \psi)$, angular rates $(p\ q\ r)$, acceleration $(\dot{U}\ \dot{V}\ \dot{W})$ and position of the aircraft $(X_e\ Y_e\ Z_e)$. The 6DoF Equation of Motion are,

$$A_{bb} = \begin{bmatrix} \dot{U}_b \\ \dot{V}_b \\ \dot{W}_b \end{bmatrix} = \frac{F_b}{m} - \Omega \times V_b \qquad \text{where,} \qquad F_b = \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix}$$

$$V_b = \begin{bmatrix} U_b \\ V_b \\ W_b \end{bmatrix} \qquad \text{where,} \qquad \Omega = \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

$$M_b = \begin{bmatrix} L \\ M \\ N \end{bmatrix} = I\dot{\Omega} + \Omega \times (I\Omega) \qquad where, \qquad I = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{yx} & I_{yy} & -I_{yz} \\ -I_{zx} & -I_{zy} & I_{zz} \end{bmatrix}$$

Equation 4-A Equation of Motion (Body Frame) [33]

A detailed representation of the flight dynamics model is illustrated below in Figure 4-4. The key signal names are tagged along the interconnections, and every unit is colour coded to its respective module in Figure 4-3. The flight dynamics model was implemented in SIMULINK by Gonzalo Valdes [32], however several modifications and alterations were made by the author to improve model correctness.

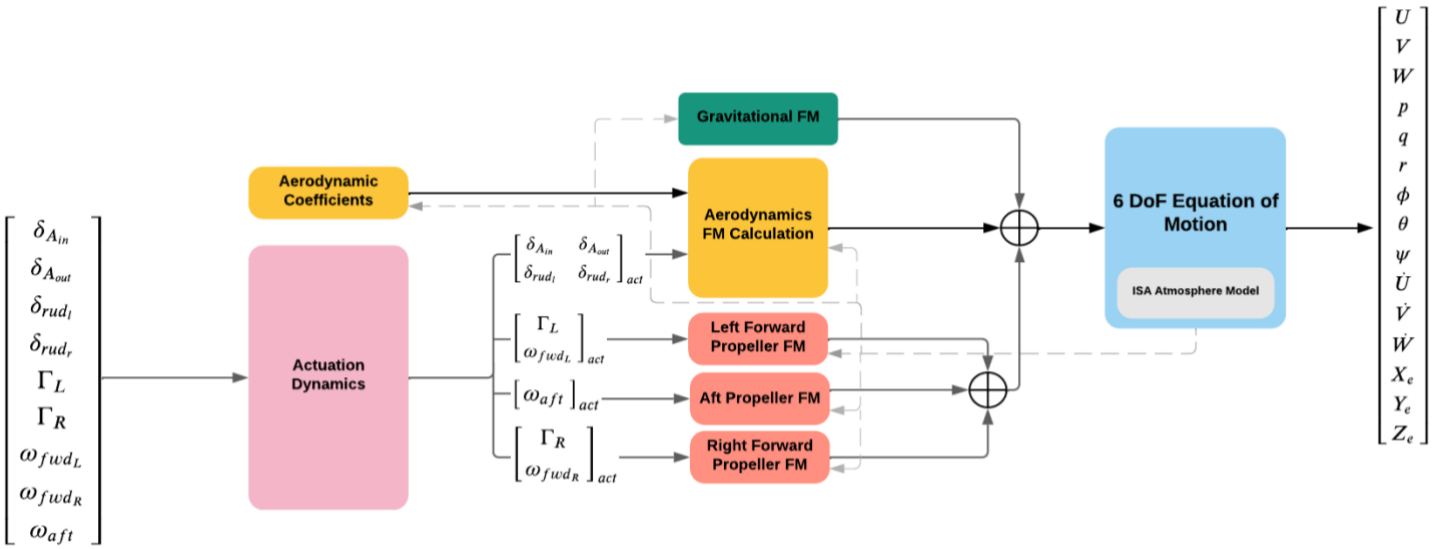


Figure 4-4 Flight Dynamics Model Architecture

4.2.3 Aerodynamics

The aerodynamics of an aircraft is what defines the flying characteristics of an aircraft. As an aircraft moves through air, the relative velocity of air flowing across the specially designed surfaces of the aircraft induces pressure force and skin friction, which are conventionally resolved into the Lift, Drag and Side forces (and its induced moments).

Characterising the nonlinear behaviour of aerodynamics forces acting on an aircraft, is a tedious and iterative process that involves significant amounts of time and resources. While modern computational capability has enabled high-fidelity aerodynamic analysis, the author of this project confined to preliminary aerodynamic analysis methods which are quick and sufficient for a conceptual study considering the project scope and timelines.

The resolving of the aerodynamic forces results in Lift and Drag (and associated moments), where lift is perpendicular to the freestream velocity vector and drag is parallel to it [28]. The aerodynamic forces are dependent on the angle of attack (α), sideslip angle (β), Mach and reynolds number, which are all factors that vary continuously for an aircraft in flight. Thus, developing a dataset that accounts for all the possible aerodynamic variations in flight is necessary for modelling flight dynamics.

4.2.3.1 Coefficients

Aerodynamic coefficients are dimensionless coefficients that represent the ability of the surface to produce lift or drag. These coefficients are obtained by normalising measured (or calculated) aerodynamics force with the product of dynamics pressure and reference area. The chosen reference lengths are used when normalising the moment coefficients from the aerodynamic moments. The aerodynamics coefficients can be obtained for different flight conditions, using various aerodynamic analysis methods, and a mapping of the coefficients and its associated flight conditions are created in lookup tables. By maintaining commonality with the aerodynamic data tables and interpolation routines, the lookup tables will be capable of computing aerodynamic coefficients for all flight conditions, including beyond the analysed data limits (at reduced fidelity).

Aircraft aerodynamic coefficients can be specified as function of the aerodynamics angles, Mach, altitude, control surface deflections and propulsion system effects. Roughly speaking, the longitudinal coefficients primarily depend on alpha, while the lateral-directional coefficient depends on alpha and beta. The fidelity of the aero coefficient data is dependent on the extent of parameter variations considered in aerodynamic analysis. The aerodynamic coefficients can be represented as,

$$C_{(\,)} = C(\alpha, \beta, M, h, \delta_{a,e,r})$$

Since aircraft aerodynamics is dependent on several factors, representing the entire flight envelope in all aircraft configurations becomes tedious and complex. For airframes with complex dynamics like Vision Volante, it could be further demanding due to large number of datapoints required to represent highly nonlinear dynamics. However, another commonly used technique for aerodynamic coefficient representation is aero-derivatives. Considering aerodynamic coefficients to be a function of flight states and deflections, the derivatives are factors associates with each variable (see Equation 4-B). This simplifies the complexity of representing aerodynamics by having a mapping of the derivatives which can be used to build the net coefficient for a given flight state and deflection. Most airplanes have flight envelopes constrained to small

angles of attack and low Mach numbers, enabling better data integrity when breaking down coefficients into sum of small derivative components[28] due to linearity in dynamics. However, at regions of high nonlinearity, the derivative representation of aerodynamics can become limited. The component build-up of the aerodynamic coefficients for the dynamics model used is given below in Equation 4-B.

Gonzalo Valdes [32] performed aerodynamic analysis using a Vortex Lattice solver called AVL to derive the aero-derivative data. Gonzalo obtained aerodynamic data for steady flight ($\alpha = 0°, \beta = 0°$) at speeds ranging from Mach 0.05 to Mach 0.35 and altitudes between 0 meters to 3000 meters for different aircraft configurations of CG and mass. It is important to note that the AVL analysis method obtains data limited to the pre-stall flight conditions, and aerodynamics characteristics beyond the stall angle of attack can be considered erroneous.

$$C_L = C_{L_0} + C_{L_\alpha}\alpha + C_{L_q}\left(\frac{qC_{ref}}{2V_T}\right) + C_{L_{\delta_{Ain}}}\delta_{Ain} + C_{L_{\delta_{Aout}}}\delta_{Aout}$$

$$C_D = C_{D_0} + KC_L^2 + C_{D_{\delta_{Ain}}}\delta_{Ain} + C_{D_{\delta_{Aout}}}\delta_{Aout} + C_{D_{\delta_{Rud_R}}}\delta_{Rud_R} + C_{D_{\delta_{Rud_L}}}\delta_{Rud_L}$$

$$C_Y = C_{Y_\beta}\beta + C_{Y_p}\left(\frac{pB_{ref}}{2V_T}\right) + C_{Y_r}\left(\frac{rB_{ref}}{2V_T}\right) + C_{Y_{\delta_{Ain}}}\delta_{Ain} + C_{Y_{\delta_{Rud_R}}}\delta_{Rud_R} + C_{Y_{\delta_{Rud_L}}}\delta_{Rud_L}$$

$$C_l = C_{l_b}\beta + C_{l_p}\left(\frac{pB_{ref}}{2V_T}\right) + C_{l_r}\left(\frac{rB_{ref}}{2V_T}\right) + C_{l_{\delta_{Ain}}}\delta_{Ain} + C_{l_{\delta_{Aout}}}\delta_{Aout} + C_{l_{\delta_{Rud_R}}}\delta_{Rud_R} + C_{l_{\delta_{Rud_L}}}\delta_{Rud_L}$$

$$C_m = C_{m_0} + C_{m_\alpha}\alpha + C_{m_q}\left(\frac{qC_{ref}}{2V_T}\right) + C_{m_{\delta_{Ain}}}\delta_{Ain} + C_{m_{\delta_{Aout}}}\delta_{Aout}$$

$$C_n = C_{n_\beta}\beta + C_{n_p}\left(\frac{pB_{ref}}{2V_T}\right) + C_{n_r}\left(\frac{rB_{ref}}{2V_T}\right) + C_{n_{\delta_{Aout}}}\delta_{Aout} + C_{n_{\delta_{Rud_R}}}\delta_{Rud_R} + C_{n_{\delta_{Rud_L}}}\delta_{Rud_L}$$

Equation 4-B Aerodynamic Coefficient Component Representation

4.2.3.2 Forces and Moments

With the characterisation of the aerodynamics in the form of coefficients, and a data structure that facilitates real-time compute of the coefficients, the net forces and moments acting on the aircraft can be given by,

$$\text{Lift} = -\frac{1}{2}\rho S_{ref} V_T^2 C_L \qquad \text{Roll Moment} = \frac{1}{2}\rho S_{ref} b_{ref} V_T^2 C_l$$

$$\text{Drag} = -\frac{1}{2}\rho S_{ref} V_T^2 C_D \qquad \text{Pitch Moment} = \frac{1}{2}\rho S_{ref} b_{ref} V_T^2 C_m$$

$$\text{Sideforce} = \frac{1}{2}\rho S_{ref} V_T^2 C_Y \qquad \text{Yaw Moment} = \frac{1}{2}\rho S_{ref} b_{ref} V_T^2 C_n$$

Equation 4-C Aerodynamic Forces and Moments

The forces and moments computed are obtained in the wind frame, and can then be transformed into the body frame as required by the EoM, using DCM_{wb} (see Equation 4.1).

As previously mentioned, the linearity of the aerodynamics has been exploited in the dynamics model which instigates a significant anomaly when in hover phase. This is due to the large aerodynamic angles during vertical flight, inducing inaccurate coefficient build-up leading to excessive aerodynamic forces and moments incorrect from the actual flight dynamics. This has been resolved by assuming the aerodynamic forces to be 0 for very small velocities (V_T <0.1 m/s) and by approximating the lift characteristics at high angles of attack. Preliminary aerodynamic tools such AVL do not analyse the post-stall aerodynamics, and the project scope limited the author to carry out an in-depth aerodynamic analysis to obtain accurate post-stall dynamics data. Thus, basing the post-stall characteristics of the wing airfoil from research [34]; [35], and with the AVL pre-stall data, the author built an approximation of the lift coefficient variation with α variation.

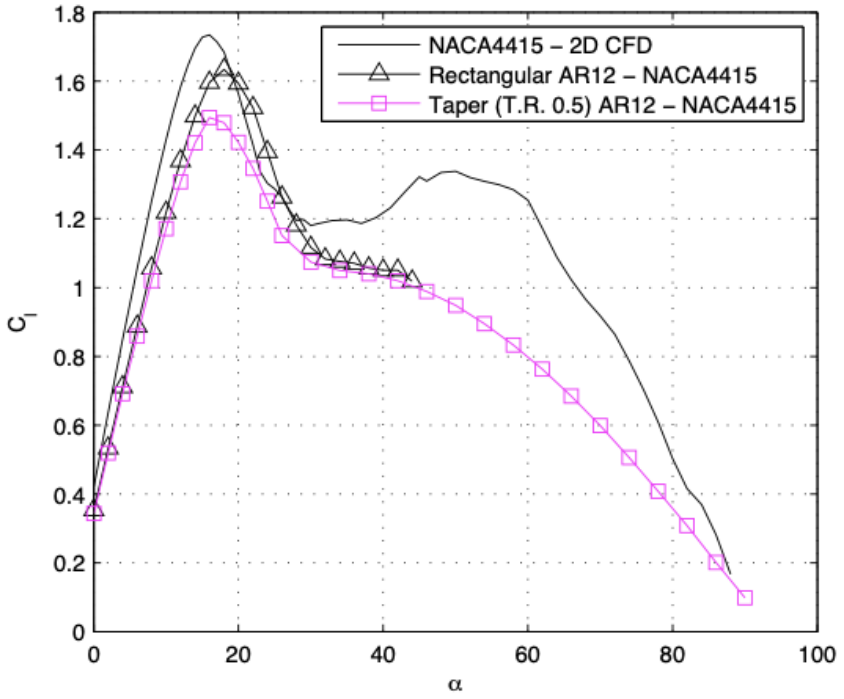


Figure 4-5 Post-Stall Characteristics [35]

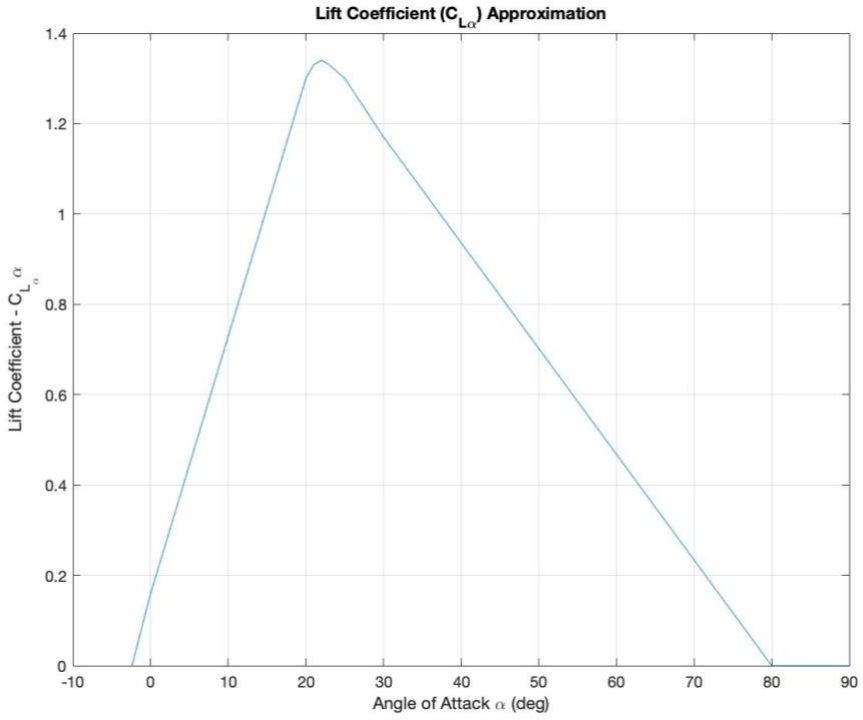


Figure 4-6 Lift Coefficient Post-Stall Approximation

The above approximation is assumed valid considering the low velocities in the hover phase and insignificant freestream flow across the wing to generate significant lift and drag forces. While the post-stall coefficient approximation is a rough estimate, it is important to note that the aircraft would not be flying at such high aerodynamic angles, and the current scope of the project limits testing controllers beyond stall condition. Hence, this approximation was considered relevant and acceptable for the scope of the author's study.

4.2.3.3 Assumptions

Flight dynamics modelling is regarded as an iterative process due to the difficulty in acquiring and modelling aerodynamic data to build an accurate representation. There are several assumptions that have been made to approve the defined dynamics model to be acceptable, and the assumptions stated below are identified to have minimal influence on the research objectives. The aerodynamic assumptions made are as follows:

1. **Propeller Wash Effects**

   The tilt-rotor propellers induce a high speed flow of air across the delta wing of the aircraft, and this airflow would modify the lift-drag forces acting on the wing. Modelling of the propeller freestream at different velocities and rotor speed, and its effects on the wing aerodynamics has to be performed.

2. **Canard Approximations**

   The degenerate model used to perform aerodynamic analysis accounted for the rather complex canard-wing structure with an equivalent tapered wing, due to the limitation of AVL [32]. This decreases the correctness of the aerodynamic influence by the canard-wing, and misses the propeller freestream effects as well. A detailed model along with powerful aerodynamic analysis methods like CFD or Wind Tunnels should be carried out, to precisely characterise the canard-wing aerodynamics.

3. **Canard Control Surfaces**

   Conceptually, the canard houses a control surface capable of generating pitch moments to improve the aircraft controllability. However, with the limited representation of the canard structure, this control surface was also not accounted for in the dynamics model. However, being an over-actuated aircraft, it has enough control surfaces to achieve controllability in all axis, and hence this oversight was deemed acceptable.

4. **Range of Aerodynamic Coefficients**

   The coefficients database that characterises aerodynamics has been obtained by running solvers for a limited set of flight conditions, and the interpolation algorithm is expected to compute coefficients for intermediate operate conditions. Practically, interpolated coefficient values could vary from actual, despite most being close approximations (especially for regions of high nonlinearity). Increasing the resolution of operating conditions to build the coefficient dataset, can significantly improve the accuracy of the modelled dynamics.

5. **Hover Phase Dynamics**

   The hover phase aerodynamics work quite differently to cruise phase aerodynamics, and an independent assessment of the hover phase dynamics should be conducted. In this model, the Hover phase aerodynamics have been assumed to be negligible and approximations have been made in computing aerodynamic forces and moments at hovering velocities (see Section 4.2.3.2). Accounting for the wing drag and lift during hover can be critical in practical development, however, it was considered acceptable to meet the project goals.

6. **No Ground Effects**

   The vortices created by downward thrust when flying close to surface, creating a cushion effect can significantly aid in characterising the dynamics in the take-off, landing and go-around scenarios. This aspect

was overlooked since near-ground control performance would not be assessed in this study.

7. **No Under Carriage Effects**

   The aerodynamic effect involved with retracted landing gear is a critical aspect purposefully overlooked considering the significance it has in terms of aiding the project aim. However, in practical development, characterising the undercarriage effects can have significant benefits.

4.2.4 Propeller Dynamics

The Vision Volante is thrusted by three coaxial propellers, with two tilting rotors between the canards, and one aft rotor that is significantly larger than the forward propellers. Modelling the propeller dynamics, and consequently computing forces and moments it generates is vital for the flight dynamics model. Like the aerodynamics, the propeller dynamic characteristics have been obtained from the development carried out by Gonzalo Valdes [32] and some modifications have been made by the author to the existing model to improve the propeller dynamics.

4.2.4.1 Performance Predictions

The propeller performance characteristics have to be identified for a given propeller geometry to model and compute the propeller induced forces and moments. QPROP [36] is an analysis program developed to predict the performance of propeller-motor combination using relatively simpler propeller models. Gonzalo Valdes [32] and Namuk Kang [37] exploited this software to compute the Thrust Coefficients (C_T), Power Coefficients (C_P) and Advance Ratio (J) of both the forward and aft propellers. More specific information concerning the working and limitations of the QPROP analysis program has been discussed in [32].

The software is input with rotor blade dimensions and propeller airfoil geometry, and the estimation software computes the thrust and power coefficients for different advance ratios. The preliminary analysis tool used to obtain the

performance data cannot be considered highly accurate considering the assumptions involved when modelling the rotor for analysis.

Firstly, the counter-rotating propeller has not been modelled when analysing in QPROP, giving rise to some inaccuracies. Secondly, the QPROP tool estimates the propeller airfoil aerodynamics, and this can affect the fidelity of the computed thrust and power coefficients. This propeller dynamics estimation also will affect the computed performance characteristics at low advance ratios [36].

Although the above-mentioned limitations exist, the QPROP software does provide reasonably valid data for building a propeller dynamic model sufficient enough to meet the requirements of the project.

4.2.4.2 Forces and Moments

The propeller forces and moments are computed in the propeller frame of reference (see section 4.2.1) which has to be transformed into the body frame of reference when input to the equations of motion. However, the propeller forces and moment equations require freestream velocity and angles in the propeller frame. This is obtained by resolving the aircraft orientation, velocities and tilting rotor angles as shown below,

$$\left[UVW_{prop}\right]_{3X1} = \left[DCM_{pb}\right]_{3X3}\left[UVW_{body}\right]_{3X1} + \left[DCM_{pb}\right]_{3X3}\left[PQR_{body} \times Pos_{prop}\right]_{3X1}$$

$$TAS_{prop} = U_{prop} \qquad \alpha_{prop} = \tan^{-1}\left(\frac{W_{prop}}{U_{prop}}\right) \qquad \beta_{prop} = \sin^{-1}\left(\frac{V_{prop}}{\sqrt{U_{prop}^2 + V_{prop}^2 + W_{prop}^2}}\right)$$

$$DCM_{pb} = \left[DCM_{bp}\right]^T = \begin{bmatrix} \cos\Gamma & 0 & -\sin\Gamma \\ 0 & 1 & 0 \\ \sin\Gamma & 0 & \cos\Gamma \end{bmatrix} = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}_{\Gamma_{aft}=90^\circ}$$

The DCM_{pb} is the direction cosine matrix that transforms the body frame vectors to propeller frame vector (see section 4.2.1), which when multiplied with the body frame freestream velocities $[UVW]_{body}$ provide propeller frame freestream velocities $[UVW]_{prop}$.

$$X_p = \rho\omega_{prop}^2 D^4 C_T \qquad\qquad \mathcal{L}_p = -\frac{\rho\omega_{prop}^2 D^5 C_P}{2\pi}$$

$$Y_p = \rho\omega_{prop}^2 D^4 C_{Z_\alpha}\beta_{prop} \qquad \mathcal{M}_p = \rho\omega_{prop}^2 D^5 C_{n_\alpha}\beta_{prop}$$

$$Z_p = \rho\omega_{prop}^2 D^4 C_{Z_\alpha}\alpha_{prop} \qquad \mathcal{N}_p = -\rho\omega_{prop}^2 D^5 C_{n_\alpha}\alpha_{prop}$$

Equation 4-D Propeller Forces and Moments

where,

ω_{prop}	Propeller speed (rad/s)
D	Propeller diameter
ρ	Density
$J = U_{prop}/nD$	Advance Ratio
C_P	Power Coefficient
C_T	Thrust Coefficient for J
C_{Z_α}	Normal Force Derivative
C_{n_α}	Pitching Moment Derivative

Calculating the thrust forces and propeller moments from Equation 4-D, the forces and moments are transformed into the propeller frame with DCM_{bp}. The total moments acting on the aircraft CG due to the propeller is obtained from the sum of the thrust torque and propeller moments given by,

$$\begin{bmatrix} L_p \\ M_p \\ N_p \end{bmatrix}_{\substack{body \\ frame}} = \text{Distance}(\text{Prop CG}, \text{Body CG}) \times \begin{bmatrix} X_p \\ Y_p \\ Z_p \end{bmatrix}_{\substack{body \\ frame}} + \begin{bmatrix} \mathcal{L}_p \\ \mathcal{M}_p \\ \mathcal{N}_p \end{bmatrix}_{\substack{body \\ frame}}$$

Finally, the sum of the forces and moments of the individual propellers calculated are output from the propeller dynamics module. Also, thrust coefficients (TC_{aft}, TC_{fwd}) that represent the relationship of rotor speed to thrust are computed for reference and model evaluation. The thrust coefficient is given by,

$$TC = \rho D^4 C_T \quad \text{such that} \quad \text{Thrust} = TC \times \omega^2$$

4.2.5 Integration

As outlined in the flight dynamics model architecture, the developed aerodynamics and propeller dynamics modules were summed together along with the gravitational module and control inputs module. The gravitational module resolves the weight force acting in the body axis as per the orientation of the aircraft, which is summed with the aerodynamics and propeller forces. The net forces and moments are then input into the equations of motion, which computes the flight states.

To account for the actuation effects, linear actuator models were implemented between control input demands and the flight dynamics model. Control input demands from the actuation models are then computed by the aerodynamics and propeller dynamics modules to compute relevant aircraft body forces and moments. Note that while the provision for incorporating actuation dynamics was implemented, the actuation system dynamics were set to be ideal to simplify the conceptual study.

4.3 Model Validation

The dynamics model was finally tested for proper function and performance by running different trimming routines and observing the flight dynamic responses and variations with different input signals. The control inputs that met the trim conditions were obtained, and the force and moments output from each module of the flight dynamics model were individually observed to assess proper functioning of every module.

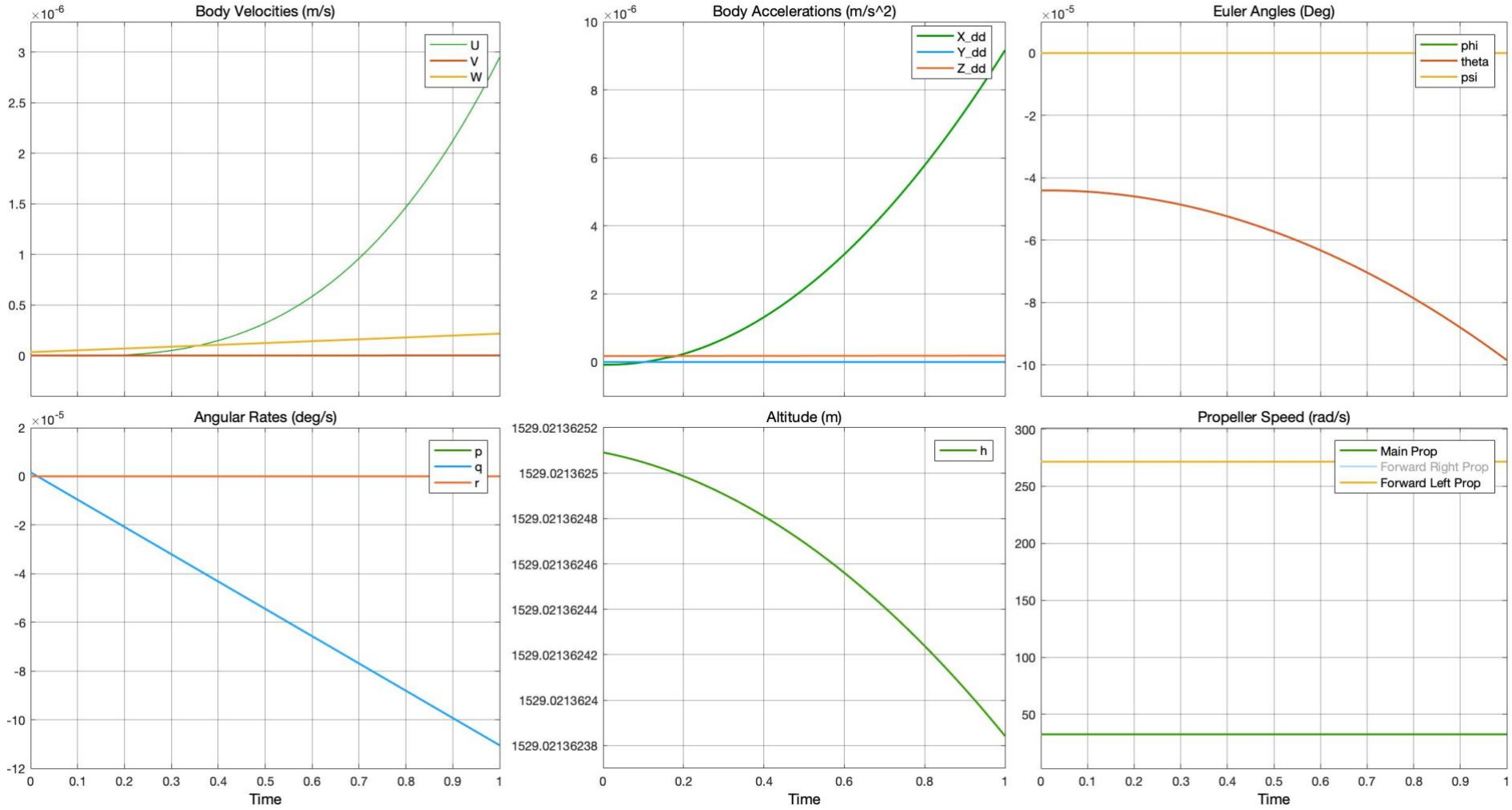
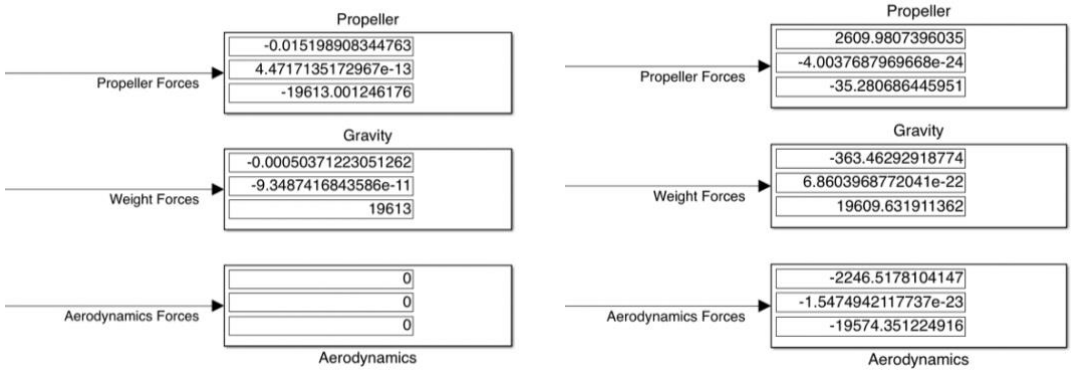


Figure 4-7 Hover Phase Trim

From the above figure, the hovering phase of the aircraft has been simulated. The trimmed state input has tilt-rotor angle at $\Gamma = 90°$. The smaller forward propellers rotate faster than the larger aft rotor, i.e., the propellers generate forces relevant to their geometry, indicating correct propeller dynamics modelling. When assessing the individual module forces and moments, the aerodynamics can be observed to produce no influence, whereas the aircraft is maintaining flight with the help of the propeller thrust.

From Figure 4-8, the cruise phase flight parameters can be observed. Here, the tilt-rotor angle is fixed at $\Gamma = 0°$, and the aft rotor can be identified to be still. The forward propellers are at 263 rad/s, propelling the aircraft to maintain its cruise velocity. The pitch attitude slightly greater that $1°$ signifies the aircraft's orientation in order to have the appropriate angle of attack for maintaining flight with aerodynamic lift. It could be noted that the delta wing configuration of the aircraft could be a potential reason for having a 1° pitch attitude to trim in cruise flight. From the right, it can be observed that the primary lift is aerodynamically induced, and the propeller primarily produces forward thrust and no lift, which validates the correctness in terms of flight dynamics.



Hover Cruise

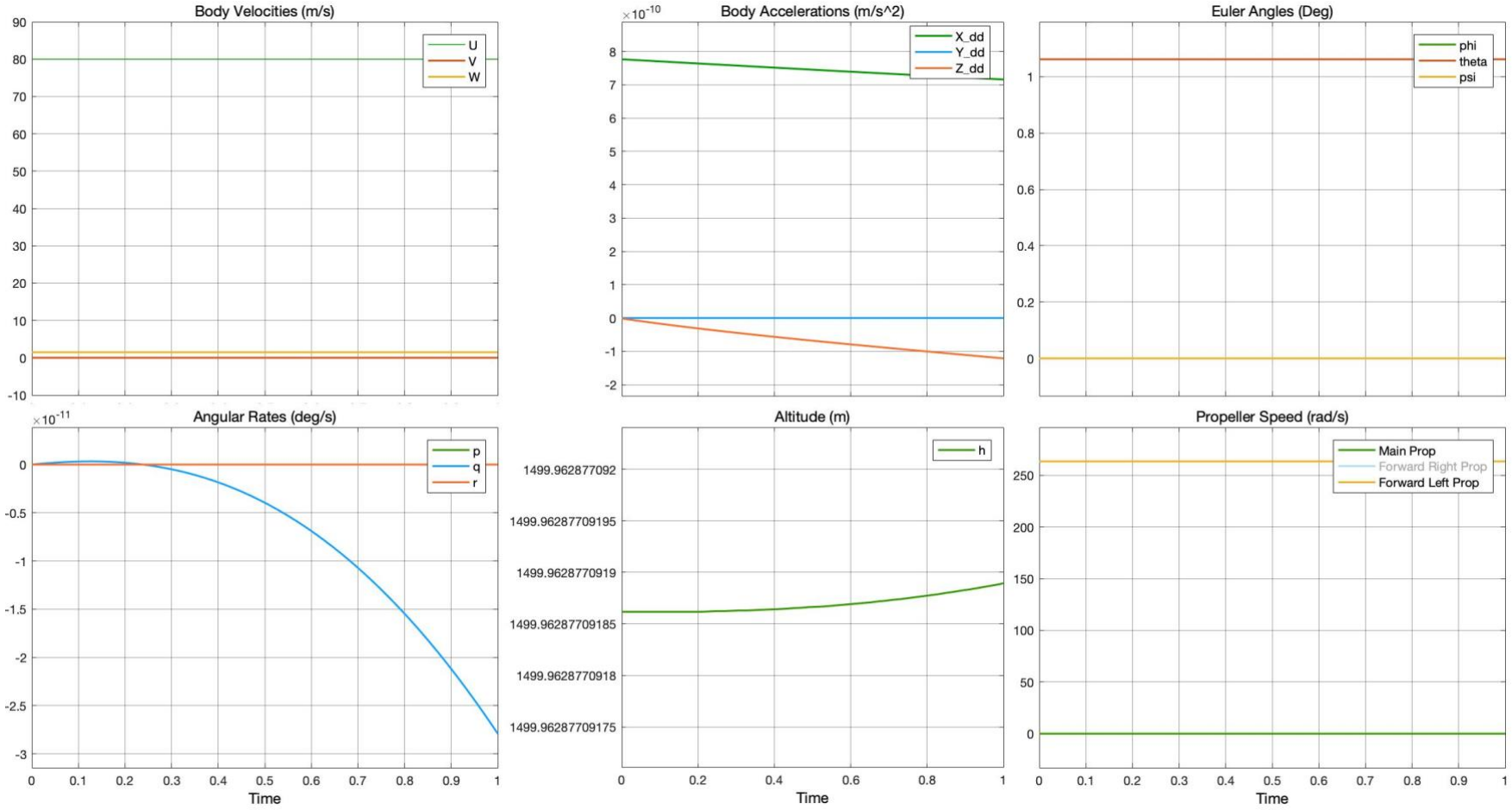


Figure 4-8 Cruise Phase Trim

4.4 Conclusion

A reasonable representation of the flight dynamics model is critical for the development of flight controllers, and the author has been capable of successfully modelling and improvising the flight dynamics model, based on the work of Gonzalo Valdes[32].

A modular architecture that provisions for iterative updates was drafted, based on which the individual modules comprising of aerodynamics and propeller dynamics were developed. The developed dynamics model was tested to perform reasonably well in the cruise and hover phases, which were the primary focus of the study. The dynamics model is to be considered preliminary and baseline, but significant for understanding and defining resolution of the control problem associated with the aircraft.

The assumptions were acknowledged, and the developed model was deemed relevant for the objectives of this research. While improving the fidelity of the model can provide an accurate representation of the non-linear flight dynamics, the current assumptions remain insignificant for validating the proposed conceptual control technique. Moreover, dynamics modelling can become an extensive process, and the scope of the project limits it to the achieved grade of accuracy.

With a baseline flight dynamics model developed, the conceptually validated control technique could be tested for functionality based on an actual aircraft model. The subsequent chapters outline the controller development based around the developed flight dynamic model.

5 DIFFERENTIAL FLATNESS CONTROLLER DESIGN

The ultimate outcome of this study is to develop flight control software that utilises the differential flatness control technique to achieve stability and control on a tilt-rotor VTOL aircraft, the Vision Volante. With an adequate representation of the flight dynamics obtained, the controller development was strategically executed to validate the viability of the proposed controller in a modelled practical system. The previously performed conceptual validation, formed the basis for the study, and the flight dynamics model was developed to understand the particular aircraft's aerodynamic and propeller dynamics to devise suitable controllers.

Controller development is an extensive process, especially when considering the VTOL flight dynamics being highly complex and nonlinear. Achieving stability and control over the entire flight envelope is critical, and requires extensive testing and validation routines to declare the controller practical and certifiable. Since this study involved proving the viability of a novel control technique, certain assumptions and simplifications were accounted to obtain a baseline controller.

Design of the controller begins with defining the controller architecture, whereby each module is identified and assigned a function in the system. The control architecture outlines the signals used, and the components required to be implemented in the individual modules. The designed architecture comprises of the differential-flatness module, disturbance controller module and control allocation module which are explained in the subsequent sections. Finally, all the controllers are tested for performance and stability in linear and non-linear representations of the flight dynamics.

5.1 Differentially Flat Equations

A non-linear system is considered differentially flat when its inputs can be represented by a function of its state differentials [3]. With inputs obtainable from state differentials, it enables control of the system for a known desired trajectory [10]. This technique of predicting control inputs from the trajectory state differentials has been shown to be of benefit in section 3.2.1, and the concept is scaled for tilt-rotor flight control as elucidated in the following sections.

Due to the nonlinear nature of aerodynamics and propeller dynamics, obtaining a function of the inputs for state differentials is impractical, i.e., aerodynamics and propeller dynamics are not differentially flat. However, the aspect of an aircraft that remains differentially flat is its kinematics, and the controller developed in this study utilises this property. The kinematics of the system is similar to the PVTOL aircraft, except that the control input sources are different. The PVTOL dynamics (see equations 3.7 – 3.12) for the Vision Volante can be expressed as,

$$\dot{U} = -g\sin\theta - qW + \frac{1}{m}\left(T_{fwd}\cos\Gamma + F_{D_{aero}}\right) \tag{5.1}$$

$$\dot{W} = g\cos\theta + qU - \frac{1}{m}\left(T_{fwd}\sin\Gamma + T_{aft} + F_{L_{aero}}\right) \tag{5.2}$$

$$\dot{q} = \frac{1}{I_{yy}}\left(T_{fwd}\sin\Gamma \times l_{fwd} + T_{aft} \times l_{aft} + M_{y_{aero}}\right) \tag{5.3}$$

where $F_{D_{aero}}$ and $F_{L_{aero}}$ are lift and drag forces resolved from the F_{aero} vector as shown in Figure 5-1, and T_{fwd} , T_{aft} are thrust forces from the forward and aft propellers. It is important to note that the propeller forces in the propeller frame x-axis is only considered as thrust, and the other component are neglected due to their negligible influence. The sum of the aerodynamic and propeller thrust forces can then be represented as,

$$F_{x_{DF}} = F_{X_{prop}} + F_{D_{aero}} = T_{fwd}\cos\Gamma + F_{D_{aero}} \tag{5.4}$$

$$F_{z_{DF}} = F_{Z_{prop}} + F_{L_{aero}} = T_{fwd}\sin\Gamma + T_{aft} + F_{L_{aero}} \tag{5.5}$$

$$M_{y_{DF}} = M_{y_{prop}} + M_{y_{aero}} = T_{fwd}\sin\Gamma \times l_{fwd} + T_{aft} \times l_{aft} + M_{y_{aero}} \tag{5.6}$$

Finally, the inverted kinematics can be represented as,

$$\begin{bmatrix} F_{x_{DF}} \\ F_{z_{DF}} \\ M_{y_{DF}} \end{bmatrix} = \begin{bmatrix} \left(\dot{U} + g\sin\theta + qW\right) \times m \\ -\left(g\cos\theta + qU - \dot{W}\right) \times m \\ -\dot{q} \times I_{yy} \end{bmatrix} = \begin{bmatrix} \left(\dot{U} + g\sin\theta + \dot{\theta}W\right) \times m \\ -\left(g\cos\theta + \dot{\theta}U - \dot{W}\right) \times m \\ -\ddot{\theta} \times I_{yy} \end{bmatrix} \tag{5.7}$$

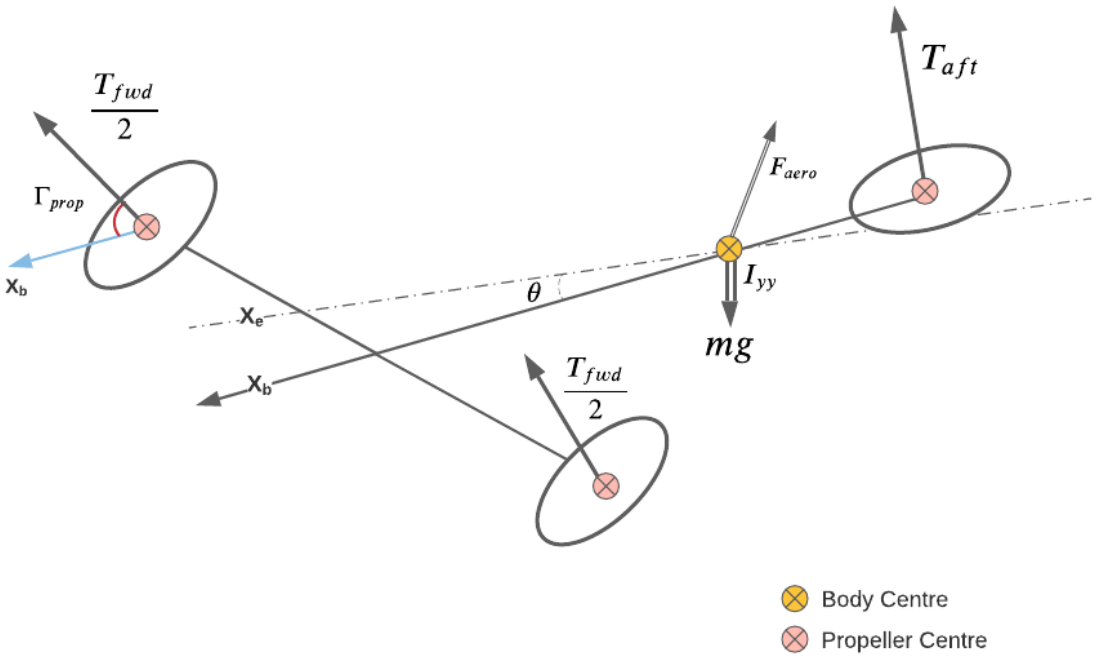


Figure 5-1 Airframe Forces Representation

From the equations, it can be identified that the control inputs can be computed from the desired trajectory state inputs, and the differentially flat system characteristics enable predicting the required forces and moments to achieve the state outputs. The forces and moments are then to be resolved in order to generate propeller and aerodynamic surface control inputs. This precalculated forces and moments along with disturbance controllers should improve the trajectory tracking performance of the flight controllers. The theoretical performance improvement was tested in the author's work as described in section 6.3.2. Note that, this study focusses on the longitudinal dynamics, and a similar work in lateral dynamics should be carried out to obtain a complete 6 DoF controller.

5.2 Control Architecture

Formulating a control architecture that describes the different control modules and its associated signals is an important step in controller development. A meticulously developed control architecture could potentially expedite controller development by providing a fundamental framework that meets the identified functional requirements. The primary aim of the control architecture is to outline the functional units with which the control demands from the differential flatness equations and the disturbance controllers can be amalgamated to meet the desired flight control performance.

For a VTOL aircraft with the ability to vector thrust between vertical and forward directions, the flight path can be categorised into three separate flight phases – Hover, Transition and Cruise. During Hover, the vertically directed thrust would generate sufficient lift to achieve flight enabling the aircraft to take-off and land without a runway. Cruise phase is when the aircraft maintains flight with aerodynamic lift. This phase is where the aircraft achieves high-speed flight with optimal efficiency. The transition phase is the intermediate phase where the aircraft transfers its lift between vertical thrust and aerodynamic lift and vice versa. With the three flight phases holding distinct dynamic characteristics, a control architecture that is robust and operable in all conditions is critical.

Additionally, design considerations to account for the over-actuated airframe and the unconventional controller demands (forces and moments) are inevitable. These factors led to devising a control architecture that was different from the architecture developed for conventional flight controllers as well as the PVTOL aircraft (see section 3.2.2).

The devised controller architecture for the Vision Volante Longitudinal Flight Controller is given below,

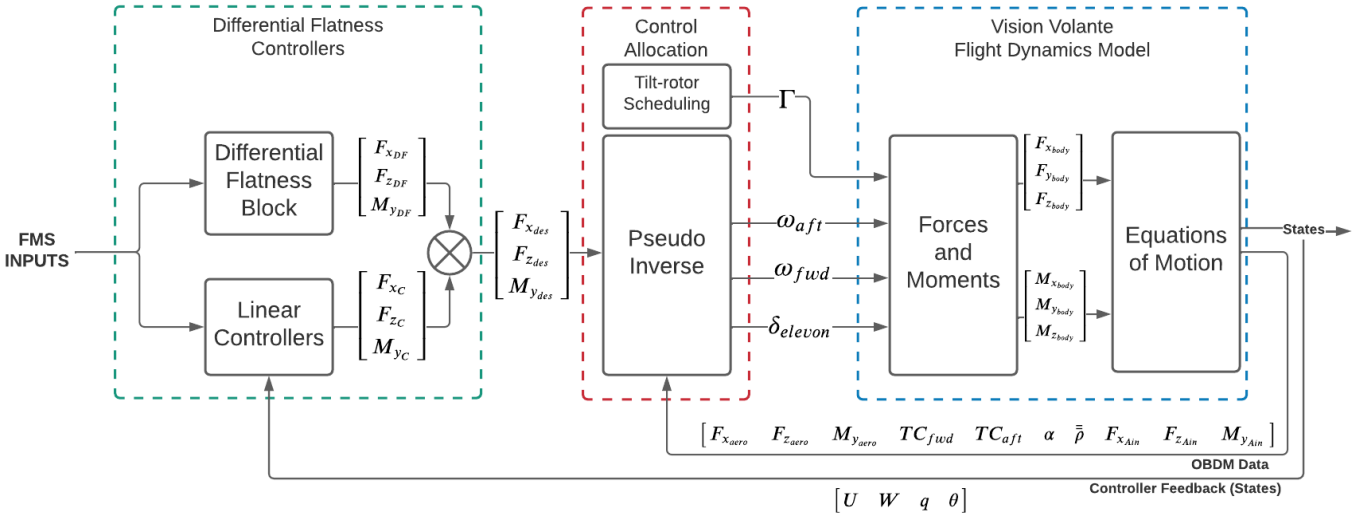


Figure 5-2 Differential Flatness Flight Control Architecture

Similar to the PVTOL controller, the differential flatness controller is based on the inverted kinematics and disturbance controllers working in tandem. However, the controllers compute the desired forces and moments to be obtained by the aerodynamics and propellers, unlike the PVTOL controller which computes the control inputs directly. The latter was made possible due to the simple dynamics structure of the PVTOL, and cannot be realised in the Vision Volante. The forces and moments signal vector required to achieve the desired trajectory, input by the Flight Management System (FMS) can be computed by the differentially flat inverted dynamics, which is then feedforward and added with feedback disturbance controllers to generate the virtual control demands. The feedforward inputs from the inverted dynamics reduces the linear controller effort by predicting the nominal control inputs required to achieve the desired trajectory. The linear controllers ensure that the aircraft maintains the desired trajectory amidst disturbances, which are oblivious to the inverted kinematics equations, ensuring robustness throughout the flight envelope while having better performance characteristics. The summation of the control demands from the inverted dynamics and disturbance controllers is identified as the controller demand.

The controller demands are then allocated into appropriate control inputs with the help of the control allocation module. A control allocation module had to be incorporated in the control scheme to,

1. compute actuator input demands from virtual (forces and moments) controller demands
2. account for varying control input effectiveness across the flight envelope, especially during transition
3. compute optimal input demands for over-actuated airframe, i.e., multiple control inputs for the same DoF

The control allocation comprises of three units, pseudo inverse control allocation, non-linear inverse control allocation and tilt-rotor scheduling. The tilt-rotor scheduling is a velocity-based lookup table that tabulates the desired tilt-rotor angle for different forward velocities, and dynamically sets the tilt-rotor angle. The pseudo inverse control allocation unit allocates the appropriate control inputs based on the desired force commands by solving for the pseudo inverse control input matrix. Due to negligible aerodynamic influence in hover, a non-linear inverse control allocation based on resolved hover dynamics is implemented for operation in hover phase. Further information on the implemented control allocation is explained in the following section.

A key unit that facilitates the operation of the control allocation unit is the On-Board Dynamics Model (OBDM). The OBDM is a reduced flight dynamics model operating as an independent unit, that computes the total aerodynamic forces and moments, propeller thrust coefficients and elevon forces and moments based on the acquired current flight states. These computed data are then fed into the control allocator for transforming the controller virtual demands to control input demands. In this study, the OBDM has been modelled as an atomic unit sampled at the same time as the simulation with a timestep delay in the data line to the control allocator. This time delay ensures that the control allocator computes control input demands based on the previous flight condition (see Appendix 7.2C.3).

Due to project time constraints, the development of the proposed novel control architecture was confined to longitudinal dynamics, and some approximations were made to restrict lateral dynamics to influence the development. The control development carried out by the author had lateral control surfaces (δ_{Aout} δ_{rudder})

fixed to its neutral setting, and all the differential rotor speed inputs and tilt-rotor inputs that would affect the lateral dynamics were ganged. These considerations significantly eased controller development as the controls problem was restricted to the longitudinal dynamics only.

$$\boldsymbol{u}_{full} = \begin{bmatrix} \Gamma_L \\ \Gamma_R \\ \omega_{aft} \\ \omega_{fwd_L} \\ \omega_{fwd_R} \\ \delta_{Ain} \\ \delta_{Aout} \\ \delta_{rud_L} \\ \delta_{rud_R} \end{bmatrix}$$

Longitudinal and Lateral Control Inputs

$$\boldsymbol{u}_{red} = \begin{bmatrix} \Gamma \\ \omega_{aft} \\ \omega_{fwd} \\ \delta_{Ain} \end{bmatrix}_{\substack{\Gamma_L=\Gamma_R \\ \omega_{fwd_L}=\omega_{fwd_R}}}$$

Longitudinal Control (Ganged) Inputs

The longitudinal control effector inputs by the control allocation unit based on controller demands are input into the flight dynamics model which computes the aircraft states. The control architecture designed is operable across all the flight phases because of the forces and moments based controller demand signals and OBDM based control allocation module, generate apt control inputs based on control effector effectiveness. However, the linear disturbance controllers are to be gain-scheduled for different operating points to ensure that the disturbance controllers function as desired throughout the flight envelope.

5.3 Control Allocation

The Vision Volante is an over-actuated system with several control inputs for controlling the aircraft. Due to the non-linearities in control input effectiveness and its coupling effects, having individual controllers manage the various redundant control inputs would become complex. Thus, a technique that would allocate the appropriate control inputs based on an abstracted control signal (virtual

commands) was deemed necessary, and is the primary idea behind control allocation.

With reference to the Vision Volante, its longitudinal dynamics can be controlled with tilt-rotor angles (Γ), aft rotor speed (ω_{aft}), forward rotor speed (ω_{fwd}) and inner aileron/elevon deflection (δ_{Ain}). Among the four inputs, it is important to note that δ_{Ain} is an aerodynamic control surface, and it is dependent on the airspeed for its effectiveness. This implies that it cannot be used to control the aircraft while hovering, but is vital for control when in cruise flight. Similarly, differential rotor speed enables attitude control in hover, but they are not practical in cruise phase. Such stark variations in control input effectiveness with flight condition are an important concern when allocating control inputs. An allocation unit that accounts for the flight conditions, and adapts its algorithm for optimal performance was deemed indispensable.

As mentioned earlier, since the redundant control inputs alter the same dynamics, a unique set of virtual control commands are necessary to design controllers. The virtual control commands can then be used to identify the appropriate control inputs demands, using the control allocation algorithm. From the control architecture (see Figure 5-2), the virtual controller demands are conveniently decided by the differential flatness module and the disturbance controller modules which are; desired forces and moments ($F_{x_{des}}\ F_{z_{des}}\ M_{y_{des}}$) the propellers and aerodynamics control surfaces should achieve. Resolving the forces and moments to allocate for control input signals are computed with the help of the pseudo-inverse algorithm, non-linear inverse, on-board dynamics model, and tilt-rotor scheduling incorporated in the control allocation module.

While over-actuation is eliminated with the control input approximation shown above, control allocation was necessary for computing actuator inputs from virtual controller demands. Additionally, further phases of the proposed controller development involving lateral dynamics will require control allocation to account for the over-actuated system, and hence the author developed a control allocation scheme which can serve as a baseline for further extensions.

5.3.1 Linear Control Allocation

The control allocation problem is to solve a system of algebraic equations with respect to control input vector from the system state equations and virtual control commands. Mathematically, the non-linear control allocation problem can be outlined as,

$$\dot{x} = f(t, \boldsymbol{x}, \boldsymbol{v}) \qquad\qquad v = h(t, \boldsymbol{x}, \boldsymbol{u})$$

where $t \geq 0$ is time, $x \in \mathbb{R}^r$ is the state vector, $u \in \mathbb{R}^m$ is the physical actuator input vector and $v \in \mathbb{R}^n$ is the virtual control commands vector [21]. Assuming that the virtual control commands are issued by a controller, its functional representation could be,

$$\boldsymbol{v}_c = k(t, x)$$

In other words, the control allocation problem can be described as the resolution of algebraic equations for control input vector $\boldsymbol{u}$ to satisfy $\boldsymbol{v} = \boldsymbol{v}_c$. For over-actuated systems, this leads to multiple solutions necessitating a cost function to find an optimal solution based on defined constraints or conditions. The cost function constraints could be based on the physical actuation parameters (deflection limits, rate limits, etc.) and functional requirements of the system.

However, resolving for the non-linear control allocation problem by running optimisation algorithms to solve control input vectors for minimum cost is computationally demanding and complex in the likelihood of non-convergence. This problem was overcome with linear control allocation, exploiting the existing analytical methods that require minimal computation power and less implementation efforts. But this means, the nonlinear control allocation problem must be dynamically linearised across the flight envelope to be resolved by the linear control allocation techniques. Hence, by linearising the function h, $\boldsymbol{v} = h(t, x, u) \approx B\boldsymbol{u}$, conventional linear control allocation techniques could be exploited [21].

5.3.1.1 Pseudo Inverse Control Allocation

Linear control allocation simplifies the nonlinear control allocation problem by characterising the control input to be linear at a particular linearisation point, and the computed control input would be satisfactory as long as the linearisation assumption is valid. Expressing the virtual commands as a linear combination of the physical control inputs would outline the linear control allocation problem, and it can be given as,

$$\boldsymbol{v} = B\boldsymbol{u} \tag{5.8}$$

Where $B \in \mathbb{R}^{m \times n}$ considering $v \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$. B is the control effectiveness matrix, which describes the relationship between the control inputs and the virtual commands. The B matrix can be obtained by solving for differentials of the forces and moments for controller input demands, which would be discussed in the section 5.3.1.2.

In practice, actuators have operable limits within which the computed control input demands must lie. $\overline{\boldsymbol{u}}$ and $\underline{\boldsymbol{u}}$ are assigned upper and lower limits of controller actuation. Mathematically, the constraint can be described as,

$$\underline{\boldsymbol{u}} \leq \boldsymbol{u} \leq \overline{\boldsymbol{u}}$$

For an over-actuated system, the existence of multiple possible solutions necessitates the requirement for a cost function $J(\boldsymbol{u})$ which accounts for the different criteria and costs to rank different combinations of $\boldsymbol{u}$**.** Linear control allocation facilitates the exploitation of analytical methods, and as per Harkegard [38], the appropriate cost function is the one that minimises or prevents actuator saturation and can be executed analytically. The control allocation problem can then be described as,

$$\min_{u} J(\boldsymbol{u}) \quad \text{subject to} \quad v_c - B\boldsymbol{u} = 0 \tag{5.9}$$

and the identified cost function is given by,

$$J(\boldsymbol{u}) = \frac{1}{2}\boldsymbol{u}^T\boldsymbol{u} \tag{5.10}$$

The above given cost function, when minimised, minimises the total actuator deflection, preventing actuator saturation by a solution that requires minimal actuation effort. In other words, minimising $\frac{1}{2}u^T u$, obtains a minimum control input demand, which can be classified as minimum distance, minimum energy or minimum deflection approach. The cost function can be analytically represented as,

$$\boldsymbol{u} = B^T(BB^T)^{-1}\boldsymbol{v} \tag{5.11}$$

The above analytic solution is key for this control allocation technique, because it enables precomputation of the matrix $B^T(BB^T)^{-1}$ without significant computational load unlike optimisation algorithms. This is called the pseudoinverse method, since $B^T(BB^T)^{-1}$ represents the pseudoinverse of B and is one of the most used methods in practical control allocation [21].

Another variation of the same method exists, called the Weighted Pseudoinverse methods [21]; [38]. This method utilises a positive diagonal matrix in the cost function to scale the different inputs, facilitating the capability to rank relevance or cost of the different physical control inputs. However, due to limited project scope, the author did not investigate the same technique in this study.

5.3.1.2 Control Input Matrix Derivation

The pseudo-inverse method can be an effective analytical control allocation method, provided an accurate representation of the *B* matrix can be obtained. From the non-linear dynamics of an aircraft, the state-space representation of its linearised dynamics is of the form,

$$\dot{x} = Ax + B^x\boldsymbol{u} = Ax + B^v\boldsymbol{u} \tag{5.12}$$

The control input matrix represents the effects of different control inputs on the aircraft states at a particular operating point. B^x represents the control input

effectiveness on the flight states, whereas B^v represents the control input effectiveness on the virtual flight commands, in this case $[F_x\ F_z\ M_y]$.

Since there exists an actual representation of the relationship between the forces and control inputs, partial differentiation of the forces by control input would provide the control input matrix [21]; [39]. This can be represented as,

$$B^v u = \begin{bmatrix} \frac{\partial F_x}{\partial_{T_{aft}}} & \frac{\partial F_x}{\partial T_{fwd}} & \frac{\partial F_x}{\partial \Gamma} & \frac{\partial F_x}{\partial \delta_{Ain}} \\ \frac{\partial F_z}{\partial_{T_{aft}}} & \frac{\partial F_z}{\partial T_{fwd}} & \frac{\partial F_z}{\partial \Gamma} & \frac{\partial F_z}{\partial \delta_{Ain}} \\ \frac{\partial M_y}{\partial_{T_{aft}}} & \frac{\partial M_y}{\partial T_{fwd}} & \frac{\partial M_y}{\partial \Gamma} & \frac{\partial M_y}{\partial \delta_{Ain}} \end{bmatrix} \begin{bmatrix} T_{aft} \\ T_{fwd} \\ \Gamma \\ \delta_{Ain} \end{bmatrix} \tag{5.13}$$

In order to simplify this partial differentiation of the forces and moments for control inputs, the tilt-rotor angle was decided to be scheduled based on trim point data obtained from the flight dynamics model. This is further described in the later section 5.3.2. Also, the rotor speed control input (ω_{aft} , ω_{fwd}) is replaced with rotor thrust (T_{aft} , T_{fwd}) to simplify the *B* matrix identification, since the relationship between rotor speed and forces could complicate the allocation scheme (see section 4.2.4.2). Removing the tilt-rotor angle (Γ) control input from equation 5.13, and differentiating force and moments equations given in 5.4, 5.5 and 5.6, the *B* matrix is defined as,

$$v = B^v u = \begin{bmatrix} 0 & \cos\Gamma & \left(\sin\alpha\, C_{L_{\delta_{Ain}}} - \cos\alpha\, C_{D_{\delta_{Ain}}}\right) S\bar{\bar{\rho}} \\ -1 & -\sin\Gamma & \left(-\sin\alpha\, C_{D_{\delta_{Ain}}} - \cos\alpha\, C_{L_{\delta_{Ain}}}\right) S\bar{\bar{\rho}} \\ -l_{main} & l_{fwd}\sin\Gamma & S_{ref} C_{ref} \bar{\bar{\rho}} C_{m_{\delta_{Ain}}} \end{bmatrix} \begin{bmatrix} T_{aft} \\ T_{fwd} \\ \delta_{Ain} \end{bmatrix} \tag{5.14}$$

The given *B* matrix represents the control input effectiveness for the current flight conditions, which are obtained from the Onboard Dynamics Model (OBDM). While the aerodynamic forces and moments comprise of several control inputs and flight parameters, the relevant term in effectiveness matrix is the one that is dependent on the desired control input (δ_{Ain}). The $\sin\alpha$ and $\cos\alpha$ terms arise from the direction cosine matrix that resolves the wind frame aerodynamic forces into the body frame.

The OBDM computes elevon aerodynamic coefficients ($C_{L_{\delta_{Ain}}} C_{D_{\delta_{Ain}}} C_{m_{\delta_{Ain}}}$), while the dynamic velocity ($\bar{\bar{\rho}}$) and angle of attack (α) are obtained from aircraft state outputs. Along with the scheduled tilt-rotor angle (Γ) inputs, the *B* matrix can be precomputed for a particular flight condition, providing a close estimate of the control input effectiveness. The author has assumed ideal actuation and motor dynamics, which in the practical sense cannot hold true, which is a limitation. However, with a proper dynamic modelling of the actuation, a similar approach can be carried out to implement this method. The fidelity of the OBDM is another limitation, which is dependent on the extent of the flight dynamics modelling effort. In the authors work, the OBDM is a replica of the flight dynamics model used to simulate the aircraft, i.e., in practical sense it is an identical replica of the actual flight dynamics. Indeed, it is impractical to model complete flight dynamics, and is to be noted as an assumption in this study.

5.3.1.3 Implementation

From the controller architecture, the differential flatness module along with the disturbance controllers compute the forces and moment demands required for flying the desired trajectory. This implies that the controller demands account for all components, including gravitational, aerodynamics and propellers. The control allocation scheme described earlier only computes control effector inputs for the forces and moments the control effectors should attain.

To compute for the desired control input forces and moments, the control allocation reduces the controller demanded inputs with current aerodynamic forces and moments without elevator deflection induced forces and moments. The aerodynamic forces and moments associated with the aircraft for a given flight condition can be obtained from the OBDM, which enables in deriving the forces and moments the control allocation scheme should resolve for appropriate control input signals.

The B matrix represents the relation between different types of control inputs, which are propeller thrusts and elevator deflection in this case. The matrix represents the effectiveness of the different control inputs for the same set of virtual commands. The dissimilar units could induce problems due to the different

order of magnitudes embedded in the unit representation, and to avoid this concern, a normalisation diagonal matrix had to be introduced. This normalisation fixed the input range from -1 to 1 regardless of the control input. The normalisation matrix E is defined, whose diagonal term is given by the maximum absolute of the actuator limit [21], i.e.

$$E_{ii} = \max(\overline{|u_i|}, \underline{|u_i|})$$

The normalised control input matrix ($\hat{B}$) is given by,

$$\hat{B} = B^v \times E$$

And the normalised control inputs computed with control allocation scheme can then be multiplied with E to obtain the control input demands in the right unit representation.

Lastly, the rotor speed control inputs demands $(\omega_{aft}\ ,\ \omega_{fwd})$ are obtained by the dividing the computed rotor thrust demands ($T_{aft}\ ,\ T_{fwd}$) and the thrust-rotation coefficients ($TC_{aft}\ ,\ TC_{fwd}$). The thrust-rotation coefficients (see section 4.2.4.2) are obtained from the OBDM, which computes them based on the current flight states. Since the author's work focussed only on the longitudinal control, the computed forward thrust (T_{fwd}) was halved for calculating the individual forward rotor speed demand, overseeing the yaw control dynamic using differential thrust. The thrust force representation, which when divided by thrust coefficient $(TC_{aft}\ ,\ TC_{fwd})$, results in the square of propeller rotation speed as shown below,

$$\omega_{\substack{aft \\ fwd}} = \sqrt{\frac{T_{\substack{aft \\ fwd}}}{TC_{\substack{aft \\ fwd}}}}$$

With all the described constituents, the control allocation scheme allocates the right control input signals to achieve the controller demands. While the pseudo inverse technique is computationally simple and supports reconfiguration in the event of actuator failure, the approach has some caveats. Firstly, it does not account for actuation saturation and secondly, it assumes linearity and is

dependent on the obtained *B* matrix. However, the pseudo inverse control allocation is an efficient and optimal analytical method that performs satisfactorily. Since the control allocation implementation accounts for the aerodynamic forces and moments based on the OBDM data, the CA facilitates a unified controller architecture that could operate throughout the VTOL flight envelope. With the current technology, implementing a control allocation scheme with an onboard dynamics model is considered feasible. The control allocation module was modelled in SIMULINK as a MATLAB function, and the code is available at 7.2Appendix B.

5.3.2 Non-linear Control Allocation

Fundamentally, control allocation is a nonlinear problem and solving it in the nonlinear domain would eliminate the need for dynamics to remain within the bounds of the linearity as required in linear allocation techniques. As elucidated above, the pseudo inverse operates based on a linearised control input matrix, which is then solved for an optimal solution based on a linear algebra. However, at low velocities, with the tilt-rotor angle at $90°$, the *B* matrix cannot be inverted due to singularity. This imposes a major concern when allocating using pseudo-inverse technique, as the control input demands would be unreasonable. Additionally, the influence of aerodynamic surfaces at low speeds are minimal, deeming the control input matrix to have the elevator column zero, making it a non-square matrix and a singular matrix as shown below.

$$B_{hover}^{v} = \begin{bmatrix} 0 & \cos\Gamma & \left(\sin\alpha\, C_{L_{\delta_{Ain}}} - \cos\alpha\, C_{D_{Ain}}\right) S\bar{\bar{\rho}} \\ -1 & -\sin\Gamma & \left(-\sin\alpha\, C_{D_{\delta_{Ain}}} - \cos\alpha\, C_{L_{\delta_{Ain}}}\right) S\bar{\bar{\rho}} \\ l_{main} & l_{fwd}\sin\Gamma & S_{ref} C_{ref} \bar{\bar{\rho}} C_{m_{\delta_{Ain}}} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ -1 & -1 & 0 \\ X_{main} & X_{fwd} & 0 \end{bmatrix}$$

SOLVING FOR NON-LINEAR CONTROL ALLOCATON

From equations 5.4, 5.5, and 5.6, the longitudinal forces and moments without the aerodynamic terms can be given as,

$$F_x = T_{fwd}\cos\Gamma \tag{5.15}$$

$$F_z = T_{fwd} \sin \Gamma + T_{aft} \tag{5.16}$$

$$M_y = T_{fwd} \sin \Gamma \times l_{fwd} + T_{aft} \times l_{aft} \tag{5.17}$$

At low-speeds in hover, with the tilt-rotors set to $90°$, the equations would be given as,

$$F_x = 0 \qquad F_z = T_{fwd} + T_{aft} \qquad M_y = T_{fwd} l_{fwd} + T_{aft} l_{aft}$$

With minimal influence of non-linear aerodynamics and forward force being a function of attitude (not thrust), the control allocation solution can be obtained from the above equations as,

$$T_{aft} = \frac{M_y - F_z \, l_{fwd}}{l_{fwd} - l_{main}} \tag{5.18}$$

After solving for T_{aft},

$$T_{fwd} = \sqrt{F_x^2 + \left(F_z - T_{aft}\right)^2} = F_z - T_{aft} \tag{5.19}$$

since $F_x = 0 \ \because \Gamma = 90°$

The above described control allocation technique was implemented along with Hover phase controllers (see section 6.2.2) at velocities below 20 m/s. The nonlinear inverse scheme remains valid below 20 m/s since there is less aerodynamic influence as well as the tilt-rotor deflections are ~$90°$, reducing the propeller force interactions in the x axis.

Both the control techniques were implemented in the same control allocation unit with an if-else condition which switches between the allocation techniques as a function of speed. It is important to note that switching can induce several complications in practical scenarios due to sudden changes in the control input parameters computed. However, for the scope of this research, this limitation was overlooked. Also, this implementation assumes negligible influence of

aerodynamics (up to 20 m/s) based on the preliminary aerodynamic data which could not be the case in practical implementations, which would require high-fidelity aerodynamic data to assess. UAM VTOL aircraft are designed to last for a very short duration at hover and conventionally do not intent to operate high-speed manoeuvres at hover, and would quickly transition to cruise once in air. Hence, this scheme can be a valid assumption to compute rotor speed inputs until it begins cruise flight.

5.3.3 Tilt-rotor angle Scheduling

The forward tilting rotors provide the Vision Volante with its capability of achieving both hover and winged flight. The precise control of the tilt-rotor angle (Γ) is important for transition from hover to cruise flight and vice versa. To simplify control allocation complexity, due to the inconvenience in partially differentiating tilt-rotor angle, a tilt-rotor angle scheduling scheme was implemented.

The tilt-rotor angle schedule functions with a look-up data containing the data of the optimal tilt-rotor angle for a given velocity. This data is obtained by trimming the flight dynamics model for an equilibrium operating point for different velocities with tilting rotor angle as a control input. The look-up data was curated by trimming for velocity range 0-80 m/s with 5 m/s intervals, and the data plot is given below

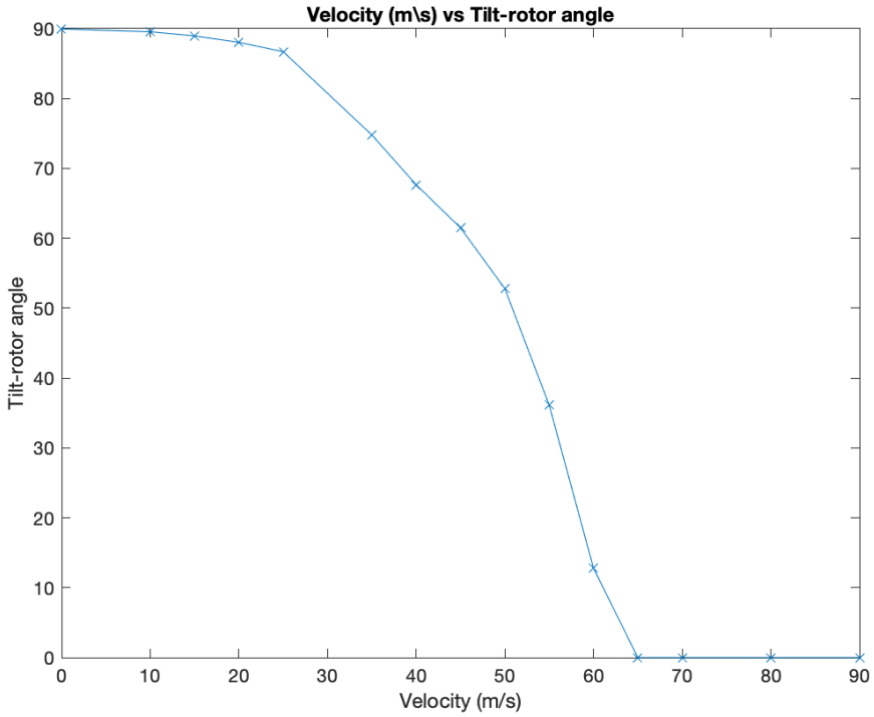


Figure 5-3 Tilt rotor angle (Γ) Scheduling

The 0-10 m/s and 65-90 m/s has the tilt-rotor angles fixed at $90°$ and $0°$ respectively. This was so that at those speed ranges, the hover and cruise flight controller can switch smoothly and operate over a range without the convoluted transition dynamics. This is was considered acceptable due to the short time duration spent in the transition manoeuvre. Trimming of the flight dynamics to obtain equilibrium operating points are described in the later section.

5.4 Dynamics Linearisation

For controller development, the foremost step is to characterise the dynamics of the plant to be controlled. For flight controllers, this is conventionally done by linearising the control inputs and output states of a flight dynamics model at a trimmed operating point. A trim point is defined as the state at which the net forces and moments acting on the aircraft body is zero, i.e., equilibrium.

Once a trim point is identified, the linearised dynamics can be obtained using MATLAB tools such as Model Lineariser, which generates a state space representation of the linear aircraft dynamics. However, identifying the trim operating point for the proposed control architecture raises some complications

due to the control allocation unit prevailing between flight controllers and flight dynamics (plant). This complexity is because the trim optimisation algorithm that searches for an equilibrium condition gets coupled with the control allocation algorithm, hindering the process of finding a valid trim point. Hence, even in the event of finding an approximate trim point, the linearised dynamics with control allocation would result in an erroneous representation of the dynamics for controller development.

However, based on the proposed controller architecture, a solution to design and tune disturbance controllers based on linear dynamics was identified. In Figure 5-2, the architecture indicates the controller to generate forces and moment demands, which the control allocation optimally allocates across the control inputs to achieve the controller demands. Building upon this concept, if the control allocator functions ideally, achieving the virtual demands (forces and moments), the dynamics model can be trimmed and linearised without control allocation, through the propeller and aerosurface forces and moments inputs. The propeller and aerosurface forces and moment inputs can then be interfaced from the aerodynamics and propeller dynamics blocks (see Figure 4-4).

BATCH LINEARISATION

For the development of controllers that operate within all speed ranges, the dynamics had to be linearised at different flight conditions. The author scripted an automated trimming routine which trim the model for different flight conditions and linearises the dynamics for each operating point, based on which the linear controllers could be designed.

The disturbance controllers are intended to compute the appropriate forces and moment demands in order to maintain zero error between the reference command and the feedback. Hence obtaining trim point with $[F_x\ F_z\ M_y]$ as control inputs, would enable deriving the dynamics based on forces and moments. However, gravity being kinematics was summed alongside the trim model control inputs. The trim model structure is given in Figure 5-4.

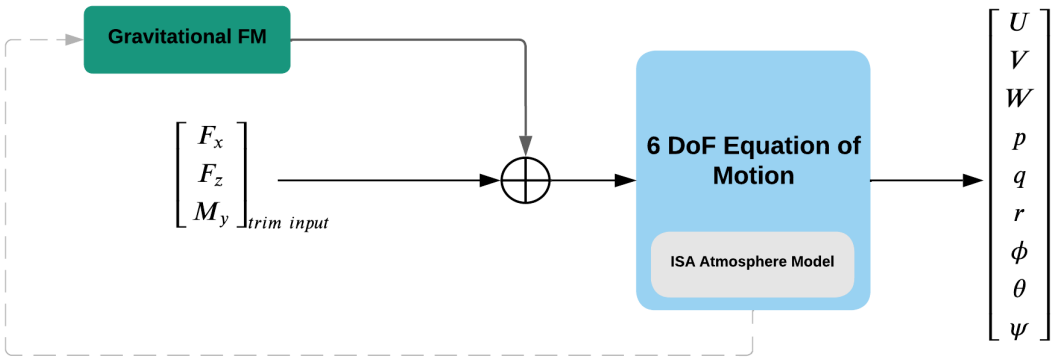


Figure 5-4 Trim Model Structure

Batch trimming was carried out for 51 velocity points between 5 to 80 m/s with a fixed set of trimming conditions as given below

States	U	V	W	ϕ	θ	ψ	p	q	r	X_e	Y_e	Z_e
Known	Vtrim	✓		✓	$[0\ to\ 2°]$	✓	✓	✓	✓			✓
Steady State	✓		✓	✓	✓	✓	✓	✓	✓			✓

Since the control development was focussed on the longitudinal dynamics, the linearised dynamics was then reduced to only represent longitudinal states, i.e., U, W and q. The linearised dynamics for U = 80 m/s are given as,

$$\frac{U}{F_x} = \frac{1}{s} \qquad \frac{W}{F_Z} = \frac{1}{s} \qquad \frac{q}{M_y} = \frac{1}{s} \qquad \frac{W}{M_y} = \frac{80}{s^2}$$

The linear dynamics remained to be identical for all trim points, except for the $\frac{W}{M_y}$ dynamics, which varies with velocity. With the trimming carried out without the aerodynamics and propulsive models, the linear dynamics represents the explicit

kinematics, which is marginally stable (poles are zero) in velocity and pitch rate for forces and moment inputs respectively.

This raises the obvious concern regarding the relevance of this linearised model since it does not account for the parameters that influence the flight dynamics. While the linearisation barely accounts for the critical factors of the flight dynamics, it is important to be reminded of the control allocation scheme that continually obtains (OBDM) the aerodynamics and propulsive forces acting on the aircraft based on current operating conditions, thus enabling the controllers to be tuned for the kinematic characteristics. The disturbance controllers are intended to be capable of generating relevant forces and moment commands, while the control allocation would appropriate compute control input demands accounting for the aerodynamics and propulsive influences.

5.5 Conclusion

The possibility of implementing a differential flatness based controller was identified with the PVTOL conceptual study. Deriving from this understanding, an architecture that can leverage the performance benefits of differential flatness concept was devised. This architecture proved the basis of this study and in implementing the controller within the flight dynamics model of the Vision Volante. The differential flatness equations based on the aircraft kinematics were derived, and its function in the architecture was identified as a feedforward loop. The feedforward inputs were designed to be summed with disturbance controllers which would be capable of dealing with plant uncertainties, ensuring stability and robustness. The controllers were chosen to operate based on forces and moment, based on the differential flatness equations.

The key requisites to utilise force and moment based control demands along with significant robustness was identified and control allocation was deemed integral for implementation. Pseudo inverse and non-linear inverse control allocation techniques were identified as optimal and implementable. The implementation of the proposed control allocation techniques for the desired airframe was devised, and OBDM was also identified necessary. The allocation logic was carefully curated and analysed for proper operation across the envelope.

Finally, an alternate model for obtaining trim linear dynamics was developed. The model was batch trimmed and linearised to facilitate controller development, which is explained in the following chapter.

6 DISTURBANCE CONTROLLER DEVELOPMENT

Controllers are systems that operate to achieve a reference command by controlling the inputs of a system. From the differential flatness control architecture devised in section 5.2, a vital component of the system is its disturbance controllers. The key role of the disturbance controllers in the defined architecture is to deal with plant and environmental uncertainties since the differential flatness feedforward loop overlooks these deviations. An aircraft is always subject to uncertainties and disturbances, and it is critical to develop controllers capable of dealing these ambiguities.

Vision Volante being a tilt-rotor VTOL aircraft, it is capable of multirotor flight as well as fixed wing flight. This introduces a control problem that requires to be tackled strategically, due to largely different dynamics in both flight phases. However, with the chosen control allocation scheme, the complexity of defining controllers for varying flight phases has been eliminated, and this is mostly because the developed control allocator operates based on an onboard dynamics model, enabling it to account for the dynamic variations at different flight conditions.

From Figure 5-2, the disturbance controllers are located alongside the differential flatness block, generating similar control signals, i.e., $[F_x \; F_z \; M_y]$. The architecture defines the controllers to compute corrective demands to the feedforward demands, in the form of forces and moments which the control allocator would recompute to actual airframe control inputs. Another critical consideration during control development was the varying dynamics and how the controllers could be refined for the complete operating range of the aircraft. This chapter elucidates on the architecture, development and implementation of the disturbance controllers. Finally, the chapter ends with the nonlinear simulation test runs of the controllers integrated within the proposed architecture.

6.1 Controller Structure

The excellence of the designed architecture alongside the unified control allocation is the simplicity it achieves with regards to disturbance controllers. As previously mentioned, with disturbance controllers generating forces and moment commands, the controllers can be straightforwardly designed for the marginally stable linear dynamics obtained in section 5.4. The basic dynamics makes it possible for simple proportional (P) controllers for forces, and proportional + integral (PI) controller for moments, as shown in the controller structure below.

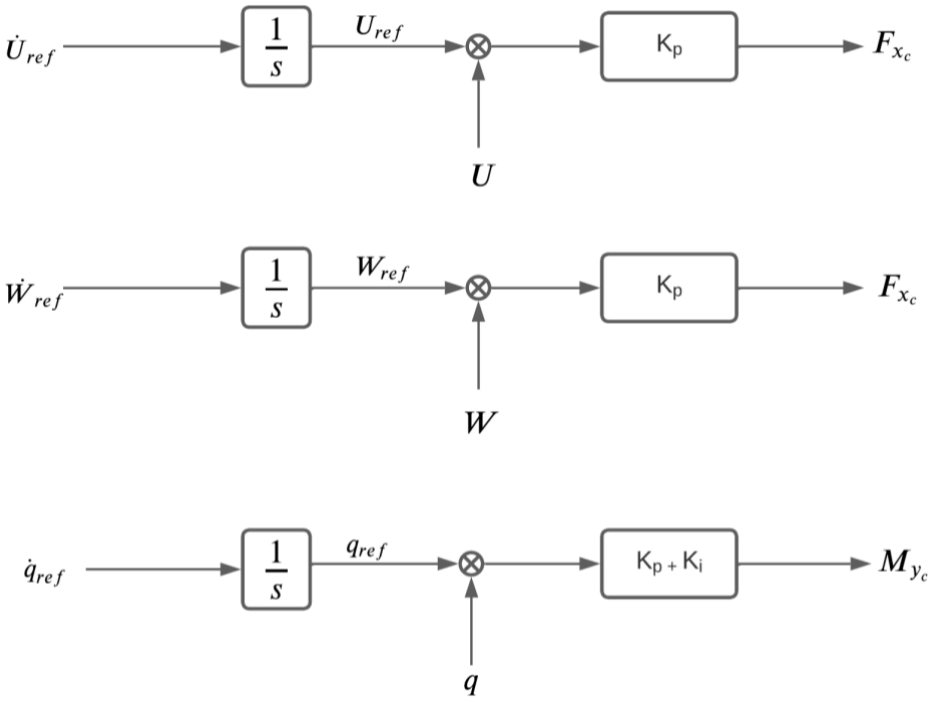


Figure 6-1 Linear Disturbance Controllers – Structure

The above controller structure satisfactorily meets the desired controller performance characteristics, as will be seen in the following sections. All the controller inputs are fed in the form of velocity accelerations ($\dot{U}$ $\dot{W}$) and pitch rate rates ($\dot{q}$), based on the input requirement for the differential flatness unit. The author decided to keep commonality with reference demands in order to make sure that both units in the control architecture operates with the same reference commands. Although the inputs are common, the controllers were designed to

function based on velocity (U W) and pitch rate (q) feedbacks, and hence the integrators were placed prior to error signal calculation.

It is important to note that conventionally, *U* and *W* control demands are not implemented due to its dependence with the orientation of the aircraft. This limitation particularly arises in fixed flight, where the aircraft would have to maintain a positive pitch attitude (angle of attack) in order to sustain lift, resulting in a non-negative longitudinal velocity (W) without actually changing altitude. Hence, commanding the *U* and *W* controller during cruise phase is not straightforward, and this aspect has not been worked to a solution by the author in this research since the Flight Management System (FMS) is assumed responsible for generating the appropriate control demands based on the desired trajectory. However, in hover phase, the *u* and *w* commands are deemed appropriate since flight is largely dependent on the propeller thrust and not aerodynamics angles.

The moment controller requires a PI controller due to constant counter-moment generated during flight due to aerodynamics or propeller dynamics. Hence an integral action that constantly ensures that the desired pitch rate is maintained was necessary, and hence the integral action was incorporated. This requisite does not imply to the velocity controllers since steady-state response isn't critical for velocities and can still achieve an almost first-order response.

Another significant advantage of the control structure along with the devised control architecture is that, both the Hover and Cruise phase controllers can function within the same structure. While the tuning of the controllers does vary with flight phase, the primacy that the control structure operates universally across the flight envelope indicates the prowess and potential of the control architecture and control allocation proposed in this study, and with further work, it could prove of significant relevance in practical control implementations.

The following section delves deeper into the implementation and testing of controllers in the linear and non-linear simulation scenarios, and about other additions that had to be incorporated in order for the system to operate satisfactorily.

6.2 Controller Development

Building upon the understanding of the linear dynamics around which controllers had to be designed, a controller structure was devised which would satisfactorily achieve the desired performance characteristics. With the structure in place, a definition of the linear controller performance requirements was required to tune controller gain for different flight phases.

6.2.1 Requirements

Considering this study to be an initial proof of concept of the suggested differential flatness-based control architecture, the control performance requirements were chosen to be in alignment with achieving baseline stability. This means that the chosen performance characteristics for tuning the controllers did not account for pilot handling quality, actuation system lag/limits, or extensive manoeuvrability.

Since the open loop linear dynamics of the system is marginally stable, the foremost goal of the controllers was to ensure closed loop stability, i.e., shifting its dynamic poles from zero to negative real numbers. Since all the system poles lie on the real line, the closed loop damping was also expected to be 1 (poles on real line). The desired closed loop step response of the controllers with linear dynamics were chosen as shown below,

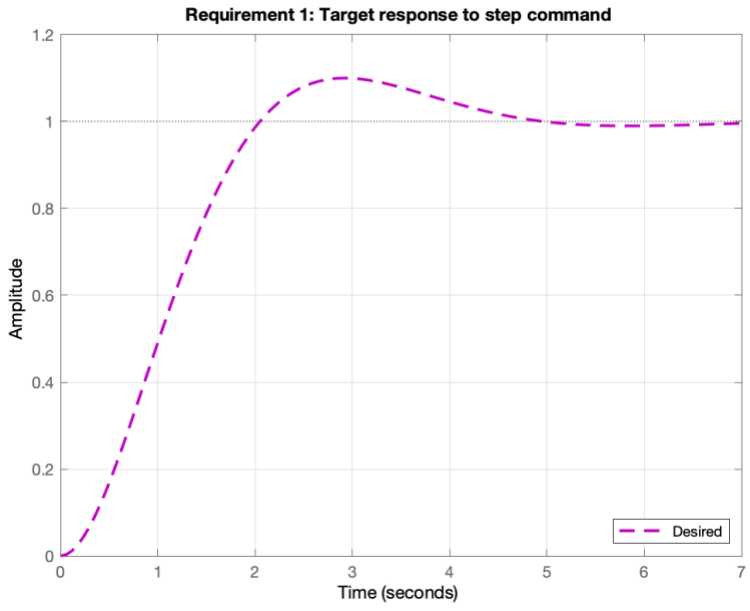


Figure 6-2 Tuning Requirement - Pitch Rate Step Response

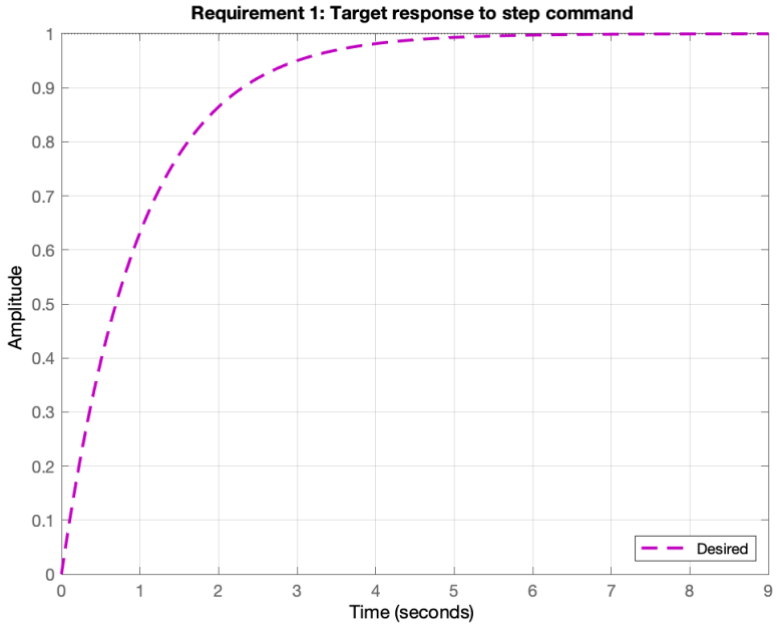


Figure 6-3 Tuning Requirement - Velocities (*u,w*) Step Response

The first order response of the velocity controller is achieved by the proportional controller due to the integral dynamics of the velocities for force input. However, with the proportional-integral controller chosen for the pitch rate controller, a second order response was chosen with 10% overshoot and 0.75 sec time period. The reduced time period for the pitch rate controller was an intentional choice to ensure the controllers maintained longitudinal stability with good response times.

6.2.2 Hover Controller

Based on the implementation in this study, the hover controller is the controller that operates when the tilt-rotor angle is greater that $85°$. At $\Gamma \geq 85°$, there is null aerodynamic surface induced forces or moments, and the control is completely dependent on the thrust generated by the forward and aft propellers. It is important to remember that tilt-rotor angle was scheduled as a function of speed, and if $\Gamma \geq 85°$, the aircraft velocity would be very small.

While it is critical to account for hover phase manoeuvrability and controllability, the author in this study assumed the purpose of the hover controllers to vary aircraft altitude and maintain stability before transitioning to cruise phase. This assumption was made to facilitate controller development within the project

constraints. Since the hover phase control allocation was based on nonlinear inverse instead of pseudo inverse, the controller had to be tuned separately in order to meet the performance characteristics with the control allocation in the loop.

Looking back to the nonlinear inverse control allocation technique (see 5.3.2), equations 5.18 and 5.19 were derived based on the hover controller condition, i.e., the tilt-rotor angle is $\sim 90°$, which implies that the forward force input does not deliver control. This is only relevant due to the fact that the controllers were not designed to achieve manoeuvrability in hover, and hence the tilting rotor variations were not accounted in the nonlinear inverse control allocation scheme. However, this rationale stands true considering the fact that the aircraft can achieve forward velocity by pitching forward in hover, after which the tilt-rotor would slowly align to cruise configuration with the increasing speed in transition.

With sufficient understanding of the functionality and performance characteristics of the hover controllers, the controller tuning was initially performed with the linear dynamics of the aircraft, after which it was tested for performance with control allocation in the loop.

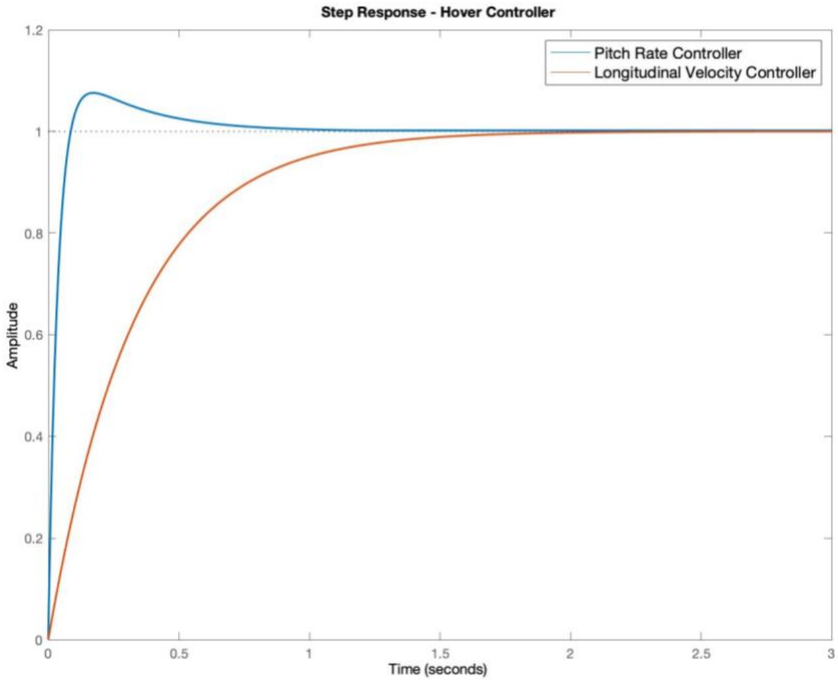


Figure 6-4 Step Response - Hover Controller (Linear Dynamics)

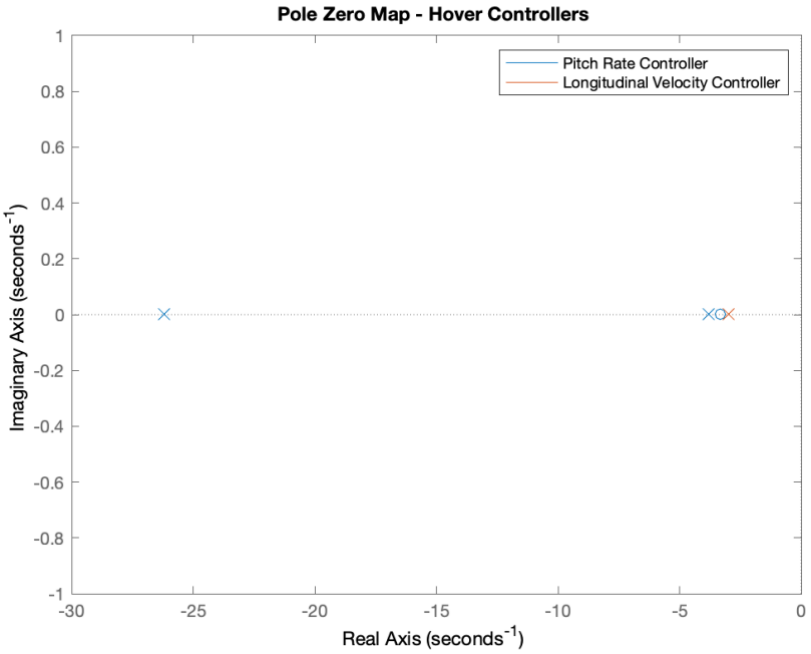


Figure 6-5 Pole Zero Map - Hover Controller (Linear Dynamics)

The linear dynamics based on the trimming model was obtained which identified integrators for the velocity and pitch rate dynamics, and controllers were tuned in order to achieve a satisfactory step response as seen in Figure 6-4. The poles of the closed loop system can be observed to be on the negative region of the real line, proving stability. The farthest pole is due to higher gain in the pitch rate controller, which was necessary to ensure the controller performed satisfactorily when validated with the control allocation.

6.2.3 Cruise Controller

Upon achieving minimum altitude in hover, the control scheme is devised to switch to cruise controllers as it transitions from hover to cruise phase. In this control implementation, it is considered cruise phase when the tilt-rotor angle is less that $85°$, which is velocity dependant (refer section 5.3.3).

The transition phase between hover to cruise and vice versa is considered complex, particularly due to its highly nonlinear dynamic characteristics. During flight, it is common preference to keep the duration of transition at minimum in order for the aircraft to quickly achieve fixed aerodynamic properties or multi-copter dynamics, whose controls are very well understood, developed and can

be better controlled. This means that transition controllers require meticulous development work in order to ensure performance characteristics. However, it is key to note that the controller architecture defined in this study fares to operate in the transitionary flight phase too. This is quite a remarkable aspect of the proposed architecture, since the kinematics based controllers are capable of handling the transition dynamics, and is largely due to the control allocator understanding and refining control input demands based on the flight condition and its changing dynamics.

Since the cruise controllers were designed to operate across a varying range of velocities, which implies varying flight dynamics, the kinematics based disturbance controllers had to be gain-scheduled across the velocity range of operation. A tuning model of the aircraft with controllers, in lieu with the defined structure (Figure 6-1) was implemented along with tuneable gain surfaces for automated tuning. The author then generated a MATLAB based script that utilises the Control System Toolbox to automatically tune the controllers based on the set requirements (see section 6.2.1). Following batch trimming and linearisation, the automated tuning script then computes appropriate control gains for every individual linearised dynamics. The tuneable gain surfaces are user defined parametrisation of gains as a function of scheduling variables (linear velocity), and the tuner automatically populates all the calculated gains as a function of the scheduling variable. The gain surface ensures a smooth variation of gains across the velocity range, eliminating control demand abnormalities due to gain switching. A plot of the varying gains as a function of velocity is illustrated below in Figure 6-6.

Observing the gains, it can be identified that there are not significant variations across the velocity range. This is largely due to fact that the controller demands are normalised for aircraft mass and inertia in implementation, due to which even minor variations in gain can have a significant performance impact. Hence, the minimal variation was regarded expected and acceptable. However, it is important to notice the pitch rate proportional gain, which vary hugely relative to other gains, and this primarily is because unlike the velocities, the pitch dynamic

is sensitive to velocity resulting in significantly differing gains to achieve control. The high integral pitch rate gain was realised important for pitch stability due to constant counter-moment force demand required to maintain the desired pitch rate.

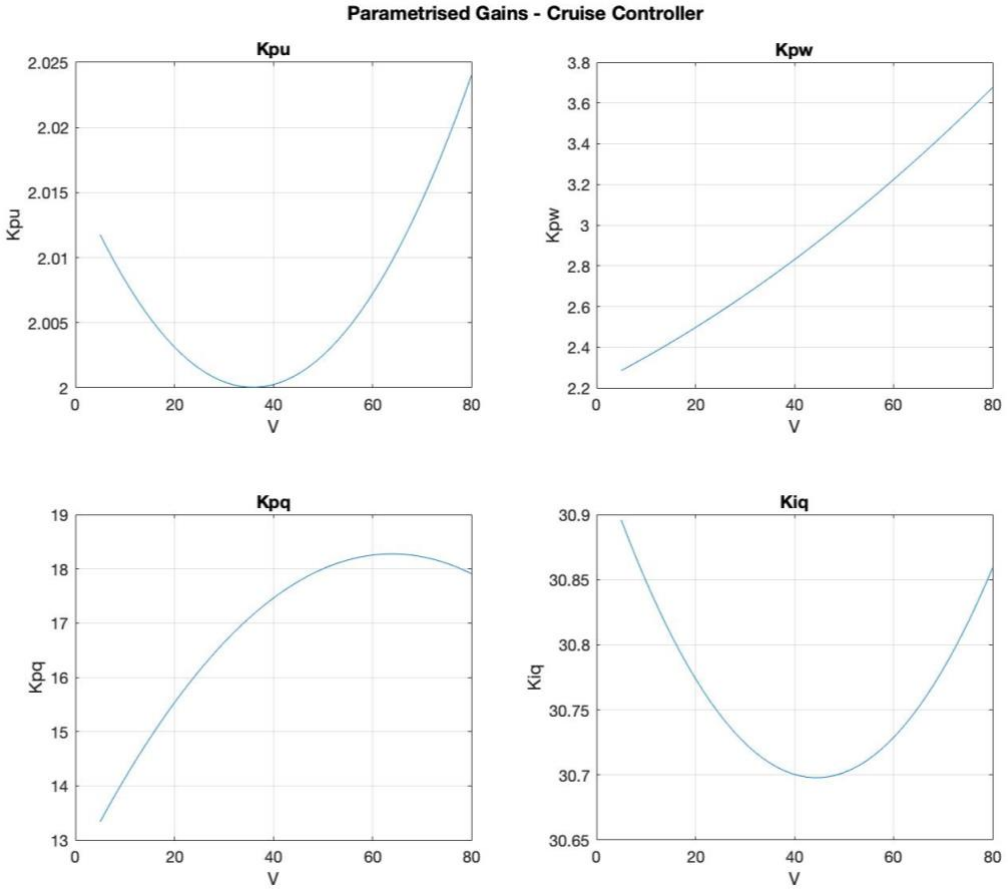


Figure 6-6 Parametrised Cruise Controller Gains

The controllers were tuned for 50 points between the range of 5 and 80 m/s, and the controller gains for the linearised dynamics at each point was obtained by Control System Tuner tool, which was then parametrised on a tuneable gain surface. Once the controllers were tuned, the closed loop step response (see Figure 6-7) of the controllers were plotted and observed to confirm that it meets the desired linear controller performance.

The single order performance of the velocities can be observed in the step response which aligns with the expected performance characteristic of the tuned dynamic. The rise times of both velocity plots lie within an acceptable range, i.e.,

less than 1 sec. The second order performance characteristics of the pitch rate controllers can also be observed to be in acceptance to the previously defined pitch rate performance requirement. The response achieves zero steady state error without overshoot beyond 10% and with a risetime lesser than 0.2 seconds. For reference, the red line in the step response plots indicates the tuned controller performance at $U = 45\ m/s$.

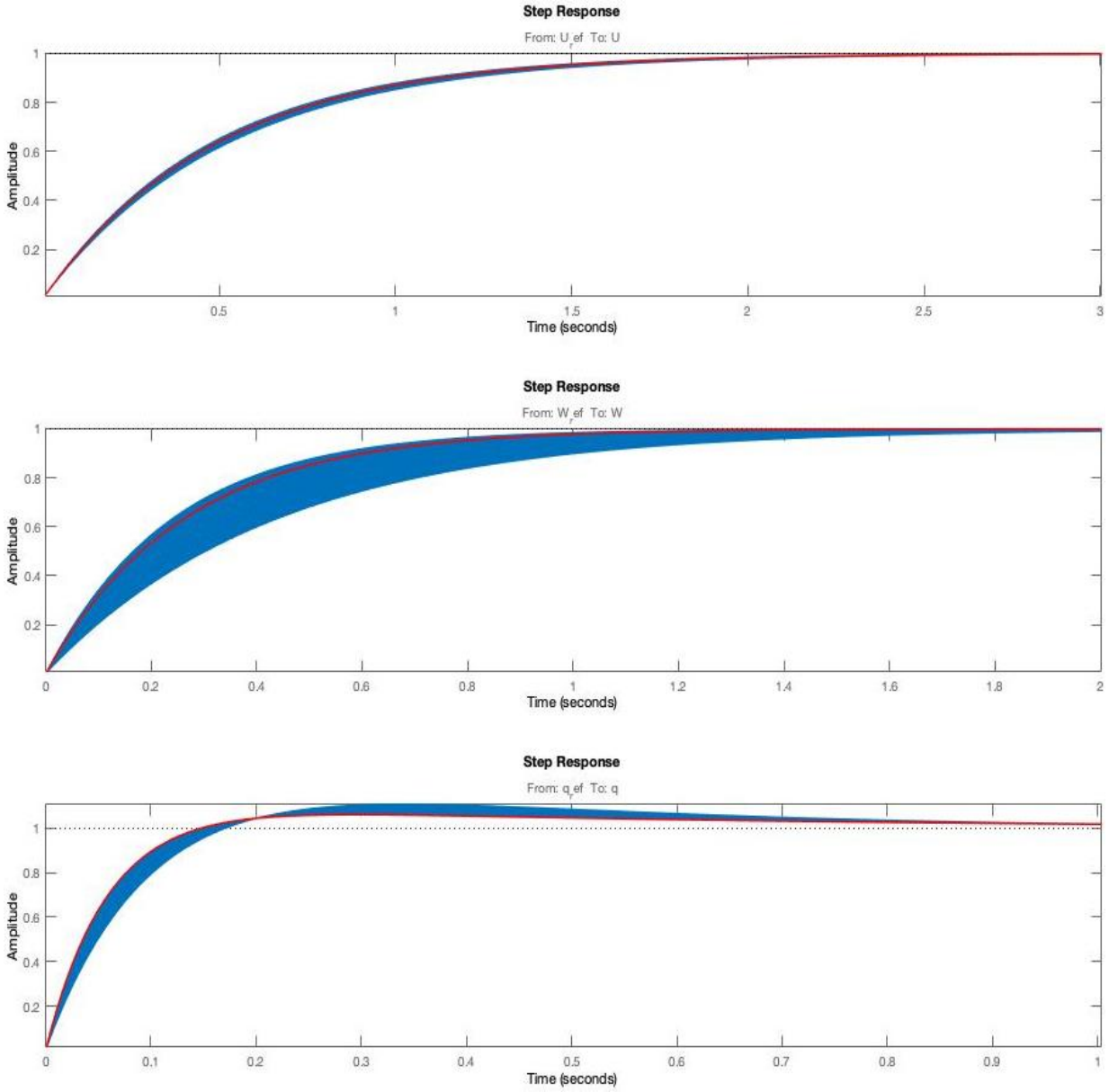


Figure 6-7 Step Response - Cruise Controllers[3]

[3] Step response of 50 tuned cruise controllers

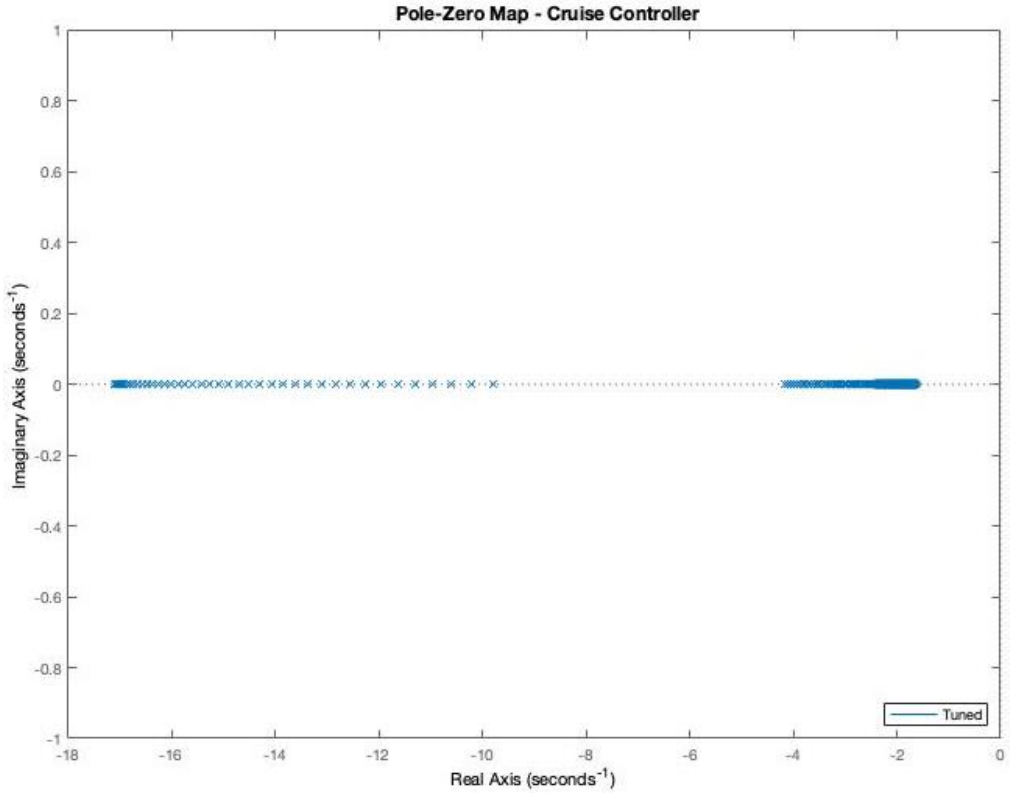


Figure 6-8 Pole Zero Map - Cruise Controller

Analysing Figure 6-8 and Figure 6-9, the closed loop dynamic poles and bode plots indicate the system to be stable and having significant gain and phase margins. The infinite gain margins provide room for introducing the denormalization gains prior to the control allocator input.

To sum up, the cruise controllers were successfully tuned for the varying velocity dynamics to meet the desired linear performance requirements with sufficient robustness. The parametrised gains provisioned for seamless gain scheduling of controllers across the velocity range, ensuring stability as the aircraft transitioned. With the linear controllers tuned, the final important step to evaluate the proposed controller architecture was to implement the architecture within complete flight dynamics model along with the control allocator.

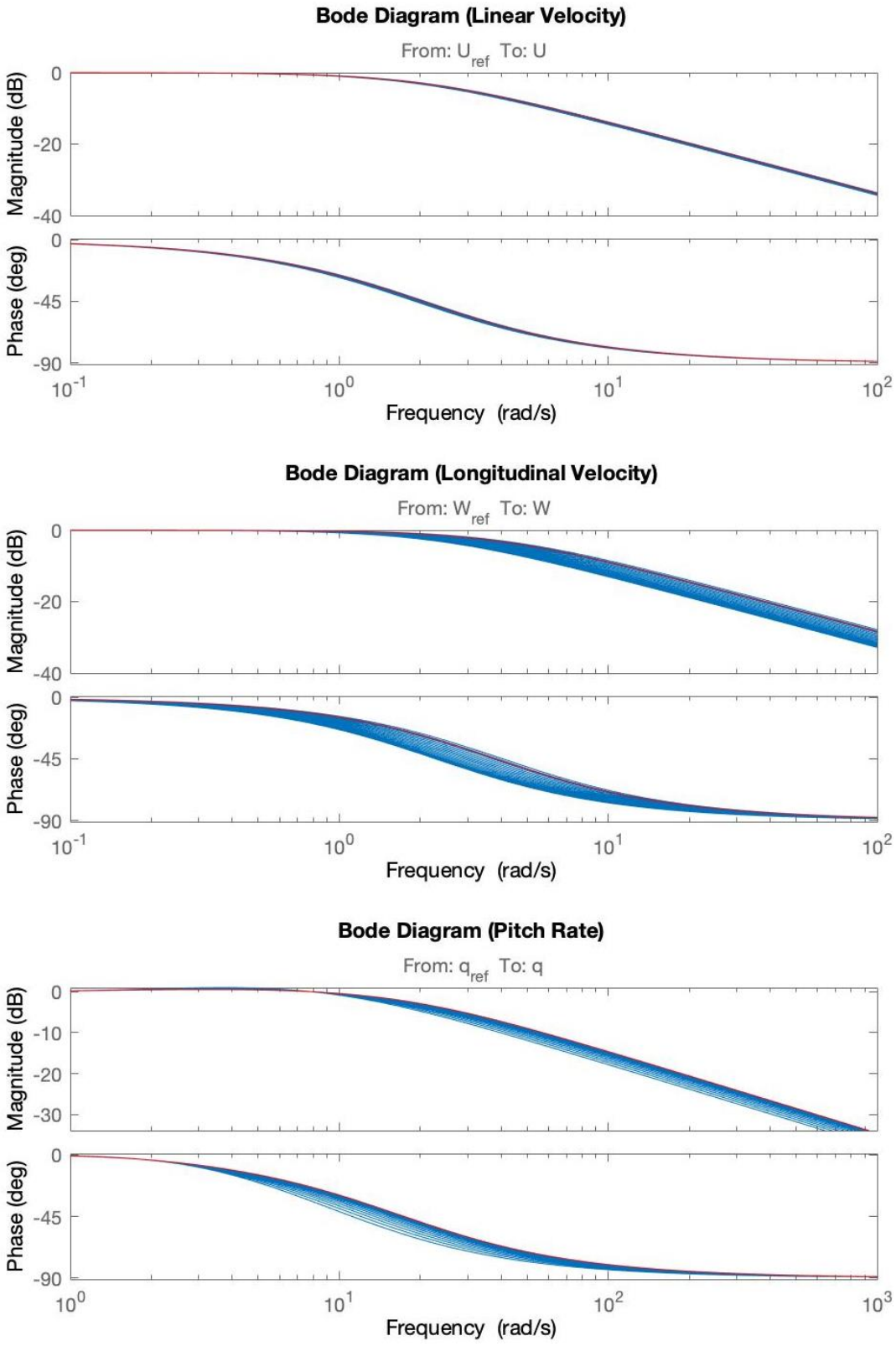


Figure 6-9 Bode Plot - Cruise Controllers

6.3 Controller Evaluation

The final element within the scope of this study was to integrate all the developed control units within the proposed control architecture along with the flight dynamics model, to assess the controller performance in nonlinear simulation. Nonlinear simulation mimics the actual flight dynamics which is largely nonlinear and dependant of several factors, giving a realistic insight into the implementation of the developed controller architecture. Despite nonlinear simulation being a close representation of practical deployment, it is completely dependent on the developed flight dynamics model and it can only represent the real flight dynamics within the bounds of the modelling fidelity.

6.3.1 Module Integration

Recalling the control architecture (Figure 5-20), the overall proposed control scheme comprises of several different units which has to come together in harmony to achieve stability and control over the aircraft. Integrating all the individually developed units in order to function within the nonlinear simulation is an important activity, which requires good understanding of the system limits and the limitations of the different controller modules.

6.3.1.1 Control Allocator

The control allocation module is positioned intermediate of the controllers and flight dynamics model, with the primary role of commanding the aircraft control inputs. The control allocation unit comprises of two allocation techniques, non-linear inversion and pseudo-inverse for the hover and cruise phases respectively. The two types of allocations were used as B matrix becomes uninvertible due to perpendicular tilt-rotor angle setting.

In implementation, the control allocation module has to be programmed in such a manner that when $\Gamma \leq 85°$, it switched the allocation logic from non-linear inverse to pseudo-inverse. This would enable a single control allocation unit to be operable in both cruise and hover flight phases. However, the author was unable to implement a solution to seamlessly switch between the allocation

methods, limiting the implementation to switching based on an if-else statement conditioned on a speed threshold.

Both of the control allocation technique has the limitation of not being able to compute controller demands within the actuator saturation limits. This meant that without defining the actuation saturations manually, the control allocation could compute control demands that might be optimal, but outside the limits of the system. Hence, all the control signals computed by the allocator module had to be dynamically saturated as a function of velocity. The dynamic saturation was key to ensure that the allocator would not erroneously push out high aft rotor speeds at cruise or large aileron deflections at low speeds. The rotor speed and aileron deflection saturation limits are shown below and these limits have been calculated from several trim points summed with a 30% margin to account for manoeuvres or controller action.

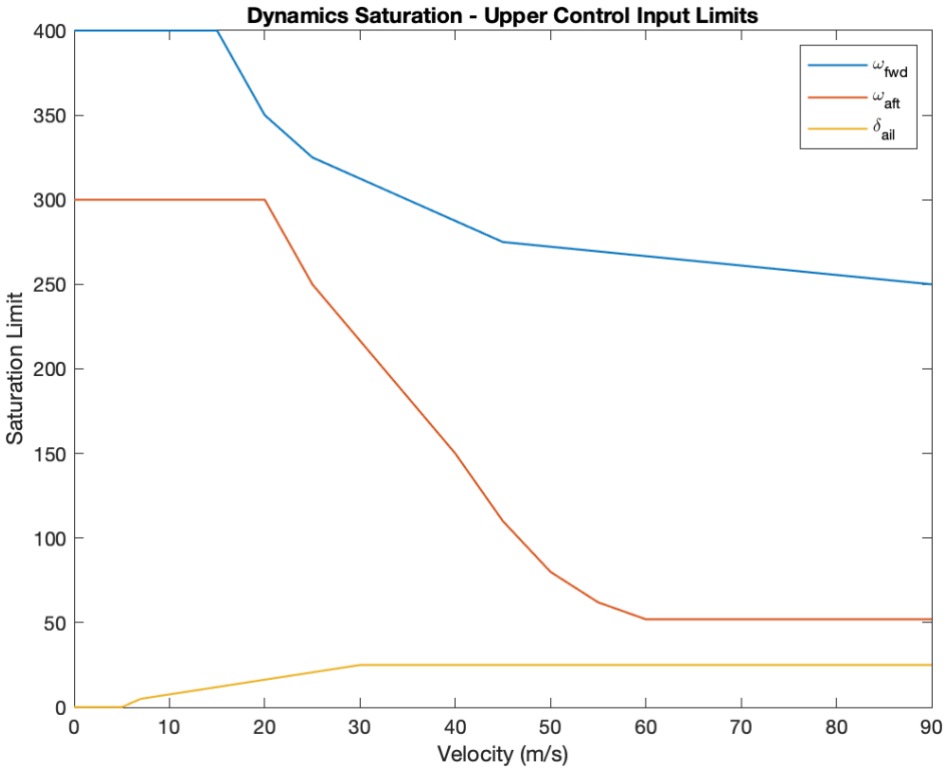


Figure 6-10 Control Input Dynamic Saturation Limits

Another implementation concern with control allocation was its requirement for Onboard dynamics model data, which in model-based simulation environment is a feedback of the aircraft simulation states. The OBDM data had to be fed through a memory block (continuous) which delay the input by a time step, and this ensured that the allocator computed the current demand from the previous aircraft states, and also eliminated algebraic loops. The memory block raised the requirement of developing an initialisation model to compute the first set of OBDM signal values for simulation start, which induces some simulation anomalies at the first simulation time step. With all discussed resolves in place, the control allocator was set up for proper operation as shown in Figure 6-11.

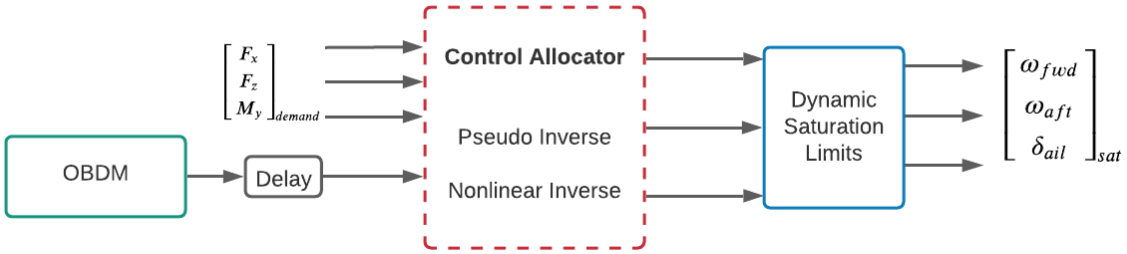


Figure 6-11 Control Allocator Implementation

6.3.1.2 Controllers

The disturbance controllers are required to operate within the entire operational velocity range, and for that reason the controllers were tuned for several trim points, and a parametric representation of gain as a function of velocity was obtained. The gains were then computed from the tuneable gain surfaces, and were built into lookup tables that interpolate across the gain values based on the aircraft velocity. The smooth variation in gains across the velocity range would ensure smooth switching of controller tunes in nonlinear simulation without the need of complex techniques like bumpless transfer. Utilising the inbuilt PID controller blocks that utilise external gain inputs, the gain scheduled cruise controllers was implemented.

Since the differential flatness feedforward loop computes control input demands normalised by mass and inertia, the hover and cruise controllers were also designed to operate with the same units. Hence, the summed feedforward and controller demands are then denormalized with gains that compute the net forces and moment demand which is then input into the control allocator. However, prior to the control allocation input, saturation limits were placed between the signal to limit the demands within the desired forces and moment operational limits. These limits were installed to ensure that the controllers would not demand the allocator outside the attainable system forces and moments. The controller implementation in SIMULINK is given in 7.2Appendix C.

6.3.2 Non-linear Simulation

With the constituent modules of the controller architecture integrated, the overall system functioning and performance were analysed. Through non-linear simulation runs of the modelled flight control architecture, the feasibility and operability of the controllers were assessed.

As discussed earlier, due to project constraints, the author was unable to implement a solution to seamlessly switch between the control allocation techniques to handle all the flight phases. Hence, a test flight simulating hover to cruise transition was not achievable, however deterministic test runs of different controllers working at different flight phases were obtained, with the cruise controller simulated to indicate stability through transition.

A common issue observed amongst all the test simulation runs were initialisation anomalies. Due to requirement for an OBDM to compute control input signals, there always tends to be an abnormal spike at the beginning of the simulation setting the aircraft states at values outside of the user defined initial simulation condition. In practical implementation, this would not be an issue of concern since such a simulation environment setup isn’t relevant and the control allocation software would be implemented in a way that it initialises all the required input appropriately before flight. This initialisation anomaly can be observed in the following test plots where the aircraft velocities might not be at the initialised value.

6.3.2.1 Hover Phase

In hover phase, the controller performance had to be tested in terms of its controllability in pitch and vertical speed since these are the pilot controlled longitudinal states. Forward speed wasn't considered a pilot input (but an autopilot input) since it is a function of the pitch attitude of the aircraft when in hover. Input demands were chosen such that it demonstrates the controller performance in a scenario whereby a pitch-rate rate inceptor inputs and vertical speed acceleration (thrust) inputs are controlled by the pilot, whereby the pilot demands inputs to zero the pitch attitude of the aircraft before increasing hover altitude. The hover phase controllers operating with the nonlinear inverse control allocation was tested with the shown input demand. This simulation run was initialised at a level setting; however, the control allocator initialisation issue puts the aircraft outside the set initial condition. The test simulation outputs with and without the feedforward differential flatness unit can be seen in Figure 6-13 and Figure 6-14.

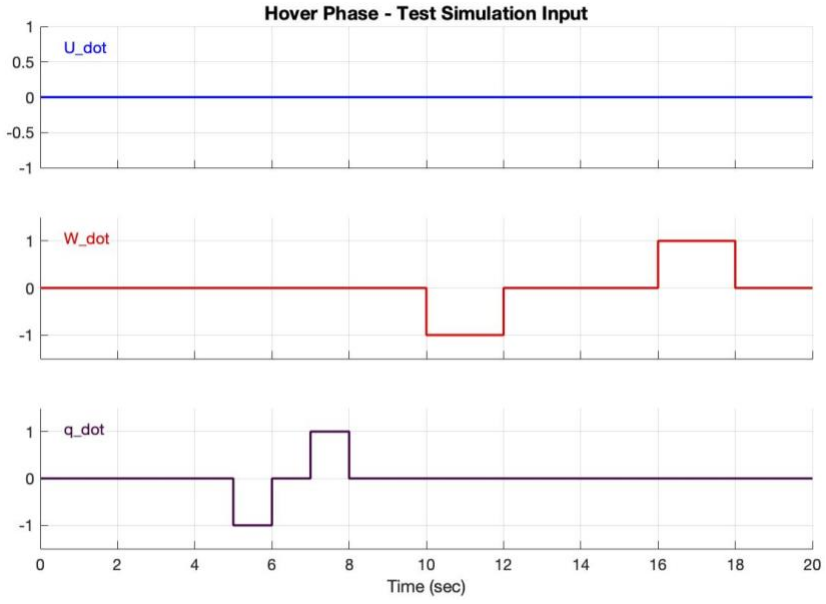


Figure 6-12 Hover Test Control Demand

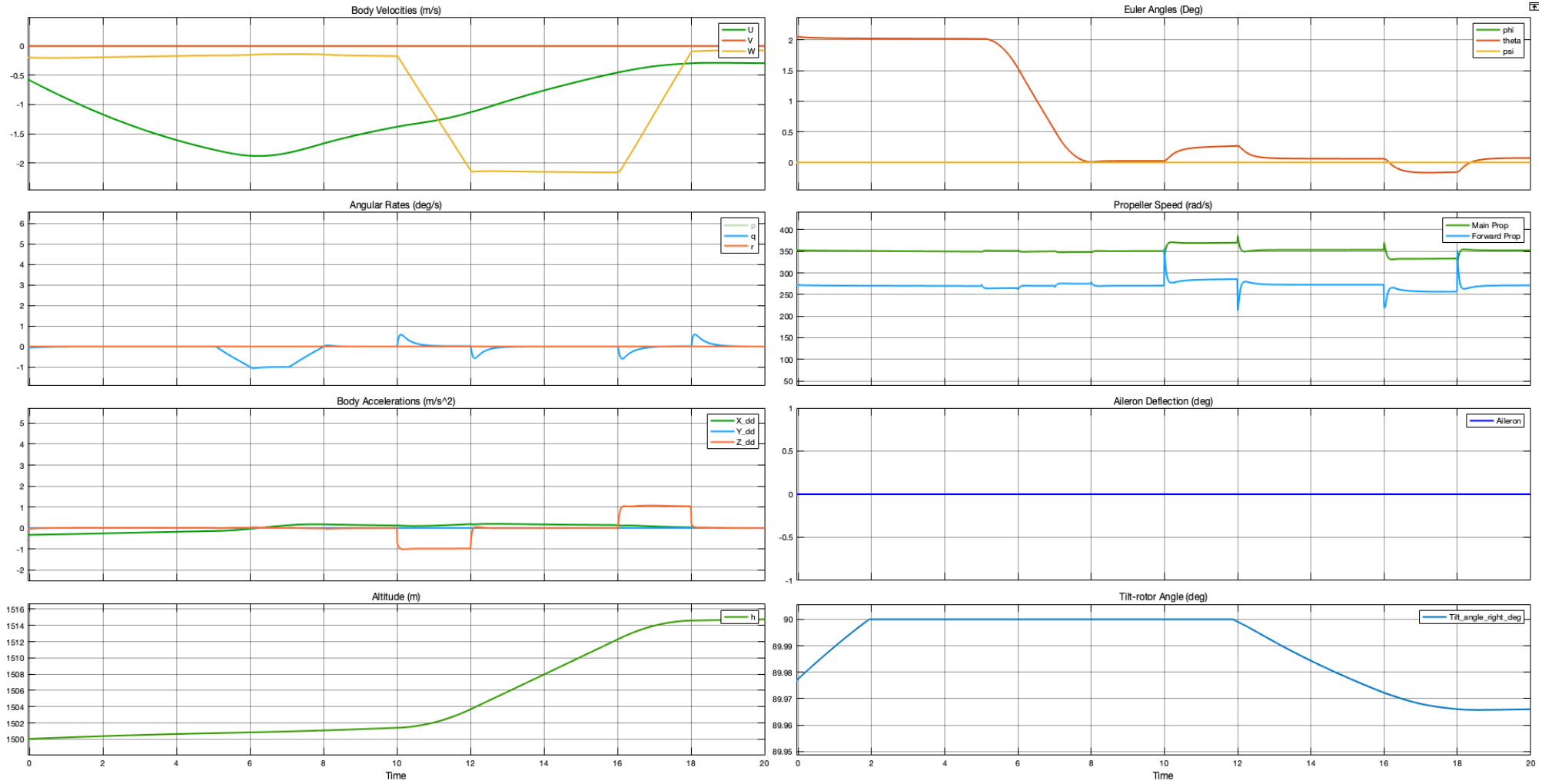


Figure 6-13 Hover Flight Test

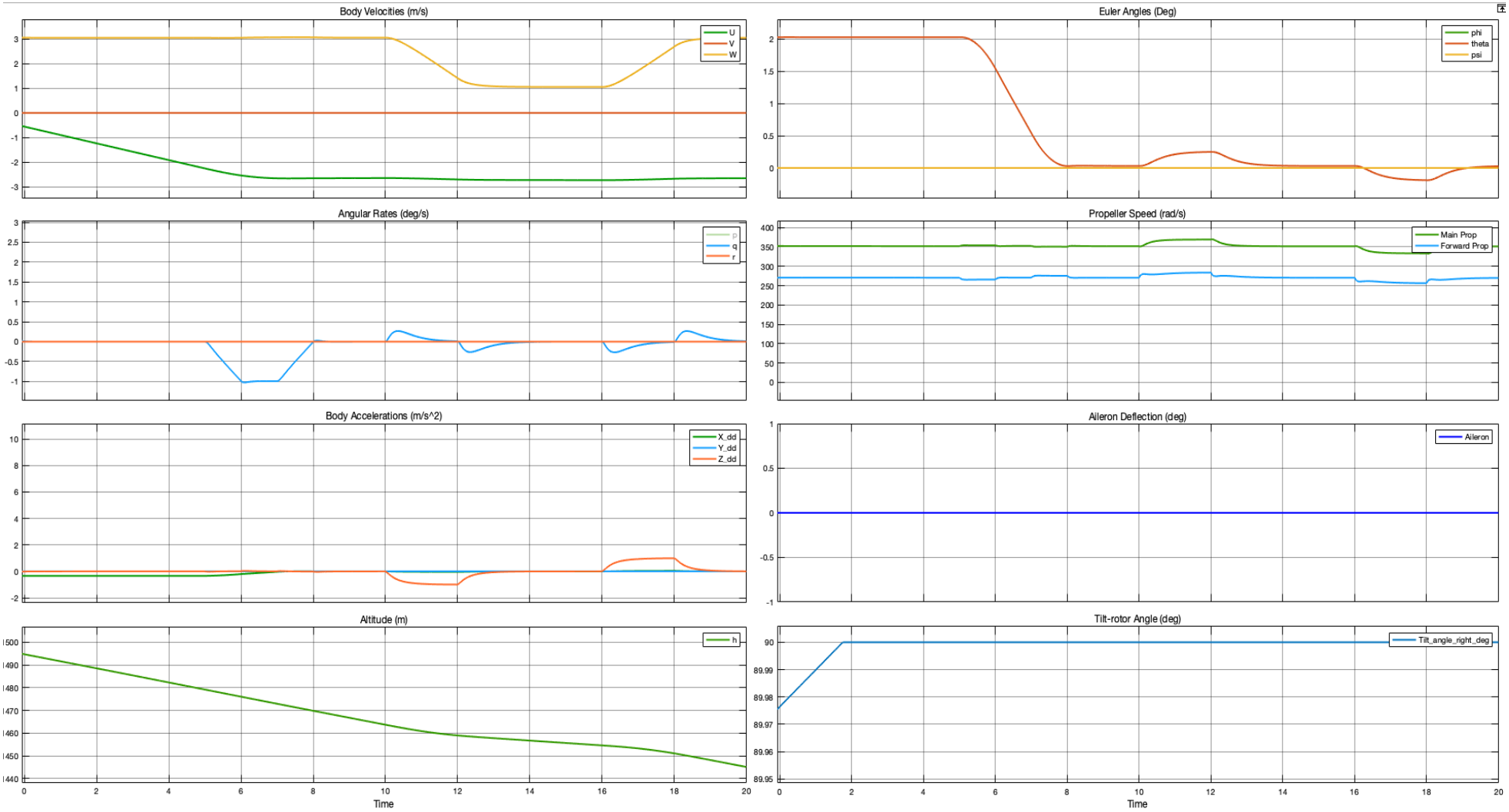


Figure 6-14 Hover Flight Test without Differential Flatness

From the above simulation results, the performance advantage of the proposed controllers are obvious. Firstly, despite the same initialisation conditions, the controllers attain stability much differently than each other, and this is because the differential flatness architecture is able to quickly resolve the initialisation disturbances and settle closer to the set initial condition. Additionally, observing the longitudinal velocity change, the proposed architecture performs significantly better with better pitch stability and precise tracking of demands, whereas a sluggish behaviour is seen without the feedforward loop. This can be explained due to fact that the summed controller demands are much more accurate for following the trajectory with the differential flatness unit feeding in the required forces and moment demand, which is not the same case without the feedforward loop where the disturbance controllers rely completely on the error term to compute the control demand. The pitch rate deviation when changing altitude is due to the control allocation demanding sudden large rotor speed inputs, however in a practical system, propulsion systems would not meet such dynamics and hence wouldn't generate such an anomaly.

The author conducted multiple tests in hover, and the hover controllers proved to be capable of functioning with the desired characteristics, rendering good pitch stability and velocity control. Most importantly, the differential flatness architecture improved performance as speculated.

6.3.2.2 Cruise Phase

The primary control of the longitudinal dynamics in the cruise phase are pitch attitude and forward velocity. The pitch attitude control enables control over the flight path angle, and the velocity control maintains speed during different manoeuvres in cruise.

The cruise phase test inputs were determined to mimic pilot pitch rate acceleration inputs and velocity acceleration inputs, which can be seen in Figure 6-15. The manoeuvre emulates the pilot setting a new flight path angle after which a new cruise speed is set. Primarily, the input test the pitch attitude stability and controllability and speed control in cruise. Additionally, the test run was initialised

at 60 m/s to observe the controller performance achieving complete switch to cruise mode ($\Gamma = 0°$) from a transitionary state. The pitch controllers with zero rates inputs also test the trim stabilisation achieved automatically using the aerodynamic control surfaces. Whilst vertical velocity controllers are vital to maintain stability of the system, in cruise, they do not have much controllability due to flight dynamic. Hence, vertical controller was not given any control input. The test results are given in the figure below.

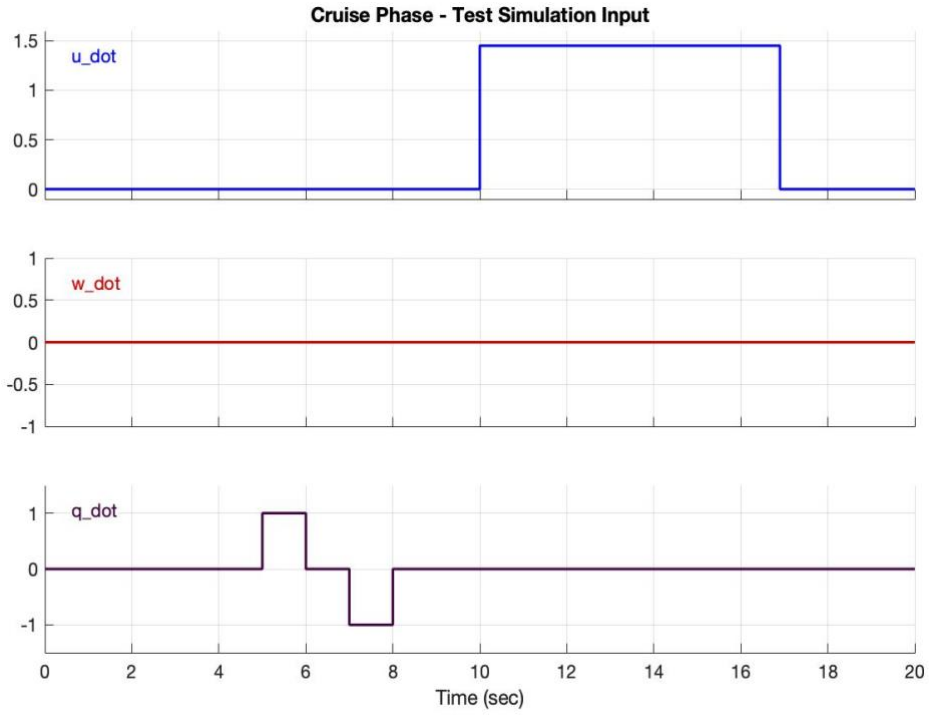


Figure 6-15 Cruise Test Input Demands

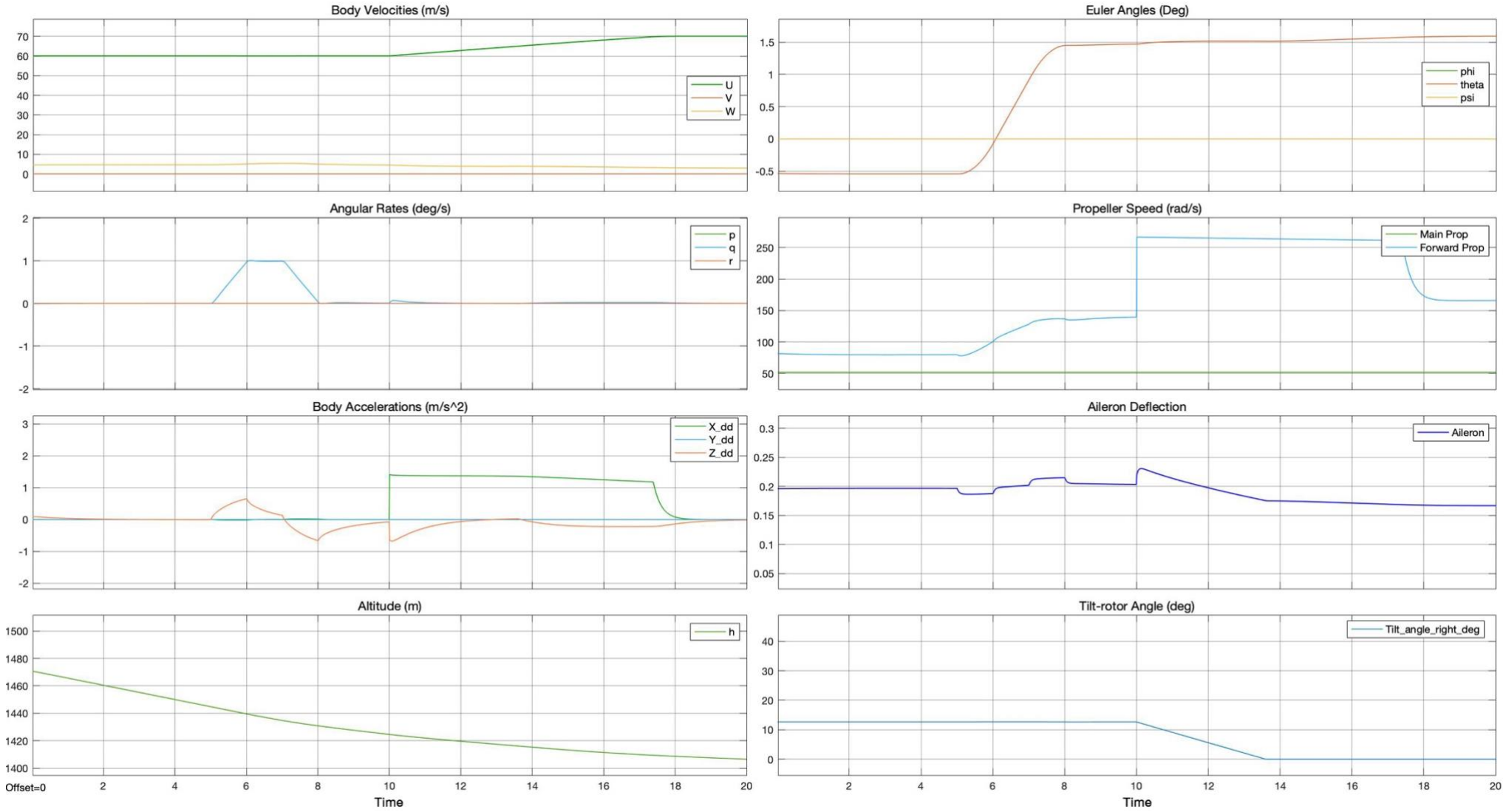


Figure 6-16 Cruise Flight Test

From the above test result, it can be concluded that the architecture performs satisfactorily in cruise phase. The cruise controllers were also tested for various velocities, and the controllers operate within the expected performance characteristics. The cruise controllers also perform reasonably well at different cruise speeds (with different tilt-rotor angles) and can be seen to smoothly increase the velocity to attain maximum cruise speed. The slight decline in the acceleration (not identical to input) is because of the saturation limit on forward thrust, limiting the aircraft to maintain the forward acceleration. The aileron deflections are observed to be very small, however, this confirms that the control allocation scheme is accurately allocating to all control input to ensure operation.

When comparing the controller performance without differential flatness, the controllers functioned almost identical. However, closely observing the angular rates and accelerations, a slight sluggishness can be observed without the feedforward loop. The reason for the less significant improvement in performance is due to larger influence of aerodynamics in the cruise flight phase, which the differential flatness unit cannot predict or account for. Hence, the disturbance controllers end up computing the majority of the control demand. Nevertheless, the cruise controllers were observed to perform well within the proposed architecture.

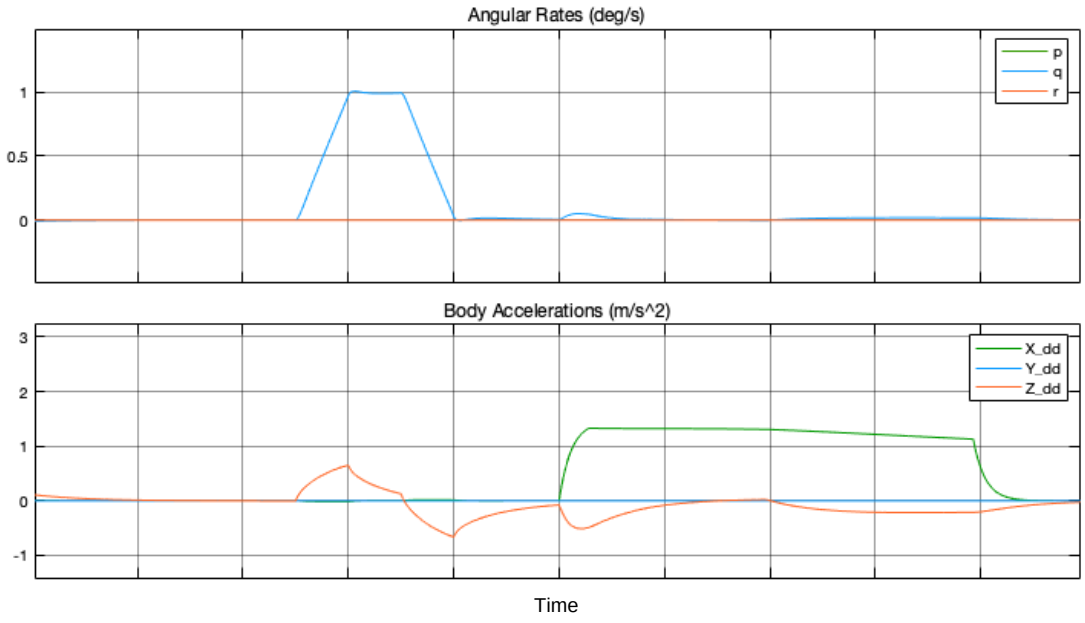


Figure 6-17 Cruise Flight Test without Differential Flatness

6.3.2.3 Transition Phase

The transition phase is the most critical and complex flight dynamics that requires to be controlled when it concerns a tilt-rotor VTOL aircraft. Achieving complete transition with proper stability characteristics involves a detailed understanding and characterisation of the airframe flight dynamics. However, the proposed architecture had a significant advantage which simplified the transition phase control development and was tested to function satisfactorily in transition.

As earlier stated, the project constraints limited the author to devise a proper implementation to ensure seamless switching between the different control allocation schemes, and hence, a single test run simulating complete transition was not possible. But a test run of the cruise controllers transitioning the aircraft to cruise was thought as a rational validation of the capability of the controller architecture.

Hence, a test simulation of transition to cruise when initialised at 30 m/s was performed. In a practical scenario, transition would be performed by maintaining the aircraft level and gradually increasing speed until it fully transitions to fixed-wing flight. A similar test was emulated whereby the pitch rate and vertical velocity inputs demands were zeroed for level flight, and a forward velocity acceleration of 1 m/s^2 was input. However, it is important to expect the controllers to the pitch the aircraft forward in hover to achieve forward velocity before tilt-rotor angles generating forward thrust. Transition tests with and without the differential flatness unit were carried out and are shown in Figure 6-18 and Figure 6-19.

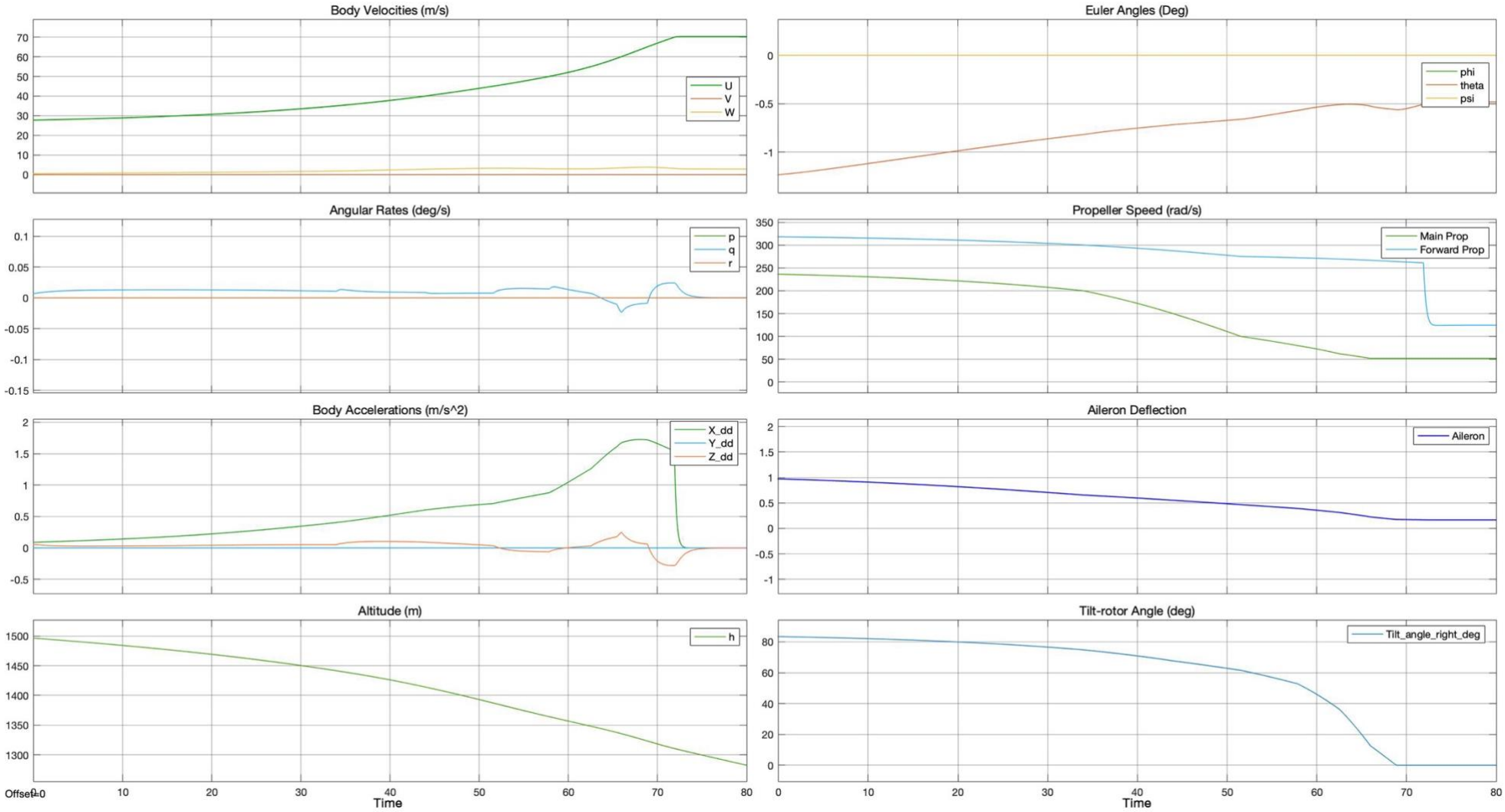


Figure 6-18 Transition Flight Test

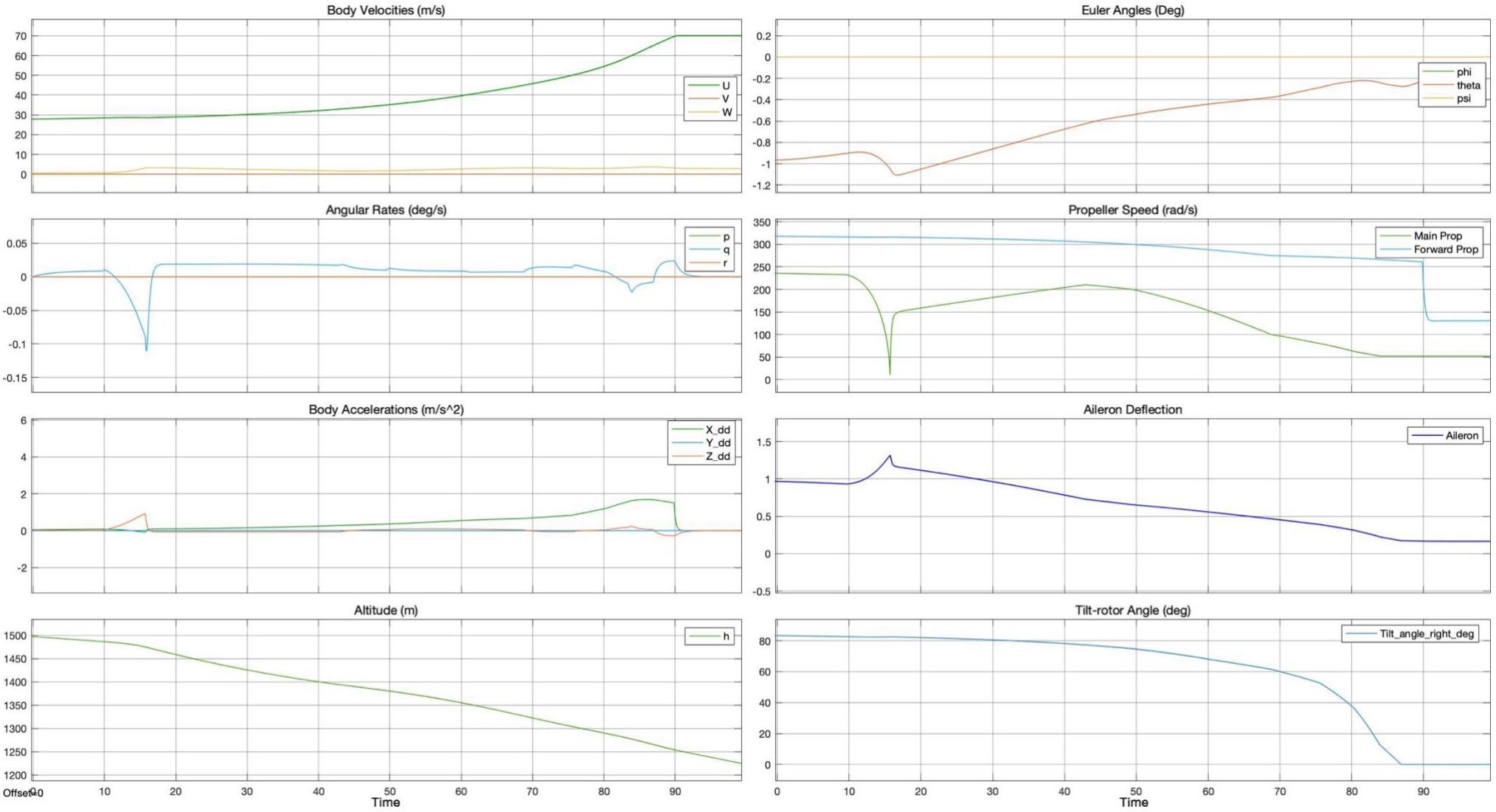


Figure 6-19 Transition Flight Test without Differential Flatness

From the test simulation plots, the controllers performed satisfactorily by maintain pitch stability while accelerating to cruise speed with the tilt-rotor angle being actuated simultaneously. The velocity increase can be observed to be not linear, and this is because of the initial tilt-rotor angle being closer to vertical orientation, and the acceleration picks up as the tilt-rotor angle approaches the cruise configuration. As is the case with all other phases, without the differential flatness feedforward loop, the controllers continue to have lower performance characteristics and can be observed to achieve the velocity much slower than with the proposed architecture. Also, while the deviations are small, the pitch stability continues to be better with the feedforward loop.

It is important to acknowledge that there are significant amount of testing routines required to be executed before confirming a controller capability to satisfactorily perform in transition, however, since the focus of this study was on the viability of the proposed architecture, extensive testing was not carried out. Nevertheless, it can be confirmed that the architecture is capable of transitioning and with more extensive development and testing, an implementable controller can be obtained.

6.4 Conclusion

The proposed control architecture incorporating differential flatness was successfully implemented and was demonstrated to be feasible for controlling tilt-rotor aircraft. The controller development involved defining a common disturbance controller structure relevant for all flight phases based on the aircraft kinematics. Conventional P and PI disturbance controllers ascertains the architecture to be easily certifiable along with being simple and practical. The controllers were then tuned for different flight conditions, and gain scheduled for varying velocities. The differential flatness module, control allocator and disturbance controllers were then integrated along with the flight dynamics model in SIMULINK to simulate and test the controller.

Various baseline tests were carried out to investigate the controller performance in hover, cruise, and transition. The simulation test inputs were determined based on assumed pilot input demands in the various flight phases, and this was done to better understand controller performance in practice. Complete transition from hover to

cruise was not possible, yet transition from low-speed flight to cruise test was also performed to demonstrate the unified architecture.

The controllers were observed to function satisfactorily, meeting baseline requirements to ascertain the proposed architecture valid along with the anticipated performance improvements. The implementation does have assumptions and several aspects to be worked and assessed (see section 7.1), however, the test simulations put forth a start to understanding the novel architecture for further research and development. Apart from performance improvements, the architecture was capable of executing transition phase proving the architecture robust amidst the complex transition flight dynamics.

7 CONCLUSION

Differential flatness is a property that has been studied for a long period of time, and it has been well implemented in control problems associated with robotics and multi-copters. Inspired from these implementations, this study was focussed on devising and implementing a control technique that leverages differential flatness of aircraft kinematics. The author of the project initiated the study by proving the concept's validity in the context of flight control by implementing a basic architecture on a PVTOL model. Upon its successful implementation, a tilt-rotor VTOL aircraft flight dynamics model was developed and a feasible control architecture was devised.

Control allocation was identified as a critical component, and significant work was done to understand and implement a control allocation scheme that works for varying dynamics and tilt-rotor angle configurations. Upon completion, disturbance controllers structure was designed based on the identified dynamics to be controlled. Hover and cruise phase controllers were tuned using automated scripts and MATLAB toolsets.

All the individual modules of the proposed control architecture were then integrated with the Vision Volante dynamics model, and was evaluated for its functionality. From the simulation runs, the controllers performed satisfactorily, providing reasonable control performance improvements in hover phase, and slight improvements in cruise. The remarkable feature of the proposed control architecture is its capability to satisfactorily handle the transition phase where flight dynamics are generally unstable and largely varying. Additionally, the proposed architecture simplifies controller development and tuning due to standard linear controllers in a basic controller structure.

The proposed control technique is identified as a novel approach in the domain of tilt-rotor VTOL flight control, and the research was concluded with a successful implementation of the proposed control concept.

7.1 Future Work

The research quite successfully validates the feasibility of the proposed architecture by implementing it within a model-based environment and performing a non-linear simulation. This study only proves the viability of the proposed concept and is only the beginning for further research and development to realising this control technique for practical implementations. Some of the key aspects that requires future work and the assumptions that should be tackled are given below.

1. **Lateral Control**

   This research only focussed on the longitudinal control of the aircraft, and a similar study assessing the capability of the architecture in lateral dynamics is critical. Practical implementations cannot be devoid of lateral controls, and significant research is necessary to equip the controllers and control allocation to operate in the lateral dynamics as well.

2. **High-Fidelity Flight Dynamics Model**

   In model-based development, the fidelity of the flight dynamics largely decides the practicality of the tuned controllers. The dynamics model used in this work was basic and had several assumptions accounted for, especially in the transitionary phase, which requires extensive modelling data. While the controllers seem to function with the current model, it is important to validate its functionality in a higher-fidelity model. Most importantly, the proposed allocation scheme is heavily reliant on the onboard dynamics model, making high fidelity dynamics model a primary requirement of the controller architecture.

3. **Control Allocation**

   The author implemented two control allocation schemes for different phases of flight, which could be eliminated if a proper dynamics model and a matrix inverse logic could be defined. While pseudo inverse has several advantages including being optimal, computationally simpler and fast, it does not perform within constraints and is dependent on linearity in dynamics. CA techniques such as weighted pseudo inverse and redistributed pseudo inverse techniques could solve these limitations. Testing the architecture with different allocation schemes could prove of significance in improving the controller performance.

4. **Control Input Demands**

   The controller has been designed to control accelerations and angular rates which might not be the appropriate control demands that would conventionally be used for piloted flight or guidance systems. Within the project constraints, the control inputs were defined based on the differential flatness unit, and an approach to account for practical inputs should be studied, especially when switching between hover and cruise flight phases.

5. **Fixed Wing Inverse Dynamics**

   As mentioned above, the cruise controller performance had minimal improvements with the proposed architecture. This is understood to be due to the little representation of aerodynamics by the inverted kinematics. However, as shown by Whidborne [10] and Lu *et al* [6], the fixed wing dynamics are also differentially flat, and exploiting this alongside the inverted kinematics could potentially improve cruise controller performance within the same control architecture.

6. **Gust and turbulence analysis**

   In practical environments, aircraft are constantly subjected to gust and turbulence making it inevitable to be overlooked when evaluating flight controllers. Along with high-fidelity dynamics modelling, detailed gust and turbulence models must be incorporated to test the proposed controller's performance in such conditions. The disturbance controllers could be further adapted with gust alleviation and more robust control techniques to maintain satisfactory performance in difficult atmospheric conditions.

7. **Extensive flight testing**

   Flight testing can be performed in simulation and experimental environments. However, considering the novelty of the proposed control scheme, extensive simulation-based flight testing must be performed to assess and analyse the controller characteristics across the flight envelope. Various manoeuvres relevant to the mission profiles and for safe piloting must carried out to evaluate the handling qualities of the designed controllers, and to identify the extent at which the control scheme could falter.

7.2 Potential Impact

Tilt-rotor VTOL aircraft are certainly going to be a part of future air travel, and developing a control algorithm that meets the requirements of performance and certifiability is key. This study introduces the feasibility of utilising a well understood mathematical concept of non-linear systems, differential flatness, to improve controller performance. The architecture is based on conventional linear controllers, raising confidence in the certifiability of its implementation. This work lays footing for further research and development to mature the differential flatness concept in VTOL flight controls. In this baseline study, the proposed architecture has shown better controller performance, and further work could potentially prove this architecture of practical implementation. Such a control technique has not been explored in the domain of VTOL flight control, and this work intends to incite research efforts into a new variant of flight controllers.

REFERENCES

[1] Porsche Consulting (2018) “The Future of Vertical Mobility - Sizing the market for passenger, inspection, and goods services until 2035,” *A Porsche Consulting study*, pp. 1–36. Available at: https://fedotov.co/wp-content/uploads/2018/03/Future-of-Vertical-Mobility.pdf.

[2] Ducard, G. J. J. and Allenspach, M. (2021) “Review of designs and flight control techniques of hybrid and convertible VTOL UAVs,” *Aerospace Science and Technology*. Elsevier Masson SAS, 118, p. 107035. doi: 10.1016/j.ast.2021.107035.

[3] Martin, P., Rouchon, P. and Murray, R. M. (2003) *Flat Systems, Equivalence and Trajectory Generation*, *CDS Technical Report*. California.

[4] FLIESS, M., LÉVINE, J., MARTIN, P. and ROUCHON, P. (1995) “Flatness and defect of non-linear systems: introductory theory and examples,” *International Journal of Control*, 61(6), pp. 1327–1361. doi: 10.1080/00207179508921959.

[5] Drouin, A., Cunha, S. S., Brandao Ramos, A. C. and Mora-Camino, F. (2011) “Differential flatness and control of nonlinear systems,” *Proceedings of the 30th Chinese Control Conference, CCC 2011*, (9), pp. 643–648.

[6] Lu, W., Mora-Camino, F. and Achaibou, K. (2005) “Differential flatness and flight guidance: A neural adaptive approach,” *Collection of Technical Papers - AIAA Guidance, Navigation, and Control Conference*, 6(August), pp. 4608–4619. doi: 10.2514/6.2005-6342.

[7] Tal, E. and Karaman, S. (2021) “Accurate Tracking of Aggressive Quadrotor Trajectories Using Incremental Nonlinear Dynamic Inversion and Differential Flatness,” *IEEE Transactions on Control Systems Technology*, 29(3), pp. 1203–1218. doi: 10.1109/TCST.2020.3001117.

[8] Murray, R. M. (2010) “Optimization-Based Control.” doi: 10.1007/978-3-662-45777-1_4.

[9] Cowling, I. D., Yakimenko, O. A., Whidborne, J. F. and Cooke, A. K. (2010) “Direct method based control system for an autonomous quadrotor,” *Journal of Intelligent and Robotic Systems: Theory and Applications*, 60(2), pp. 285–316. doi: 10.1007/s10846-010-9416-9.

[10] Whidborne, J. F., Cowling, I. D. and Yakimenko, O. A. (2008) “A Direct Method for {UAV} Guidance and Control,” *Proc 23rd International Conference on Unmanned Air Vehicle Systems*, (0), pp. 37.1-37.13.

[11] Kumar, R., Nemati, A., Kumar, M., Sharma, R., Cohen, K. and Cazaurang, F. (2017) “Tilting-rotor quadcopter for aggressive flight maneuvers using differential flatness based flight controller,” *ASME 2017 Dynamic Systems and Control Conference, DSCC 2017*, 3(October). doi: 10.1115/DSCC2017-5241.

[12] Zeng, J., Kotaru, P. and Sreenath, K. (2019) “Geometric control and differential flatness of a quadrotor uav with load suspended from a pulley,” *Proceedings of the American Control Conference*, 2019-July(3), pp. 2420–2427. doi: 10.23919/acc.2019.8815173.

[13] Martin, P., Devasia, S. and Paden, B. (1996) “A different look at output tracking: Control of a VTOL aircraft,” *Automatica*, 32(1), pp. 101–107. doi: 10.1016/0005-1098(95)00099-2.

[14] Hauser, J., Shankar, S. and George, M. (1992) “Nonlinear Control Design for Slightly Non-minimum Phase Systems: Application to V/STOL Aircraft,” *Automatica*, 28(4), pp. 665–679.

[15] Sira-Ramirez, H. and Fliess, M. (1998) “Regulation of non-minimum phase outputs in a PVTOL aircraft,” *Proceedings of the IEEE Conference on Decision and Control*, 4(December), pp. 4222–4227. doi: 10.1109/cdc.1998.761966.

[16] Petit, N., Milam, M. B. and Murray, R. M. (2001) “Inversion Based Constrained Trajectory Optimization,” *IFAC Proceedings Volumes*, 34(6), pp. 1211–1216. doi: 10.1016/s1474-6670(17)35349-1.

[17] Theys, B., de Vos, G. and de Schutter, J. (2016) “A control approach for transitioning VTOL UAVs with continuously varying transition angle and

controlled by differential thrust," *2016 International Conference on Unmanned Aircraft Systems, ICUAS 2016*, pp. 118–125. doi: 10.1109/ICUAS.2016.7502519.

[18] Johansen, T. A. and Fossen, T. I. (2013) "Control allocation - A survey," *Automatica*. Elsevier Ltd, 49(5), pp. 1087–1103. doi: 10.1016/j.automatica.2013.01.035.

[19] Durham, W., Bordignon, K. A. and Beck, R. (2016) *Aircraft Control Allocation*. First, *John Wiley & Sons*. First. Chichester, UK: John Wiley & Sons, Ltd. doi: 10.1002/9781118827789.

[20] Spannagl, L. and Ducard, G. (2020) "Control allocation for an unmanned hybrid aerial vehicle," *2020 28th Mediterranean Conference on Control and Automation, MED 2020*, pp. 709–714. doi: 10.1109/MED48518.2020.9182865.

[21] Gasco, P. S. (2012) *Development of a Dual Axis Tilt Rotorcraft UAV: Modelling, Simulation and Control*.

[22] Bordignon, K. and Bessolo, J. (2002) "Control allocation for the X-35B," *2002 Biennial International Powered Lift Conference and Exhibit*, (November), pp. 1–10. doi: 10.2514/6.2002-6020.

[23] Werink, R. (2019) "On the Control Allocation of Fully-Actuated and Over-Actuated Multirotor UAVs," p. 56. Available at: http://essay.utwente.nl/77102.

[24] Chen, C., Zhang, J., Zhang, D. and Shen, L. (2017) "Control and flight test of a tilt-rotor unmanned aerial vehicle," *International Journal of Advanced Robotic Systems*, 14(1), pp. 1–12. doi: 10.1177/1729881416678141.

[25] Lombaerts, T., Kaneshige, J., Schuet, S., Hardy, G., Aponso, B. and Shish, K. (2019) "Nonlinear dynamic inversion based attitude control for a hovering quad tiltrotor evtol vehicle," *AIAA Scitech 2019 Forum*, (January), pp. 1–26. doi: 10.2514/6.2019-0134.

[26] Yang, H. and Morales, R. (2021) "Robust Full-Envelope Flight Control Design for an eVTOL Vehicle," in *AIAA Scitech 2021 Forum*. Reston, Virginia: American Institute of Aeronautics and Astronautics. doi: 10.2514/6.2021-0254.

[27] Whidborne, J. (2020) “Dynamics & Control of Multirotors – An Introduction,” (September).

[28] Stevens, B. L., Lewis, F. L. and Johnson, E. N. (2015) *Aircraft control and simulation: Dynamics, controls design, and autonomous systems: Third edition*. 3rd edn, *Aircraft Control and Simulation: Dynamics, Controls Design, and Autonomous Systems: Third Edition*. 3rd edn. John Wiley & Sons. doi: 10.1002/9781119174882.

[29] *The Aston Martin Volante Vision Concept | Aston Martin (United Kingdom)* (no date ac). Available at: https://www.astonmartin.com/en-gb/models/special-projects/the-aston-martin-volante-vision-concept-2019 (Accessed: August 5, 2021).

[30] *Aston Martin Volante Vision* (no date ad). Available at: https://evtol.news/aston-martin-volante/ (Accessed: August 5, 2021).

[31] Diez, B. (2020) *Flight Control Design and Handling Qualities assessment for UAM VTOL vehicle*.

[32] Valdes, G. (2020) *Urban Air Mobility Flight Dynamics*, *Cranfield University*.

[33] *6DOF (Euler Angles)* (2006) *MathWorks*. Available at: https://uk.mathworks.com/help/aeroblks/6dofeulerangles.html (Accessed: August 6, 2021).

[34] Ostowari, C. and Naik, D. (1985) “Post-stall wind tunnel data for NACA 44XX series airfoil sections,” *Solar Energy Research Institute*, (4807), pp. 1–171. Available at: http://www.osti.gov/energycitations/product.biblio.jsp?osti_id=5791328.

[35] Petrilli, J., Paul, R., Gopalarathnam, A. and Frink, N. T. (2013) “A CFD database for airfoils and wings at post-stall angles of attack,” *31st AIAA Applied Aerodynamics Conference*, (919). doi: 10.2514/6.2013-2916.

[36] Drela, M. (2007) *QPROP User Guide*. Available at: https://web.mit.edu/drela/Public/web/qprop/qprop_doc.txt (Accessed: August 10, 2021).

[37] Kang, N. (2021) *Tilt-schedule flight control system of VTOL PAV for a flight naive pilot*.

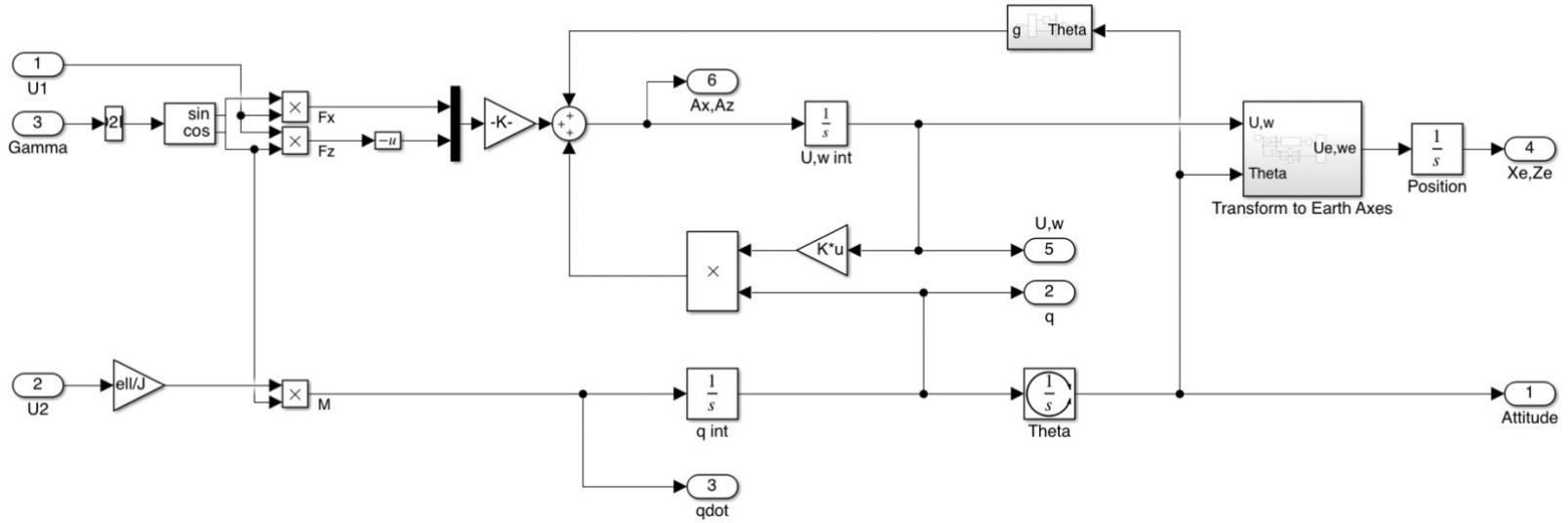
[38] Härkegård, O. (2004) “Dynamic Control Allocation Using Constrained Quadratic Programming,” *Journal of Guidance, Control, and Dynamics*, 27(6), pp. 1028–1034. doi: 10.2514/1.11607.

[39] Zhang, J., Bhardwaj, P., Raab, S. A., Saboo, S. and Holzapfel, F. (2018) “Control allocation framework for a tilt-rotor vertical take-off and landing transition aircraft configuration,” *2018 Applied Aerodynamics Conference*, pp. 1–19. doi: 10.2514/6.2018-3480.

APPENDICES

Appendix A Conceptual Validation

A.1 PVTOL Dynamics Model (SIMULINK)



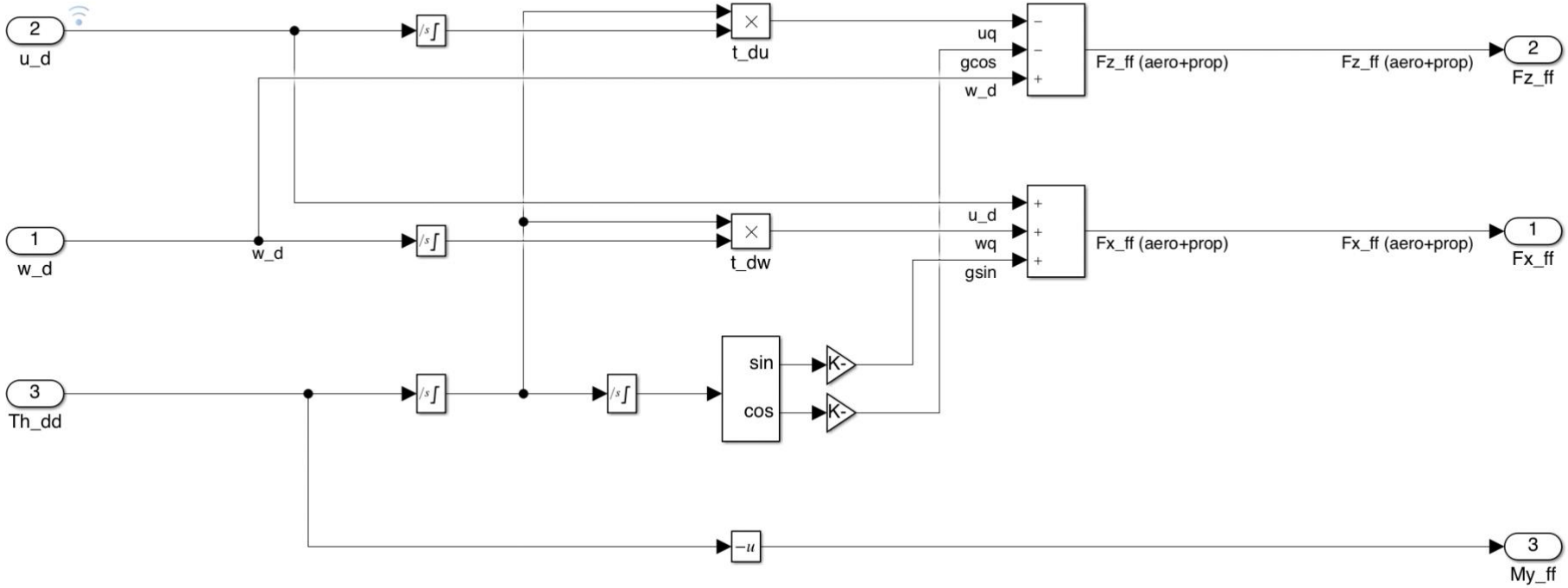
Appendix B Control Allocation

B.1 Function Code

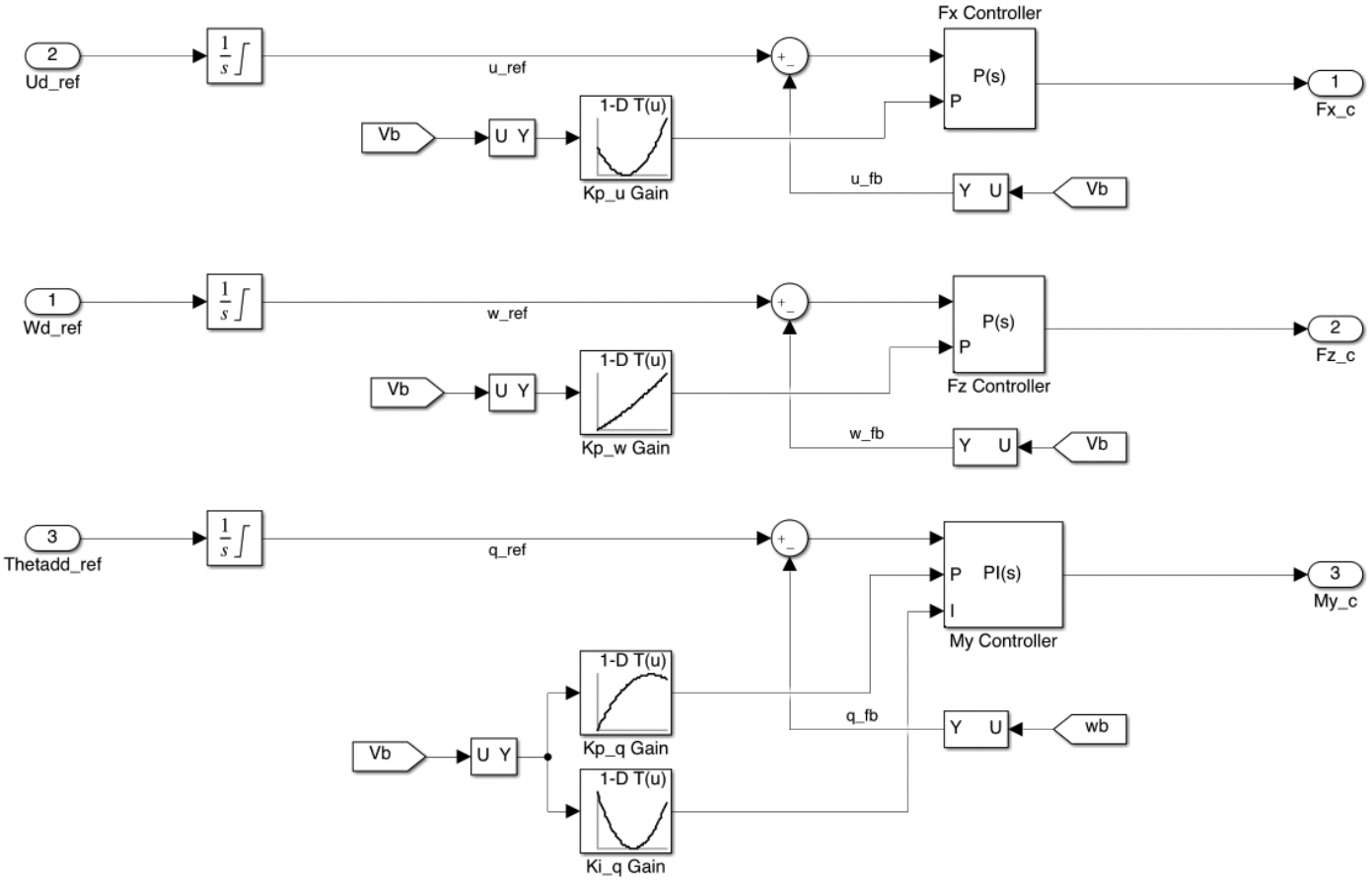
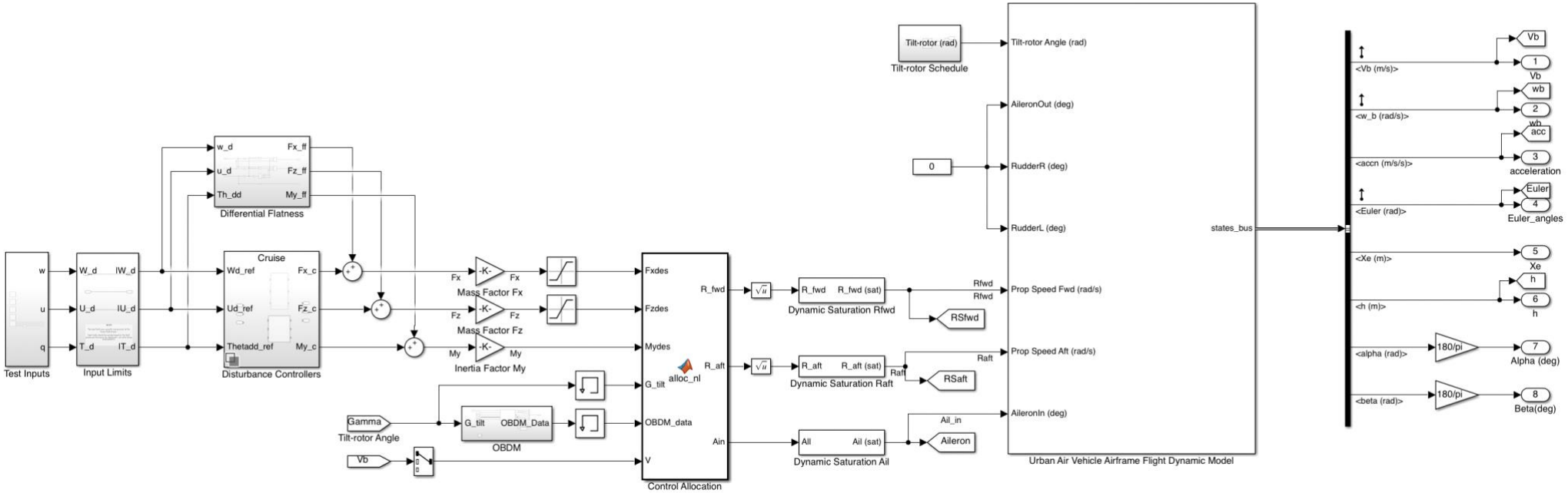
```
function [R_fwd, R_aft, Ain] = alloc_nl(Fxdes, Fzdes,
Mydes,G_tilt, OBDM_data, V)

Xmain = -0.9080;        % Moment arm length (Aft Prop)
Xfwd =  3.3853;         % Moment arm length (Fwd Prop)
Sref = 15.8431;         % Wing span area (m^2)
Cref = 1.9462;

% Obtaining data from the OBDM for CA
CL_dinA = OBDM_data(1);
Cm_dinA = OBDM_data(2);
CD_dinA = OBDM_data(3);
DP      = OBDM_data(4);
Fx_aero = OBDM_data(5);
Fz_aero = OBDM_data(6);
My_aero = OBDM_data(7);
TC_aft  = OBDM_data(8)/(4*pi^2);
TC_fwd  = OBDM_data(9)/(4*pi^2);
alpha   = OBDM_data(10);
Fx_grav = OBDM_data(11);
Fz_grav = OBDM_data(12);

% Desired Forces and Moments (v_desired)
v = [Fxdes - (Fx_aero + Fx_grav);
     Fzdes - (Fz_aero + Fz_grav);
     Mydes - My_aero];

if V < 20
    v = [Fxdes - (Fx_aero);
     Fzdes - (Fz_aero);
     Mydes - My_aero];

    T_aft = (v(3) - v(2)*Xfwd)/(Xfwd - Xmain);
    T_fwd = sqrt(v(1)^2 + (v(2)- T_aft)^2);
    Ain = 0;
else
    % Control Effectiveness Matrix
    B_actual = [0 cos(G_tilt) -((sin(alpha)*CL_dinA  -
cos(alpha)*CD_dinA)*Sref*DP);
                -1 -sin(G_tilt) -((sin(alpha)*CD_dinA +
cos(alpha)*CL_dinA)*Sref*DP);
               -Xmain (Xfwd*sin(G_tilt))
Sref*Cref*DP*Cm_dinA];

    % Normalisation Factor
    E = [5000 0 0;
        0 5000 0;
        0 0 25];

    % Normalised B Matrix
    B = B_actual*E;
    Binv = B*B';

    %Pseudo-inverse Matrix
    u_norm = (B'/Binv)*v;
    u = E*u_norm;

    T_aft = u(1);
    T_fwd = u(2);
    Ain   = u(3);

end
R_fwd = abs((0.5*T_fwd)/TC_fwd); %Rotor speed input (fwd)
R_aft = abs((T_aft)/TC_aft);%Rotor speed input (aft)

end
```

Appendix C Controller Implementation

C.1 Differential Flatness Unit – SIMULINK Implementation



C.2 Cruise Disturbance Controller – Gain Scheduling



C.3 Differential Flatness Controller – SIMULINK Implementation