School of Science and Engineering
Department of Mechanical Engineering

Dynamics and Control of a Multi-Rotor Aircraft

Sean Anthony Kava

This thesis is presented for the Degree of
Master of Philosophy (Mechanical Engineering)
of
Curtin University

May 2016

Declaration

To the best of my knowledge and belief this thesis contains no material previously published by any other person except where due acknowledgment has been made.

This thesis contains no material which has been accepted for the award of any other degree or diploma in any university.

Signature: Saankara ……………………………………………

Date: 13/05/2016 ………………………...

Abstract

This thesis presents the development and implementation of a stochastic control scheme to stabilise an underactuated quadrotor vertical take-off and landing (VTOL) unmanned aircraft, under both deterministic and stochastic environmental loads, to track a reference trajectory in three-dimensional (3-D) space. In practice, environmental loads acting on aircraft are induced by both laminar and turbulent airflow (deterministic and stochastic loads). In spite of this, loads are either ignored or considered deterministic in all existing works on controlling this type of aircraft. In practice, control performance is deteriorated as a result of simplifying the aircrafts aerodynamic properties in such a manner. The proposed stochastic control scheme has been designed from the outset to address this unique problem.

The stochastic control as a result of this work is based on the fusion of several recent results developed for stability and control of stochastic systems. Combining one-step-ahead backstepping with standard backstepping and Lyapunov's direct design method in conjunction with output state feedback a stochastically practically asymptotically stable controller is achieved.

A combination of Euler angles and Modified Rodrigues Parameters (MRP) are used for the aircrafts attitude representation to achieve an effective control design, and to achieve stable path-tracking control results. The modified Rodrigues parameters are used for attitude representation to reduce singularities in comparison with the use of Euler-angles and Rodrigues parameters. Weak and strong nonlinear Lyapunov functions are introduced to overcome difficulties caused by underactuation and Hessian terms induced by the stochastic differentiation rule. To overcome the inherent underactuation of the aircraft, the roll and pitch angles of the aircraft are considered as immediate controls. Introducing projection functions allows the design of estimates for both the deterministic components and covariance's of the stochastic components of the disturbances acting on the system.

Numerical evaluation of the proposed stochastic control scheme and comparison with a deterministic controller based on a similar design methodology is achieved through simulation. Furthermore, both deterministic and stochastic control schemes are implemented on a real quadrotor for attitude stabilisation.

Table of Contents

List of Tables

List of Figures

Glossary

Arduino	Is the brand name of a family of form factor microcontrollers designed for robotic system development.
CASA	‘Civil Aviation Safety Authority’. The body governing and regulating the laws for civil and commercial aviation in Australia.
Deterministic	Referring to a ‘deterministic system’. A system in which no randomness is involved in the development of future states of the system.
ESC	‘Electronic Speed controller’. A small device that takes a reference signal and voltage source and produces a signal to drive a three phase brushless DC motor.
Gaussian	Referring to ‘Gaussian Distribution’. A theoretical distribution with known mean and variance represented as a symmetrical bell shaped graph.
GPS	‘Global Positioning System’. A series of space based satellites used for the communication of position and time.
GNSS	‘Global Navigational Satellite System’. A global navigation system using positioning satellites and sometimes supplemented with inertial measurements.
IMU	‘Inertial Measurement Unit’. A device which uses a combination of accelerometers, gyroscopes and magnetometers to provide velocity and orientation information.
Jacobian	Referring to ‘Jacobin Matrix’. A matrix of the first order partial derivatives of one vector with respect to another.
MATLAB	‘Matrix Laboratory’. A programming language for numerical computing, with an emphasis on matrix mathematics.
MRP	‘Modified Rodrigues Parameter’. A method for representing the angular position about an axis.
Stochastic	Referring to a ‘stochastic system’. A system having a random probability distribution or pattern that may be analysed statistically but may not be predicted precisely.
VTOL	‘Vertical Take-Off and Landing’. Is an aircraft which can hover and take off and land vertically such example is the helicopter.
UAV	‘Unmanned Aerial Vehicle’. Is an aircraft with no pilot on board.
XBee	Is The brand name of a family of form factor compatible radio modules from Digi International.

Nomenclature

Symbol	Definition
$\boldsymbol{\alpha}_1$	*Virtual control signal for linear velocity.*
$\boldsymbol{\alpha}_2$	*Virtual control signal for angular positions Euler representation.*
α_φ	*Virtual control for the pitch angle.*
α_θ	*Virtual control for the roll angle.*
α_ψ	*Virtual control for the yaw angle.*
$\boldsymbol{\alpha}_q$	*Virtual control signal for angular positions MRP representation.*
α_{q_1}	*Virtual control for the MRP 1.*
α_{q_2}	*Virtual control for the MRP 2.*
α_{q_3}	*Virtual control for the MRP 3.*
$\boldsymbol{\alpha}_\omega$	*Virtual control signal for angular velocity.*
Δ_i	*Noise covariance matrix* i.
γ_i	*Controller positive constant gain* i.
δ	$\|\Delta(t)\Delta^T(t)\|_\infty^2$.
$\hat{\delta}$	*Estimate of* δ.
$\tilde{\delta}$	*Error between* δ *and* $\hat{\delta}$.
ε_i	*Young's inequality positive constant parameter* i.
ς	*Velocity vector for both linear velocity and angular velocity in the body-fixed frame.*
ς_a	*Disturbance air velocity vector for both linear velocity and angular velocity in the body-fixed frame.*
ς_r	*Relative velocity vector between* ς *and* ς_a.
$\boldsymbol{\eta}_1$	*position vector of the quadrotor expressed in the inertial frame.*
$\boldsymbol{\eta}_{1d}$	*Given trajectory for the position.*
$\boldsymbol{\eta}_{1e}$	*Tracking error between position and given trajectory.*
$\hat{\boldsymbol{\eta}}_1$	*Position vector estimate of the quadrotor expressed in the inertial frame.*

$\tilde{\boldsymbol{\eta}}_1$	*Error between the actual and estimate of the quadrotors position expressed in the inertial frame.*
$\boldsymbol{\nu}_1$	*Velocity vector of the quadrotor expressed in the inertial frame.*
$\boldsymbol{\nu}_{1e}$	*Tracking error between velocity and its virtual control.*
$\hat{\boldsymbol{\nu}}_1$	*Velocity vector estimate of the quadrotor expressed in the inertial frame.*
$\tilde{\boldsymbol{\nu}}_1$	*Error between the actual and estimate of the quadrotors velocity expressed in the inertial frame.*
$\boldsymbol{\eta}_2$	*Attitude of the quadrotor expressed in Euler angles after the xyz convention.*
$\boldsymbol{\eta}_{2e}$	*Tracking error between attitude and its virtual control.*
$\boldsymbol{\omega}$	*Angular velocity of the quadrotor expressed in the body-fixed frame.*
$\boldsymbol{\omega}_e$	*Tracking error between angular velocity and its virtual control.*
$\lambda_{\mathrm{m}}(\cdot)$	*Minimum eigenvalue of matrix*$(\cdot)$.
$\lambda_{\mathrm{M}}(\cdot)$	*Maximum eigenvalue of matrix*$(\cdot)$.
φ	*Roll angle of the quadrotor.*
θ	*Pitch angle of the quadrotor.*
$\sigma(\cdot)$	*Saturation function of*$(\cdot)$.
$\boldsymbol{\tau}$	*Control torque of the quadrotor expressed in the body-fixed frame.*
$\boldsymbol{\tau}_{Aero}$	*Aerodynamic disturbance torque expressed in the body-fixed frame.*
$\bar{\boldsymbol{\tau}}_{Aero}$	*Deterministic component of aerodynamic disturbance torque expressed in the body-fixed frame.*
$\tilde{\boldsymbol{\tau}}_{Aero}$	*Stochastic component of aerodynamic disturbance torque expressed in the body-fixed frame.*
$\hat{\boldsymbol{\tau}}_{Aero}$	*Estimate of the disturbance torque expressed in the body-fixed frame.*
$\boldsymbol{\tau}_{Aero,de}$	*Error between the actual disturbance torque and its estimate expressed in the body-fixed frame.*
ψ	*Yaw angle of the quadrotor.*
Ω_i	*Motor angular velocity of motor* i.
C_L	*Aerodynamic coefficient of lift.*

C_D	*Aerodynamic coefficient of drag.*
C_T	*Aerodynamic coefficient of thrust.*
C_R	*Aerodynamic coefficient of rotation.*
$\mathbf{C}_B$	*Coriolis and Centripetal matrix of aircraft.*
$\mathbf{C}_{B1}$	*Coriolis matrix of aircraft.*
$\mathbf{C}_{B2}$	*Centripetal matrix of aircraft.*
$\mathbf{C}_A$	*Coriolis and Centripetal matrix for added mass and inertia of aircraft.*
$\mathbf{C}_{A1}$	*Coriolis matrix for added mass of aircraft.*
$\mathbf{C}_{A2}$	*Centripetal matrix for added inertia of aircraft.*
$\mathbf{D}_i$	*Damping matrix i.*
f	*Control force of the quadrotor expressed in the body-fixed frame.*
$\boldsymbol{f}_{Aero}$	*Aerodynamic disturbance force expressed in the inertial-fixed frame.*
$\bar{\boldsymbol{f}}_{Aero}$	*Deterministic component of aerodynamic disturbance force expressed in the inertial-fixed frame.*
$\tilde{\boldsymbol{f}}_{Aero}$	*Stochastic component of aerodynamic disturbance force expressed in the inertial-fixed frame.*
$\hat{\boldsymbol{f}}_{Aero}$	*Estimate of the aerodynamic disturbance force.*
$\boldsymbol{f}_{Aero,de}$	*Error between the actual aerodynamic disturbance force and its estimate.*
g	*Gravitational acceleration experienced by the quadrotor.*
I_x	*Intertie of quadrotor about its X axis.*
I_y	*Intertie of quadrotor about its Y axis.*
I_z	*Intertie of quadrotor about its Z axis.*
$\mathbf{I}_H$	*Inertia matrix.*
$\mathbf{K}_i$	*Control gain matrix i.*
L	*Quadrotor radius.*
L_x	*Distance along quadrotor x axis from centre of gravity to motors 2,3,5 and 6.*
L_y	*Distance along quadrotor Y axis from centre of gravity to motors 2,3,5 and 6.*

m	*Mass of quadrotor.*
$\boldsymbol{q}$	*Attitude of the quadrotor expressed in MRP.*
q_1	*MRP 1.*
q_2	*MRP 2.*
q_3	*MRP3.*
$\boldsymbol{q}_e$	*Tracking error between attitude and its virtual control.*
$\mathbf{R}_1$	*Transformation matrix of angular position from B-frame to E-frame.*
$\mathbf{T}$	*Transformation matrix of angular velocity from B-frame to E-frame with Euler angles.*
$\mathbf{R}_2$	*Transformation matrix of angular velocity from B-frame to E-frame with MRP.*
$\boldsymbol{S}(\cdot)$	*Skew symmetric matrix.*
$\boldsymbol{U_1}$	*Sum of thrust forces produced by each motor.*
$\boldsymbol{U_2}$	*Pitching moment produced by motors.*
$\boldsymbol{U_3}$	*Rolling moment produced by motors.*
$\boldsymbol{U_4}$	*Yawing moment produced by motors.*
$\boldsymbol{v^b}$	*Linear velocity vector expressed in the body - fixed frame.*
$\boldsymbol{v}_a^b$	*Linear velocity of the airflow expressed in the body - fixed frame.*
$\overline{\boldsymbol{v}}_a^b$	*Deterministic component of the linear velocity of the airflow expressed in the body - fixed frame.*
$\widetilde{\boldsymbol{v}}_a^b$	*Stochastic component of the linear velocity of the airflow expressed in the body - fixed frame.*
V_i	*Lyapunov function i.*

Acknowledgements

To my Mother, Father and Sisters, for their support and encouragement throughout my life, thank you for being behind me in everything that I pursued.

To John Kava for time spent proof reading this thesis.

To my supervisor Duc, you have been truly inspirational during this journey, your passion for this discipline has been infectious and it has been a joy to complete this work under your guidance.

To my co-supervisor Lei, for allowing me the opportunity to start this project under his supervision.

To the staff of the Mechatronics department during my time at Curtin – Brad, Euan, Duc, Jonathan, and Lei. You made mechatronics an exciting and challenging field and no doubt helped shape me into the engineer I am today.

To the musical entities Tommy Emanuel and Metallica for proving inspiring music that helped in the writing of this thesis.

1 Introduction

Figure 1-1: Quadrotor helicopter.

Multi-rotor aircraft are an attractive Vertical Take-Off and Landing (VTOL) with multiple applications in both civilian and military settings. A typical multi-rotor type Unmanned Aerial Vehicle (UAV), the quadrotor has a rigid cross frame body equipped with four fixed pitch propellers or rotors (see Figure 1-1). The four rotors are configured such that two counter-rotate relative to the other two. (i.e. Two rotors turn clockwise and the other two counter-clockwise). The paired opposing rotation compensates for the reactive torques. The vertical motion (altitude) is obtained by the collective speed increase or decrease of all four rotors. The pitch and roll motion are obtained by changing the speed of front-rear pair and the left-right pair of rotors, respectively. Yaw motion is realised by the difference in the reactive torques between the two pairs of rotors. Horizontal motion (latitude and longitude) result from the combination of the pitch, roll and vertical motions. Each fixed pitch rotor is directly connected to its motor and there is no capability to change the direction of rotation of the rotors. The motions of the quadrotor aircraft are nonlinearly coupled. Furthermore, the aircraft is underactuated because there are only four independent control inputs as each motor is fixed with only its rotor able to rotate and not the motor housing as to the pitch of the propellers. However, there are six degrees of freedom (latitude, longitude, altitude, pitch, roll, yaw) to be controlled, see [1] for more details on controlling other underactuated mechanical systems. Difficulties in controlling the motions of the quadrotor aircraft arise from the nonlinear coupling and underactuation mentioned along with the non-deterministic (chaotic) nature of turbulent airflow being applied to the airframe. A brief review of the related works on controlling multi-rotor aircraft is given below to motivate contributions of the present thesis.

1.1 Literature Review

Due to the challenging nature of this problem, controlling a VTOL aircraft was initially restricted to a vertical plane. In works by [2], [3] and [4] an input-output linearization approach was used to develop controllers for stabilization and output tracking/regulation of a VTOL aircraft. By choosing to stabilise the output of a VTOL at a fixed point with respect to the aircraft's body, for example at the Huygens centre of oscillation, several controllers for stabilizing the aircraft have been designed and presented by [5], [6], [7] and [8]. Because the quadrotor aircraft operates in three-dimensional (3D) space, controlling all six degrees of freedom has recently attracted attention of researchers in the control and robotics communities. Firstly, attitude control was obtained with the use of nonlinear feedback to ensure global asymptotic stability in [9] and exponential stability in [10]. Local position control was achieved by multiple controllers for instance a saturation controller was developed in[11], full state backstepping was used in [12], a command filtered backstepping technique was employed in [13]. Position control has also been addressed and several controllers designed using a backstepping controller without velocity measurements in [14] and an adaptive controller with disturbance estimation was presented in [15]. Local position control was achieved on a physical quadrotor with differing controllers, for example a backstepping like feedback linearization controller was designed and implemented in [16]. A quadrotor with wall climbing capability was presented in [17],. A haptic teleoperation control architecture was presented in [18]. While a unified nonlinear optimal control was presented in [19].

In addition, with there being only four independent inputs and six outputs to be controlled when flying a quadrotor in the 3D space, there is a second problem if Euler angles are used to represent the aircrafts attitude, singularities occur, in the kinematic equations describing the motions of the aircraft. This usage limits the working space of the aircraft (the pitch angle needs to be within $\left(\pm\frac{\pi}{2}\right)$, (see for example [16], [17], [18], [19]). In addition to the above works, control of quadrotor aircraft under bounded control inputs has also been considered in several bodies of work such as [20], [21], [22], [23] based on the use of nested saturation control design method [24] and its alternatives. However, a one-step ahead backstepping controller and the unit quaternion were used in [25] to obtain global asymptotic control. In all these works on control of multi-rotor aircraft and others not listed here, the loads (forces and moments) induced by airflow on the aircraft were neglected or assumed to be deterministic. In practice these assumptions do not hold, as wind loads acting on the aircraft always contain both laminar and turbulent flow. Moreover, airflow can be both steady and unsteady. Steady flow refers to the condition where the fluid properties at a point in the system do not change over time. Otherwise, flow is considered unsteady. Airflow can be steady or unsteady depending on the chosen frame of reference. For instance, laminar flow over a sphere is steady in the frame of reference that is stationary with respect to the sphere. In a frame of reference that is stationary with respect to a background airflow, the airflow is unsteady. Turbulence is flow characterized by recirculation, eddies, and apparent randomness. Flow in which turbulence is not exhibited is called laminar. By definition turbulence is unready airflow

This thesis treats these loads as having both deterministic and stochastic components respectively. Neglecting these components can result in significant deterioration of the control performance. As such, stochastic differential equations (SDE) will be used to model the dynamics of the aircraft. SDE's are useful for modelling systems, with high degrees of uncertainty which is not reasonable to be ignored as is the case for turbulent airflow about an object.

1.2 Motivation

In the case of the quadrotor, the majority of stochastic disturbance will occur at higher frequencies from the vibrating mechanical structure, motors and turbulent airflow over the airframe and across the rotor blades. Linear dampening is present in the mechanical structure of the aircraft, as a direct result of viscous friction between the airframe and surrounding air. With this in mind, it is possible to think of the quadrotor system as a low pass filter. It is therefore reasonable to ask, *"Why consider the aircraft under stochastic loads when high frequency noise is filtered out by a low pass filter"*. The answer is: that the stochastic disturbance acting on the system is in practice both additive and multiplicative. Additive noise in practice will be filtered out by the low pass filter, however, multiplicative noise will pass through a low pass filter as pointed out by Lindsey [26]. This phenomenon can be demonstrated if we consider the following system:

$$\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x}, \quad \boldsymbol{x}(t_0) \in \mathbb{R}^n, \tag{1-1}$$

where, $\mathbf{A}$ is a Hurwitz matrix. This system is globally exponentially stable at the origin. Assuming that the system (1-1) is perturbed by some noise and the stochastic perturbed system is given by the following stochastic ordinary differential equation (SODE):

$$d\boldsymbol{x} = \mathbf{A}\boldsymbol{x}dt + \mathbf{B}\boldsymbol{x}dw, \quad \boldsymbol{x}(t_0) \in \mathbb{R}^n, \tag{1-2}$$

where, $\mathbf{B}$ is a skew symmetric matrix, and w is standard one dimensional wiener process. The SDE (1-2) has the following solution

$$\boldsymbol{x}(t) = x_0 + e^{\left(\mathbf{A}-0.5\mathbf{B}^2\right)(t-t_0)} + \mathbf{B}\big(w(t) - w(t_0)\big)\boldsymbol{x}(t_0), \qquad \forall t \geq 0. \tag{1-3}$$

Since $\mathbf{B}$ is a skew symmetric matrix, $\mathbf{B}^2$ is negative definite. Thus, convergence rate of $\boldsymbol{x}(t)$ to zero in probability reduces, and even $\boldsymbol{x}(t)$ tends to infinity almost surely exponentially if $\mathbf{B}^2$ is such that $(\boldsymbol{A} - 0.5\,\boldsymbol{B}^2)$ is positive definite, i.e., the solution can become almost surely exponentially unstable if perturbed by a stochastic noise. Since the Coriolis, centripetal matrix of aircraft dynamics section 2.6 deterioration and instability of closed loop systems due to stochastic perturbation under existing control algorithms will be similar to those in the above example.

The above discussion motivates contributions of this paper on a constructive design of controllers for a quadrotor aircraft to track a reference path in 3D space under both deterministic and stochastic disturbances with bounded control.

1.3 Proposed Control Design

The aim of this thesis is the development and verification of a control law for a quadrotor aircraft operating under both deterministic and stochastic conditions. This control law will use the one-step ahead backstepping method, standard backstepping method and Lyapunov's direct method. The controller will allow the quadrotor to follow a linear position and yaw angle trajectory autonomously under differing disturbances. The stochastic control design is based on the following main ideas:

Firstly, the aircraft attitude is represented by a combination of Euler angles and the Modified Rodrigues Parameters (MRP). The MRP are chosen for the attitude representation of the aircraft, to reduce the effect of singularities on the system dynamics, in comparison with the use of Euler-angles and Rodrigues parameters. While similar to these two representations, the MRP allows all the Eigen axis rotations in the range of [0, 2π), i.e., there is only one singularity, which is moved as far from the equilibrium point (the origin) as possible yielding an effective control design.

Second, to overcome difficulties caused by under-actuation and Hessian terms induced by the stochastic differentiation rule, weak and strong nonlinear Lyapunov functions are introduced. The weak nonlinear Lyapunov function is referred to as the case where the function grow no faster than linearly of its argument. The strong case is where the function grows faster than square of its argument.

Third, design a deterministic controller based on standard backstepping to obtain attitude stabilization of the quadrotor, by using a disturbance observer to compensate for aerodynamic disturbances acting on the system. Then, design a stochastic controller based on standard backstepping, to obtain attitude stabilization of the quadrotor. Using projection algorithms estimate and compensate for both deterministic and stochastic aerodynamic disturbances acting on the aircraft.

Fourth, validate and contrast the use of both control strategies presented in the previous point by numerical simulation in MATLAB. Next, examine the performance of the deterministic controller under the stochastic conditions to validate the decision to design a stochastic controller.

Fifth, validate and contrast the use of both control strategies presented in the third point by testing on a physical quadrotor.

Sixth, to design both a deterministic and stochastic controller, to allow the quadrotor to follow a linear position and yaw angle trajectory, based on a deterministic and stochastic model respectively. These two controllers are to be based on the one-step ahead backstepping and standard backstepping methods. The controllers are to build on those identified in the third point. These controllers will employ a state estimator to estimate the linear velocity of the aircraft as full state feedback is not possible on a physical system. The roll and pitch angles of the aircraft will be used as immediate controls, to overcome the inherent under-actuation in the aircraft. For the deterministic controller's predefined upper limit of the disturbances acting on the system are granted by the introduction of a disturbance observer. For the stochastic controller, both predefined upper and lower limit estimates of mean values and covariance of wind loads, are ensured by an introduction of projection algorithms.

Finally, validate the deterministic and final new stochastic controller by numerical simulations in MATLAB, to examine the validity of the new proposed controller. By comparing and contrasting the behaviour of the deterministic controller and stochastic controller implemented on the stochastic system, the effects of ignoring the stochastic components of disturbances in the control design can be highlighted.

1.4 Thesis Structure

The contents of the chapters comprising this thesis are as follows;

Chapter 2 Preliminaries, contains an introduction to Lyapunov theory for a deterministic system, followed by an introduction to stochastic differential equations and an extension of Lyapunov theory to stochastic systems. Both backstepping and one-step ahead backstepping control theory are presented followed by state and disturbance estimation methods. An example of a simple system stabilised by the backstepping method when acted upon by deterministic disturbances is presented, simulated and analysed. Finally, the analysis of the dynamics and equations of motion for the quadrotor are presented.

In Chapter 3 Attitude Control, the control design of the quadrotor attitude system is presented operating under deterministic conditions. Specific model assumptions are followed by stability analysis of the presented control scheme.

In Chapter 4 Multi-rotor Control, the control design for the complete quadrotor system under both stochastic and deterministic conditions are presented. The corresponding control objectives along with stability analysis for each controller are also presented.

In Chapter 5 Numerical Simulation, numerical simulation results are presented and discussed to validate the works of chapter 3 and 4.

In Chapter 6 Physical Implementation, a physical quadrotor is presented with all parts identified and catalogued. A test rig construction is presented to allow testing of the proposed deterministic control schemes from chapter 3. Results from the experimental testing are reviewed and discussed.

Finally, Chapter 7 Discussion & Conclusions, the final discussion and conclusions are presented.

1.5 Published works

At the point of submitting this thesis a conference article titled “Dynamics and Control of a Multi-Rotor Aircraft” has been submitted to the 20th Australasian Fluid Mechanics Conference (AFMC). However, notification on the acceptance of this conference article is not made until after the submission of this thesis.

2 Preliminaries

This chapter discusses the methods required for the control design and is broken up into eight sections.

Firstly, the principles of nonlinear systems are presented and discussed including the comparison principle, autonomous systems, non-autonomous systems, along with Lyapunov's direct and indirect methods.

Secondly, a general overview of Lyapunov theory is presented and conditions for stability of deterministic systems.

Thirdly, an overview of stochastic systems and stability of such systems is presented. This is an expansion of the conditions for stability, presented in the second section, into the stochastic situation.

Fourthly, the Lyapunov control design methods of standard backstepping and one-step ahead backstepping are presented and explained.

Fifthly, a set of governing equations are developed and presented for modelling the multi-rotor.

Sixthly, state estimation is presented for dealing with unmeasured system states. A disturbance observer for estimating external disturbances acting on the system for a deterministic aircraft model is presented. The projection algorithm for estimating external disturbances acting on the system for a stochastic aircraft model is developed and presented.

Finally, the control schemes are presented in relation to a simple deterministic system with disturbance estimation, simulated and studied.

2.1 Definitions

2.1.1 Comparison Principle

Consider the general system (2-1):

$$\dot{x} = f(t, x), \quad \text{(2-1)}$$

where $x \in \mathbb{R}^n$ is the state vector and $f(t, x): D \to \mathbb{R}^n$ is a locally Lipschitz map in the domain $D \subset \mathbb{R}^n$ into $\mathbb{R}^n$.

In certain situations, it is necessary to compute the bounds of $x(t)$ without actually computing the solution itself, in such a situation we can use the comparison lemma presented below. The principle applies to situations where the derivative of a scalar differentiable function $v(t)$ satisfies the differential inequality $\dot{v}(t) \leq f\big(t, v(t)\big)$ for all t in a certain time interval. The lemma compares the differential inequality $\dot{v}(t) \leq f\big(t, v(t)\big)$ with the differential equation $\dot{u} = f(t, u)$. The lemma applies even when $v(t)$ is not differentiable, but has an upper right-hand derivative $D^{+}v(t)$ which satisfies a differential inequality.

Lemma 2.1 (Comparison Lemma)

$$\dot{u} = f(t, u), u(t) = u_0, \quad \text{(2-2)}$$

where $f(t;\ u)$ is continuous in t and locally Lipschitz in u, for all $t \geq 0$ and all $u \in J \subset \mathbb{R}$. Let $[t0;\ T)$ (T could be infinity) be the maximal interval of existence of the solution $u(t)$, and suppose $u(t) \in J \forall t \in [t0;\ T)$. Let $v(t)$ be a continuous function whose upper right-hand derivative $D^{+}v(t)$ satisfies the differential inequality

$$D^{+}v(t) \leq f\big(t, v(t)\big) \quad v(t) \leq u_0, \quad \text{(2-3)}$$

with $v(t) \in J \forall\ t \in [t0;\ T)$. Then, $v(t) \leq u(t),\ \forall t \in [t0;\ T)$.

Proof see *Khalil* [27]

From this lemma if there is a function $v(t)$ consisting of the state vector x it can be shown that the function and state are bounded. Moreover, depending on the design of $v(t)$ we will be able to make conclusions to the bound of x.

2.1.2 Autonomous System

Consider a dynamic system, which satisfies

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, t), \qquad t \geq 0\,, \quad \text{(2-4)}$$

$$\boldsymbol{x}(t_0) = \boldsymbol{x}_0, \qquad t_0 \geq 0, \quad \text{(2-5)}$$

where $\boldsymbol{x} \in \mathbb{R}$, $\boldsymbol{f}$ is a given nonlinear continuous function in $\boldsymbol{x}$ with $\boldsymbol{f}(\boldsymbol{x}): D \to \mathbb{R}^n$ is a locally Lipschitz map in the domain $D \subset \mathbb{R}^n$ into $\mathbb{R}^n$ and t and $t \in \mathbb{R}$ and it is assumed that $\boldsymbol{f}(\boldsymbol{x}, t)$ satisfies the standard conditiond for existence and uniqueness of solutions. The nonlinear system in

(2-4) is called autonomous if $\boldsymbol{f}\,(t, \boldsymbol{x})$ does not depend explicitly on time, i.e., if the system can be written as:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}), \qquad t \geq 0. \tag{2-6}$$

Otherwise, the system is a non-autonomous system. A system state $\boldsymbol{x}$ is said to be an equilibrium point of the system if the real roots of the equation:

$$\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{0}, \tag{2-7}$$

where $\boldsymbol{f}(\boldsymbol{x})$ satisfies $\boldsymbol{f}(\boldsymbol{0}) = \boldsymbol{0}$ meaning that the system has an equilibrium point at $\boldsymbol{x} = \boldsymbol{0}$.

2.2 Lyapunov Theory

When designing a control law, it is important to consider the stability of the resulting closed loop system. Normally the approach is to prove a specific form of stability. However, in this thesis the approach and focus is on the stability of equilibrium points. This is because we are concerned with how well the quadrotor behaves during flight and studying its behaviour and ability to reach equilibrium will provide evidence of this. At equilibrium, the derivatives of the state will become zero implying the states are non-changing and the system has reached the desired state.

The Lyapunov stability criterion is a general and useful approach to analyse the stability of nonlinear systems. Lyapunov stability theory includes two different methods of approaching a problem,

(i) Lyapunov indirect method, and
(ii) Lyapunov direct method.

Lying at the core of the indirect Lyapunov method is the idea that linearising the system around a particular point allows for local stability, with small stability regions about this point.

In comparison the fundamental concept of Lyapunov's direct method is; if the total energy of a system is continuously dissipating, then the system will eventually reach an equilibrium point and remain at that point. This method consists of two steps. Firstly, a suitable scalar function is chosen and this function is referred to as Lyapunov function [27]. Secondly, is the evaluation of the first order time derivative of the Lyapunov function along the trajectory of the system. If the derivative of a Lyapunov function is decreasing along the system trajectory as time increases, then the system energy is dissipating and the system will finally settle down to equilibrium. This method, while more intensive than the indirect method, is applied directly to a nonlinear system without the need to linearise the system, thus allowing the controller to achieve both local and global stability. As such, Lyapunov's direct method is used for the remainder of this thesis.

The remainder of this section explains Lyapunov's direct method for the stability analysis of equilibrium points. For convenience, the equilibrium point hereafter will always be at the origin of the system, as a coordinate transformation can always move the equilibrium point to the origin without loss of generality.

Definition 2.1 *The equilibrium point* $\boldsymbol{x} = \mathbf{0}$ *of* (2-6) *is:*

- *Stable if, for each ε > 0, there is δ = δ(ε) > 0 such that:*

$$\|\boldsymbol{x}(0)\| < \delta \Rightarrow \|\boldsymbol{x}(t)\| < \varepsilon, \forall\, t \geq 0. \quad \text{(2-8)}$$

- *Unstable if not stable.*
- *Asymptotically stable if it is stable and δ can be chosen such that:*

$$\|\boldsymbol{x}(0)\| < \delta \Rightarrow \lim_{t\to\infty} \boldsymbol{x}(t) = 0. \quad \text{(2-9)}$$

Proof see Khalil page 112 [27]

This means that the equilibrium point is stable if, for a neighbourhood of arbitrarily small ε, there is a neighbourhood of δ such that a trajectory $\|\boldsymbol{x}(t)\|$ with origin $\boldsymbol{x}(t_0)$ belonging to δ stays in ε for all future time as the trajectory tends to the origin.

The equilibrium point is unstable if there is a neighbourhood ε around the origin such that no neighbourhood of δ exists such that the initial state $\boldsymbol{x}(t_0)$ ensures that $\|\boldsymbol{x}(t)\| < \varepsilon$.

The equilibrium is asymptotically stable if the conditions for a stable equilibrium are met and the system converges to the origin for an initial state $\boldsymbol{x}(t_0)$, which belongs to the neighbourhood of δ of the equilibrium point.

Traditionally energy functions were used to examine the stability of an equilibrium point. However, in 1892 Aleksandr Mikhailovich Lyapunov (1857-1918) introduced a method to analyse the stability of equilibrium points presented in (2-4) by using alternate functions, commonly known as Lyapunov functions. These functions can be considered to be a generalization of energy functions, which can be easier to obtain in cases where energy functions are complex. Lyapunov's approach is based on the assumption that a system is (asymptotically) stable at the origin, and such a function can be found which has a minimum at the origin, which is (strictly) decreasing along trajectories of the system.

Theorem 2.1 Let $\boldsymbol{x} = \boldsymbol{0}$ be an equilibrium point for (2-6) and $D \subset \mathbb{R}^n$ be a domain containing $\boldsymbol{x} = \boldsymbol{0}$. Let $V: D \rightarrow \mathbb{R}^n$ be a continuously differentiable function, such:

$$V(\boldsymbol{0}) = 0 \text{ and } V(\boldsymbol{x}) > 0 \text{ in } D - \{0\}, \tag{2-10}$$

$$\dot{V}(\boldsymbol{x}) \leq 0 \text{ in } \boldsymbol{D}. \tag{2-11}$$

Then $\boldsymbol{x} = \boldsymbol{0}$ *is stable. Moreover, if:*

$$\dot{V}(\boldsymbol{x}) < 0 \text{ in } \boldsymbol{D} - \{\boldsymbol{0}\}, \tag{2-12}$$

then $\boldsymbol{x} = \boldsymbol{0}$ *is asymptotically stable.*

Proof see Khalil page 14 [27]

The continuously differentiable function $V(\boldsymbol{x})$ is called the *Lyapunov function of the system*. If the function $V(x)$ satisfies condition (2-10) then the system has a minimum at the equilibrium point $\boldsymbol{x} = \boldsymbol{0}$. Furthermore, it must conform to condition (2-11), implying that $\boldsymbol{x} = \boldsymbol{0}$ is the only minimum of $V(\boldsymbol{x})$ in the domain $\boldsymbol{D}$. The surface $V(\boldsymbol{x}) = c$ for some $c > 0$, is called a *Lyapunov surface* or a *levelled surface.* If $V(\boldsymbol{x}) = c$ and $\dot{V}(\boldsymbol{x}) \leq 0$ in the domain D, the function $V(\boldsymbol{x})$ can only stay within the current Lyapunov surface and can never come out again. That is the function remains constant or it moves towards its minimum and never away. If the function $V(\boldsymbol{x})$ satisfies these two conditions in the domain D, then the equilibrium point of the system is stable. Furthermore, if a function $V(\boldsymbol{x})$ satisfies condition (2-12) then the equilibrium point is asymptotically stable as the function $V(\boldsymbol{x})$ continuously moves towards the origin through the *Lyapunov surfaces* and never remains on a given surface for a time period until it reaches the origin.

Conditions (2-10) - (2-12) are sufficient to prove *local asymptotical* stability of the origin in the sense of Lyapunov. To prove *global asymptotic* stability, the *Lyapunov function* $V(\boldsymbol{x})$

needs to be *radially unbounded*. That is $\|\boldsymbol{x}\| \to \infty \Rightarrow V(\boldsymbol{x}) \to \infty$, this can be summarised in the following theorem:

Theorem 2.2 let $\boldsymbol{x} = \boldsymbol{0}$ be an equilibrium point of (2-6) and let $V: \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable function such that:

$$V(0) = 0 \text{ and } V(\boldsymbol{x}) > 0, \forall\, \boldsymbol{x} \neq 0, \quad \text{(2-13)}$$

$$\|\boldsymbol{x}\| \to \infty \Rightarrow V(\boldsymbol{x}) \to \infty, \quad \text{(2-14)}$$

$$\dot{V}(\boldsymbol{x}) < 0, \forall\, \boldsymbol{x} \neq 0, \quad . \quad \text{(2-15)}$$

then $\boldsymbol{x} = \boldsymbol{0}$ *is globally asymptotically stable*

Proof see Khalil page 124 [27]

In conjunction with conditions (2-10) - (2-12) holding the *Lyapunov function* $V(\boldsymbol{x})$ must have a limit at infinity as $\boldsymbol{x} \to \infty$ to prove that the function $V(\boldsymbol{x})$ is radially unbound.

Furthermore, proof of exponential stability requires finding a boundary for $\|\boldsymbol{x}(t)\|$ that moves exponentially to the origin.

Theorem 2.3 Let $\boldsymbol{x} = \boldsymbol{0}$ be an equilibrium point for (2-6) furthermore and $\boldsymbol{D} \subset \mathbb{R}^n$ be a domain containing $\boldsymbol{x} = 0$. Let $V: [0, \infty)\, D \to \mathbb{R}$ be a continuously differentiable function such that:

$$k_1\|\boldsymbol{x}\|^c \leq V(\boldsymbol{x}) \leq k_2\|\boldsymbol{x}\|^c, \quad \text{(2-16)}$$

$$\frac{\partial V}{\partial t} + \frac{\partial V}{\partial \boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}) \leq -k_3\|\boldsymbol{x}\|^c, \quad \text{(2-17)}$$

for some positive constants k1, k2, k3 and c. Then, $\boldsymbol{x} = \boldsymbol{0}$ *is exponentially stable. Moreover, if the assumptions hold globally, then* $\boldsymbol{x} = \boldsymbol{0}$ *is globally exponentially stable.*

Proof see Khalil page 154 [27].

2.3 Stochastic Systems

Unlike the deterministic differential equations, in general, it is not possible to give explicit expressions for the solution to the SDE and numerical solution is a cumbersome affair. It is therefore of great interest to be able to characterise at least qualitatively the behaviour of the solutions.

- Is there a unique solution and is it defined for all times?
- Is the solution bounded in a suitable stochastic sense?
- Is the solution stable in a suitable stochastic sense?
- Does the solution approach a stationary (perhaps even periodic) process? In this case, what properties does this limiting process have?

In this sub section, we are interested in these qualitative properties than in the exact form of the solutions, e.g. stability analysis of control systems. As previously mentioned Lyapunov in 1892 described a method for determining stability without solving the state equation. We will extend the theories presented in the previous subsection to that of a stochastic system:

$$d\boldsymbol{x}(t) = \boldsymbol{f}(x(t), t)dt + \mathbf{G}(\boldsymbol{x}, t)\Delta(t)d\boldsymbol{w}, \qquad \forall t \geq t_0, \tag{2-18}$$

where the state$x$ belongs to Euclidean space $\boldsymbol{x} = \mathbb{R}^n$, $\boldsymbol{w}$ is an r-dimensional independent standard Wiener process, $\Delta(t): \mathbb{R}_+ \to \mathbb{R}^{r\times r}$ is Borel measurable, bounded and nonnegative definite for each $t \in \mathbb{R}_+$. The functions $\boldsymbol{f}: \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}^n$ and $\mathbf{G}: \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}^{n\times r}$ are locally Lipschitz continuous in $\boldsymbol{x}$, uniformly in $t \in \mathbb{R}_+$, and locally bounded.

2.3.1 Ito's Formula and Infinite Generator

Consider the non-linear stochastic system (2-18) and let $\boldsymbol{y}: \mathbb{R}^n \to \mathbb{R}$ be a C^2 function of $\boldsymbol{x}(t)$. Then the stochastic differentiation (Ito's formula), [28], of $y\big(\boldsymbol{x}(t)\big)$ is given by:

$$dy\big(\boldsymbol{x}(t)\big) \leq \left[\frac{\partial y}{\partial x}\boldsymbol{f}(\boldsymbol{x}, t) + \frac{1}{2}\mathrm{Tr}\left\{\Delta^{\mathrm{T}}(t)\mathbf{G}^{\mathrm{T}}(\boldsymbol{x}, t)\frac{\partial^2 y}{\partial \boldsymbol{x}^2}\mathbf{G}(\boldsymbol{x}, t)\Delta(t)\right\}\right]dt + \frac{\partial y}{\partial x}\mathbf{G}(\boldsymbol{x}, t)\Delta(t)d\boldsymbol{w}, \tag{2-19}$$

where $\mathrm{Tr}(\bullet)$ dentoes the trace operator of •.

For the nonlinear stochastic system of (2-18), the infinite generator $\mathcal{L}V(\boldsymbol{x})$ of a C^2 function $V(\boldsymbol{x})$ is defined as:

$$\mathcal{L}V\big(\boldsymbol{x}(t)\big) \leq \frac{\partial V}{\partial x}\boldsymbol{f}(\boldsymbol{x}, t) + \frac{1}{2}\mathrm{Tr}\left\{\Delta^{\mathrm{T}}(t)\mathbf{G}^{\mathrm{T}}(\boldsymbol{x}, t)\frac{\partial^2 V}{\partial \boldsymbol{x}^2}\mathbf{G}(\boldsymbol{x}, t)\Delta(t)\right\}. \tag{2-20}$$

2.3.2 Stability in Probability

Throughout this section, it should be known that the initial value $\boldsymbol{x}_0$ is a vector in $\mathbb{R}^n$ and not a random variable. For the justification please refer to Mao, [28].

Definition 2.2 *The origin of* (2-18) *is said to be:*

1. *Stochastically stable or stable in probability if for every pair of $\varepsilon \in (0,1)$ and $r > 0$ there exists a $\delta = \delta(\varepsilon, r, t_0) > 0$ such that*

$$P\{\|\boldsymbol{x}(t; t_0, \boldsymbol{x}_0)\| < r, \qquad \forall t \geq t_0\} \geq 1 - \varepsilon, \tag{2-21}$$

   where $\|x(t_0)\| < \delta$ and $r > 0$ otherwise it is stochastically unstable

2. *Stochastically asymptotically stable if it is stochastically stable and for every $\varepsilon \in (0,1)$ there exists a $\delta_0 = \delta_0(\varepsilon, t_0) > 0$ such that:*

$$P\left\{\lim_{t\to\infty} \boldsymbol{x}(t; t_0, \boldsymbol{x}_0) = 0\right\} \geq 1 - \varepsilon, \tag{2-22}$$

   whenever $\|x_0\| < \delta_0$.

3. *Stochastically asymptotically stable in the large if the conditions for stochastically asymptotically stable is met and for all $x_0 \in \mathbb{R}^n$:*

$$P\left\{\lim_{t\to\infty} \boldsymbol{x}(t; t_0, \boldsymbol{x}_0) = 0\right\} = 1. \tag{2-23}$$

4. *Globally stable in probability if for all $\varepsilon > 0$ there exist a class $\mathcal{K}$ function $\gamma(\cdot)$ such that:*

$$P\{\|\boldsymbol{x}(t)\| < \gamma\|\boldsymbol{x}(t_0)\|\} \geq 1 - \varepsilon, \quad \forall t \geq t_0 \geq 0, \forall \boldsymbol{x}(t_0) \in \mathbb{R}^n \backslash 0\}. \tag{2-24}$$

5. *Globally asymptotically stable in probability if for all $\varepsilon > 0$ there exist a class $\mathcal{KL}$ function $\beta(\cdot)$ such that:*

$$P\{\|\boldsymbol{x}(t)\| < \beta(\boldsymbol{x}(t_0)), t - t_0\} \geq 1 - \varepsilon, \quad \forall t \geq t_0 \geq 0, \forall \boldsymbol{x}(t_0) \in \mathbb{R}^n \backslash 0\}. \tag{2-25}$$

Proof. See Mao page 110, [28].

The following theorem extends Lyapunov theory presented in Theorem 2.1 to the stochastic case.

Theorem 2.4 If there exists a positive definite function $V(\boldsymbol{x}, t) \in C^{2,1}(S_h \times [t_0, \infty))$

$$V(0, t) = 0 \text{ and } V(\boldsymbol{x}, t) > 0 \text{ in } \boldsymbol{D} - \{\boldsymbol{0}\}, \tag{2-26}$$

$$\mathcal{L}V(\boldsymbol{x}, t) \leq 0, \qquad \forall (\boldsymbol{x}, t) \in \mathbf{S}_h \times [t_0, \infty). \tag{2-27}$$

The origin of (2-18) *is stochastically stable furthermore if:*

$$\mathcal{L}V(\boldsymbol{x}, t) < 0 \text{ in } \boldsymbol{D} - \{\boldsymbol{0}\}, \tag{2-28}$$

then $\boldsymbol{x} = \boldsymbol{0}$ the origin of (2-18) is *stochastically asymptotically stable.*

That is *if there exists a positive definite function* $V(\boldsymbol{x},t) \in C^{2,1}(S_h \times [t_0, \infty))$ *the origin of* (2-18) is *stochastically stable. More over if* there exists a positive definite decrescent function $V(x,t) \in C^{2,1}(S_h \times [t_0, \infty))$ *the origin of* (2-18) *is stochastically asymptotically stable.*

Proof. See Mao [28].

Theorem 2.5 let $\boldsymbol{x} = \boldsymbol{0}$ be an equilibrium point of (2-18) and let $V: \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable function such that:

$$V(0,t) = 0 \textit{ and } V(\boldsymbol{x},t) > 0, \forall\, \boldsymbol{x} \neq \boldsymbol{0}, \tag{2-29}$$

$$\|\boldsymbol{x}\| \to \infty \Rightarrow V(\boldsymbol{x},t) \to \infty, \tag{2-30}$$

$$\mathcal{L}V(\boldsymbol{x},t) < 0, \forall\, \boldsymbol{x} \neq \boldsymbol{0}, \tag{2-31}$$

then $\boldsymbol{x} = \boldsymbol{0}$ *the origin of* (2-18) *is stochastically asymptotically stable in the large.*

Proof. See Mao [28].

2.3.3 Global Stability

In this section, the conditions for global stability in terms of Lyapunov are presented.

Theorem 2.6 Suppose there exists a C^2 function $V(\boldsymbol{x}) \in C^{2,1}(\mathbb{R}^d)\times: \mathbb{R}^n \to \mathbb{R}_+$, a constant c, class $\mathcal{K}_\infty$, functions γ_1, γ_2 and a Borel measurable and increasing function ϱ such that:

$$\gamma_1(\|\boldsymbol{x}\|) \leq V(\boldsymbol{x}) \leq \gamma_2(\|\boldsymbol{x}\|), \tag{2-32}$$

$$\mathcal{L}V(\boldsymbol{x}) \leq \frac{\partial V(\boldsymbol{x})}{\partial x} \boldsymbol{f}(\boldsymbol{x},t) + \frac{1}{2}\mathrm{Tr}\left\{\Delta^T(t)\mathbf{G}^T(\boldsymbol{x},t)\frac{\partial^2 V(\boldsymbol{x})}{\partial \boldsymbol{x}^2}\mathbf{G}(\boldsymbol{x},t)\Delta(t)\right\} \leq -W(\boldsymbol{x}), \tag{2-33}$$

where $W(x): \mathbb{R}^n \to \mathbb{R}_+$, is continuous and nonnegative. Then there exists a unique solution of the system and the equilibrium $\boldsymbol{x} \equiv \boldsymbol{0}$ is globally stable in probability and satisfies:

$$P\left\{\lim_{t\to\infty} W(\boldsymbol{x}(t)) = 0\right\} = 1, \forall \boldsymbol{x}(t_0) \in \mathbb{R}^n. \tag{2-34}$$

Moreover, if $W(\boldsymbol{x})$ *is a class* $\mathcal{K}$, function then the equilibrium $\boldsymbol{x} \equiv \boldsymbol{0}$ is globally asymptotically stable in probability and satisfies

Proof. See [29]

Theorem 2.7 Suppose there exists a C^2 function $V(\boldsymbol{x}): \mathbb{R}^n \to \mathbb{R}_+$, a constant c, class $\mathcal{K}_\infty$, functions γ_1, γ_2 and a Borel measurable and increasing function ϱ such that:

$$\gamma_1(\|\boldsymbol{x}\|) \leq V(\boldsymbol{x}) \leq \gamma_2(\|\boldsymbol{x}\|), \tag{2-35}$$

$$\begin{aligned}\mathcal{L}V(\boldsymbol{x},t,\Delta(t)) &\leq \frac{\partial V(\boldsymbol{x})}{\partial x} \boldsymbol{f}(\boldsymbol{x},t) + \frac{1}{2}\mathrm{Tr}\left\{\Delta^T(t)\mathbf{G}^T(\boldsymbol{x},t)\frac{\partial^2 V(\boldsymbol{x})}{\partial \boldsymbol{x}^2}\mathbf{G}(\boldsymbol{x},t)\Delta(t)\right\} \\ &\leq -cV(\boldsymbol{x}) + \varrho\|\Delta(t)\Delta^T(t)\|,\end{aligned} \tag{2-36}$$

for all $\boldsymbol{x} \in \mathbb{R}^n, t \geq t_0 \geq 0$, and all nonnegative definite matrices $\Delta(t) \in \mathbb{R}^{r \times r}$. Then there exist a unique strong solution of the system for each $\boldsymbol{x}(t_0) \in \mathbb{R}^n$ and satisifies

$$\mathbb{E}\big[V\big(\boldsymbol{x},(t)\big)\big] \leq \mathrm{e}^{-c(t-t_0)} V\big(\boldsymbol{x}(t_0)\big) + \frac{1}{c} \varrho \left(\sup_{\mathrm{t}_0 \leq \boldsymbol{\tau} \leq \mathrm{t}} \|\Delta(\boldsymbol{\tau}) \Delta^T(\boldsymbol{\tau})\| \right). \tag{2-37}$$

Proof. See Mao [28].

2.4 Selected Control Design Strategies

In the succeeding two subsections, we will present the methodology behind the congenital standard backstepping and the one-steep ahead backstepping methods both of which will be used throughout the course of this paper.

2.4.1 Standard Backstepping

Backstepping is a method for designing a stabilizing controller for a class of nonlinear strict-feedback or pure-feedback systems. To explain the concept an example is provided, with the methodology explained after . We consider the system:

$$\dot{\boldsymbol{x}}_1 = \boldsymbol{f}_1(\boldsymbol{x}_1) + \boldsymbol{g}_1(\boldsymbol{x}_1)\boldsymbol{x}_2 \,, \tag{2-38}$$

$$\dot{\boldsymbol{x}}_2 = \boldsymbol{f}_2(\boldsymbol{x}_1, \boldsymbol{x}_2) + \boldsymbol{g}_2(\boldsymbol{x}_1, \boldsymbol{x}_2)\boldsymbol{u}, \tag{2-39}$$

where $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathbb{R}^n$ are the states and $\boldsymbol{u} \in \mathbb{R}^n$ is the input the functions $\boldsymbol{f}_1$, $\boldsymbol{g}_1 : D \rightarrow \mathbb{R}^n$ are differentiable one in a domain $D \subset \mathbb{R}^n$ that contains $\boldsymbol{x}_1 = \boldsymbol{0}$ and $\boldsymbol{f}_1(\boldsymbol{0}) = \boldsymbol{f}_2(\boldsymbol{0}) = \boldsymbol{0}$. Furthermore, the functions $\boldsymbol{g}_1$ and $\boldsymbol{g}_2$ are square and invertible matricides, and it is assumed that all functions are known. The system can be thought of as a cascade of two subsystems as shown in Figure 2-1 the first component is (2-38) with $\boldsymbol{x}_2$ as the input and (2-39) with $\boldsymbol{u}$ as the input.

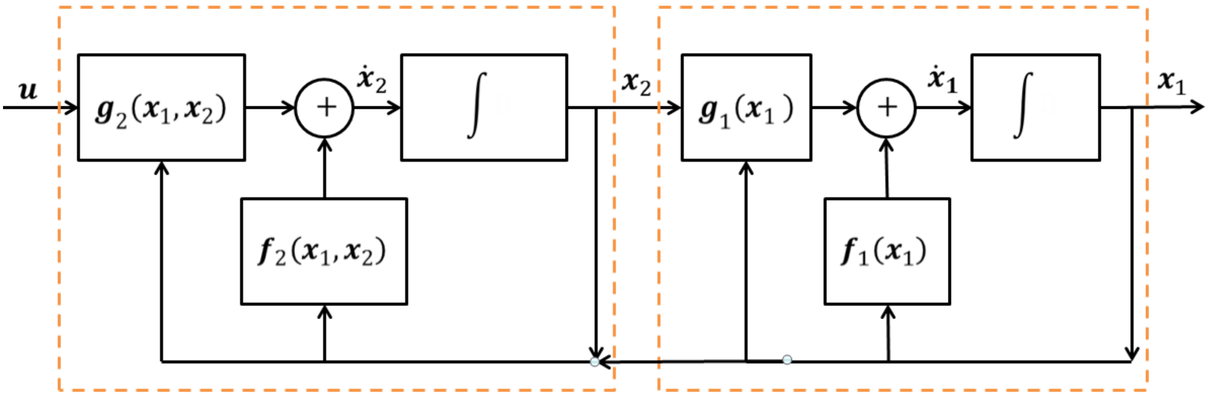

Figure 2-1: Time block diagram of system (2-39) - (2-38).

The method assumes that a stabilizing controller for the first state $\boldsymbol{x}_1$ can be found using the second state $\boldsymbol{x}_1$ as a virtual control, most often through the application Lyapunov methods.

Stabilization of the system (2-38) - (2-39) using the backstepping method consists of two steps. In the first step, we will asymptotically and exponentially stabilize the first subsystem (2-38) at the origin using a virtual control of the second stage as the input. In the second step, we will design the input, such that the second stage tracks the virtual control, which means the tracking error of the second stage will be asymptotically and exponentially stabilized at the origin. The chosen form of the derivatives of the Lyapunov functions will achieve asymptotic and exponential stabilization V_i. The Lyapunov function and derivative we design later are not necessary in general but we will use them, because we need this form for the actual control design later.

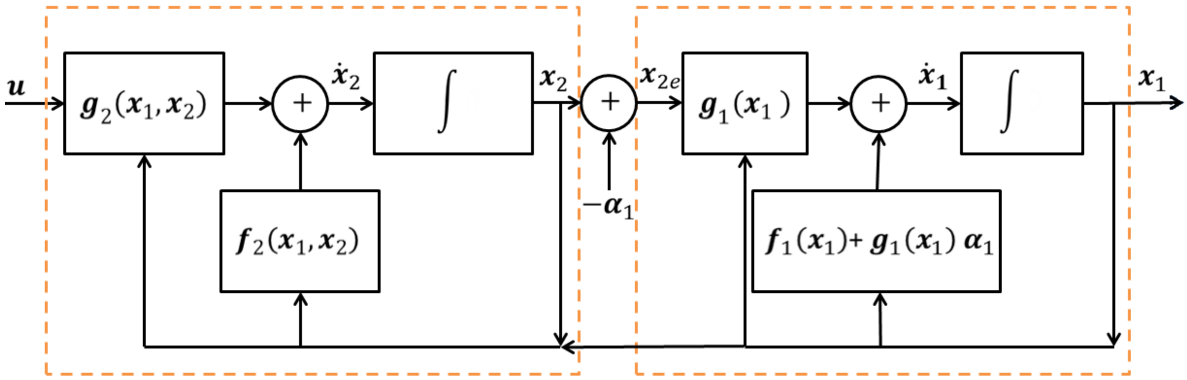

Figure 2-2: Time block diagram of system (2-41) and (2-39) with $f_2 = \mathbf{0}$ and $g_2 = \mathrm{I}_{3\times3}$.

Step 1: To use the virtual control $\boldsymbol{\alpha}_1$ the following coordinate transformation is introduced,

$$\boldsymbol{x}_{2e} = \boldsymbol{x}_2 - \boldsymbol{\alpha}_1. \tag{2-40}$$

Substituting (2-40) into (2-38) yields the transformed system:

$$\dot{\boldsymbol{x}}_1 = \boldsymbol{f}_1(\boldsymbol{x}_1) + \boldsymbol{g}_1(\boldsymbol{x}_1)\boldsymbol{x}_{2e} + \boldsymbol{g}_1(\boldsymbol{x}_1)\boldsymbol{\alpha}_1. \tag{2-41}$$

Figure 2-2 shows the block diagram for the transformed system (2-41) with (2-39). The only difference is the addition of the virtual control $\boldsymbol{\alpha}_1$ inside the first subsystem and subtracted at the input of it.

To analyse the stability of the origin of $\boldsymbol{x}_1$ the Lyapunov function candidate:

$$V_1 = \frac{1}{2}\|\boldsymbol{x}_1\|^2, \tag{2-42}$$

is considered. The function V_1 is positive definite for $\boldsymbol{x}_1 \in \mathbb{R}^n \backslash \{\mathbf{0}\}$ and $V_1(\mathbf{0}) = 0$. The derivative is:

$$\dot{V}_1 = \boldsymbol{x}_1^{\mathrm{T}}(\boldsymbol{f}_1(\boldsymbol{x}_1) + \boldsymbol{g}_1(\boldsymbol{x}_1)\boldsymbol{x}_{2e} + \boldsymbol{g}_1(\boldsymbol{x}_1)\boldsymbol{\alpha}_1). \tag{2-43}$$

By designing $\boldsymbol{\alpha}_1$ the first term of (2-43) can be made negative definite. The second term will be considered in the second step and is ignored for the design of $\boldsymbol{\alpha}_1$. To achieve exponential stability, we need to design the virtual control such that $\dot{V}_1 < 0$ for $\boldsymbol{x}_1 \in D\backslash\{\mathbf{0}\}$ Therefore we choose

$$\boldsymbol{\alpha}_1 = \boldsymbol{g}_1^{-1}(\boldsymbol{x}_1)\big(-\mathbf{K}_1\boldsymbol{x}_1 - \boldsymbol{f}_1(\boldsymbol{x}_1)\big), \tag{2-44}$$

where $\mathbf{K}_1$ is positive defininte. This implies:

$$\dot{V}_1 = -\boldsymbol{x}_1^{\mathrm{T}}\mathbf{K}_1\boldsymbol{x}_1 + \boldsymbol{x}_1^{\mathrm{T}}\boldsymbol{g}_1(\boldsymbol{x}_1)\boldsymbol{x}_{2e}. \tag{2-45}$$

Substituting (2-44) back into (2-41) yields:

$$\dot{\boldsymbol{x}}_1 = -\mathbf{K}_1\boldsymbol{x}_1 + \boldsymbol{g}_1(\boldsymbol{x}_1)\boldsymbol{x}_{2e}. \tag{2-46}$$

Step 2: Differentiating both sides of (2-40) along the solution (2-39) yields

$$\dot{\boldsymbol{x}}_{2e} = \boldsymbol{f}_2(\boldsymbol{x}_1, \boldsymbol{x}_2) + \boldsymbol{g}_2(\boldsymbol{x}_1, \boldsymbol{x}_2)\boldsymbol{u} - \dot{\boldsymbol{\alpha}}_1, \quad (2\text{-}47)$$

with

$$\dot{\boldsymbol{\alpha}}_1 = \frac{\delta \boldsymbol{\alpha}_1}{\delta \boldsymbol{x}_1}(-\mathbf{K}_1\boldsymbol{x}_1 + \boldsymbol{g}_1(\boldsymbol{x}_1)\boldsymbol{x}_{2e}), \quad (2\text{-}48)$$

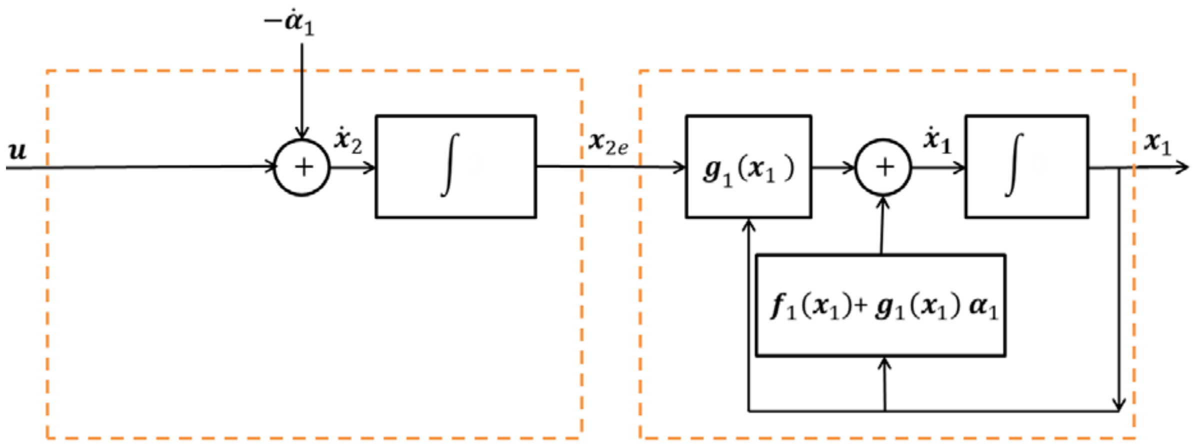

Figure 2-3: Time block diagram of system (2-41) and (2-47) with $f_2 = \mathbf{0}$ and $g_2 = \mathrm{I}_{3\times 3}$.

Figure 2-3 shows a block diagram of the complete transformed system consisting of (2-44) and (2-47). It is visible here, how the virtual control $\boldsymbol{\alpha}_1$ is shifted in front of the integrator. This gives the name to this method, the "backstepping" of $\boldsymbol{\alpha}_1$.

To analyse the stability of the origin of $\boldsymbol{x}_{2e}$ the Lyapunov function candidate

$$V_2 = V_1 + \frac{1}{2}\|\boldsymbol{x}_2\|^2. \quad (2\text{-}49)$$

Taking the derivative of both sides of (2-49) along the solutions of (2-45) and (2-47) is

$$\dot{V}_2 = -\boldsymbol{x}_1^{\mathrm{T}}\mathbf{K}_1\boldsymbol{x}_1 + \boldsymbol{x}_1^{\mathrm{T}}\boldsymbol{g}_1(\boldsymbol{x}_1)\boldsymbol{x}_{2e} + \boldsymbol{x}_{2e}^{\mathrm{T}}(\boldsymbol{f}_2(\boldsymbol{x}_1, \boldsymbol{x}_2) + \boldsymbol{g}_2(\boldsymbol{x}_1, \boldsymbol{x}_2)\boldsymbol{u} - \dot{\boldsymbol{\alpha}}_1), \quad (2\text{-}50)$$

we choose:

$$\boldsymbol{g}_1^{\mathrm{T}}(\boldsymbol{x}_1)\boldsymbol{x}_1 + \boldsymbol{f}_2(\boldsymbol{x}_1, \boldsymbol{x}_2) + \boldsymbol{g}_2(\boldsymbol{x}_1, \boldsymbol{x}_2)\boldsymbol{u} - \dot{\boldsymbol{\alpha}}_1 = -\mathbf{K}_2\boldsymbol{x}_{2e}, \quad (2\text{-}51)$$

to achieve exponential stability for the closed loop system we design the input as

$$\boldsymbol{u} = \boldsymbol{g}_2^{-1}(\boldsymbol{x}_1, \boldsymbol{x}_2)\left(-\mathbf{K}_2\boldsymbol{x}_{2e} - \boldsymbol{g}_1^{\mathrm{T}}(\boldsymbol{x}_1)\boldsymbol{x}_1 - \boldsymbol{f}_2(\boldsymbol{x}_1, \boldsymbol{x}_2) + \dot{\boldsymbol{\alpha}}_1\right), \quad (2\text{-}52)$$

where $\mathbf{K}_1$ is positive defininte. This implies:

$$\dot{\boldsymbol{x}}_{2e} = -\mathbf{K}_2\boldsymbol{x}_{2e} - \boldsymbol{g}_1^{\mathrm{T}}(\boldsymbol{x}_1)\boldsymbol{x}_1. \quad (2\text{-}53)$$

This results in the closed loop system consisting of (2-46) and (2-53) obeys:

$$\begin{bmatrix}\dot{\boldsymbol{x}}_1\\ \dot{\boldsymbol{x}}_{2e}\end{bmatrix} = \left(\begin{bmatrix}-\mathbf{K}_1 & \mathbf{0}\\ \mathbf{0} & -\mathbf{K}_2\end{bmatrix} + \begin{bmatrix}\mathbf{0} & \boldsymbol{g}_1(\boldsymbol{x}_1)\\ -\boldsymbol{g}_1^{\mathrm{T}}(\boldsymbol{x}_1) & \mathbf{0}\end{bmatrix}\right)\begin{bmatrix}\boldsymbol{x}_1\\ \boldsymbol{x}_{2e}\end{bmatrix}, \tag{2-54}$$

hence the derivative of (2-49) is:

$$\dot{V}_2 = -\boldsymbol{x}_1^{\mathrm{T}}\mathbf{K}_1\boldsymbol{x}_1 - \boldsymbol{x}_{2e}^{\mathrm{T}}\mathbf{K}_2\boldsymbol{x}_{2e}. \tag{2-55}$$

Refereeing back to (2-49) it is clear that:

$$V_2(0) = 0 \textit{ and } V_2(x) > 0 \text{ in } D - \{0\}, \tag{2-56}$$

in addition, from (2-49) it can be said that:

$$V_2(\boldsymbol{x}) < 0 \text{ in } D - \{0\}. \tag{2-57}$$

Inequalities (2-56) and (2-57) satisfy conditions (2-13) and (2-15) of Theorem 2.2. It can therefore be stated that the closed loop system (2-54) is asymptotically stable. Foreshore for this system it is possible to prove exponential stability, refereeing back to (2-49) it can be stated

$$\left(\frac{1}{2} - \delta\right)\|(\boldsymbol{x}_1, \boldsymbol{x}_{2e})\|^2 \leq V_2 \leq \left(\frac{1}{2} + \delta\right)\|(\boldsymbol{x}_1, \boldsymbol{x}_{2e})\|^2, \tag{2-58}$$

with $\delta > 0$ and $\|\boldsymbol{x}_1\|^2 + \|\boldsymbol{x}_{2e}\|^2 = \|(\boldsymbol{x}_1, \boldsymbol{x}_{2e})\|^2$. From (2-55) it can be stated:

$$\dot{V}_2 \leq -\min\big(\lambda_m(\mathbf{K}_1), \lambda_m(\mathbf{K}_2)\big)\|(\boldsymbol{x}_1, \boldsymbol{x}_{2e})\|^2. \tag{2-59}$$

The inequalities (2-57) and (2-58) fulfil the conditions (2-16) and (2-17) of Theorem 2.3 where $a = 2$, $k_1 = \left(\frac{1}{2} - \delta\right)$, $k_2 = \left(\frac{1}{2} + \delta\right)$ and $k_3 = \min\big(\lambda_m(\mathbf{K}_1), \lambda_m(\mathbf{K}_2)\big)$. Therefore, we can state that the closed loop system (2-54) is exponentially stable.

2.4.2 One-step Ahead Backstepping

The one-step ahead backstepping methods first presented by Do in [30] and is summarised here.

Consider the following second order system

$$\dot{\boldsymbol{x}}_1 = \boldsymbol{f}_1(t, \boldsymbol{x}_1, \boldsymbol{x}_2) + \boldsymbol{g}_1(\boldsymbol{x}_1)\boldsymbol{x}_2\,, \tag{2-60}$$

$$\dot{\boldsymbol{x}}_2 = \boldsymbol{f}_2(t, \boldsymbol{x}_1, \boldsymbol{x}_2) + \boldsymbol{g}_2(\boldsymbol{x}_1, \boldsymbol{x}_2)\boldsymbol{u}, \tag{2-61}$$

where t denotes the time, $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are the states, $\boldsymbol{u}$ is the control input we assume that

$$|f_1(\cdot)| \leq \varrho_{11}, \left|\frac{\partial f_1(\cdot)}{\partial t}\right| \leq \varrho_{12}, \left|\frac{\partial f_1(\cdot)}{\partial \boldsymbol{x}_1}\right| \leq \varrho_{13}, \left|\frac{\partial f_1(\cdot)}{\partial \boldsymbol{x}_1}\boldsymbol{x}_2\right| \leq \varrho_{14}, \left|\frac{\partial f_1(\cdot)}{\partial \boldsymbol{x}_2}\right| \leq \varrho_{15}, |f_2(\cdot)| \leq \varrho_{21}, \tag{2-62}$$

for all $t \in \mathbb{R}^+, \boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathbb{R}^n$ and $\cdot$ denotes $(t, \boldsymbol{x}_1, \boldsymbol{x}_2)$ and $\varrho_{1i}, i = 1 \ldots 5$ and ϱ_{21} are nonegative constants with ϱ_{15} is strictly less than 1. The system can be thought of as a cascade of two subsystems as shown in Figure 2-4 the first component is (2-60) with $\boldsymbol{x}_2$ as the input and (2-61) with $\boldsymbol{u}$ as the input.

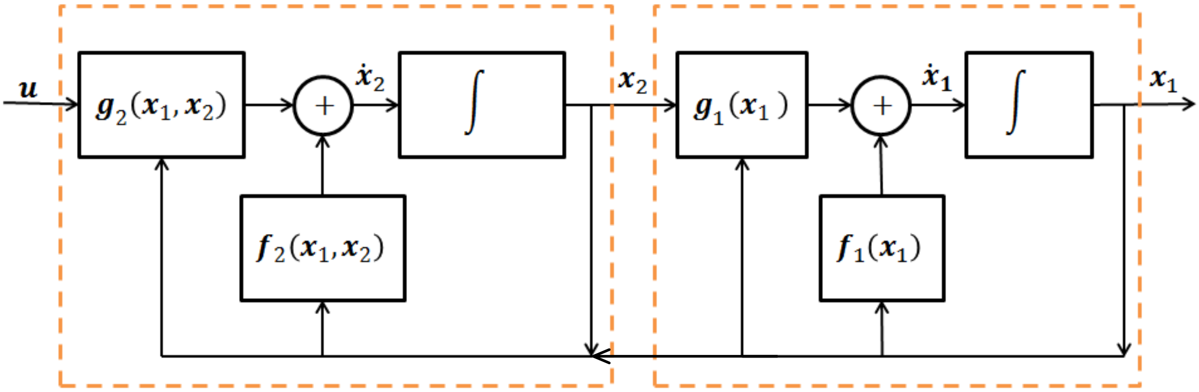

Figure 2-4: Time block diagram of system (2-60) -(2-61).

To make a bounded control $\boldsymbol{u}$ to globally asymptotically and locally exponentially stabilize (2-60) and (2-61). At the origin we must impose conditions (2-62) on $\boldsymbol{f}_1(t,\boldsymbol{x}_1,\boldsymbol{x}_2)$ and $\boldsymbol{f}_2(t,\boldsymbol{x}_1,\boldsymbol{x}_2)$. If $\boldsymbol{f}_2(t,\boldsymbol{x}_1,\boldsymbol{x}_2)$ is not bounded by a constant, it is not possible to design a bounded control input $\boldsymbol{u}$ to globally assynmptotically and locally exponentially stabilize (2-60) and (2-61) at the origin because $\boldsymbol{u}$ needs to cancel $\boldsymbol{f}_2(t,\boldsymbol{x}_1,\boldsymbol{x}_2)$. In this section, a smooth saturation function and a one-step ahead backstepping method is presented, which will be used in the control design and stability analysis in chapter 4.

Definition 2.3 The function $\sigma(x)$ is said to be a smooth saturation function if it possesses the following properties:

$$\sigma(0) = 0, \qquad \sigma(x)x > 0, \qquad \forall x \in \{\mathbb{R} - 0\}, \tag{2-63}$$
$$(x-y)[\sigma(x) - \sigma(y)] > 0, \qquad \forall (x,y) \in \mathbb{R}^2, \tag{2-64}$$
$$\sigma(-x) = -\sigma(x), \qquad |\sigma(x)| < 1, \qquad \frac{|\sigma(x)|}{x} < 1, \qquad \frac{\partial\sigma(x)}{\partial x} \leq 1 > 0, \qquad \forall x \in \mathbb{R}, \tag{2-65}$$
$$\frac{\partial\sigma(x)}{\partial x} > 0, \qquad \forall x \in (-\infty, \infty). \tag{2-66}$$

Such a function satisfying the above properties is $\sigma(x) = \frac{x}{\sqrt{1+x^2}}$, this is the saturation function that will be used throughout thief remainder of this thesis. For the vector $\boldsymbol{x} = [x_1, x_2, x_3]^{\mathrm{T}}$, the notation $\sigma(\boldsymbol{x}) = [\sigma(x_1), \sigma(x_2), \sigma(x_3)]^{\mathrm{T}}$ will be used to denote the smooth saturation function of the vector $\boldsymbol{x}$.

The one-step ahead backstepping control design method consists of two steps.

Step 1: Define

$$\boldsymbol{x}_{2e} = \boldsymbol{x}_2 - \boldsymbol{\alpha}_1, \tag{2-67}$$

where $\boldsymbol{\alpha}_1$ is the virtual control of $\boldsymbol{x}_2$. Let us consider the Lyapunov function candidate

$$V_1 = \int_0^{x_{1e}} \sigma(s)\, ds, \tag{2-68}$$

where $\sigma(s)$ is a smooth saturation function Definition 2.3. By differentiating both sides of (2-68) the virtual control signal is chosen as:

$$\dot{V}_1 = \sigma^{\mathrm{T}}(\boldsymbol{x}_{1e})(\boldsymbol{f}_1(t, \boldsymbol{x}_1, \boldsymbol{x}_2) + \boldsymbol{x}_{2e} + \boldsymbol{\alpha}_1), \tag{2-69}$$

$$\boldsymbol{\alpha}_1 = -\mathbf{K}_1 \frac{\sigma(\boldsymbol{x}_{1e})}{\Delta_1(\boldsymbol{x}_2)} + \boldsymbol{f}_1(t, \boldsymbol{x}_1, \boldsymbol{x}_2), \tag{2-70}$$

where $\mathbf{K}_1$ is a possiative deffiinete matrix and $\lambda_{Max}(\mathbf{K}_1) + \varrho_{15} < 1$, the function $\Delta_1(\boldsymbol{x}_2)$ is chosen such that $\Delta_1(\boldsymbol{x}_2) = 1 + \frac{1}{2}\boldsymbol{x}_2^{\mathrm{T}}\boldsymbol{x}_2$. By substituting (2-70) and (2-67) into (2-69) results in:

$$\dot{V}_1 = -\frac{\sigma^{\mathrm{T}}(\boldsymbol{x}_{1e})\mathbf{K}_1\sigma(\boldsymbol{x}_{1e})}{\Delta_1(\boldsymbol{x}_2)} + \sigma^{\mathrm{T}}(\boldsymbol{x}_{1e})\boldsymbol{x}_{2e} \tag{2-71}$$

Step 2: Consider the Lyapunov function:

$$V_2 = \gamma_1 V_1 + \tfrac{1}{2}\|\boldsymbol{x}_{2e}\|^2, \tag{2-72}$$

where γ_1 is a positive constant. By differentiating both sides of (2-72) and selecting the control input $\boldsymbol{u}$ as:

$$\mathbf{G}_1 = \left(\mathbf{I}_{3\times3} - \frac{\partial \boldsymbol{\alpha}_1}{\partial \boldsymbol{x}_2}\right), \tag{2-73}$$

$$\boldsymbol{u} = \mathbf{G}_1^{\mathrm{T}}\left(-\mathbf{K}_2\sigma(\boldsymbol{x}_{2e}) - \gamma_1\sigma(\boldsymbol{x}_{1e}) + \frac{\partial \boldsymbol{\alpha}_1}{\partial t} + \frac{\partial \boldsymbol{\alpha}_1}{\partial \boldsymbol{x}_1}(\boldsymbol{f}_1(t, \boldsymbol{x}_1, \boldsymbol{x}_2) + \boldsymbol{x}_2)\right) - \boldsymbol{f}_2(t, \boldsymbol{x}_1, \boldsymbol{x}_2), \tag{2-74}$$

results in:

$$\dot{V}_2 = -\frac{\sigma^{\mathrm{T}}(\boldsymbol{x}_{1e})\mathbf{K}_1\sigma(\boldsymbol{x}_{1e})}{\Delta_1(\boldsymbol{x}_2)} - \boldsymbol{x}_{2e}^{\mathrm{T}}\mathbf{K}_2\sigma(\boldsymbol{x}_{2e}). \tag{2-75}$$

Since

$$\left\|\frac{\partial \boldsymbol{\alpha}_1}{\partial \boldsymbol{x}_2}\right\| \le \lambda_{Max}(\mathbf{K}_1) + \varrho_{15}, \qquad \forall \boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathbb{R}^n, \qquad \lambda_{Max}(\mathbf{K}_1) + \varrho_{15} < 1, \tag{2-76}$$

the control input $\boldsymbol{u}$ given in(2-74) is well defined (2-68), (2-72) and (2-75) it is easily shown that the closed loop system consisting of (2-60), (2-61) and (2-74) is forward complete and is globally asymptotically and locally exponentially stable at the origin. From (2-74), a calculation shows that

$$\|\boldsymbol{u}\| = \frac{\lambda_M(\mathbf{K}_2) + \gamma_1 + \varrho_{12} + \lambda_M(\mathbf{K}_1) + \varrho_{14} + (\lambda_M(\mathbf{K}_1) + \varrho_{13})\varrho_{11}}{1 - \lambda_M(\mathbf{K}_1) - \varrho_{15}} + \varrho_{21}, \qquad \forall t \ge t_0 \ge 0. \tag{2-77}$$

The above bound means that the magnitude of the control input $\boldsymbol{u}$ is bounded by a positive constant for all initial conditions $\boldsymbol{x}_1(t_0) \in \mathbb{R}^n$ and $\boldsymbol{x}_2(t_0) \in \mathbb{R}^n$.

Remark 2.1

The main difference between the above control design and the standard backstepping method section 2.4.1 is that the virtual control $\boldsymbol{\alpha}_1$ in (2-70) is a function of both $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$. This is crucial to allow us to design the bounded control input $\boldsymbol{u}$ in (2-74), see the term $\frac{\partial \alpha_1}{\partial x_1}\boldsymbol{x}_2$.

2.5 Basic Quadrotor Motion

The quadrotor is a special form of a helicopter it has a rigid body with six degrees of freedom (6-DOF) that can move in all three dimensions of space. Let $\boldsymbol{O}_E = [x_E \; y_E \; z_E]$ be the inertial east bound frame (E frame) and $\boldsymbol{O}_B = [x_B \; y_B \; z_B]$ be the body bound frame (B frame) of the quadrotor, where the x_B and y_B axis are orthogonal to each other and parallel to the quadrotor frame. Figure 2-5 below shows the orientation of these two frames of reference with respect to each other.

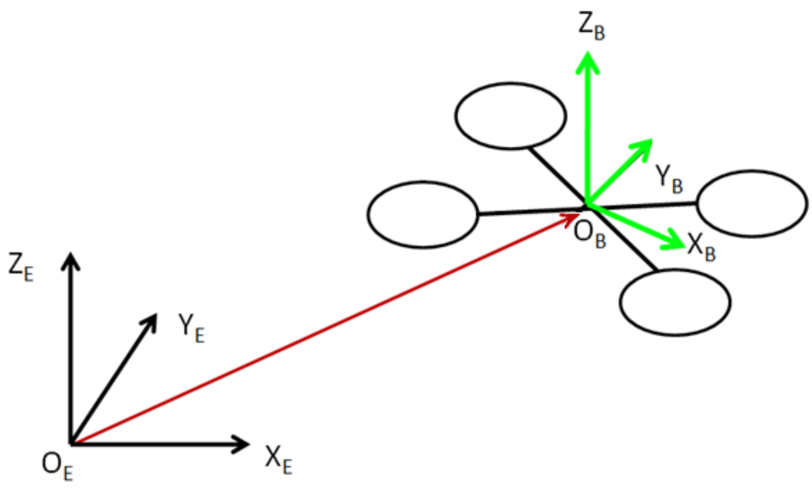

Figure 2-5: Quadrotor frames of reference.

For a non-rotated system, the axis of $\boldsymbol{O}_B$ match the direction of the axis of $\boldsymbol{O}_E$. In this report the quadrotor aircraft configuration will be considered a simple cross structure with 4 equally spaced electric motor powered propeller/rotor-disc units as shown in Figure 2-6.

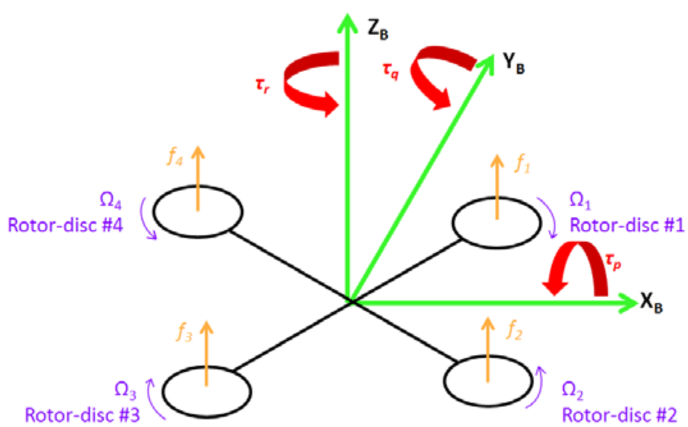

Figure 2-6: Simplified Quadrotor in hovering.

Figure 2-6 above the airframe as well as the motor propeller units of the quadrotor aircraft is shown in black, while the thrust force produced by each motor rotor-disc unit is shown in orange and the direction of rotation and speed of each rotor-disc is shown in purple and the torque about each axis is shown in red. Resultant thrust and torque are achieved by each rotor-disc rotating at an angular velocity Ω [rads^{-1}].

In the figure above the odd numbered rotor-discs #1 and #3 rotate in a clockwise manner when looking straight down the normal Z_B axis while even numbered rotor-discs #2 and #4 rotate in a counter clockwise direction. When compared to a fixed wing aircraft with rudder, elevator, and aileron control surfaces or to a conventional helicopter with cyclic control of main rotor-disc along with pitch control of main and tail rotor blades, the quadrotor aircraft has no obvious adjustable aerodynamic surfaces that can be used to control the aircraft. Also unlike fixed wing aircraft the quadrotor airframe has no aerodynamic shape to provide static or dynamic stability to return it to its original attitude (e.g. angle of attack) when acted upon by external forces like wind gusts, wind shear, thermals or vortices from other aircraft. Therefore, from the figure above it can be seen for this aircraft type the only forces available to provide any flight control around the six degrees of freedom with respect to its three local or body axes are;

- longitudinal axis: - forward displacement and rolling motion around the longitudinal axis,
- lateral axis: - side displacement and pitching motion around the lateral axis, and,
- normal axis: - “vertical” displacement and yawing motion around the normal axis

The resultant rotor-disc thrust and torque is achieved by adjusting individual motor and hence rotor-disc speeds. A further constraint is the co-plainer rotor-disc configuration (with all four propellers having parallel axes of rotation) which means there is only one rotor-disc thrust direction (along the normal axis) that can be delivered from any of the four motor rotor-disc units, (see Figure 2-6). The specific motions around the six degrees of freedom are achieved by controlling the rotor-discs in the following manner;

- Aircraft “altitude” Normal or $\mathbf{Z_E}$ axis displacement motion is achieved by increasing/decreasing the thrust from all rotor-discs. At hovering conditions, the sum total of vertical rotor-disc thrust is equal to the weight of the quadrotor.
- <u>Aircraft Yaw</u> motion or rotation around the $\mathbf{Z_B}$ normal axis either clockwise or anticlockwise) is achieved by unbalancing the resultant anticlockwise or clockwise torque around this aircraft body axis by varying the relative rotor speeds and hence reaction torques of the anticlockwise versus clockwise rotating rotor-discs.
- <u>Aircraft Roll</u> (rotation around the longitudinal Y_B axis) say to the left is achieved by increasing the thrust from rotor-discs #2 and #3 while also reducing the thrust from rotor-discs #1 and #4.
- <u>Aircraft Pitch</u> (rotation around the lateral X_B axis) say nose down is achieved by increasing the thrust from rotor-discs #1 and #2 while also reducing the thrust from rotor-discs #3 and #4.
- <u>Aircraft forward air speed </u>or Longitudinal $\mathbf{Y_E}$ axis displacement motion is achieved by first pitching the aircraft forward (rotation around the X_B axis as explained above) and then increasing the overall thrust on all six rotor-discs to reach the desired air speed.
- <u>Aircraft sideways air speed </u>or Lateral $\mathbf{X_E}$ axis displacement motion is achieved by first rolling the aircraft (rotation around the Y_B axis as explained above) toward the direction of travel and then increasing the thrust on all six rotor-discs to reach the desired air speed.

2.6 Equations of Motion

Modelling and analysis of the quadrotor helicopter's 6-DOF (six degrees of freedom) can be done with the application of the Newton-Euler method, which takes into account both the kinematics and kinetics of the aircraft, we will then present the model using the Modified Rodrigues Parameters.

2.6.1 Newton Euler Model.

The position of the quadrotor's centre of gravity is given by the vector $\boldsymbol{\eta}_1 = [x\ y\ z]^{\mathrm{T}}$ expressed in the inertial E-frame. The attitude (orientation) of the quadrotor is given by the vector $\boldsymbol{\eta}_2 = [\phi\ \theta\ \psi]^{\mathrm{T}}$, where ϕ, θ and ψ are Euler angles after the xyz convention and ϕ is the pitch angle, θ the roll angle and ψ the yaw angle of the quadrotor. Figure 2-5 shows the earth fixed frame with the vector $\boldsymbol{\eta}_1$ to the body fixed frame and Figure 2-5 shows the body fixed frame with the pitch angle ϕ, the roll angle θ and the yaw angle ψ. From the pitch, roll and yaw rotation the following rotation matrix $\mathbf{R}_1(\boldsymbol{\eta}_2)$ from B-frame to the E-frame is obtained:

$$\mathbf{R}_1(\boldsymbol{\eta}_2) = \begin{bmatrix} c_\Psi c_\Theta & s_\Psi c_\Phi + c_\Psi s_\Theta s_\Phi & s_\Psi s_\Phi - c_\Psi s_\Theta c_\Phi \\ s_\Psi c_\Theta & c_\Psi c_\Phi + s_\Psi s_\Theta s_\Phi & c_\Psi s_\Phi + s_\Psi s_\Theta c_\Phi \\ -s_\Theta & c_\Theta s_\Phi & c_\Theta c_\Phi \end{bmatrix} \quad \text{(2-78)}$$

Where $s. = \sin(\cdot)$ and $c. = \cos(\cdot)$.

Taking the aerodynamics of the aircraft into consideration, the dynamics of the quadrotor can be described with its velocity, acceleration, angular velocity and angular acceleration by the following formulas:

$$m\dot{\boldsymbol{v}}_1 + mg\boldsymbol{e}_3 = \boldsymbol{f} + \boldsymbol{f}_{Aero}, \quad \text{(2-79)}$$
$$\mathbf{I}_H\dot{\boldsymbol{\omega}} + \boldsymbol{\omega}\times(\mathbf{I}_H\boldsymbol{\omega}) = \boldsymbol{\tau} + \boldsymbol{\tau}_{Aero}, \quad \text{(2-80)}$$

Thereby we use the force vector $\boldsymbol{f} = [f_x\ \ f_y\ \ f_z]^{\mathrm{T}}$ and the torque vector $\boldsymbol{\tau} = [\tau_p\ \ \tau_q\ \ \tau_r]^{\mathrm{T}}$, the aerodynamic disturbance force vector $\boldsymbol{f}_{Aero} = [f_{Aero,x}\ \ f_{Aero,y}\ \ f_{Aero,z}]^{\mathrm{T}}$ and torque vector $\boldsymbol{\tau}_{Aero} = [\boldsymbol{\tau}_{Aero,p}\ \ \boldsymbol{\tau}_{Aero,e}\ \ \boldsymbol{\tau}_{Aero,q}]^{\mathrm{T}}$ the mass m, the inertia matrix $\mathbf{I}_{\mathrm{H}}$, the earth acceleration g and the vector $\boldsymbol{e}_3 = [0\ 0\ 1]^{\mathrm{T}}$. The acceleration $\ddot{\boldsymbol{\eta}}_1$ is expressed in the inertia frame and the angular velocity $\boldsymbol{\omega}$ and its derivative the angular acceleration $\dot{\boldsymbol{\omega}}$ are expressed in the body fixed frame [31]. To include all degrees of freedom and the full dynamics in a state model we define the states position $\boldsymbol{\eta}_1$, velocity $\boldsymbol{v}_1$, attitude $\boldsymbol{\eta}_2$ and angular velocity $\boldsymbol{\omega}$. This results in the following state model:

$$s_1 = \begin{cases} \dot{\boldsymbol{\eta}}_1 = \boldsymbol{v}_1, \\ \dot{\boldsymbol{v}}_1 = -g\boldsymbol{e}_3 + \frac{f}{m}\mathbf{R}(\boldsymbol{\eta}_2)\boldsymbol{e}_3 + \frac{1}{m}\boldsymbol{f}_{Aero}, \end{cases} \quad \text{(2-81)}$$
$$s_2 = \begin{cases} \dot{\boldsymbol{\eta}}_2 = \mathbf{T}(\boldsymbol{\eta}_2)\boldsymbol{\omega}, \\ \dot{\boldsymbol{\omega}} = -\mathbf{I}_H^{-1}\boldsymbol{\omega}\times(\mathbf{I}_H\boldsymbol{\omega}) + \mathbf{I}_H^{-1}\boldsymbol{\tau} + \mathbf{I}_H^{-1}\boldsymbol{\tau}_{Aero}, \end{cases} \quad \text{(2-82)}$$

where $\mathbf{R}_1(\boldsymbol{\eta}_2)e_3$ is the last column of the rotation matrix and therefore the direction of the thrust and $\mathbf{T}(\boldsymbol{\eta}_2)$ is the so called Euler matrix and is given by

$$\mathbf{T}(\boldsymbol{\eta}_2) = \begin{bmatrix} 1 & s_\Phi t_\theta & -s_\theta t_\theta \\ 0 & c_\Phi & -s_\Phi \\ 0 & \frac{s_\Phi}{c_\theta} & \frac{c_\Phi}{c_\theta} \end{bmatrix} \tag{2-83}$$

Listed below is the system inertia matrix:

$$\mathbf{I}_H = \begin{bmatrix} I_X & 0 & 0 \\ 0 & I_Y & 0 \\ 0 & 0 & I_Z \end{bmatrix}. \tag{2-84)}$$

Remark 2.2

1. *The rotation matrix* $\mathbf{T}(\boldsymbol{\eta}_2)$ *has singularies at* $\theta = \pm n\pi/2$ *with* $n = 1,2,3$ ...

2.6.2 Newton Modified Rodrigues Parameter Model

For this stochastic modelling of the quadrotor we will use the modified Rodrigues parameter for angle representation, the reason for this is that this method of angle representation moves the singularity occurring in the Newton Euler representation to $\pm 2\pi$We will use the modified Rodrigues vector $\boldsymbol{q} = col(q_1, q_2, q_3)$ to represent the attitude of the aircraft and thus we will define:

$$\boldsymbol{q}(\boldsymbol{\eta}_2) = \begin{bmatrix} \dfrac{\sin(\frac{\phi}{2})\cos(\frac{\theta}{2})\cos(\frac{\psi}{2}) - \cos(\frac{\phi}{2})\sin(\frac{\theta}{2})\sin(\frac{\psi}{2})}{1 + \cos(\frac{\phi}{2})\cos(\frac{\theta}{2})\cos(\frac{\psi}{2}) + \sin(\frac{\phi}{2})\sin(\frac{\theta}{2})\sin(\frac{\psi}{2})} \\ \dfrac{\cos(\frac{\phi}{2})\sin(\frac{\theta}{2})\cos(\frac{\psi}{2}) + \sin(\frac{\phi}{2})\cos(\frac{\theta}{2})\sin(\frac{\psi}{2})}{1 + \cos(\frac{\phi}{2})\cos(\frac{\theta}{2})\cos(\frac{\psi}{2}) + \sin(\frac{\phi}{2})\sin(\frac{\theta}{2})\sin(\frac{\psi}{2})} \\ \dfrac{\cos(\frac{\phi}{2})\cos(\frac{\theta}{2})\sin(\frac{\psi}{2}) - \sin(\frac{\phi}{2})\sin(\frac{\theta}{2})\cos(\frac{\psi}{2})}{1 + \cos(\frac{\phi}{2})\cos(\frac{\theta}{2})\cos(\frac{\psi}{2}) + \sin(\frac{\phi}{2})\sin(\frac{\theta}{2})\sin(\frac{\psi}{2})} \end{bmatrix}, \tag{2-85}$$

[32]. As mentioned before there are two subsystems too representing the quadrotor by altering the equations of (3-84) and (3-84) to the MRP in place of Euler angles it can be stated.:.

$$s_1 = \begin{cases} \dot{\boldsymbol{\eta}}_1 = \boldsymbol{v}_1, \\ \dot{\boldsymbol{v}}_1 = -\mathbf{D}_1\boldsymbol{v}_1 - g\boldsymbol{e}_3 + \mathbf{R}_1(\boldsymbol{q})\boldsymbol{e}_3 + \frac{1}{m}\bar{\boldsymbol{f}}_{Aero}, \end{cases} \tag{2-86}$$

$$s_2 = \begin{cases} \dot{\boldsymbol{q}} = \mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}, \\ \dot{\boldsymbol{\omega}} = \mathbf{I}_H^{-1}(-\mathbf{D}_2\boldsymbol{\omega} - \boldsymbol{\omega}\times\mathbf{I}_H\boldsymbol{\omega} + \boldsymbol{\tau} + \bar{\boldsymbol{\tau}}_{Aero}). \end{cases} \tag{2-87}$$

where we define

$$\mathbf{R}_1(\boldsymbol{q}) = \mathbf{I}_{3\times3} + \frac{8}{(1 + \|\boldsymbol{q}\|^2)^2}\boldsymbol{S}(\boldsymbol{q})\left(\mathbf{S}(\boldsymbol{q}) - -\frac{1 - \|q\|^2}{2}\mathbf{I}_{3\times3}\right), \tag{2-88}$$

$$\mathbf{R}_2(\boldsymbol{q}) = \frac{1}{2}\left(\mathbf{I}_{3\times3} + \boldsymbol{q}\boldsymbol{q}^{\mathrm{T}} + \mathbf{S}(\boldsymbol{q}) - \frac{1 + \|\boldsymbol{q}\|^2}{2}\mathbf{I}_{3\times3}\right), \tag{2-89}$$

$$\dot{\boldsymbol{q}} = \mathbf{R}_2(\boldsymbol{q})\mathbf{T}(\boldsymbol{\eta}_2)^{-1}\dot{\boldsymbol{\eta}}_2, \tag{2-90}$$

where $\mathbf{I}_{3\times3}$ is a 3×3 identity matrix, $\mathbf{T}(\boldsymbol{\eta}_2)$ was defined in (2-83) and the skew-symmetric matrix $\mathbf{S}(\boldsymbol{x})$ of the vector $\boldsymbol{x} = col(x_1, x_2, x_3) \in \mathbb{R}^3$ is defined as:

$$\mathbf{S}(\boldsymbol{x}) = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix}, \tag{2-91}$$

Note that the matrices $\mathbf{R}_1(\boldsymbol{q})$ and $\mathbf{R}_2(\boldsymbol{q})$ are invertible with properties

$$\mathbf{R}_1^{-1}(\boldsymbol{q}) = \mathbf{R}_1^{\mathrm{T}}(\boldsymbol{q}), \tag{2-92}$$

$$\mathbf{R}_2^{-1}(\boldsymbol{q}) = \frac{16}{(1 + \|\boldsymbol{q}\|^2)^2}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{q}), \tag{2-93}$$

$$\mathbf{R}_1(\boldsymbol{q}) = \mathbf{R}_1(\boldsymbol{\eta}_2). \tag{2-94}$$

The equations of motion for the aircraft in the body fixed frame with no external disturbances acting on the system is defined as follows:

$$\mathbf{M}_b\dot{\varsigma} + \mathbf{D}\varsigma + \mathbf{C}_b(\varsigma)\varsigma + \mathbf{G}_1(q) = \begin{bmatrix} \boldsymbol{f}^b \\ \boldsymbol{\tau} \end{bmatrix}, \tag{2-95}$$

where we define

$$\varsigma = \begin{bmatrix} \boldsymbol{v}^b \\ \boldsymbol{\omega} \end{bmatrix}, \quad \mathbf{M}_b = \begin{bmatrix} m & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{I}_H \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} \mathbf{D}_1 & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{D}_2 \end{bmatrix},$$
$$\mathbf{C}_b(\varsigma)\varsigma = \begin{bmatrix} \mathbf{0}_{3\times3} & \boldsymbol{v}^b \times m\boldsymbol{\omega} \\ \boldsymbol{v}^b \times m\boldsymbol{v}^b & \boldsymbol{\omega} \times \mathbf{I}_H\boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{C}_{B12}(\boldsymbol{\omega})\boldsymbol{v}^b \\ \mathbf{C}_{B21}(\boldsymbol{\omega})\boldsymbol{v}^b & \mathbf{C}_{B2}(\boldsymbol{\omega})\boldsymbol{\omega} \end{bmatrix},$$
$$\mathbf{G}_1(q) = \begin{bmatrix} mg\mathbf{R}_1^{-1}(q)\mathbf{e}_3 \\ \mathbf{0}_{3\times1} \end{bmatrix}. \tag{2-96}$$

and m is the mass of the aircraft, $\mathbf{I}_H$ is the inertia of the aircraft, g is the acceleration due to gravity and the vector $\boldsymbol{e}_3 = [0\ 0\ 1]^{\mathrm{T}}$.

The relative velocity vector

$$\varsigma_r = \varsigma - \varsigma_a \tag{2-97}$$

where:

$$\varsigma_a = \begin{bmatrix} \boldsymbol{v}_a^b \\ \mathbf{0}_{3\times1} \end{bmatrix}, \tag{2-98}$$

is the air current velocity vector. The force and moment vector due to the loading effect of added mass and inertia due to the aircraft moving thought the air and having to push this air out of the way is defined as:

$$\begin{bmatrix} \mathbf{M}_{A1} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{I}_A \end{bmatrix}\begin{bmatrix} \dot{\boldsymbol{v}}_r^b \\ \dot{\boldsymbol{\omega}} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{3\times3} & \boldsymbol{v}_r^b \times \mathbf{M}_{A1}\boldsymbol{\omega} \\ \boldsymbol{v}_r^b \times \mathbf{M}_{A1}\boldsymbol{v}_r^b & \boldsymbol{\omega} \times \mathbf{I}_A\boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}_r^b \\ \boldsymbol{\tau}_r \end{bmatrix}, \tag{2-99}$$

which can be rewritten as:

$$\mathbf{M}_A\dot{\varsigma}_r + \mathbf{C}_A(\varsigma_r)\varsigma_r = \begin{bmatrix} \boldsymbol{f}_r^b \\ \boldsymbol{\tau}_r \end{bmatrix}, \tag{2-100}$$

where:

$$\varsigma_a = \begin{bmatrix} \boldsymbol{v}_a^b \\ \mathbf{0}_{3\times1} \end{bmatrix}, \qquad \mathbf{M}_A = \begin{bmatrix} \mathbf{M}_{A1} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{I}_A \end{bmatrix},$$
$$\mathbf{C}_A(\varsigma_a)\varsigma_a = \begin{bmatrix} \mathbf{0}_{3\times3} & \boldsymbol{v}_a^b \times \mathbf{M}_{A1}\boldsymbol{\omega} \\ \boldsymbol{v}_a^b \times \mathbf{M}_{A1}\boldsymbol{v}_a^b & \boldsymbol{\omega}\times\mathbf{I}_A\boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{C}_{A12}(\boldsymbol{v}^b)\boldsymbol{\omega} \\ \mathbf{C}_{A21}(\boldsymbol{v}^b)\boldsymbol{v}^b & \mathbf{C}_{A22}(\boldsymbol{\omega})\boldsymbol{\omega} \end{bmatrix}. \tag{2-101}$$

is the load due to the air current velocity v_c. The wind force and moment vector $\boldsymbol{f}_{Aero}$, $\boldsymbol{\tau}_{Aero}$ is given by:

$$\begin{bmatrix} \boldsymbol{f}_{Aero} \\ \boldsymbol{\tau}_{Aero} \end{bmatrix} = \mathbf{g}_{\text{wind}}\gamma_{rw}V_{rw}^2, \tag{2-102}$$

where $\mathbf{g}_{\text{wind}}$ is a vector depending on the air density ρ_a, wind coefficients, and frontal and lateral projected areas and V_w is the relative wind speed to the aircraft. Considering the fact that in practice airflow will comprise of both laminar and turbulent flow, which will be consider as deterministic and stochastic respectively we can state:

$$\boldsymbol{v}_a^b = \bar{\boldsymbol{v}}_a^b + \tilde{\boldsymbol{v}}_a^b, \tag{2-103}$$
$$\varsigma_a = \bar{\varsigma}_a + \tilde{\varsigma}_a, \tag{2-104}$$
$$\boldsymbol{f}_{Aero} = \bar{\boldsymbol{f}}_{Aero} + \tilde{\boldsymbol{f}}_{Aero}, \tag{2-105}$$
$$\boldsymbol{\tau}_{Aero} = \boldsymbol{\tau}_{Aero} + \tilde{\boldsymbol{\tau}}_{Aero}. \tag{2-106}$$

Where $\bar{\bullet}$ and $\tilde{\bullet}$ denote the mean-value (deterministic) and zero-mean turbulent (stochastic) components of $\bullet$ respectively. The deterministic components can be treated as unknown constants. The stochastic components are regarded as Gaussian random disturbances. By substituting (2-97) into the expressions for $\mathbf{M}_A\dot{\varsigma}_r$, $\mathbf{D}\varsigma_r$ and $\mathbf{C}_A(\varsigma_r)\varsigma_r$ Results in:

$$\mathbf{M}_A\dot{\varsigma}_r = \mathbf{M}_A\dot{\varsigma} - \mathbf{M}_A\dot{\varsigma}_a, \tag{2-107}$$
$$\mathbf{D}\varsigma_r = \mathbf{D}\varsigma - \mathbf{D}\varsigma_a, \tag{2-108}$$
$$\mathbf{C}_A(\varsigma_r)\varsigma_r = \mathbf{C}_A(\varsigma)\varsigma - (\mathbf{C}_A(\varsigma) + \bar{\mathbf{C}}_A(\varsigma))\varsigma_a + \bar{\mathbf{C}}_A(\varsigma_a)\varsigma_a, \tag{2-109}$$
$$\bar{\mathbf{C}}_A(\varsigma) = \begin{bmatrix} \boldsymbol{S}(\boldsymbol{\omega})\mathbf{M}_{A1} & \mathbf{0}_{3\times3} \\ \boldsymbol{S}(\boldsymbol{v}^b)\mathbf{M}_{A1} & \boldsymbol{S}(\boldsymbol{\omega})\mathbf{I}_A \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{C}}_{A11} & \bar{\mathbf{C}}_{A12} \\ \bar{\mathbf{C}}_{A21} & \bar{\mathbf{C}}_{A22} \end{bmatrix}, \tag{2-110}$$

In derivation of equation (2-109), we have used the property of the skew symmetric matrix, i.e., $\mathbf{S}(\boldsymbol{x})\boldsymbol{y} = -\mathbf{S}(\boldsymbol{y})\boldsymbol{x}$ for all $\boldsymbol{x} =\in \mathbb{R}^3$ and $\boldsymbol{y} =\in \mathbb{R}^3$. Substituting (2-109) into (2-95) yields

$$\dot{\varsigma} = \mathbf{M}_b^{-1}\left(-\mathbf{D}\varsigma - (\mathbf{C}_b(\varsigma) + \mathbf{C}_A(\varsigma))\varsigma - \mathbf{G}_1(\boldsymbol{q}) + \begin{bmatrix} \boldsymbol{f}^b \\ \boldsymbol{\tau} \end{bmatrix}\right)$$
$$+\mathbf{M}_b^{-1}(\mathbf{M}_A\dot{\varsigma}_a + (\mathbf{C}_A(\varsigma) + \bar{\mathbf{C}}_A(\varsigma) + \mathbf{D})\varsigma_a - \bar{\mathbf{C}}_A(\varsigma_a)\varsigma_a). \tag{2-111}$$

If we now alter the above system of equations so that linear position system is represented in the earth inertial frame while leaving the angular position system in the body frame (2-111) becomes.

$$\dot{\boldsymbol{\eta}}_1 = \boldsymbol{\nu}_1,$$
$$\dot{\boldsymbol{\nu}}_1 = -\mathbf{D}_1\boldsymbol{\nu}_1 - g\boldsymbol{e}_3 + \frac{1}{m}f\mathbf{R}_1(\boldsymbol{q})\boldsymbol{e}_3 + \frac{1}{m}\boldsymbol{f}_{Aero}, \tag{2-112}$$
$$\dot{\boldsymbol{q}} = \mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}$$
$$\dot{\boldsymbol{\omega}} = (\mathbf{I}_A + \mathbf{I}_H)^{-1}(-\mathbf{D}_2\boldsymbol{\omega} - \mathbf{C}_{A21}(\boldsymbol{v}^b)\boldsymbol{v}^b - \mathbf{C}_{B22}(\boldsymbol{\omega})\boldsymbol{\omega} - \mathbf{C}_{A22}(\boldsymbol{\omega})\boldsymbol{\omega} + \boldsymbol{\tau} + \bar{\boldsymbol{\tau}}_{Aero}) \tag{2-113}$$

Since

$$\mathbf{C}_{B1}(\boldsymbol{v}^b)\boldsymbol{v}^b = \mathbf{0}_{3\times1}. \tag{2-114}$$

Because $\boldsymbol{v}_a^b = \bar{\boldsymbol{v}}_a^b + \tilde{\boldsymbol{v}}_a^b$ we have:

$$\mathbf{C}_A(\varsigma_r)\varsigma_r = \mathbf{C}_A(\bar{\varsigma}_a)\bar{\varsigma}_a - \left(\mathbf{C}_A(\bar{\varsigma}_a) + \bar{\mathbf{C}}_A(\bar{\varsigma}_a)\right)\bar{\varsigma}_a + \bar{\mathbf{C}}_A(\bar{\varsigma}_a)\bar{\varsigma}_a, \tag{2-115}$$

substituting (2-115) into (2-111) gives:

$$\begin{aligned} d\varsigma = \mathbf{M}_b^{-1}\left(-\mathbf{D}\varsigma - (\mathbf{C}_b(\varsigma) + \mathbf{C}_A(\varsigma))\varsigma - \mathbf{G}_1(\boldsymbol{q}) + \begin{bmatrix}\boldsymbol{f}^b \\ \boldsymbol{\tau}\end{bmatrix}\right) \\ + \mathbf{M}_b^{-1}\left(\mathbf{M}_A\dot{\bar{\varsigma}}_a + (\mathbf{C}_A(\varsigma) + \bar{\mathbf{C}}_A(\varsigma) + \mathbf{D})\bar{\varsigma}_a - \bar{\mathbf{C}}_A(\varsigma_a)\bar{\varsigma}_a\right) \\ + \mathbf{M}_b^{-1}\left(\mathbf{M}_A\dot{\tilde{\varsigma}}_a + (\mathbf{C}_A(\varsigma) + \bar{\mathbf{C}}_A(\varsigma) + \mathbf{D})\bar{\tilde{\varsigma}}_a - \bar{\mathbf{C}}_A(\varsigma_a)\tilde{\varsigma}_a\right) \end{aligned} \tag{2-116}$$

If we now alter the above system of equations o that the linear position system is represented in the earth inertial frame while leaving the angular position system in the body frame (2-116) becomes:

$$\begin{aligned} &\dot{\boldsymbol{\eta}}_1 = \boldsymbol{v}_1, \\ &\qquad d\boldsymbol{v}_1 = \left(-\mathbf{D}_1\boldsymbol{v}_1 - g\boldsymbol{e}_3 + \tfrac{1}{m}f\mathbf{R}_1(\boldsymbol{q})\boldsymbol{e}_3 + \tfrac{1}{m}\bar{\boldsymbol{f}}_{Aero}\right)dt + \tfrac{1}{m}\Delta_1(t)d\boldsymbol{w}_1, \end{aligned} \tag{2-117}$$

$$\begin{aligned} &\dot{\boldsymbol{q}} = \mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega} \\ &d\boldsymbol{\omega} = (\mathbf{I}_A + \mathbf{I}_H)^{-1}\left(-\mathbf{D}_2\boldsymbol{\omega} - \mathbf{C}_{A21}(\boldsymbol{v}^b)\boldsymbol{v}^b - \mathbf{C}_{B22}(\boldsymbol{\omega})\boldsymbol{\omega} - \mathbf{C}_{A222}(\boldsymbol{\omega})\boldsymbol{\omega} + \boldsymbol{\tau} + \boldsymbol{\tau}_{Aero}\right)dt \\ &\quad +(\mathbf{I}_A + \mathbf{I}_H)^{-1}\left(\mathbf{C}_{A21}(\boldsymbol{v}^b) + \mathbf{C}_{A22}(\boldsymbol{\omega}) + \boldsymbol{S}(\boldsymbol{v}^b)\mathbf{M}_A + \boldsymbol{S}(\boldsymbol{\omega})\mathbf{I}_A + \mathbf{D}_2\right)\Delta_2(t)d\boldsymbol{w}_2. \end{aligned} \tag{2-118}$$

where $\boldsymbol{\Delta}_1(t), \boldsymbol{\Delta}_2(t)$ and $\boldsymbol{\Delta}_3(t)$ are (time-varying) covariance matrices, and $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ are three-dimensional vectors of standard Wiener process.

Remark 2.3

1. *The rotation matrix* $\mathbf{R}_2(\boldsymbol{q})$ *has singularities at* $\phi, \theta\psi = \pm n2\pi$ *with* $n = 1,2,3 \ldots$
2. Since this thesis focuses on a trajectory tracking control design for the quadrotor under stochastic wind loads, the system parameters m, $\mathbf{M}_A$, $\mathbf{I}_H$, $\mathbf{I}_H$, $\mathbf{D}_1$, and $\mathbf{D}_2$ are assumed to be known, and the mean value $\bar{\boldsymbol{f}}_{Aero}$ of the aerodynamic loads are assumed to be constant. Including consideration of the unknown system parameters and time-varying mean values of the aerodynamic loads will increase complexity of this thesis. However, this inclusion is possible since the control design proposed in this thesis is based on the Lyapunov direct method. Treatment of unknown $\mathbf{D}_1$, and $\mathbf{D}_2$ is given in [13] while time-varying mean values of the wind loads are considered in [33] for quadrotor aircraft under deterministic wind loads.
3. *If* $\frac{dw}{dt}$ *is bounded, the aircraft system* (2-117) *and* (2-118) *becomes deterministic. There exist many control robust adaptive control design techniques for this type of deterministic systems for both known and unknown* $\sup_{t\geq0}\left\|\frac{dw}{dt}\right\|$, *See for example Corless and Leitmann* [34], *Krstic et al.* [35], *Khalil* [27]*However, as mentioned above w(t) is treated in this paper as a Wiener*

standard process vector, i.e., $\sup_{t\geq 0}\left\|\frac{dw}{dt}\right\|$ *unbounded, thus the control design techniques (such as those in Corless and Leitmann* [34]*, Krstic et al.* [35]*, Khalil* [27]*,for deterministic systems are not applicable to design a proper controller that can stabilize the system (2-117) and (2-118) at the origin in probability.*

4. *The stochastic term* $\left(\boldsymbol{C}_{A21}(\boldsymbol{v}^b)+\boldsymbol{C}_{A22}(\boldsymbol{\omega})+\boldsymbol{S}(\boldsymbol{v}^b)\boldsymbol{M}_A+\boldsymbol{S}(\boldsymbol{\omega})\boldsymbol{I}_A+\boldsymbol{D}_2\right)\Delta_{\mathbf{2}}(t)$ *creates difficulties in control design and stability analysis of resulting closed loop systems. The difficulties are due to the fact that Ito's differentiation rule (Pardoux (1975)) introduces both gradient and Hessian terms. If the stochastic term* $\left(\boldsymbol{C}_{A21}(\boldsymbol{v}^b)+\boldsymbol{C}_{A22}(\boldsymbol{\omega})+\boldsymbol{S}(\boldsymbol{v}^b)\boldsymbol{M}_A+\boldsymbol{S}(\boldsymbol{\omega})\boldsymbol{I}_A+\boldsymbol{D}_2\right)\Delta_{\mathbf{2}}(t)$ *is ignored, the stochastic aircraft system (2-117) and (2-118) becomes deterministic and there is a number of existing controllers. If these controllers are applied to the stochastic aircraft system (2-117) and (2-118) the control performance is deteriorated and the closed loop system is even unstable, see the illustrative example in Section 1.2. This instability can be seen from (2-109) that the term* $\boldsymbol{C}_A(\varsigma_r)$ *due to the Coriolis-centripetal matrix of the added mass contains* $\left(\boldsymbol{C}_A(\varsigma)+\overline{\boldsymbol{C}}_A(\varsigma)\right)\varsigma_a$*, which can be written as* $\left(\boldsymbol{C}_A(\varsigma)+\overline{\boldsymbol{C}}_A(\varsigma)\right)\varsigma_a=\left(\boldsymbol{C}_A(\varsigma_a)+\overline{\boldsymbol{C}}_A(\varsigma_a)\right)\varsigma$ *which is non vanishing. Since matrix* $\boldsymbol{C}_A(\varsigma)$ *is skew symmetric, the instability due to stochastic perturbation in the example from Section 1.2 holds for the stochastic quadrotor aircraft system (2-117) and (2-118).*

2.6.3 Motor Speed to Control Force and Torque Vector

The force f and the torque $\boldsymbol{\tau}=[\tau_p \quad \tau_q \quad \tau_r]^{\mathrm{T}}$, can be calculated from the propeller speed squared $[\Omega_1^2 \quad \Omega_2^2 \quad \Omega_3^2 \quad \Omega_4^2]^{\mathrm{T}}$ as:

$$\begin{bmatrix} f \\ \boldsymbol{\tau} \end{bmatrix}=\begin{bmatrix} f \\ \tau_p \\ \tau_q \\ \tau_r \end{bmatrix}=\begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix}=\begin{bmatrix} c_T & c_T & c_T & c_T \\ -\frac{\sqrt{2}}{2}c_T L & \frac{\sqrt{2}}{2}c_T L & \frac{\sqrt{2}}{2}c_T L & -\frac{\sqrt{2}}{2}c_T L \\ \frac{\sqrt{2}}{2}c_T L & \frac{\sqrt{2}}{2}c_T L & -\frac{\sqrt{2}}{2}c_T L & -\frac{\sqrt{2}}{2}c_T L \\ c_R L & -c_R L & c_R L & -c_R L \end{bmatrix}\begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{bmatrix} \tag{2-119}$$

where c_T represents the propellers coefficient of thrust and c_R the propellers coefficient of reactive torque, L the radius of the aircraft in th body x-y plane, Ω_1, Ω_2, Ω_3, Ω_4 repressent the angulare velocity of each propeller. Hence we can state the control signals as:

$$\begin{aligned} U_1 &= c_T(\Omega_1^2+\Omega_2^2+\Omega_3^2+\Omega_4^2), \\ U_2 &= \frac{\sqrt{2}}{2}c_T L(-\Omega_1^2+\Omega_2^2+\Omega_3^2-\Omega_4^2), \\ U_3 &= \frac{\sqrt{2}}{2}c_T L\left(\Omega_2^2+\Omega_3^2-\Omega_5^2-\Omega_6^2\right), \\ U_4 &= c_R L(\Omega_1^2-\Omega_2^2+\Omega_3^2-\Omega_4^2), \end{aligned} \tag{2-120}$$

and the calculation for the required motor speed as:

$$\begin{cases} \Omega_1^2 = \frac{1}{4c_T}U_1 - \frac{\sqrt{2}}{4c_T L}U_2 + \frac{\sqrt{2}}{4c_T L}U_3 + \frac{1}{4c_R L}U_4, \\ \Omega_2^2 = \frac{1}{4c_T}U_1 + \frac{\sqrt{2}}{4c_T L}U_2 + \frac{\sqrt{2}}{4c_T L}U_3 - \frac{1}{4c_R L}U_4, \\ \Omega_3^2 = \frac{1}{4c_T}U_1 + \frac{\sqrt{2}}{4c_T L}U_2 - \frac{\sqrt{2}}{4c_T L}U_3 + \frac{1}{4c_R L}U_4, \\ \Omega_4^2 = \frac{1}{4c_T}U_1 - \frac{\sqrt{2}}{4c_T L}U_2 - \frac{\sqrt{2}}{4c_T L}U_3 - \frac{1}{4c_R L}U_4. \end{cases} \quad (2\text{-}121)$$

2.7 State Estimation

In this section methods for estimating unmeasured states and unknown external disturbances are presented

2.7.1 Disturbance Observer

The following disturbance observer is based closely on that presented in [36]. Consider the following system:

$$\begin{aligned}\dot{\boldsymbol{x}}_1 &= \boldsymbol{f}_1(t, \boldsymbol{x}_1, \boldsymbol{x}_2),\\ \dot{\boldsymbol{x}}_2 &= \boldsymbol{f}_2(t, \boldsymbol{x}_1, \boldsymbol{x}_2) + \mathbf{G}(t, \boldsymbol{x}_1)\boldsymbol{d}(t),\end{aligned} \tag{2-122}$$

where $t \in \mathbb{R}^+$, $x_1 \in \mathbb{R}^n$, $x_2 \in \mathbb{R}^n$, $\boldsymbol{f}_1(t, \boldsymbol{x}_1, \boldsymbol{x}_2)$ and $\boldsymbol{f}_2(t, \boldsymbol{x}_1, \boldsymbol{x}_2)$ are known vector functions of $t, \boldsymbol{x}_1$ and $\boldsymbol{x}_2$. $\mathbf{G}(t, \boldsymbol{x}_1)$ is a matrix whose elements are functions of t and $\boldsymbol{x}_1$, and $\boldsymbol{d}(t)$ is a vector of unknown disturbances. The system (2-122)satisfies the following assumption:

Assumption 2.1

1. *The disturbance vector $\boldsymbol{d}(t)$ and its derivative are bounded, i.e., there exist nonnegative constants d_M and d_{1M} such that $\|\boldsymbol{d}(t)\| \le d_M$ and $\|\dot{\boldsymbol{d}}(t)\| \le d_{1M}, \forall t \ge t_0 \ge 0$.*
2. *The system* (2-122) *is well-posed for all $t \ge t_0 \ge 0$.*
3. *The matrix $\mathbf{G}(t, \boldsymbol{x}_1)$ is invertible for all $t \ge t_0 \ge 0$ and $\boldsymbol{x}_1 \in \mathbb{R}^n$, and is differentiable with respect to t and $\boldsymbol{x}_1$.*

The disturbance observer is given in the following lemma [36]:

Lemma 2.2 Following the conditions specified in Assumption 2.1 the disturbance observer estimate $\widehat{\boldsymbol{d}}$ of the disturbance $\boldsymbol{d}(t)$ is given as follows:

$$\begin{aligned}\widehat{\boldsymbol{d}} &= \boldsymbol{\xi} + \mathbf{K}\mathbf{G}^{-1}(t, \boldsymbol{x}_1)\boldsymbol{x}_2,\\ \dot{\boldsymbol{\xi}} &= -\mathbf{K}\boldsymbol{\xi} - \mathbf{K}\left(\frac{\partial \mathbf{G}^{-1}(t, \boldsymbol{x}_1)}{\partial t} + \frac{\partial \mathbf{G}^{-1}(t, \boldsymbol{x}_1)}{\partial \boldsymbol{x}_1}\boldsymbol{f}_1(t, \boldsymbol{x}_1, \boldsymbol{x}_2)\right)\boldsymbol{x}_2\\ &\qquad - \mathbf{K}\mathbf{G}^{-1}(t, \boldsymbol{x}_1)(\boldsymbol{f}_2(t, \boldsymbol{x}_1, \boldsymbol{x}_2) + \mathbf{G}(t, \boldsymbol{x}_1)\mathbf{K}\mathbf{G}^{-1}(t, \boldsymbol{x}_1)\boldsymbol{x}_2),\\ \boldsymbol{\xi}(t_0) &= -\mathbf{K}\mathbf{G}^{-1}\big(t_0, \boldsymbol{x}_1(t_0)\big)\boldsymbol{x}_2(t_0),\end{aligned} \tag{2-123}$$

where $\mathbf{K}$ is a symmetric and positive definite matrix. To prove that the system (2-123) guarantees that the disturbance observer error $\boldsymbol{d}_e = \boldsymbol{d} - \widehat{\boldsymbol{d}}$ and the disturbance observer $\widehat{\boldsymbol{d}}$ remain bounded for all $t \ge t_0 \ge 0$. We will present the proof from [36], we first consider $\boldsymbol{d}_e$, by differentiating the disturbance observer error $\boldsymbol{d}_e$ along the solution of (2-123) we obtain:

$$\dot{\boldsymbol{d}}_e = \dot{\boldsymbol{d}} - \mathbf{K}\boldsymbol{d}_e \tag{2-124}$$

If we consider the following Lyapunov function:

$$V_e = \frac{1}{2}\|\boldsymbol{d}_e\|^2. \tag{2-125}$$

Differentiating (2-125) along the solution of (2-124) yields:

$$\dot{V}_e = -\frac{1}{2}\boldsymbol{d}_e^{\mathrm{T}}(\mathbf{K}+\mathbf{K}^{\mathrm{T}})\boldsymbol{d}_e + \boldsymbol{d}_e\dot{\boldsymbol{d}}. \tag{2-126}$$

Since $\mathbf{K}$ is a positive definite matrix then given the minimum Eigen value of $\mathbf{K}$, $\lambda_m(\mathbf{K})$ it can be said that:

$$\frac{1}{2}\boldsymbol{d}_e^{\mathrm{T}}(\mathbf{K}+\mathbf{K}^{\mathrm{T}})\boldsymbol{d}_e \geq \lambda_m(\mathbf{K})\|\boldsymbol{d}_e\|^2. \tag{2-127}$$

Substituting (2-127) into (2-126) yields:

$$\dot{V}_e \leq -\lambda_m(\mathbf{K})\|\boldsymbol{d}_e\|^2 + \boldsymbol{d}_e d_{1M}. \tag{2-128}$$

By completing, the square it can be said that:

$$\dot{V}_e \leq -\lambda_m(\mathbf{K})V_e + d_{1M}^2. \tag{2-129}$$

With regards to inequality (2-129) it can be seen that V_e exponentially converges to a ball centred at the origin. To prove this, we can rewrite (2-129) as follows:

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(V_e - \frac{d_{1M}^2}{2\lambda_m^2(\mathbf{K})}\right) \leq -\lambda_m(\mathbf{K})\left(V_e - \frac{d_{1M}^2}{2\lambda_m^2(\mathbf{K})}\right). \tag{2-130}$$

Solving (2-130) gives:

$$V_e \leq \left(V_e(t_0) - \frac{d_{1M}^2}{2\lambda_m^2(\mathbf{K})}\right)e^{-\lambda_m(\mathbf{K})(t-t_0)} + \frac{d_{1M}^2}{2\lambda_m^2(\mathbf{K})}, \quad \forall t \geq t_0 \geq 0. \tag{2-131}$$

By using the definition of V_e as defined in (2-125), the following bound on $\boldsymbol{d}_e$ is obtained:

$$\|\boldsymbol{d}_e\| \leq \sqrt{\left(\|\boldsymbol{d}_e(t_0)\|^2 - \frac{d_{1M}^2}{\lambda_m^2(\mathbf{K})}\right)e^{-\lambda_m(\mathbf{K})(t-t_0)} + \frac{d_{1M}^2}{\lambda_m^2(\mathbf{K})}}, \quad \forall t \geq t_0 \geq 0. \tag{2-132}$$

Therefore, it has been proven that $\boldsymbol{d}_e$ remains bounded to prove $\widehat{\boldsymbol{d}}$ remains bounded first the disturbance observer $\widehat{\boldsymbol{d}}$ is differentiated along the solution of (2-122) and (2-123) we obtain:

$$\dot{\widehat{\boldsymbol{d}}} = -\mathbf{K}\widehat{\boldsymbol{d}} + \mathbf{K}\boldsymbol{d}. \tag{2-133}$$

Considering the Lyapunov function candidate:

$$V = \frac{1}{2}\left\|\widehat{\boldsymbol{d}}\right\|^2, \tag{2-134}$$

whose derivative along the solution of (2-133) is:

$$\dot{V} = -\widehat{\boldsymbol{d}}^{\mathrm{T}}\mathbf{K}\widehat{\boldsymbol{d}} + \widehat{\boldsymbol{d}}^{\mathrm{T}}\mathbf{K}\boldsymbol{d}, \\ \dot{V} \leq -\lambda_m(\mathbf{K})V + \frac{\lambda_M^2(\mathbf{K})}{2\lambda_m(\mathbf{K})}d_M^2. \tag{2-135}$$

With regards to inequality (2-135) it can be seen that V_e exponentially converges to a ball centred at the origin [36]. To prove this, we can rewrite (2-135) as follows:

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(V - \frac{\lambda_M^2(\mathbf{K})}{2\lambda_m^2(\mathbf{K})}\right) \leq -\lambda_m(\mathbf{K})\left(V - \frac{\lambda_M^2(\mathbf{K})}{2\lambda_m^2(\mathbf{K})}d_M^2\right), \tag{2-136}$$

Solving (2-136) gives:

$$V \leq \left(V(t_0) - \frac{\lambda_M^2(\mathbf{K})}{2\lambda_m^2(\mathbf{K})}d_M^2\right)e^{-\lambda_m(\mathbf{K})(t-t_0)} + \frac{\lambda_M^2(\mathbf{K})}{2\lambda_m^2(\mathbf{K})}d_M^2, \quad \forall t \geq t_0 \geq 0. \tag{2-137}$$

By using the definition of V as in (2-134), we can obtain the following bound on $\widehat{\boldsymbol{d}}$:

$$\|\widehat{\boldsymbol{d}}\| \leq \sqrt{\left(\|\widehat{\boldsymbol{d}}(t_0)\|^2 - \frac{\lambda_M^2(\mathbf{K})}{\lambda_m^2(\mathbf{K})}d_M^2\right)e^{-\lambda_m(\mathbf{K})(t-t_0)} + \frac{\lambda_M^2(\mathbf{K})}{\lambda_m^2(\mathbf{K})}d_M^2}, \quad \forall t \geq t_0 \geq 0. \tag{2-138}$$

Since $\|\widehat{\boldsymbol{d}}(t_0)\|^2 = \mathbf{0}$, (2-138) becomes:

$$\|\widehat{\boldsymbol{d}}\| \leq \frac{\lambda_M(\mathbf{K})}{\lambda_m(\mathbf{K})}d_M, \quad \forall t \geq t_0 \geq 0. \tag{2-139}$$

Therefore, it has been proven that both $\boldsymbol{d}_e$ and $\widehat{\boldsymbol{d}}$ remains bounded. We can therefore state:

$$\|\boldsymbol{d}_e\| \leq \sqrt{\left(\|\boldsymbol{d}_e(t_0)\|^2 - \frac{d_{1M}^2}{\lambda_m^2(\mathbf{K})}\right)e^{-\lambda_m(\mathbf{K})(t-t_0)} + \frac{d_{1M}^2}{\lambda_m^2(\mathbf{K})}}, \\ \|\widehat{\boldsymbol{d}}\| \leq \frac{\lambda_M(\mathbf{K})}{\lambda_m(\mathbf{K})}d_M. \tag{2-140}$$

2.7.2 Unmeasured State Estimation

In works presented by [37], the issue of a class of stochastic output-feedback nonlinear systems driven by noise whose covariance is time varying and bounded, but whose bound is not known was addressed. It was assumed that the measurable states derivative contained the same Wiener noise and unknown covariance in the final system state where the control signal appeared. Also in works presented by Ji and Xi, in [38] the adaptive stabilization and tracking problems for a class of output feedback canonical systems driven by Wiener noises of unknown covariance was addressed. However, the presented works required that the derivative of the measured state, contain the unknown deterministic disturbance and the Wiener noises of unknown covariance present in order to estimate the Wiener noise in the unmeasured state. In both situations adaptive control was obtained via the use of backstepping. However, for the case of the quadrotor we consider no noise to be present in the derivative of the position, only in the derivative of the velocity. The reason why no noise is considered in the velocity state is because we are presenting a velocity state estimator that will work for the quadrotor model developed in section 2.6.2 and presented in equation (2-117). This model does not contain noise in the velocity state only in the acceleration state. As such, it is not possible to obtain an adaptive control. Instead, we will treat the unknown but bounded deterministic and stochastic disturbances as constants and develop a robust control scheme. Consider the following system of equations:

$$\begin{aligned} d\boldsymbol{x}_1 &= \boldsymbol{x}_2 dt, \\ d\boldsymbol{x}_2 &= (-\mathbf{D}_1\boldsymbol{x}_2 + \boldsymbol{f}_1(t,\boldsymbol{x}_1) + \mathbf{G}(t,\boldsymbol{x}_1)\boldsymbol{u})dt + \boldsymbol{g}_1(t)d\boldsymbol{w}, \end{aligned} \tag{2-141}$$

where the state $\boldsymbol{x}_1$ is known and $\boldsymbol{x}_2$ is unknown, we wish to design a practical controller to force the system to track a reference signal $\boldsymbol{x}_{1d}$. We will define the following state estimate errors as:

$$\begin{aligned} \tilde{\boldsymbol{x}}_1 &= \boldsymbol{x}_1 - \hat{\boldsymbol{x}}_1, \\ \tilde{\boldsymbol{x}}_2 &= \boldsymbol{x}_2 - \hat{\boldsymbol{x}}_2, \end{aligned} \tag{2-142}$$

we state,

$$\begin{aligned} d\hat{\boldsymbol{x}}_1 &= (\mathbf{K}_1\tilde{\boldsymbol{x}}_{01} + \hat{\boldsymbol{x}}_2 + \boldsymbol{h}_1)dt, \\ d\hat{\boldsymbol{x}}_2 &= (-\mathbf{D}_1\hat{\boldsymbol{x}}_2 + \mathbf{K}_{02}\tilde{\boldsymbol{x}}_1 + \boldsymbol{f}_1(t,\boldsymbol{x}_1) + \boldsymbol{h}_2 + \mathbf{G}(t,\boldsymbol{x}_1)\boldsymbol{u})dt, \end{aligned} \tag{2-143}$$

and

$$\begin{aligned} d\tilde{\boldsymbol{x}}_1 &= (\tilde{\boldsymbol{x}}_2 - \mathbf{K}_{01}\tilde{\boldsymbol{x}}_1 - \boldsymbol{h}_1)dt, \\ d\tilde{\boldsymbol{x}}_2 &= (-\mathbf{D}_1\tilde{\boldsymbol{x}}_2 - \mathbf{K}_{02}\tilde{\boldsymbol{x}}_1 - \boldsymbol{h}_2)dt + \boldsymbol{g}_1(t)d\boldsymbol{w}, \end{aligned} \tag{2-144}$$

the coordinate transformation;

$$\begin{aligned} \boldsymbol{x}_{1e} &= \boldsymbol{x}_1 - \boldsymbol{x}_{1d}, \\ \boldsymbol{x}_{2e} &= \hat{\boldsymbol{x}}_2 - \alpha_1. \end{aligned} \tag{2-145}$$

We will consider the Lyapunov function candidate:

$$V_1 = \frac{1}{2}(\|\boldsymbol{x}_{1e}\|^2 + \|\boldsymbol{x}_{2e}\|^2 + \|\tilde{\boldsymbol{x}}_1\|^2 + \|\tilde{\boldsymbol{x}}_2\|^2), \tag{2-146}$$

The infinitesimal generator of (2-146) is:

$$\begin{aligned} \mathcal{L}V_1 = {} & \boldsymbol{x}_{1e}^{\mathrm{T}}(\boldsymbol{x}_2 - \dot{\boldsymbol{x}}_{1d}) + \boldsymbol{x}_{2e}^{\mathrm{T}}\big(\dot{\hat{\boldsymbol{x}}}_2 - \dot{\alpha}_1\big) + \tilde{\boldsymbol{x}}_1^{\mathrm{T}}\big(\boldsymbol{x}_2 - \dot{\hat{\boldsymbol{x}}}_1\big) \\ & + \tilde{\boldsymbol{x}}_2^{\mathrm{T}}\big(-\mathbf{D}_1\boldsymbol{x}_2 + \boldsymbol{f}_1(t,\boldsymbol{x}_1,\boldsymbol{x}_2) + \mathbf{G}(t,\boldsymbol{x}_1)\boldsymbol{u} - \dot{\hat{\boldsymbol{x}}}_2\big) + \frac{1}{2}\mathrm{Tr}\{\boldsymbol{g}_1(t)\boldsymbol{g}_1^{\mathrm{T}}(t)\}, \end{aligned} \tag{2-147}$$

this can be re written as:

$$\begin{aligned} \mathcal{L}V_1 = {} & \boldsymbol{x}_{1e}^{\mathrm{T}}(\tilde{\boldsymbol{x}}_2 + \boldsymbol{x}_{2e} + \alpha_1 - \dot{\boldsymbol{x}}_{1d}) + \boldsymbol{x}_{2e}^{\mathrm{T}}\big(\dot{\hat{\boldsymbol{x}}}_2 - \dot{\alpha}_1\big) - \tilde{\boldsymbol{x}}_1^{\mathrm{T}}\boldsymbol{h}_1 - \tilde{\boldsymbol{x}}_1^{\mathrm{T}}\mathbf{K}_{01}\tilde{\boldsymbol{x}}_1 - \tilde{\boldsymbol{x}}_2^{\mathrm{T}}\mathbf{D}_1\tilde{\boldsymbol{x}}_2 \\ & + \tilde{\boldsymbol{x}}_1^{\mathrm{T}}(\mathbf{I}_{3\times3} - \mathbf{K}_{02})\tilde{\boldsymbol{x}}_1 - \tilde{\boldsymbol{x}}_2^{\mathrm{T}}\boldsymbol{h}_2 + \frac{1}{2}\mathrm{Tr}\{\boldsymbol{g}_1(t)\boldsymbol{g}_1^{\mathrm{T}}(t)\}. \end{aligned} \tag{2-148}$$

Choosing the virtual control signal as:

$$\boldsymbol{\alpha}_1 = -\mathbf{K}_1\boldsymbol{x}_{1e} + \dot{\boldsymbol{x}}_{1d}, \tag{2-149}$$

Implies that (2-148) becomes:

$$\begin{aligned}\mathcal{L}V_1 = &-\boldsymbol{x}_{1e}^{\mathrm{T}}\mathbf{K}_1\boldsymbol{x}_{1e} + \boldsymbol{x}_{2e}^{\mathrm{T}}(\boldsymbol{x}_{1e} - \mathbf{D}_1\widehat{\boldsymbol{x}}_2 + \mathbf{K}_{02}\widetilde{\boldsymbol{x}}_1 + \boldsymbol{f}_1(t,\boldsymbol{x}_1) + \boldsymbol{h}_2 + \mathbf{G}(t,\boldsymbol{x}_1)\boldsymbol{u} \\ &+\mathbf{K}_1(\widetilde{\boldsymbol{x}}_2 + \boldsymbol{x}_{2e} + \alpha_1) - \ddot{\boldsymbol{x}}_{1d}) - \widetilde{\boldsymbol{x}}_1^{\mathrm{T}}\boldsymbol{h}_1 - \widetilde{\boldsymbol{x}}_1^{\mathrm{T}}\mathbf{K}_{01}\widetilde{\boldsymbol{x}}_1 + \widetilde{\boldsymbol{x}}_1^{\mathrm{T}}(\mathbf{I}_{3\times 3} - \mathbf{K}_{02})\widetilde{\boldsymbol{x}}_1 - \widetilde{\boldsymbol{x}}_2^{\mathrm{T}}\mathbf{D}_1\widetilde{\boldsymbol{x}}_2 \\ &-\widetilde{\boldsymbol{x}}_2^{\mathrm{T}}(\boldsymbol{h}_2 - \boldsymbol{x}_{1e}) + \frac{1}{2}\mathrm{Tr}\{\boldsymbol{g}_1(t)\boldsymbol{g}_1^{\mathrm{T}}(t)\},\end{aligned} \tag{2-150}$$

Choosing the interlace terms as:

$$\boldsymbol{h}_1 = \mathbf{K}_{02}^{\mathrm{T}}\boldsymbol{x}_{2e}, \tag{2-151}$$
$$\boldsymbol{h}_2 = \boldsymbol{x}_{1e} + \mathbf{K}_{01}^{\mathrm{T}}\boldsymbol{x}_{2e}, \tag{2-152}$$

moreover, we choose:

$$\mathbf{K}_{02} = \mathbf{I}_{3\times 3}, \tag{2-153}$$
$$2\boldsymbol{x}_{1e} - \mathbf{D}_1\widehat{\boldsymbol{x}}_2 + \boldsymbol{f}_1(t,\boldsymbol{x}_1) + \mathbf{K}_{01}^{\mathrm{T}}\boldsymbol{x}_{2e} + \mathbf{G}(t,\boldsymbol{x}_1)\boldsymbol{u} + \mathbf{K}_1(\boldsymbol{x}_{2e} - \mathbf{K}_1\boldsymbol{x}_{1e}) - \ddot{\boldsymbol{x}}_{1d} = -\mathbf{K}_2\boldsymbol{x}_{1e}, \tag{2-154}$$

Choosing the control signal as:

$$\boldsymbol{u} = \mathbf{G}^{-1}(t,\boldsymbol{x}_1)\big(\mathbf{D}_1\widehat{\boldsymbol{x}}_2 - \mathbf{K}_2\boldsymbol{x}_{1e} - \mathbf{K}_1\boldsymbol{x}_{2e} + \mathbf{K}_1\mathbf{K}_1\boldsymbol{x}_{1e} - 2\boldsymbol{x}_{1e} - \boldsymbol{f}_1(t,\boldsymbol{x}_1) - \mathbf{K}_{01}^{\mathrm{T}}\boldsymbol{x}_{2e} + \ddot{\boldsymbol{x}}_{1d}\big). \tag{2-155}$$

this implies:

$$\mathcal{L}V_1 = -\boldsymbol{x}_{1e}^{\mathrm{T}}\mathbf{K}_1\boldsymbol{x}_{1e} - \boldsymbol{x}_{2e}^{\mathrm{T}}\mathbf{K}_2\boldsymbol{x}_{1e} - \widetilde{\boldsymbol{x}}_1^{\mathrm{T}}\mathbf{K}_{01}\widetilde{\boldsymbol{x}}_1 - \widetilde{\boldsymbol{x}}_2^{\mathrm{T}}\mathbf{D}_1\widetilde{\boldsymbol{x}}_2 + \frac{1}{2}\mathrm{Tr}\{\boldsymbol{g}_1(t)\boldsymbol{g}_1^{\mathrm{T}}(t)\}, \tag{2-156}$$
$$\begin{aligned}\mathcal{L}V_1 \leq\ &-\lambda_m(\mathbf{K}_1)\|\boldsymbol{x}_{1e}\|^2 - \lambda_m(\mathbf{K}_2)\|\boldsymbol{x}_{2e}\|^2 - \lambda_m(\mathbf{K}_{01}) - \lambda_m(\mathbf{D}_1)\|\widetilde{\boldsymbol{x}}_2\|^2 \\ &+\frac{1}{2}\mathrm{Tr}\{\boldsymbol{g}_1(t)\boldsymbol{g}_1^{\mathrm{T}}(t)\}.\end{aligned} \tag{2-157}$$

Hence it can be stated:

$$\mathcal{L}V_1 \leq\ -cV_1 +\ \lambda, \tag{2-158}$$

where

$$c = \frac{\min\big(1, \lambda_m(\mathbf{K}_1), \lambda_m(\mathbf{K}_2), \lambda_m(\mathbf{K}_{01}), \lambda_m(\mathbf{K}_{02}), \lambda_m(\mathbf{D}_1)\big)}{\max\left(1, \frac{1}{2}\mathrm{Tr}\{\boldsymbol{g}_1(t)\boldsymbol{g}_1^{\mathrm{T}}(t)\}\right)}, \tag{2-159}$$
$$\lambda = \sup_{t\in R^n}\left(\frac{1}{2}\mathrm{Tr}\{\boldsymbol{g}_1(t)\boldsymbol{g}_1^{\mathrm{T}}(t)\}\right),$$

Hence

$$V_1(t) \leq \left(V_1(t_0) - \frac{\lambda}{c}\right)\mathrm{e}^{-c(t-t_0)} + \frac{\lambda}{c}, \tag{2-160}$$

Therefore, we can conclude that the system is asymptotically stable in probability.

2.7.3 Projection Algorithm

The projection algorithm presented in [39] and [40] is a method for estimating unknown system parameters, as such we will explain the projection algorithm in accordance with these works as follows, consider the following system:

$$\begin{aligned} \dot{\boldsymbol{x}}_i &= \boldsymbol{f}_i(t, \boldsymbol{x}_1, \dots \boldsymbol{x}_i) + \boldsymbol{g}_i(t, \boldsymbol{x}_1, \dots \boldsymbol{x}_i)\boldsymbol{x}_{i+1}, \\ \dot{\boldsymbol{x}}_n &= \boldsymbol{f}_n(t, \boldsymbol{x}_1, \dots \boldsymbol{x}_n) + \boldsymbol{g}_n(t, \boldsymbol{x}_1, \dots \boldsymbol{x}_n)u\,, \end{aligned} \tag{2-161}$$

where $t \in \mathbb{R}^+$, $x_1, \dots x_n \in \mathbb{R}^n$, $\boldsymbol{f_i}$ and g_i are unknown nonlinear system functions of $t, x_1, \dots x_n$ which are parameterized nonlinearly. Furthermore, we say:

$$\begin{aligned} \boldsymbol{f}_i(t, \boldsymbol{x}_1, \boldsymbol{x}_2) &= \theta^T_{\boldsymbol{f}_i(t)} \psi_{\boldsymbol{f}_i(t)}(\boldsymbol{x}_i), \\ \boldsymbol{g}_i(t, \boldsymbol{x}_1, \boldsymbol{x}_2) &= \theta^T_{\boldsymbol{g}_i(t)} \psi_{\boldsymbol{g}_i(t)}(\boldsymbol{x}_i), \end{aligned} \tag{2-162}$$

where $\psi_{\boldsymbol{f}_i(t)}(\boldsymbol{x}_i)$ and $\psi_{\boldsymbol{g}_i(t)}(\boldsymbol{x}_i)$ are known functions, while $\theta_{\boldsymbol{f}_i(t)} \in \mathbb{R}^{pi}$ and $\theta_{g_i(t)} \in \mathbb{R}^{pi}$ are unknown time varying parameters belonging to a known compact set $\Omega_i \in \mathbb{R}^{pi}$ and for simplicity will be considered a closed ball of known radius r_Ω centred at the origin. We summarise as follows

Assumption 2.2

1. *The unknown parameter θ and its derivative are bounded, i.e., there exist nonnegative constants θ_M and θ_{1M} such that $\|\theta(t)\| \leq \theta_M$ and $\|\dot{\theta}(t)\| \leq \theta_{1M}, \forall t \geq t_0 \geq 0$*
2. *The system* (2-162) *is well-posed for all $t \geq t_0 \geq 0$.*
3. *The function $\psi(t, \boldsymbol{x}_1)$ is invertible for all $t \geq t_0 \geq 0$ and $\boldsymbol{x}_i \in \mathbb{R}^n$, and is differentiable with respect to t and $\boldsymbol{x}_i$.*

We now define the tracking errors

$$\begin{aligned} &\boldsymbol{z}_1 = \boldsymbol{x}_1 - \boldsymbol{x}_{1d}, \qquad \boldsymbol{z}_2 = \boldsymbol{x}_2 - \alpha_1(\boldsymbol{x}_1, \dot{\boldsymbol{x}}_{1d}, \hat{\theta}_{g_1}, \hat{\theta}_{g_1}), \\ &\boldsymbol{z}_i = \boldsymbol{x}_i - \alpha_{1-1}(\boldsymbol{x}_1 \dots \boldsymbol{x}_i, \dot{\boldsymbol{x}}_{1d} \dots \boldsymbol{x}^{\boldsymbol{i}-1}_{1d}, \hat{\theta}_{f_1} \dots \hat{\theta}_{f_{i-1}}, \hat{\theta}_{g_1} \dots \hat{\theta}_{g_{i-2}}), \end{aligned} \tag{2-163}$$

where $\boldsymbol{x}_1$ denotes the system state and α_i denotes the virtual control signals. Based on the above objective we will define the update law for the parameter estimation:

$$\begin{aligned} \dot{\hat{\theta}}_{f_i} &= \gamma \mathrm{proj}(\|\boldsymbol{z}_1\|^2 \psi_{\boldsymbol{f}_i}(\boldsymbol{x}_i), \hat{\theta}_{f_i}), \\ \dot{\hat{\theta}}_{g_i} &= \gamma \mathrm{proj}(\|\boldsymbol{z}_1\|^2 \psi_{g_i}(\boldsymbol{x}_i), \hat{\theta}_{g_i}). \end{aligned} \tag{2-164}$$

Lemma 2.3 The operator 'proj' denotes the smooth projection algorithm in [39] as follows:

$$\begin{aligned} &\mathrm{proj}(\boldsymbol{\varpi}, \hat{\boldsymbol{\omega}}) = \boldsymbol{\varpi} && \text{if} \quad \Xi(\hat{\boldsymbol{\omega}}) \leq 0, \\ &\mathrm{proj}(\boldsymbol{\varpi}, \hat{\boldsymbol{\omega}}) = \boldsymbol{\varpi} && \text{if} \quad \Xi(\hat{\boldsymbol{\omega}}) > 0 \text{ and } \Xi(\hat{\boldsymbol{\omega}})\hat{\omega}\boldsymbol{\varpi} \leq 0, \\ &\mathrm{proj}(\boldsymbol{\varpi}, \hat{\boldsymbol{\omega}}) = (1 - \Xi(\hat{\boldsymbol{\omega}}))\boldsymbol{\varpi} && \text{if} \quad \Xi(\hat{\boldsymbol{\omega}}) > 0 \text{ and } \Xi(\hat{\boldsymbol{\omega}})_{\hat{\omega}}\boldsymbol{\varpi} > 0, \end{aligned} \tag{2-165}$$

where:

$$\Xi(\hat{\boldsymbol{\omega}}) = \frac{\|\hat{\boldsymbol{\omega}}\|^2 - \omega_{\mathrm{M}}^2}{(\xi^2 + 2\xi\omega_{\mathrm{M}})}, \quad \Xi(\hat{\boldsymbol{\omega}})_{\hat{\omega}} = \frac{\partial\, \Xi(\hat{\boldsymbol{\omega}})}{\partial \hat{\boldsymbol{\omega}}}, \tag{2-166}$$

where ξ is an arbitrary small positive constant and $\|\widehat{\boldsymbol{\omega}}\| \leq \omega_{\mathrm{M}}$. The projection algorithm is such that if $\dot{\widehat{\boldsymbol{\omega}}} = \boldsymbol{\Upsilon}\mathrm{proj}(\boldsymbol{\varpi}, \widehat{\boldsymbol{\omega}})$, where $\boldsymbol{\Upsilon}$ is a symmetric positive definite matrix and $\|\widehat{\boldsymbol{\omega}}\| \leq \omega_{\mathrm{M}}$ then:

a.) $\widehat{\boldsymbol{\omega}} \leq \omega_{\mathrm{M}} + \xi, \quad \forall t_0 \leq t \leq \infty.$ **(2-167)**
b.) $\mathrm{proj}(\boldsymbol{\varpi}, \widehat{\boldsymbol{\omega}}) = \boldsymbol{\varpi}$ is continuous.
c.) $\|\mathrm{proj}(\boldsymbol{\varpi}, \widehat{\boldsymbol{\omega}})\| \leq \|\boldsymbol{\varpi}\|$.
d.) $\widetilde{\boldsymbol{\omega}}^T \mathrm{proj}(\boldsymbol{\varpi}, \widehat{\boldsymbol{\omega}}) \geq \widetilde{\boldsymbol{\omega}}^T \boldsymbol{\varpi}$, with $\widetilde{\boldsymbol{\omega}} = \boldsymbol{\varpi} - \widehat{\boldsymbol{\omega}}$,

For original source see [40].

2.8 Control Design Strategies Examples with Trigonometric Functions

This subsection demonstrates through numerical simulation, the ability of the standard backstepping controller and disturbance observer presented subsections 2.4.1 and 2.7.1 to stabilise a simple system subject to a non-vanishing disturbance. Secondly, the effectiveness of the one-step ahead backstepping controllers presented in subsection 2.4.2 to stabilise a simple system. All simulations are done by the use of MATLAB.

2.8.1 Standard Backstepping Example with Disturbances

Consider the pure feedback system:

$$\dot{x}_1 = \tan(x_2)\,, \qquad \dot{x}_2 = u + d. \tag{2-168}$$

the input u of the system (2-168) will be designed to force x_1 to track a reference signal x_{1d} and stabilise x_2 at the origin while compensating for the disturbance d using the standard backstepping method. This system has been chosen because it is similar to the angular position system of a quadrotor in that there is appearance of the tan function which linking the two states together. While this system is not in the form of a non-linear strict feedback system it has as required by backstepping it does share the same singularity property with the angular position subsystem of a quadrotor This method consists of two steps. In the first step, we will design a virtual control α_1 derived from the second state to locally asymptotically and exponentially stabilize the state x_1 and the tracking of a reference signal x_{1d}. In the second step, we will design the control input u to stabilize the tracking error of the second stage and ensure that the system tracks the reference signal as closely as possible. Then we will specify the bounds on the disturbance observer gain to ensure exponential convergence of the system within a range of operation. Furthermore, we make the following assumptions on the disturbance $d(t)$.

Assumption 2.3

1. *The disturbance vector $d(t)$ and its derivative are bounded, i.e., there exist nonnegative constants d_M and d_{1M} such that* $\|d(t)\| \le d_M$ and $\|\dot{d}(t)\| \le d_{1M}$, $\forall t \ge t_0 \ge 0$.
2. *The reference signal x_{1d} and its derivative are bounded i.e. there is a constant ϱ_1 such that:*

$$\sup_{t\in R^n}\|\dot{x}_{1d}(t)\| \le \varrho_1 \;\; \sup_{t\in R^n}\|\ddot{x}_{1d}(t)\| \le \varrho_2, \forall t \ge t_0 \ge 0. \tag{2-169}$$

Step 1

First we will need to introduce the error coordinate transformation as:

$$z_1 = x_1 - x_{1d}, \tag{2-170}$$

$$z_2 = x_2 - \alpha_1, \tag{2-171}$$

where x_{1d} is the desired trajectory, α_1 is the virtual control of x_2, yielding the transformed equation for the first state:

$$\dot{z}_1 = \tan(z_2 + \alpha_1) - \dot{x}_{1d}, \tag{2-172}$$

$$\dot{z}_1 = \tan(\alpha_1) - (\tan(\alpha_1) - \tan(z_2 + \alpha_1)) - \dot{x}_{1d}. \quad (2\text{-}173)$$

We will analyse the stability of z_1 by using the Lyapunov function candidate:

$$V_1 = \frac{1}{2} z_1^2, \quad (2\text{-}174)$$

taking the derivatives of both sides of (2-174) along the solution of (2-173) yields:

$$\dot{V}_1 = z_1 \tan(\alpha_1) + z_1(-\tan(\alpha_2) + \tan(z_2 + \alpha_1)). \quad (2\text{-}175)$$

To achieve asymptotic stability for the closed loop system we choose the virtual control as:

$$\tan(\alpha_2) - \dot{x}_{1d} = -k_1 z_1, \quad (2\text{-}176)$$
$$\alpha_1 = -\tan^{-1}(k_1 z_1 - \dot{x}_{1d}), \quad (2\text{-}177)$$

where k_1 is a positive constant, implying:

$$\|\alpha_1\| < \frac{\pi}{2}, \quad \forall z_1, \dot{x}_{1d} \in \mathbb{R}. \quad (2\text{-}178)$$

Therefore, we rewrite (2-175) as:

$$\dot{V}_1 = -k_1 z_1^2 + z_1(1 - \tan(z_2 + \alpha_1)). \quad (2\text{-}179)$$

We will consider the last term of (2-179) in Step 2 and for the meantime, is assumed zero. Substituting (2-177) into (2-173) yields:

$$\dot{z}_1 = -\tan(\alpha_2) + \tan(z_2 + \alpha_1) - k_1 z_1. \quad (2\text{-}180)$$

Step 2

Referring back to the system defined in (2-171) and differentiating both sides we obtain:

$$\dot{z}_2 = \dot{x}_2 - \dot{\alpha}_1. \quad (2\text{-}181)$$

Using (2-177) we obtain:

$$\dot{\alpha}_1 = \frac{\partial \alpha_1}{\partial z_1}(\dot{z}_1) = \frac{-k_1(-\tan(\alpha_2) + \tan(z_2 + \alpha_1)) + k_1^2 z_1 + \ddot{x}_{1d}}{1 + (-k_1 z_1 + x_{1d})^2}. \quad (2\text{-}182)$$

rewriting (2-181) as:

$$\dot{z}_2 = u - \dot{\alpha}_1 + \hat{d} + d_e. \quad (2\text{-}183)$$

To analyse the stability of the origin of z_2 we consider the Lyapunov function:

$$V_2 = V_1 + \tfrac{1}{2} z_2^2, \quad (2\text{-}184)$$

taking the derivate of both sides of (2-184):

$$\dot{V}_2 = -k_1 z_1^2 + z_2\left(z_1 \frac{-\tan(\alpha_1) + \tan(\alpha_1 + z_2)}{z_2} + u + \hat{d} + d_e - \dot{\alpha}_1\right). \quad (2\text{-}185)$$

Let us choose the control input as:

$$u = \dot{\alpha}_1 - k_2 z_2 - \hat{d} + z_1 \frac{\tan(\alpha_1) - \tan(\alpha_1 + z_2)}{z_2}, \quad (2\text{-}186)$$

where k_2, is a positive constant, it is easily shown that the last term of (2-186) is well defined for $z_2 = 0$ and will therefore not be presented the proof. We can therefore say:

$$\dot{z}_2 = -k_2 z_2 + d_e, \quad (2\text{-}187)$$

this ensures that (2-185) becomes:

$$\dot{V}_2 = -k_1 z_1^2 - k_2 z_2^2 + z_2 d_e. \quad (2\text{-}188)$$

By using Young's inequality (2-188) becomes

$$\dot{V}_2 \leq -k_1 z_1^2 - k_2 z_2^2 + \varepsilon_1 z_2^2 + \tfrac{1}{4\varepsilon_1} d_e^2. \quad (2\text{-}189)$$

We now design the disturbance observer:

$$\begin{aligned} \hat{d} &= \xi + k_d z_2, \\ \dot{\xi} &= k_d\left(k_2 z_2 + z_1 \frac{\tan(\alpha_1) - \tan(\alpha_1 + z_2)}{z_2}\right), \\ \xi(t_0) &= -k_d z_2(t_0), \end{aligned} \quad (2\text{-}190)$$

where k_d is a positive constant, based on Assumption 2.3 system (2-190) guarantees that the disturbance observer error $d_e = d - \hat{d}$ and the disturbance observer estimate $\hat{d}$ remain bounded for all $t \geq t_0 \geq 0$. By differentiating both sides of the first equation of (2-190) along the solution of the second equation of (2-190) we obtain:

$$\dot{\hat{d}} = k_d d_e, \quad (2\text{-}191)$$

this implies that the derivative of the error d_e is:

$$\dot{d}_e = -k_d d_e + \dot{d}. \quad (2\text{-}192)$$

Step 3

To deal with the last term of (2-189) we choose the following Lyapunov function.

$$V_3 = V_2 + \tfrac{\delta_1}{2} d_e^2. \quad (2\text{-}193)$$

Taking the derivative of (2-193) along the solutions of (2-189) and (2-192) yields:

$$\dot{V}_3 \leq -k_1 z_1^2 - (k_2 - \varepsilon_1) z_2^2 - \left(\delta_1 k_d - \frac{1}{4\varepsilon_1}\right) d_e^2 + \delta_1 d_e \dot{d}. \tag{2-194}$$

By using Young's inequality (2-194) becomes:

$$\dot{V}_3 \leq -k_1 z_1^2 - (k_2 - \varepsilon_1) z_2^2 - \left(\delta_1 k_d - \frac{1}{4\varepsilon_1} - \delta_1 \varepsilon_2\right) d_e^2 + \frac{\delta_1}{4\varepsilon_2} \dot{d}^2, \tag{2-195}$$

which means that we can now choose the values of k_2, k_d, δ_1, ε_1 and ε_2 such that:

$$k_2 - \varepsilon_1 > 0, \tag{2-196}$$

$$\delta_1 k_d - \frac{1}{4\varepsilon_1} - \delta_1 \varepsilon_2 \leq \frac{\gamma}{2}. \tag{2-197}$$

Substituting (2-197) into (2-195) gives:

$$\dot{V}_3 \leq -k_1 z_1^2 - (k_2 - \varepsilon_1) z_2^2 - \frac{\gamma}{2} d_e^2 + \frac{\delta_1}{4\varepsilon_2} \dot{d}^2, \tag{2-198}$$

letting:

$$c = \frac{\min\left(k_1, (k_2 - \varepsilon_1), \frac{\gamma_2}{2}\right)}{\max\left(\frac{1}{2}, \frac{\delta_1}{2}\right)}, \tag{2-199}$$

$$\lambda = \frac{\delta_1}{4\varepsilon_2} d_{1M}^2,$$

implies that:

$$\dot{V}_3 \leq cV + \lambda. \tag{2-200}$$

Solving the differential inequality (2-200) shows that V_3 , (2-198) exponentially converges to a ball of radius $r = \lambda / c$, while $\|x_2\| < \pi/2$ that is $\|z_2 + \alpha_1\| < \pi/2$.

We need to prove now the conditions under which $\|x_2\| < \pi/2$. Firstly, the Lyapunov function (2-193) is positive definite and well defined if $\|z_2 + \alpha_1\| < \pi/2$. In this case the closed loop system is locally asymptotically and exponentially stable. To show the conditions under which this holds we have to define the bounds on the initial states and control gains such that:

$$\|z_2 + \alpha_1\| < \pi/2, \tag{2-201}$$

$$\|x_2\| < \pi/2. \tag{2-202}$$

Referring back to (2-200) it can be concluded that:

$$V_3(t) \leq \left(V_3(t_0) - \frac{\lambda}{c}\right) e^{-c(t - t_0)} + \frac{\lambda}{c}. \tag{2-203}$$

To ensure exponential boundedness of the system we need to choose the initial conditions such that the following holds true

$$\frac{\lambda}{c} \leq z_1^2(t_0) + z_2^2(t_0) + \delta_1 d_e^2(t_0), \tag{2-204}$$

Referring back to (2-200) it can be concluded that:

$$z_1^2(t) + z_2^2(t) + \delta_1 d_e^2(t) \leq z_1^2(t_0) + z_2^2(t_0) + \delta_1 d_e^2(t_0) + \frac{\lambda}{c} := \Omega_0, \tag{2-205}$$

this in turn implies that we are able to state:

$$\|z_1(t), z_2(t), d_e(t)\| \leq \|z_1(t_0), z_2(t_0), d_e(t_0)\| \mathrm{e}^{-c(t-t_0)/2} + \sqrt{\frac{\lambda}{c}}, \tag{2-206}$$

It can therefore be said that:

$$\|z_1(t)\| \leq \sqrt{\Omega_0}, \qquad \|z_2(t)\| \leq \sqrt{\Omega_0}, \qquad \|d_e(t)\| \leq \sqrt{\frac{\Omega_0}{\delta_1}}, \qquad \frac{\lambda}{c} \leq \Omega_0. \tag{2-207}$$

From (2-132) we know that:

$$\|d_e(t)\| \leq \sqrt{\left(\|d_e(t_0)\|^2 - \frac{d_{1M}^2}{2k_d^2}\right) e^{-k_d(t-t_0)} + \frac{d_{1M}^2}{2k_d^2}} \leq d_M < \sqrt{\frac{\Omega_0}{\delta_1}}, \quad \forall t \geq t_0 \geq 0 \tag{2-208}$$

This implies that we can specify the disturbance observer gain k_d and the constant δ_1 by:

$$d_M < \frac{\Omega_0}{\delta_1}, \tag{2-209}$$

it can be said that:

$$\|z_2(t)\| \leq \|z_1(t), z_2(t), d_e(t)\|. \tag{2-210}$$

By using (2-209), (2-201) and (2-177) we can obtain:

$$\|z_2(t)\| \leq \|z_1(t_0), z_2(t_0), d_e(t_0)\| \mathrm{e}^{-c(t-t_0)/2} + \sqrt{\frac{\lambda}{c}} < \frac{\pi}{2}, \tag{2-211}$$

furthermore:

$$\left\|z_1(t_0), \left(x_2(t_0) + \arctan\left(k_1 z_1(t_0) - \dot{x}_{1d}(t_0)\right)\right), \delta_1 d_e(t_0)\right\| + \sqrt{\frac{\lambda}{c}} < \frac{\pi}{2}, \tag{2-212}$$

where the constant Ω_0 and ϱ_1 are defined in (2-205) and (2-169) respectively. Hence we can see that the possible initial value of the state x_2 depends not only on its own initial value but also that of x_1, the reference trajectory x_{1d} and its derivative $\dot{x}_{1d}$. Rearranging (2-212) gives

$$k_1 < \frac{1}{z_1(t_0)}\left(\|\dot{x}_{1d}(t_0)\| + \left\|\tan\left(\sqrt{\left(\frac{\pi}{2} - \sqrt{\frac{\lambda}{c}}\right)^2 - z_1^2(t_0) - \delta_1 d_e^2(t_0)} + \|x_2(t_0)\|\right)\right\|\right). \tag{2-213}$$

where the constant ϱ_1 is defined in (2-169). This implies that we have the following bound on the control gain k_1.

$$k_1 < \frac{1}{z_1^M(t_0)}\left(\varrho_1 + \left\|\tan\left(\sqrt{\left(\frac{\pi}{2} - \sqrt{\frac{\lambda}{c}}\right)^2 - z_1^2(t_0) - \delta_1 d_e^2(t_0)} + \|x_2(t_0)\|\right)\right\|\right). \tag{2-214}$$

Furthermore, it can be said that for (2-213) to be well defined the following must hold true:

$$\left(\frac{\pi}{2} - \sqrt{\frac{\lambda}{c}}\right)^2 - z_1^2(t_0) - \delta_1 d_e^2(t_0) \geq 0, \tag{2-215}$$

this implies that we can obtain the following bound on the initial value of z_1:

$$\|z_1(t_0)\| < \sqrt{\left(\frac{\pi}{2} - \sqrt{\frac{\lambda}{c}}\right)^2 - \delta_1 d_e^2(t_0)}. \tag{2-216}$$

Therefore, we have a means of choosing a suitable value for the constants δ_1, ε_1, ε_2 and k_2 to stabilise the system. Furthermore, we have provided initial conditions on the state error z_1, z_2 the disturbance error d_e, the state x_2, the initial rate of change of the reference signal $\dot{x}_{1d}$ along with the control and observer gains k_1 , k_2 and k_d. Therefore, with (2-204), (2-212), (2-214), (2-216), and (2-208), conditions (2-201) and (2-202) are satisfied. Implying that all state and disturbance errors exponentially converging to 0.

2.8.1.2 Simulation

This section illustrates the effectiveness of the proposed local tracking controller through a numerical simulation using MATLAB, the code for the following simulations can be found in Appendix A. The system parameters as to control and observer gains are chosen as:

$$k_1 = 3, \qquad k_2 = 2.5, \qquad k_d = 200, \qquad \delta_1 = 0.001, \qquad \varepsilon_1 = 2.4, \varepsilon_2 = 0.1. \tag{2-217}$$

The initial conditions are chosen as

$$x_1(t_0) = -0.5, \qquad x_2(t_0) = 0, \qquad x_{1d} = 1.255t, \qquad \dot{x}_{1d} = 1.255. \tag{2-218}$$

The disturbance is modeled as:

$$\begin{aligned} d &= \sum_{i=1}^{15} \sin(i * t) + \sin(x_2)\sin\left(i * \frac{t}{2}\right), \\ d_M &= 18.3, \\ d_{1M} &= 16.5180. \end{aligned} \tag{2-219}$$

The control gains, initial conditions and disturbance observer parameters have been chosen so that conditions (2-196), (2-197), (2-204), (2-205), (2-208), (2-212), (2-214) and (2-216) hold true.

Furthermore, the parameters:

$$c = \frac{\min\left(k_1, (k_2 - \varepsilon_1), \frac{\gamma_2}{2}\right)}{\max\left(\frac{1}{2}, \frac{\delta_1}{2}\right)} = 0.1915,$$
$$\lambda = \frac{\delta_1}{4\varepsilon_2} d_{1M}^2 = 0.0284,$$
$$\frac{\lambda}{c} := 0.1484. \quad \text{(2-220)}$$

The control gain k_1:

$$k_1 = 2 < \frac{1}{0.5}\left(1.255 + \left\|\tan\left(\sqrt{\left(\frac{\pi}{2} - \sqrt{0.14849}\right)^2 - 0.5 - 18.3^2 * 10^{-3}}\right)\right\|\right) := 5. \quad \text{(2-221)}$$

This implies that condition (2-213) is satisfied. The time history of the states z_1, z_2 and d_e are shown in Figure 2-7 we can clearly see that the state errors converge to zero:

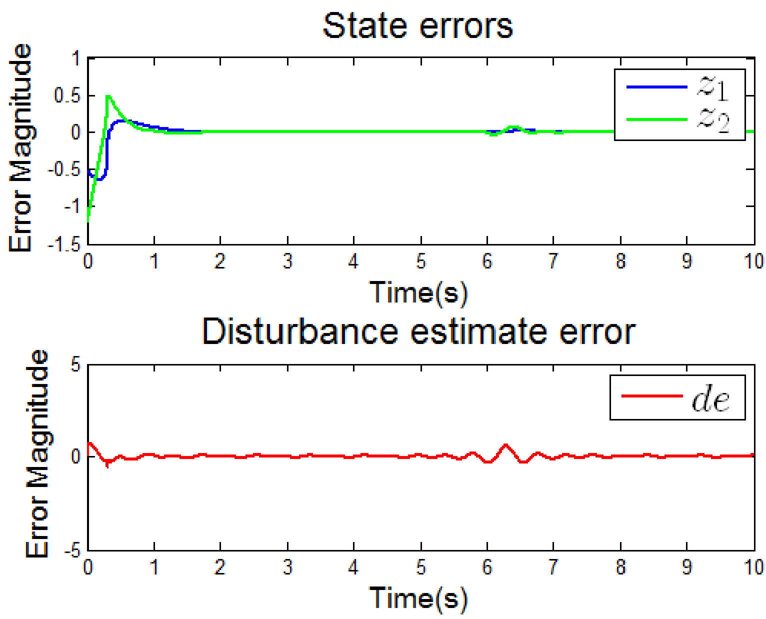

Figure 2-7: Time history of states and disturbance estimate errors.

furthermore $\|x_2\| < \pi/2$.

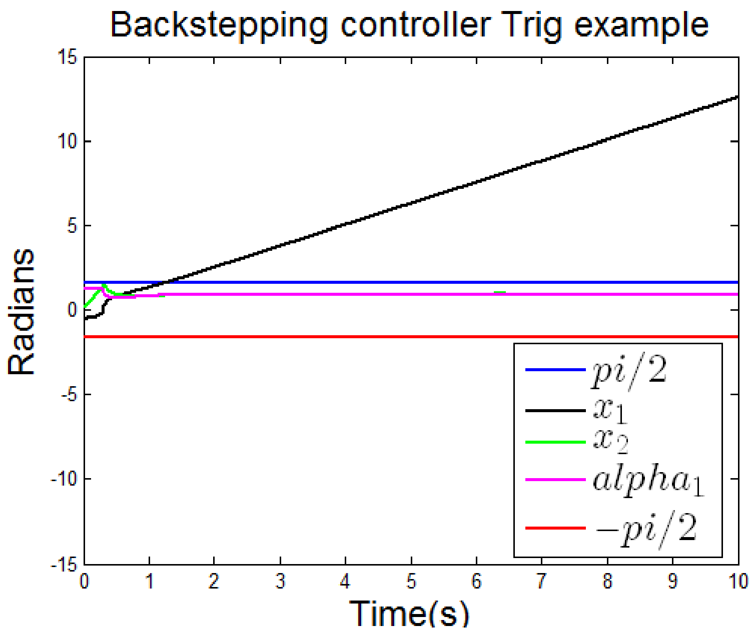

Figure 2-8: Time history of state and virtual control signal.

Furthermore, inspecting the Lyapunov function V_3 in Figure 2-9, it is strictly less than the bound:

$$V_{3,bound} = \left(V_3(t_0) - \frac{\lambda}{c}\right) \mathrm{e}^{-c(t-t_0)} + \frac{\lambda}{c}. \quad \text{(2-222)}$$

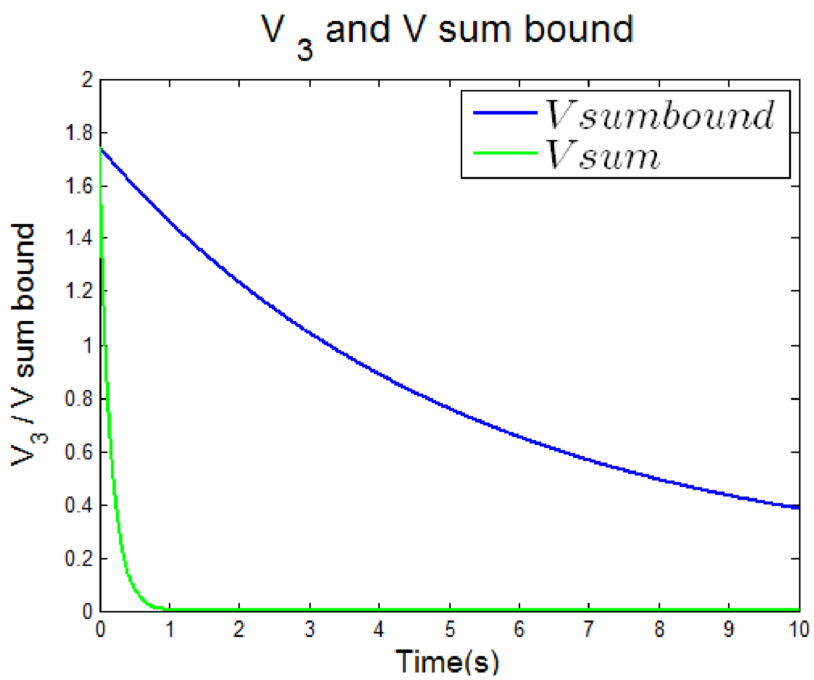

Figure 2-9: Time history of V_3 and $V_{3\ bound}$.

in addition, the control signal u is bounded as shown below:

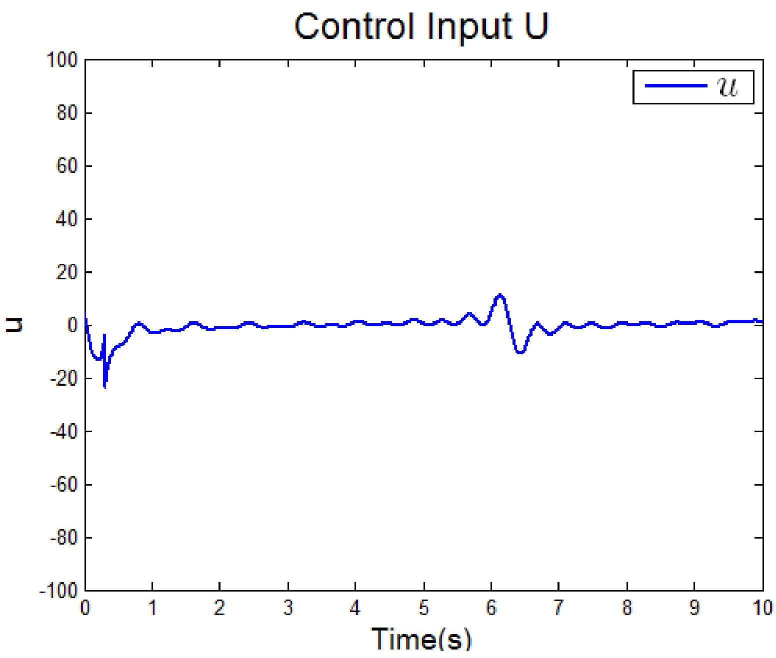

Figure 2-10: Time history of control signal u.

We will show the importance of the initial conditions by simulating the system once more with the initial conditions and parameters as follows

$$k_1 = 5.5, \quad k_2 = 2.5, \quad k_d = 200, \quad \delta_1 = 0.001, \quad \varepsilon_1 = 2.4, \varepsilon_2 = 0.1. \quad \text{(2-223)}$$

The initial conditions are chosen as

$$x_1(t_0) = -0.5, \quad x_2(t_0) = 0, \quad x_{1d} = 1.255t, \quad \dot{x}_{1d} = 1.255, \quad \text{(2-224)}$$

The initial conditions are very similar to the first simulation with the exception of k_1. The disturbance is modeled as:

$$\begin{aligned} d &= \sum_{i=1}^{15} \sin(i * t) + \sin(x_2) \sin\left(i * \frac{t}{2}\right), \\ d_M &= 18.3, \\ d_{1M} &= 16.5180, \end{aligned} \quad \text{(2-225)}$$

Furthermore, the parameters

$$\begin{aligned} c &= \frac{\min\left(k_1, (k_2 - \varepsilon_1), \frac{\gamma_2}{2}\right)}{\max\left(\frac{1}{2}, \frac{\delta_1}{2}\right)} = 0.1915, \\ \lambda &= \frac{\delta_1}{4\varepsilon_2} d_{1M}^2 = 0.0314, \\ \frac{\lambda}{c} &:= 0.1639. \end{aligned} \quad \text{(2-226)}$$

The control gains, initial conditions and disturbance observer parameters have been chosen to be the same as for the previous simulation parameters(page45) except for k_1 which has been chossen to not comply with condition (2-214). The control gain k_1:

$$k_1 = 5.2 > \frac{1.255 + \left\| \tan\left(\sqrt{\left(\frac{\pi}{2} - \sqrt{0.1639}\right)^2 - 0.5 - 18.3^2 * 10^{-3}} \right) \right\|}{0.5} := 5.1. \quad \text{(2-227)}$$

This implies that condition (2-213) is not satisfied. The time history of the states z_1, z_2 and x_2 are shown in Figure 2-11 we can see that the state errors converge to zero furthermore $x_2 \,! < \pi/2$.

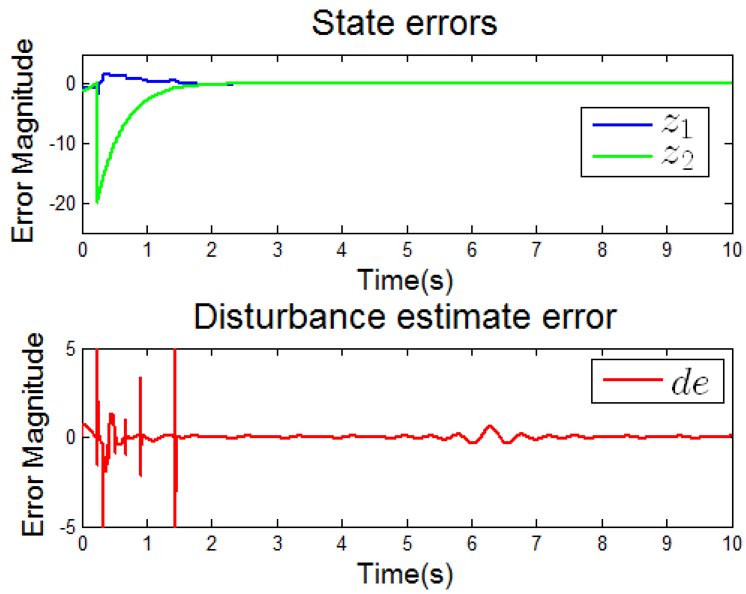

Figure 2-11: Time history of z_1, z_2, x_2 and α_1 for unstable system.

furthermore $\|x_2\| > \pi/2$.:

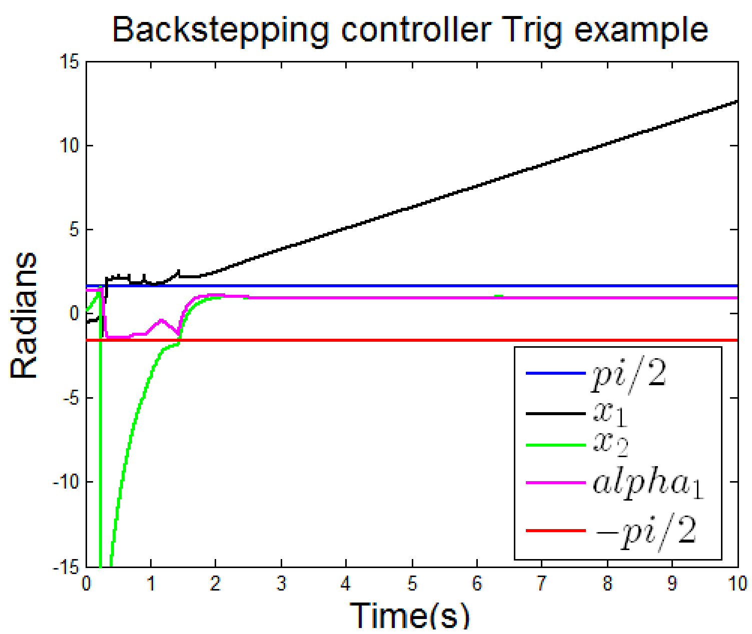

Figure 2-12: Time history of state and virtual control signal. For unstable system.

Furthermore, on inspection of the Lyapunov function V_3 in Figure 2-13we see that while it settles to less than the bound:

$$V_{3,bound} = \left(V_3(t_0) - \frac{\lambda}{c}\right) e^{-c(t-t_0)} + \frac{\lambda}{c}. \tag{2-228}$$

We see that the function does not exponentially converge to a steady solution due to a spike occurring between 0 and 1 second.

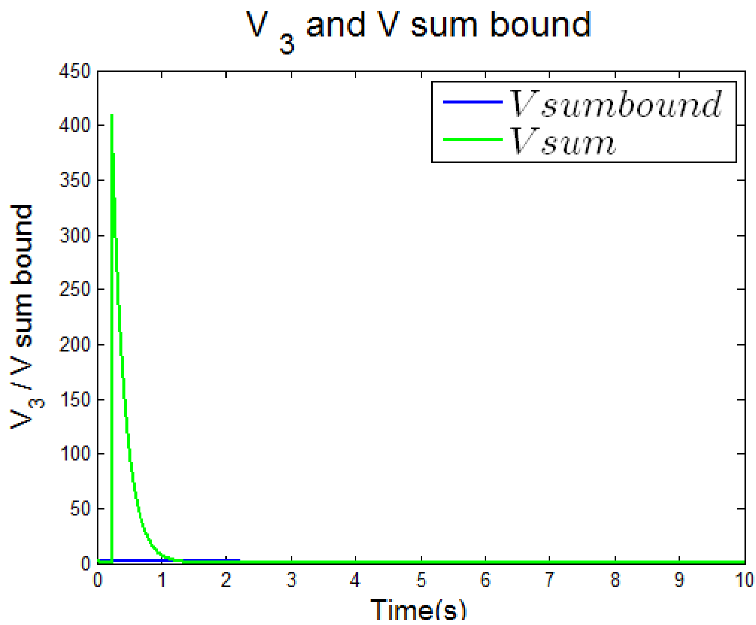

Figure 2-13: Time history of V_3 and $V_{3\ bound}$.

in addition, the control signal u is unbounded as shown below:

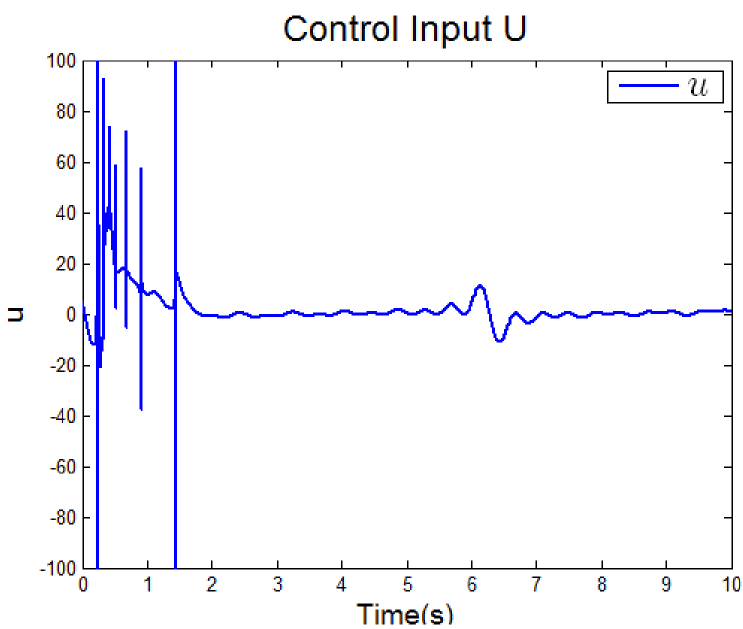

Figure 2-14: Time history of control signal u.

Comparing Figure 2-14 with Figure 2-10 it can be seen that the control input is unbounded at the time that $x_2 = \frac{\pi}{2}$ (see Figure 2-12). Comparing the Lyapunov function V_2 for both situations Figure 2-9 and Figure 2-13 it can be seen that for the case where the control gains meet the condition outlined by (2-214) exponential stability is achieved, while for the case where condition (2-214) is not met exponential stability is not achieved. While the system does eventually stabilise it does so after entering a region outside the range of the controller.

2.8.2 One-step ahead backstepping trig example

We will now present the one-step ahead backstepping method with an example. The system we will use will differ from the previous example in section 2.8.1 as the virtual control signal in that example could be unbounded. This section will use a system where the virtual control signal must remain bounded to highlight the usefulness of the one-step ahead backstepping method. This situation is similar to the design conditions for the quadrotors linear position subsystem. Consider the system:

$$\begin{aligned} \dot{x}_1 &= \sin(x_2), \\ \dot{x}_2 &= -Dx_2 + u + d, \end{aligned} \tag{2-229}$$

the input u of the system (2-229) is designed to stabilize x_1 and x_2 at the origin while compensating for the disturbance d using the one-step ahead backstepping method. This method consists of two steps. In the first step, a virtual control α_1 is derived from the second state, to globally asymptotically and locally exponentially stabilise the origin of x_1. In the second step, the control input u is designed to stabilize the tracking error of the second stage globally asymptotically and locally exponentially at the origin [41]. Furthermore, the following assumptions on the disturbance $d(t)$are made:

Assumption 2.4

1. *The disturbance vector $d(t)$ and its derivative are bounded, i.e., there exist nonnegative constants d_M and d_{1M} such that $\|d(t)\| \leq d_M$ and $\|\dot{d}(t)\| \leq d_{1M}, \forall t \geq t_0 \geq 0$.*

Step 1

Firstly a change of coordinates is introduced by the following coordinate transformation. The error is defined as:

$$x_1 = x_1, \tag{2-230}$$

$$z_2 = x_2 - \alpha_1, \tag{2-231}$$

Where α_1 is the virtual control of x_2, this yields the transformed equation for the first state:

$$\dot{x}_1 = \sin(z_2 + \alpha_1), \tag{2-232}$$

$$=> \dot{x}_1 = \sin(z_2)\cos(\alpha_1) + \sin(\alpha_1)\cos(z_2), \tag{2-233}$$

$$=> \dot{x}_1 = \sin(z_2)\cos(\alpha_1) + (\cos(z_2) - 1)\sin(\alpha_1) + \sin(\alpha_1). \tag{2-234}$$

Let:

$$\Delta = \sin(z_2)\cos(\alpha_1) + (\cos(z_2) - 1)\sin(\alpha_1), \tag{2-235}$$

and substituting (2-235) into (2-230) yields:

$$\dot{x}_1 = \sin(\alpha_1) + \Delta. \tag{2-236}$$

We will analyse the stability of x_1 by using the Lyapunov function candidate:

$$V_1 = \int_0^{x_1} \sigma(s)\, ds, \tag{2-237}$$

where we define the smooth function $\sigma(s)$ as:

$$\sigma(\boldsymbol{s}) = \frac{\boldsymbol{s}}{\sqrt{1 + \|\boldsymbol{s}\|^2}}. \tag{2-238}$$

Taking the derivatives of both sides of (2-237) yields:

$$\dot{V}_1 = \sigma(x_1)\sin(\alpha_2) + \sigma(x_1)\Delta. \tag{2-239}$$

To achieve asymptotic stability for the closed loop system we choose the virtual control as:

$$\sigma(x_1)\sin(\alpha_1) = \sigma(x_1)\left(-k_1\frac{\sigma(x_1)}{\Delta_1(x_2)}\right), \tag{2-240}$$

$$\alpha_1 = \sin^{-1}\left(-k_1\frac{\sigma(x_1)}{\Delta_1(x_2)}\right), \tag{2-241}$$

where we define $\Delta_1(x_2) = 1 + \frac{1}{2}\|x_2\|^2$ and K_1 is a positive constant, to ensure that α_1 is defined $\forall x_1 \in \mathbb{R}$ and $\forall x_2 \in \mathbb{R}$ it is necessary that:

$$\left\| k_1\frac{\sigma(x_1)}{\Delta_1(x_2)} \right\| < 1. \tag{2-242}$$

where, k_1 is a positive constant and is strictly less than or equal to 1, and therefore:

$$\dot{V}_1 = -k_1\frac{\sigma(x_1)^2}{\Delta_1(x_2)} + \sigma(x_1)\Delta. \tag{2-243}$$

The last term of the equation will be considered in Step 2 and in the meantime, is assumed equal to zero. Substituting (2-241) into (2-234) yields:

$$\dot{x}_1 = \Delta - k_1\frac{\sigma(x_1)}{\Delta_1(x_2)}. \tag{2-244}$$

<u>Step 2</u>

We now design the disturbance observer

$$\begin{aligned} \hat{d} &= \xi + k_d x_2, \\ \dot{\xi} &= -k\xi - k_d(u + k_d x_2), \\ \xi(t_0) &= -k_d x_2(t_0), \end{aligned} \tag{2-245}$$

where k is a symmetric and positive constant, based on Assumption 2.4 system (2-245) guarantees that the disturbance observer error $d_e = d - \hat{d}$ and the disturbance observer $\hat{d}$ remain bounded for all $t \geq t_0 \geq 0$. By differentiating both sides of (2-245) along the solution of the second equation of (2-245) we obtain the as

$$\dot{\hat{d}} = k_d d_e, \tag{2-246}$$

this implies that the derivative of the error d_e is:

$$\dot{d}_e = -k_d d_e + \dot{d}. \tag{2-247}$$

By differentiating both sides of (2-231) we obtain:

$$\dot{z}_2 = \dot{x}_2 - \dot{\alpha}_1, \tag{2-248}$$

in addition, using (2-241) we obtain:

$$\dot{\alpha}_1 = \frac{\partial \alpha_1}{\partial x_2}(\dot{x}_2) + \frac{\partial \alpha_1}{\partial x_1}(\dot{x}_1). \tag{2-249}$$

Calculating the derivatives:

$$\frac{\partial \alpha_1}{\partial x_2} = \frac{-k_1 \frac{\sigma(x_1)}{\Delta_1(x_2)^2} x_2}{\sqrt{1 - \left(-k_1 \frac{\sigma(x_1)}{\Delta_1(x_2)}\right)^2}} < \frac{1}{\sqrt{2}}, \quad \frac{\partial \alpha_1}{\partial x_1} = \frac{-k_1 \frac{1}{\Delta_1(x_2)}\left(\frac{1}{(1 + x_1^2)^{1.5}}\right)}{\sqrt{1 - \left(-k_1 \frac{\sigma(x_1)}{\Delta_1(x_2)}\right)^2}} < 1. \tag{2-250}$$

Furthermore, it can be stated that:

$$\left\|\frac{\partial \alpha_1}{\partial x_2}\right\| < \frac{1}{\sqrt{2}}, \quad \left\|\frac{\partial \alpha_1}{\partial x_1}\right\| < 1, \tag{2-251}$$

allowing (2-248) to be rewritten as:

$$\dot{z}_2 = -Dx_2 + u - \frac{\partial \alpha_1}{\partial x_2}(u) + \frac{\partial \alpha_1}{\partial x_1}(\dot{x}_1) + \hat{d} + d_e - \frac{\partial \alpha_1}{\partial x_2}(\hat{d} + d_e). \tag{2-252}$$

To analyse the stability of the origin of x_2 we consider the Lyapunov function:

$$V_2 = \gamma V_1 + \tfrac{1}{2} z_2^2, \tag{2-253}$$

where γ is a positive constant, taking the derivate of both sides of (2-253):

$$\dot{V}_2 = \gamma_1\left(-k_1 \tfrac{\sigma(x_1)^2}{\Delta_1(x_2)} + \sigma(x_1)\Delta\right) + z_2(u + \hat{d} + d_e \pm \dot{\alpha}_2), \tag{2-254}$$

$$\dot{V}_2 = -\gamma_1 \mathrm{K}_1 \tfrac{\sigma(x_1)^2}{\Delta_1(x_2)} + z_2\left(\gamma_1 \frac{\Delta\sigma(x_1)}{z_2} + \left(1 - \frac{\partial \alpha_1}{\partial x_2}\right)\left(-Dx_2 + u + (\hat{d} + d_e)\right) - \dot{\alpha}_2\right), \tag{2-255}$$

$$\dot{V}_2 = -\gamma k_1 \frac{\sigma(x_1)^2}{\Delta_1(x_2)} + z_2\left(\gamma_1 \frac{\Delta\sigma(x_1)}{z_2} + \left(1 - \frac{\partial \alpha_1}{\partial x_2}\right)(u + \hat{d} + d_e) + \frac{\partial \alpha_1}{\partial x_2} Dx_2 - Dz_2 - D\alpha_1 - \frac{\partial \alpha_1}{\partial x_1}(\dot{x}_1)\right). \tag{2-256}$$

Let:

$$\gamma_1 \frac{\Delta\sigma(x_1)}{z_2} + \left(1 - \frac{\partial\alpha_1}{\partial x_2}\right)(u + \hat{d}) + \frac{\partial\alpha_1}{\partial x_2} Dx_2 - D\alpha_1 - \frac{\partial\alpha_1}{\partial x_1}(\dot{x}_1) = -k_2\sigma(z_2) - \frac{k_3(2\alpha_1 + z_2)}{\Delta_1(x_2)}, \tag{2-257}$$

recall that:

$$\Delta = \sin(z_2)\cos(\alpha_2) + (\cos(z_2) - 1)\sin(\alpha_1). \tag{2-258}$$

Now consider the fact:

$$\lim_{z_2 \to 0} \frac{\sin(z_2)}{z_2} = \lim_{z_2 \to 0} \frac{\sin(z_2)'}{{z_2}'} = \lim_{z_2 \to 0} \frac{\cos(z_2)}{1} = 1, \tag{2-259}$$

$$\lim_{z_2 \to 0} \frac{\cos(z_2) - 1}{z_2} = \lim_{z_2 \to 0} \frac{(\cos(z_2) - 1)'}{{z_2}'} = \lim_{z_2 \to 0} \frac{\sin(z_2)}{1} = 0. \tag{2-260}$$

The consequence of the equations above (2-259) and (2-260) is that (2-257) is well defined for all z_2 to achieve global asymptotic stability:

$$\left(1 - \frac{\partial\alpha_1}{\partial x_2}\right)u = \frac{\partial\alpha_1}{\partial x_1}(\dot{x}_1) - k_2\sigma(z_2) - \frac{k_3(2\alpha_1 + z_2)}{\Delta_1(x_2)} - \gamma_1 \frac{\Delta\sigma(x_1)}{z_2} - \frac{\partial\alpha_1}{\partial x_2} Dx_2 + D\alpha_1 - \left(1 - \frac{\partial\alpha_1}{\partial x_2}\right)\hat{d}, \tag{2-261}$$

$$u = \left(1 - \frac{\partial\alpha_1}{\partial x_2}\right)^{-1}\left(\frac{\partial\alpha_1}{\partial x_1}(\dot{x}_1) - k_2\sigma(z_2) - \frac{k_3(2\alpha_1 + z_2)}{\Delta_1(x_2)} - \gamma_1\sigma(x_1)\frac{\Delta}{z_2} - \frac{\partial\alpha_1}{\partial x_2} Dx_2 + D\alpha_1\right) - \hat{d}, \tag{2-262}$$

where k_2 is a positive constant, this implies:

$$\dot{z}_2 = -\gamma_1\sigma(z_1)\frac{\Delta}{z_2} - k_2\sigma(z_2) - Dz_2 - -\frac{k_3(2\alpha_1 + x_2)}{\Delta_1(x_2)}, \tag{2-263}$$

this ensures that:

$$\dot{V}_2 = -\gamma_1 k_1 \frac{\sigma(x_1)^2}{\Delta_1(x_2)} - k_2\sigma(z_2)z_2 - Dz_2^2 - \frac{k_3 z_2(2\alpha_1 + z_2)}{\Delta_1(x_2)} + z_2\left(1 - \frac{\partial\alpha_1}{\partial x_2}\right)d_e, \tag{2-264}$$

Step 3

To deal with the last term of (2-264) we choose the following Lyapunov function.

$$V_3 = V_2 + \tfrac{1}{2}\delta_1 d_e^2. \tag{2-265}$$

Taking the derivative of (2-265) along the solutions of (2-264) and (2-247) yields:

$$\dot{V}_3 = -\gamma_1 k_1 \frac{\sigma(x_1)^2}{\Delta_1(x_2)} - k_2\sigma(z_2)z_2 - Dz_2^2 - \frac{k_3 z_2(2\alpha_1 + z_2)}{\Delta_1(x_2)} + z_2\left(1 - \frac{\partial\alpha_1}{\partial x_2}\right)d_e$$
$$-\delta_1 d_e\left(k_d d_e - \dot{d}\right). \tag{2-266}$$

By applying Young's inequality to (2-266):

$$\dot{V}_3 \leq -\gamma_1 k_1 \frac{\sigma(x_1)^2}{\Delta_1(x_2)} - k_2\sigma(z_2)z_2 - \left(D - \frac{\varepsilon_1}{2}\right)z_2^2 - \frac{k_3 z_2(2\alpha_1 + z_2)}{\Delta_1(x_2)}$$
$$- \delta_1\left(\delta_1 k_d + \delta_1\varepsilon_2 - \frac{1}{\varepsilon_1\delta_1}\left(1 - \frac{\partial\alpha_1}{\partial x_2}\right)^2\right)d_e^2 + \frac{\delta_1}{4\varepsilon_2}\dot{d}^2 \tag{2-267}$$

Through completing the square (2-267) becomes:

$$\dot{V}_3 \leq -\frac{\gamma_1 k_1 - \frac{\delta_1}{4\varepsilon_1}\dot{d}^2(1 + 0.5\alpha_1^2)}{\Delta_1(x_2)(1 + \|x_1\|^2)}\mathrm{x}_1^2 - k_2\sigma(z_2)z_2 - \left(k_3 - \frac{\delta_1}{4\varepsilon_1}\dot{d}^2\right)\frac{z_2(2\alpha_1 + z_2)}{\Delta_1(x_2)}$$
$$-\left(D - \frac{\varepsilon_1}{2}\right)z_2^2 - \left(\delta_1 k_d - \delta_1\varepsilon_2 - \frac{1}{\varepsilon_1}\left(1 - \frac{\partial\alpha_1}{\partial x_2}\right)^2\right)d_e^2 + \frac{\delta_1}{4\varepsilon_1}\frac{\dot{d}^2(1 + 0.5\alpha_1^2)}{\Delta_1(x_2)(1 + x_1^2)}. \tag{2-268}$$

To ensure that (2-241) is well defined, we have to find initial states and control gains such that condition (2-242) holds. The condition can be rewritten as:

$$k_1 < 1. \tag{2-269}$$

The values of $\delta_1, \varepsilon_1\ \varepsilon_2, c$ and γ_1 can mow be chosen such that:

$$\gamma_1 k_1 - \frac{\delta_1}{4\varepsilon_1}\dot{d}^2(1 + 0.5\alpha_1^2) > 0 := b_1, \tag{2-270}$$

$$D - \frac{\varepsilon_1}{2} > 0 := b_2, \tag{2-271}$$

$$\delta_1 k_d - \delta_1\varepsilon_2 - \frac{1}{\varepsilon_1}\left(1 - \frac{\partial\alpha_1}{\partial x_2}\right)^2 > 0 := b_3, \tag{2-272}$$

$$\frac{\delta_1}{4\varepsilon_1}\dot{d}^2(1 + 0.5\alpha_1^2) = b_4, \tag{2-273}$$

$$k_3 - \frac{\delta_1}{4\varepsilon_1}\dot{d}^2 = 0, \tag{2-274}$$

where the symbol γ_2 is a positive constant. As a consequence of the one-step ahead backstepping method the term $\frac{\partial\alpha_1}{\partial x_2}$ in (2-251) is bounded which gives:

$$\dot{V}_3 \leq -b_1\frac{\sigma(x_1)^2}{\Delta_1(x_2)} - k_2\sigma(z_2)z_2 - b_2 z_2^2 - b_3 d_e^2 + \frac{b_4}{\Delta_1(x_2)(1 + x_1^2)}. \tag{2-275}$$

Therefore, it can be said that:

$$\dot{V}_3 \leq -\frac{b_1 x_1^2 - b_4}{\Delta_1(x_2)(1 + x_1^2)} - k_2\sigma(z_2)z_2 - b_2 z_2^2 - b_3 d_e^2, \tag{2-276}$$

letting:

$$c = \min\left(\frac{b_1}{\gamma_1}\frac{1}{\Delta_1(x_2(t_0))(1+x_1^2(t_0))}, \frac{b_2}{2}, \frac{b_3}{2}\right),$$
$$\lambda = \max\left(\frac{b_4}{\Delta_1(x_2(t_0))(1+x_1^2(t_0))}\right), \tag{2-277}$$

This implies that:

$$\dot{V}_3 \le c(z_2^2 + d_e^2) + \lambda. \tag{2-278}$$

Solving the differential inequality (2-278) shows that V_3, (2-275) exponentially converges to a ball of radius $r = \lambda/c$, which implies that the system is forward complete. Furthermore, if the derivative of the disturbance $\dot{d}$ is zero then the closed loop system is globally asymptotically stable.

Hence $\dot{V}_2 < 0\ \forall x_1 \in \mathbb{R}$ and $\forall z_2 \in \mathbb{R}$ furthermore since $V_3 = \gamma V_1 + (\frac{1}{2})z_2^2 + \frac{1}{2}\delta_1 d_e^2$, this implies that V_3 is radically unbounded and hence we have globally asymptotically stabilized the system.

The upper bound of the control input u is deffiend by:

$$u = \left(1 - \frac{\partial\alpha_1}{\partial x_2}\right)^{-1}\left(\frac{\partial\alpha_1}{\partial x_1}(\dot{x}_1) - k_2\sigma(z_2) - \frac{k_3(2\alpha_1 + z_2)}{\Delta_1(x_2)} - \gamma_1\sigma(x_1)\frac{\Delta}{z_2} - \frac{\partial\alpha_1}{\partial x_2}Dx_2 + D\alpha_1\right) - \hat{d} \le u^M, \tag{2-279}$$

Which implies that

$$\frac{\partial\alpha_1}{\partial x_1}(\dot{x}_1) - k_2\sigma(z_2) - \frac{k_3(2\alpha_1 + z_2)}{2\Delta_1(x_2)} - \gamma_1\sigma(x_1)\frac{\Delta}{z_2} - \frac{\partial\alpha_1}{\partial x_2}Dx_2 + D\alpha_1 \le \left(1 - \frac{\partial\alpha_1}{\partial x_2}\right)(u^M - \hat{d}^M), \tag{2-280}$$

Which gives the following upper bound on k_2 as:

$$k_2 \le (1 - k_1)(u^M - \hat{d}^M) - k_1 - 0.3551k_3 - \gamma_1 - 0.5k_1D - \left\|D - \frac{k_3}{2}\right\|\alpha_1^M, \tag{2-281}$$

2.8.2.1 Simulation

This section illustrates the effectiveness of the proposed local tracking controller through a numerical simulation using MATLAB, the code for the following simulation can be found in Appendix B. The system parameters as to control and observer gains are chosen as:

$$k_1 = 0.8, \quad k_2 = 2.9, \quad k_d = 200, \quad \delta_1 = 0.001, \quad \gamma_1 = 1,$$
$$k_3 = 0.0165, \quad \varepsilon_1 = \frac{3.9}{2}D, \quad \varepsilon_2 = 0.1, \quad D = 0.2, \quad u^M = 20, \tag{2-282}$$

The initial conditions are chosen as:

$$x_1(t_0) = -0.5, \quad x_2(t_0) = 0,. \tag{2-283}$$

The disturbance is modeled as:

$$d = \sum_{i=1}^{15} \sin(i * t) + \sin(x_2)\sin\left(i * \frac{t}{2}\right),$$
$$d_M = 18.3,$$
$$d_{1M} = 16.5180. \qquad \text{(2-284)}$$

Furthermore, the parameters:

$$c = \min\left(\frac{b_1}{\gamma_1}\frac{1}{\Delta_1(x_2(t_0))(1+x_1^2(t_0))}, \frac{b_2}{2}, \frac{b_3}{2}\right) = 0.2,$$
$$\lambda = \frac{b_4}{\Delta_1(x_2(t_0))(1+x_1^2(t_0))} = 0.0891,$$
$$\frac{\lambda}{c} := 0.4455. \qquad \text{(2-285)}$$

The control gains, initial conditions, disturbance observer parameters and disturbance profile have been chosen so that conditions(2-242), (2-251), (2-269), (2-270) - (2-274) and (2-281) hold trueThe time history of the states x_1, z_2 and d_e are shown in Figure 2-15 we can clearly see that the state errors converge to zero:

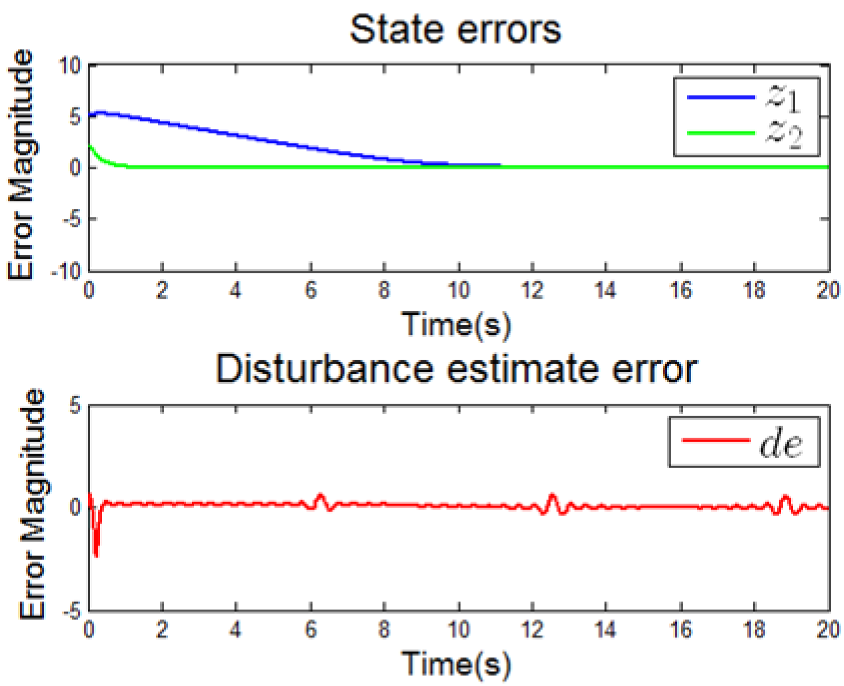

Figure 2-15: Time history of states and disturbance estimate errors.

furthermore $\lim_{t\to\infty}\|x_2(t)\| = 0$.

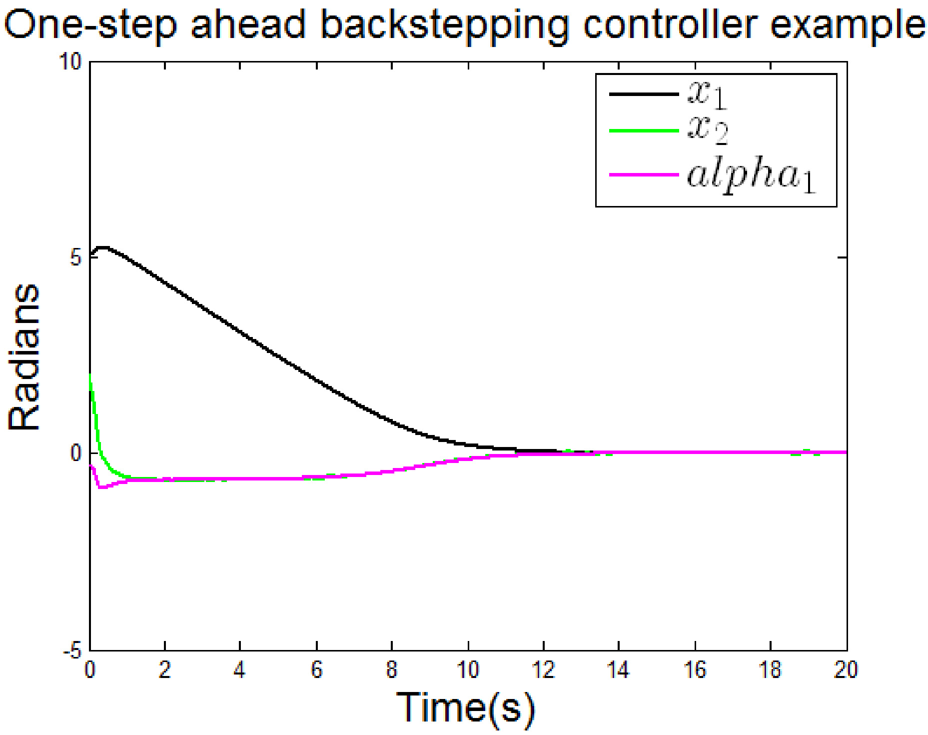

Figure 2-16: Time history of state and virtual control signal.

Furthermore, inspecting the Lyapunov function V_3 in Figure 2-17, it is strictly less than the bound:

$$V_{3,bound} = \left(V_3(t_0) - \frac{\lambda}{c}\right) \mathrm{e}^{-c(t-t_0)} + \frac{\lambda}{c}. \tag{2-286}$$

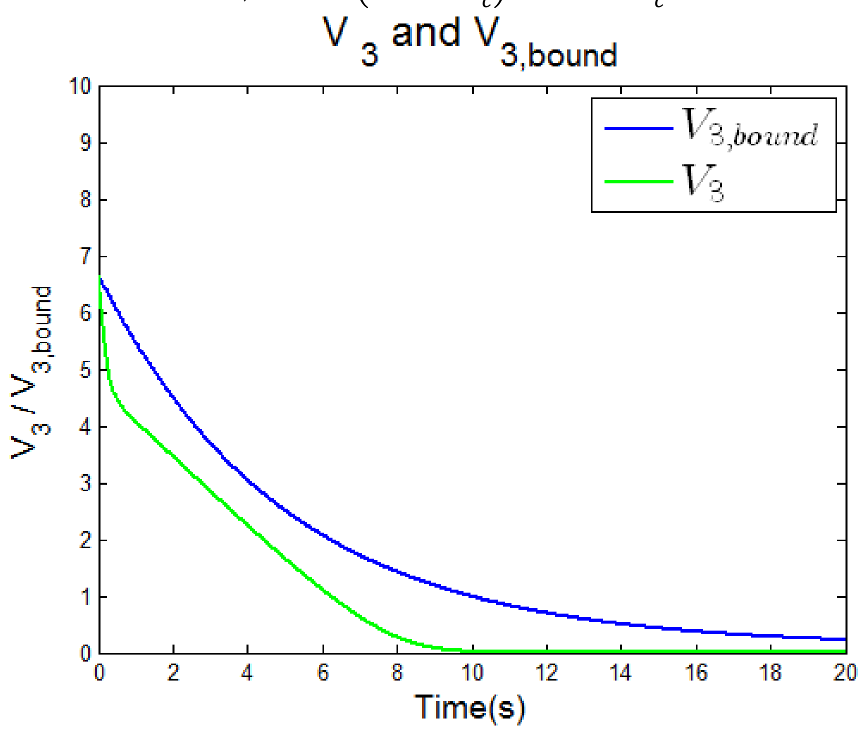

Figure 2-17: Time history of V_3 and $V_{3\ bound}$.

in addition, the control signal u is bounded as shown below:

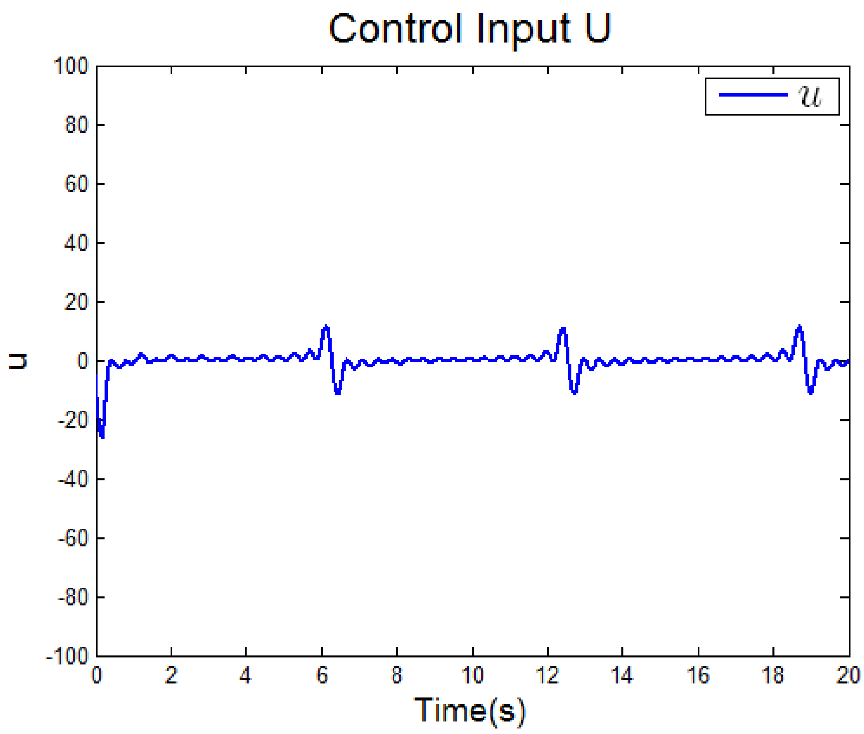

Figure 2-18: Time history of control signal u.

We will show the importance of the initial conditions by simulating the system once more with the initial conditions and parameters as follows

$$k_1 = 3, \qquad k_2 = 2.9, \qquad k_d = 200, \qquad \delta_1 = 0.001, \qquad \gamma_1 = 1,$$
$$k_3 = 0.0165\,, \quad \varepsilon_1 = \frac{3.9}{2}D, \qquad \varepsilon_2 = 0.1, \qquad D = 0.2\,, \quad u^M = 20, \tag{2-287}$$

The initial conditions are chosen as

$$x_1(t_0) = -0.5, \qquad x_2(t_0) = 0,. \tag{2-288}$$

The disturbance is modeled as:

$$d = \sum_{i=1}^{15} \sin(i * t) + \sin(x_2)\sin\left(i * \frac{t}{2}\right),$$
$$d_M = 18.3,$$
$$d_{1M} = 16.5180. \tag{2-289}$$

Furthermore, the parameters:

$$c = \min\left(\frac{\mathrm{b}_1}{\gamma_1}\frac{1}{\Delta_1\big(x_2(t_0)\big)\big(1 + x_1^2(t_0)\big)}, \frac{b_2}{2}, \frac{b_3}{2}\right) = 0.2,$$
$$\lambda = \frac{b_4}{\Delta_1\big(x_2(t_0)\big)\big(1 + x_1^2(t_0)\big)} = 0.0891,$$
$$\frac{\lambda}{c} := 0.4455. \tag{2-290}$$

The difference between this simulation and the previous simulation is the value of the control gain k_1 from 0.8 to 3 which implies that condition (2-269), is no longer true. The time history of the states x_1, z_2 and d_e are shown in Figure 2-19 we can clearly see that the state errors converge to zero:

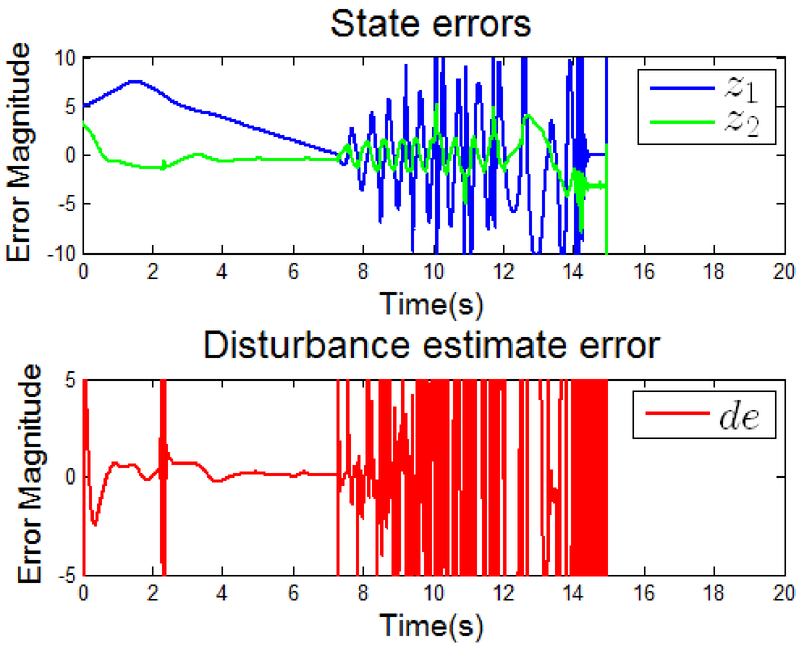

Figure 2-19: Time history of states and disturbance estimate errors.

furthermore $\lim_{t\to\infty} x_2(t) \,! = .$

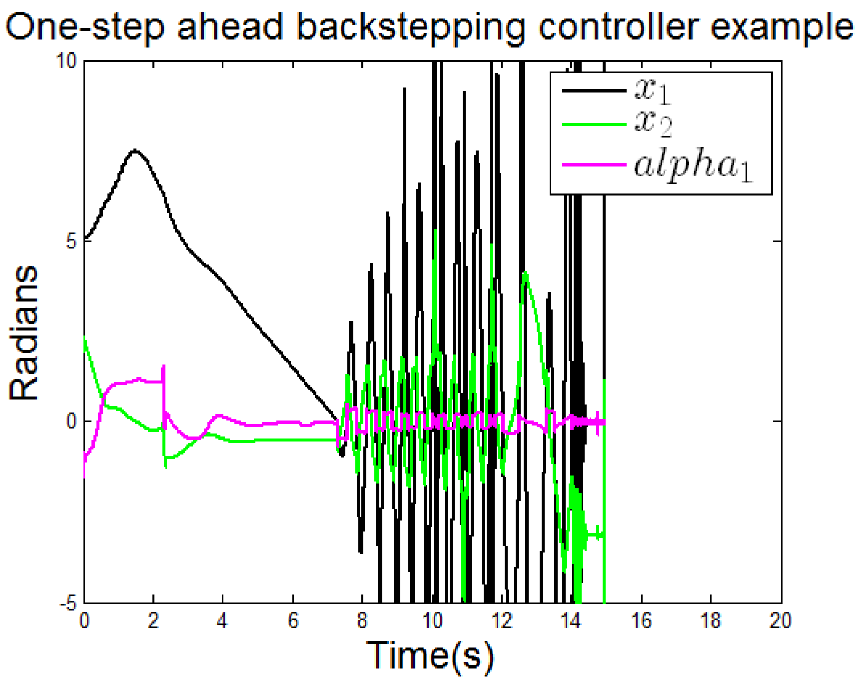

Figure 2-20: Time history of state and virtual control signal.

Furthermore, inspecting the Lyapunov function V_3 in Figure 2-21 Figure 2-17, it is strictly less than the bound:

$$V_{3,bound} = \left(V_3(t_0) - \frac{\lambda}{c}\right) e^{-c(t-t_0)} + \frac{\lambda}{c}. \tag{2-291}$$

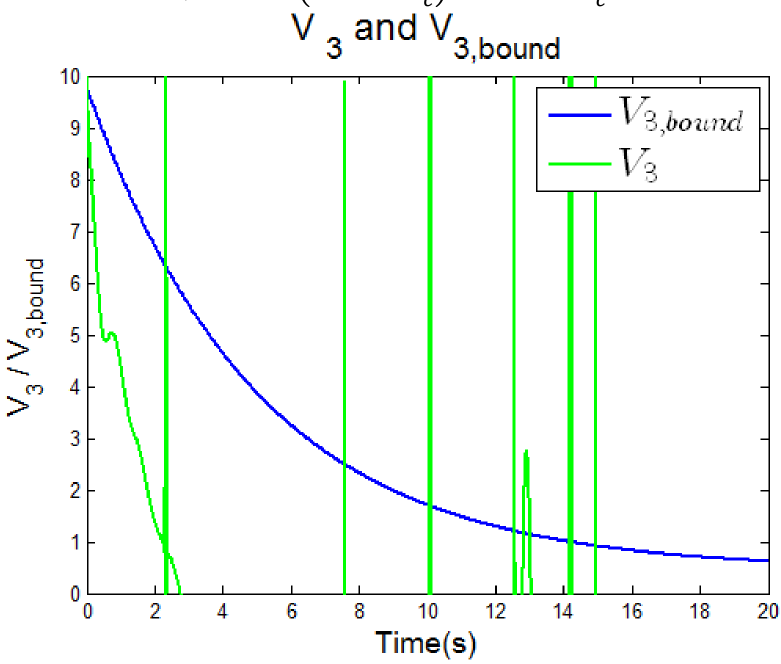

Figure 2-21: Time history of V_3 and $V_{3\ bound}$.

In addition, the control signal u is bounded as shown below:

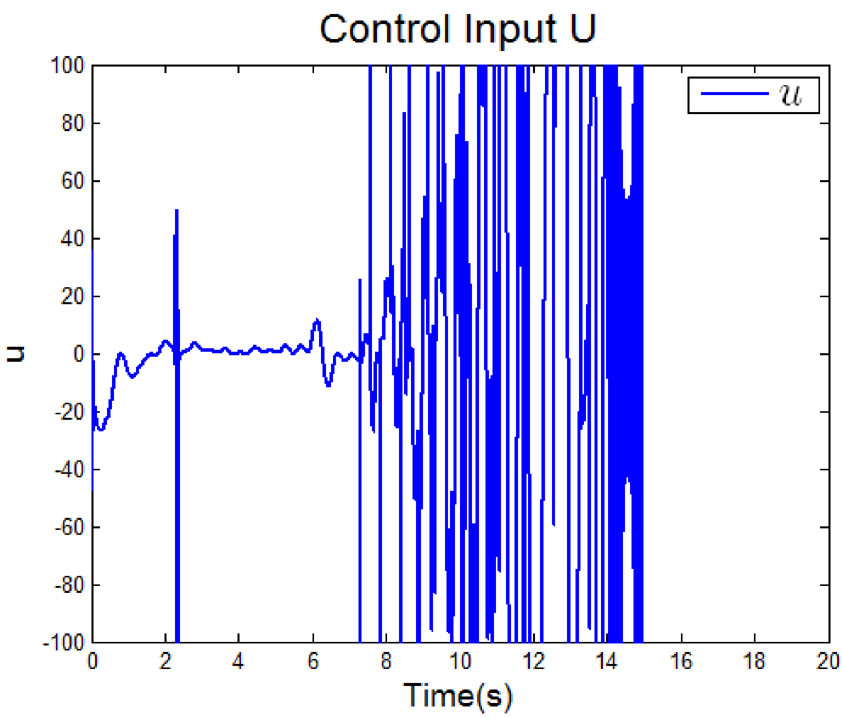

Figure 2-22: Time History of Control Signal u.

Comparing Figure 2-22 with Figure 2-18 it can be seen that the control input is unbounded at the time that $x_2 = -\frac{\pi}{2}$ (see Figure 2-20). Comparing the Lyapunov function V_2 for both situations Figure 2-21 and Figure 2-17 it can be seen that for the case where the control gains meet the condition outlined by (2-269) stability is achieved, while for the case where condition (2-269) is not met stability is not achieved.

3 Attitude Control

In this chapter the control design and stability analysis of the quadrotor aircraft attitude system is presented. This chapter is split up into two separate sections. Firstly, a deterministic system is considered and the control design to asymptotically stabilise the aircraft's attitude is presented. Secondly, a stochastic system is considered and a control algorithm is designed to stochastically asymptotically stabilise the system.

The first section is broken up into three subsections. Firstly, the model assumptions for the deterministic system are presented. Secondly, a control law is presented which employs Lyapunov's direct method and backstepping to design a control law that stabilises the attitude of the aircraft while tracking a reference signal and rejecting disturbances. Finally, the stability analysis of the completed control law is presented, outlining the initial conditions, regions of operation and control gains.

The second section is broken up into three subsections. Firstly, the model assumptions for the stochastic system are presented. Secondly, a control law is presented which employs Lyapunov's direct method and backstepping to design a control law that stabilises the attitude of the aircraft. This control law allows the tracking of a reference signal, rejecting disturbances and overcoming difficulties with Hessian terms introduced by differentiation rule. Finally, the stability analysis of the completed control law is presented, outlining the initial conditions, regions of operation and control gains.

3.1 Deterministic Attitude system

In this section we present the analysis and control of the attitude system of the quadrotor.

We will design a control law for the system described in equation (2-84). For convenience we restate the equations here:

$$s_2 = \begin{cases} \dot{\boldsymbol{\eta}}_2 = \mathbf{T}(\boldsymbol{\eta}_2)\boldsymbol{\omega}, \\ \dot{\boldsymbol{\omega}} = -\mathbf{I}_H^{-1}\boldsymbol{\omega}\times(\mathbf{I}_H\boldsymbol{\omega}) + \mathbf{I}_H^{-1}\boldsymbol{\tau} + \mathbf{I}_H^{-1}\boldsymbol{\tau}_{Aero}, \end{cases} \tag{3-1}$$

where the aerodynamic induced moment $\boldsymbol{\tau}_{Aero}$ is a nonlinear vector function of the aircraft's angle of attack with respect to the oncoming airflow. For the purpose of control design, we will consider $\boldsymbol{\tau}_{Aero}$ to be deterministic in nature. Recall section 2.6 the definition:

$$\mathbf{T}(\boldsymbol{\eta}_2) = \begin{bmatrix} 1 & s_\phi t_\theta & c_\phi t_\theta \\ 0 & c_\phi & -s_\phi \\ 0 & \dfrac{s_\phi}{c_\theta} & \dfrac{c\phi}{c_\theta} \end{bmatrix}, \quad \mathbf{T}(\boldsymbol{\eta}_2)^{-1} = \begin{bmatrix} 1 & 0 & -s_\theta \\ 0 & c_\phi & c_\theta s_\phi \\ 0 & -s_\phi & c_\theta c_\phi \end{bmatrix}, \tag{3-2}$$

and we will define:

$$\dot{\mathbf{T}}(\boldsymbol{\eta}_2)^{-1} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -s_\phi & c_\theta c_\phi \\ 0 & -c_\phi & -c_\theta s_\phi \end{bmatrix}\dot{\phi} + \begin{bmatrix} 0 & 0 & -c_\theta \\ 0 & 0 & -s_\theta s_\phi \\ 0 & 0 & -s_\theta c_\phi \end{bmatrix}\dot{\theta}, \tag{3-3}$$

$$\boldsymbol{\eta}_{2d} = [\phi_d \quad \theta_d \quad \psi_d]^{\mathrm{T}}, \tag{3-4}$$

$$\dot{\boldsymbol{\eta}}_{2d} = [\dot{\phi}_d \quad \dot{\theta}_d \quad \dot{\psi}_d]^{\mathrm{T}}, \tag{3-5}$$

$$\boldsymbol{\alpha}_\omega = [\alpha_p \quad \alpha_q \quad \alpha_r]^{\mathrm{T}}, \tag{3-6}$$

and further define the following:

$$\boldsymbol{\eta}_{2e} = \boldsymbol{\eta}_2 - \boldsymbol{\eta}_{2d}, \tag{3-7}$$

$$\dot{\boldsymbol{\eta}}_{2e} = \mathbf{T}(\boldsymbol{\eta}_2)\boldsymbol{\alpha}_\omega - \dot{\boldsymbol{\eta}}_{2d} + \mathbf{T}(\boldsymbol{\eta}_2)\boldsymbol{\omega}_e, \tag{3-8}$$

$$\boldsymbol{\omega}_e = \boldsymbol{\omega} - \boldsymbol{\alpha}_\omega, \tag{3-9}$$

$$\dot{\boldsymbol{\omega}}_e = -\mathbf{I}_H^{-1}\boldsymbol{\tau} - \mathbf{I}_H^{-1}\boldsymbol{\omega}\times\mathbf{I}_H\boldsymbol{\omega} + \mathbf{I}_H^{-1}(\hat{\boldsymbol{\tau}}_{Aero} + \boldsymbol{\tau}_{de}) - \dot{\boldsymbol{\alpha}}_\omega, \tag{3-10}$$

where $\hat{\boldsymbol{\tau}}_{Aero}$ deontes the estimate of the disturbance torque and $\boldsymbol{\tau}_{de}$ denotes the error between the actual disturbance torque and its estimate.

3.1.1 Model Assumptions

Assumption 3.1

Assume that the reference attitude trajectory $\boldsymbol{\eta}_{2d}(t) = [\phi_d(t) \quad \theta_d(t) \quad \psi_d(t)]^{\mathrm{T}}$ is sufficiently smooth, i.e., the first two derivatives exist and are bounded, there exists non-negative constants ϱ_1 and ϱ_2 such that:

$$\sup_{t\in R^n} \|\dot{\boldsymbol{\eta}}_{2d}(t)\| \le \varrho_1, \sup_{t\in R^n} \|\ddot{\boldsymbol{\eta}}_{2d}(t)\| \le \varrho_2. \tag{3-11}$$

Assumption 3.2

Assume that the reference roll trajectory is bounded by a positive constant ϱ_3, *between* $\pm\pi/2$ *such that:*

$$\sup_{t\in R^n}\|\theta_d(t)\| \leq \varrho_3, \qquad \varrho_3 < \frac{\pi}{2}. \tag{3-12}$$

Assumption 3.3

Assume that the aerodynamic disturbance torque vector $\boldsymbol{\tau}_{Aero}$ *and its derivative are bounded, i.e., there exist nonnegative constants such that:*

$$\sup_{t\in R^n}\|\boldsymbol{\tau}_{Aero}(t)\| \leq \varrho_\tau, \qquad \sup_{t\in R^n}\left\|\dot{\boldsymbol{\tau}}_{Aero,\boldsymbol{\eta}}(t)\right\| \leq \varrho_{1\tau}, \forall t \geq t_0 \geq 0. \tag{3-13}$$

Furthermore, there exists constants $\varrho_{\tau 1}$, $\varrho_{\tau 2}$ *and* $\varrho_{\tau 3}$ *such that:*

$$\sup_{t\in R^n}\left\|\boldsymbol{\tau}_{Aero,\boldsymbol{p}}(t)\right\| \leq \varrho_{\tau 1}, \sup_{t\in R^n}\left\|\boldsymbol{\tau}_{Aero,\boldsymbol{q}}(t)\right\| \leq \varrho_{\tau 2} \sup_{t\in R^n}\left\|\boldsymbol{\tau}_{Aero,\boldsymbol{r}}(t)\right\| \leq \varrho_{\tau 3}, \tag{3-14}$$

And constants $\varrho_{\tau 4}$, $\varrho_{\tau 5}$ *and* $\varrho_{\tau 6}$ *such that:*

$$\sup_{t\in R^n}\left\|\dot{\boldsymbol{\tau}}_{Aero,\boldsymbol{p}}(t)\right\| \leq \varrho_{\tau 4}, \sup_{t\in R^n}\left\|\dot{\boldsymbol{\tau}}_{Aero,\boldsymbol{q}}(t)\right\| \leq \varrho_{\tau 5}, \sup_{t\in R^n}\left\|\dot{\boldsymbol{\tau}}_{Aero,\boldsymbol{r}}(t)\right\| \leq \varrho_{\tau 6}. \tag{3-15}$$

Under the above assumptions, it is possible to design the control input vector $\boldsymbol{\tau} = [U_2 \quad U_3 \quad U_4]$, in such a manner that the angular position $\boldsymbol{\eta}_1(t)$ of the aircraft asymptotically track their reference trajectories $\boldsymbol{\eta}_{1d}(t)$, i.e.,

$$\lim_{t\to\infty}\left(\boldsymbol{\eta}_2(t) - \boldsymbol{\eta}_{1d}(t)\right) = 0. \tag{3-16}$$

Control Objective 3.1

Based on the above assumptions there is two objectives to achieve:

1. Design the control input vector $\boldsymbol{\tau}$ and estimate laws for $\boldsymbol{\tau}_{Aero}$ to force the aircraft's attitude vectors $\boldsymbol{\eta}_2$ to asymptotically track their reference trajectory vectors $\boldsymbol{\eta}_{2d}$.
2. Keep the aircraft's attitude between the range of $\pm\pi/2$.

3.1.2 Attitude System Analysis

Step 1: To analyse the stability of the origin of $\boldsymbol{\eta}_{2e}$ we consider the Lyapunov function candidate as follows:

$$V_1 = \frac{1}{2}\boldsymbol{\eta}_{2e}^{\mathrm{T}}\boldsymbol{\eta}_{2e}. \tag{3-17}$$

Taking the derivative of both sides of (3-17) along the solution of (3-8) gives:

$$\dot{V}_1 = +\boldsymbol{\eta}_{2e}^{\mathrm{T}}(\mathbf{T}(x_3)\boldsymbol{\alpha}_{\omega} - \dot{\boldsymbol{\alpha}}_2) + \boldsymbol{\eta}_{2e}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\eta}_2)\boldsymbol{\omega}_e. \tag{3-18}$$

To stabilise $\boldsymbol{\eta}_{2e}$ and ensures $\dot{V}_1$ is asymptotically stable we will implement the standard backstepping method to design the virtual control signal $\boldsymbol{\alpha}_{\boldsymbol{\omega}}$ as follows:

$$\boldsymbol{\eta}_{2e}^{\mathrm{T}}(\mathbf{T}(\boldsymbol{\eta}_2)\boldsymbol{\alpha}_{\omega} - \dot{\boldsymbol{\eta}}_{2d}) = -\boldsymbol{\eta}_{2e}^{\mathrm{T}}\mathbf{K}_1\boldsymbol{\eta}_{2e}, \tag{3-19}$$

where the matrix $\mathbf{K_1} \in \mathbb{R}^{3\times3}$ is positive definite, $\mathbf{K_1} = diag\left(k_{1,1}, k_{1,2}, k_{1,3}\right)$ where $k_{1,1}$, $k_{1,2}$ and $k_{1,3}$ are positive constants to be defined later. Thus $\boldsymbol{\alpha}_{\omega}$ is defined as follows:

$$\boldsymbol{\alpha}_{\boldsymbol{\omega}} = \mathbf{T}^{-1}(\boldsymbol{\eta}_2)(\dot{\boldsymbol{\eta}}_{2d} - \mathbf{K}_1\boldsymbol{\eta}_{2e}), \tag{3-20}$$

Substituting (3-19) back into (3-18) yields $\dot{V}_1$ redefined as:

$$\dot{V}_1 = -\boldsymbol{\eta}_{2e}^{\mathrm{T}}\mathbf{K_1}\boldsymbol{\eta}_{2e} + \boldsymbol{\eta}_{2e}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\eta}_2)\boldsymbol{\omega}_e. \tag{3-21}$$

Substituting (3-20) into (3-8) yields $\dot{\boldsymbol{\eta}}_{2e}$ being defined as follows:

$$\dot{\boldsymbol{\eta}}_{2e} = -\mathbf{K}_1\boldsymbol{\eta}_{2e} + \mathbf{T}(\boldsymbol{\eta}_2)\boldsymbol{\omega}_e, \tag{3-22}$$

To prepare for the next step we calculate $\dot{\boldsymbol{\alpha}}_{\boldsymbol{\omega}}$ by differentiating both sides of (3-20) along the solution of (3-22) we obtain:

$$\dot{\boldsymbol{\alpha}}_{\boldsymbol{\omega}} = \mathbf{T}^{-1}(\boldsymbol{\eta}_2)(-\mathbf{K}_3\dot{\boldsymbol{\eta}}_{2e} + \ddot{\boldsymbol{\eta}}_{2d}) + \dot{\mathbf{T}}^{-1}(\boldsymbol{\eta}_2)(\dot{\boldsymbol{\eta}}_{2d} - \mathbf{K}_1\boldsymbol{\eta}_{2e}), \tag{3-23}$$

where $\dot{\mathbf{T}}^{-1}(\boldsymbol{\eta}_2)$ is defined in (3-3).

Step 2: Now we will design a control law for the control torque vector $\boldsymbol{\tau}$ to stabilize $\boldsymbol{\omega}_e$ at the origin. If we denote the error between the actual and the observed aerodynamic torque disturbance as

$$\boldsymbol{\tau}_{de} = \boldsymbol{\tau}_{Aero} - \hat{\boldsymbol{\tau}}_{Aero}. \tag{3-24}$$

To analyse the stability of $\boldsymbol{\omega}_e$ at the origin of we consider the Lyapunov function:

$$V_2 = V_1 + \frac{1}{2}\boldsymbol{\omega}_e^{\mathrm{T}}\boldsymbol{\omega}_e. \tag{3-25}$$

Taking the derivative of both sides of (3-25) along the solution of (3-21) and (3-10) gives:

$$\dot{V}_2 = -\boldsymbol{\eta}_{2e}^{\mathrm{T}}\mathbf{K}_3\boldsymbol{\eta}_{2e} + \boldsymbol{\eta}_{2e}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\eta}_2)\boldsymbol{\omega}_e + \boldsymbol{\omega}_e^{\mathrm{T}}\left(\mathbf{I}_H^{-1}\boldsymbol{\tau} - \mathbf{I}_H^{-1}\boldsymbol{\omega}\times\mathbf{I}_H\boldsymbol{\omega} + \mathbf{I}_{\boldsymbol{H}}^{-1}(\hat{\boldsymbol{\tau}}_{Aero} + \boldsymbol{\tau}_{de}) - \dot{\boldsymbol{\alpha}}_{\boldsymbol{\omega}}\right), \tag{3-26}$$

where $\dot{\boldsymbol{\alpha}}_{\boldsymbol{\omega}}$ is defined in (3-23), (3-26) suggests the choosing of:

$$\boldsymbol{\omega}_e^{\mathrm{T}}\left(\mathbf{T}(\boldsymbol{\eta}_2)\boldsymbol{\eta}_{2e} + \mathbf{I}_H^{-1}\boldsymbol{\tau} - \mathbf{I}_H^{-1}\boldsymbol{\omega}\times\mathbf{I}_H\boldsymbol{\omega} + \mathbf{I}_{\boldsymbol{H}}^{-1}(\hat{\boldsymbol{\tau}}_{Aero}) - \dot{\boldsymbol{\alpha}}_{\boldsymbol{\omega}}\right) = -\,\boldsymbol{\omega}_e^{\mathrm{T}}\mathbf{K}_2\boldsymbol{\omega}_e, \tag{3-27}$$

where the matrix $\mathbf{K}_2 \in \mathbb{R}^{3\times3}$, $\mathbf{K}_2 = diag(k_{2,1}, k_{2,2}, k_{2,3})$ where $k_{2,1}$, $k_{2,2}$ and $k_{2,3}$ are positive constants to be defined later. Choosing the control torque input as:

$$\boldsymbol{\tau} = \boldsymbol{I}_H\left(\boldsymbol{I}_H^{-1}\boldsymbol{\omega}\times\boldsymbol{I}_H\boldsymbol{\omega} - \boldsymbol{I}_{\boldsymbol{H}}^{-1}\hat{\boldsymbol{\tau}}_{Aero} - \mathbf{T}(\boldsymbol{\eta}_2)\boldsymbol{\eta}_{2e} + \dot{\boldsymbol{\alpha}}_\omega - \mathbf{K}_2\boldsymbol{\omega}_e\right), \tag{3-28}$$

and substituting (3-28) into (3-10)yields:

$$\dot{\boldsymbol{\omega}}_e = -\mathbf{T}(\boldsymbol{\eta}_2)\boldsymbol{\eta}_{2e} - \mathbf{K}_2\boldsymbol{\omega}_e + \mathbf{I}_{\boldsymbol{H}}^{-1}\boldsymbol{\tau}_{de}. \tag{3-29}$$

Furthermore, substituting (3-28) back into (3-26) gives:

$$\dot{V}_2 = -\boldsymbol{\eta}_{2e}^{\mathrm{T}}\mathbf{K}_1\boldsymbol{\eta}_{2e} - \boldsymbol{\omega}_e^{\mathrm{T}}\mathbf{K}_2\boldsymbol{\omega}_e + \boldsymbol{\omega}_e^{\mathrm{T}}\mathbf{I}_{\boldsymbol{H}}^{-1}\boldsymbol{\tau}_{de}. \tag{3-30}$$

We must now define the disturbance observer, by implementing the disturbance observer presented in (2-124) we obtain:

$$\begin{aligned} \hat{\boldsymbol{\tau}}_{Aero} &= \boldsymbol{\xi}_2 + \mathbf{K}_{2d}\mathbf{I}_{\boldsymbol{H}}\boldsymbol{\omega}_{\boldsymbol{e}}, \\ \dot{\boldsymbol{\xi}}_2 &= \mathbf{K}_{2d}\mathbf{I}_{\boldsymbol{H}}(\mathbf{T}(\boldsymbol{\eta}_2)\boldsymbol{\eta}_{2e} + \mathbf{K}_4\boldsymbol{\omega}_e), \\ \boldsymbol{\xi}_2(t_0) &= -\mathbf{K}_{2d}\boldsymbol{\omega}_{\boldsymbol{e}}(t_0), \end{aligned} \tag{3-31}$$

where the matrix $\mathbf{K}_\mathrm{d} \in \mathbb{R}^{3\times3}$ is positive definite, $\mathbf{K}_\mathrm{d} = diag(k_{d,1}, k_{d,2}, k_{d,3})$ where $k_{d,1}$, $k_{d,2}$ and $k_{d,3}$ are positive constants to be defined later. Due to the selection of $\boldsymbol{\xi}_2$ in (3-31) and taking the total derivative of both side of (3-24) gives:

$$\dot{\boldsymbol{\tau}}_{de} = -\mathbf{K}_{2d}\boldsymbol{\tau}_{de} + \dot{\boldsymbol{\tau}}_{Aero}. \tag{3-32}$$

Furthermore, we can state:

$$\begin{aligned} \|\boldsymbol{\tau}_{de}\| &\le \sqrt{\left(\|\boldsymbol{\tau}_{de}(t_0)\|^2 - \frac{\tau_{1dN}^2}{2\lambda_m^2(\mathbf{K}_d)}\right)e^{-\lambda_m(\mathbf{K}_{2d})(t-t_0)} + \frac{\tau_{1dN}^2}{2\lambda_m^2\mathbf{K}_d}} := \varrho_{3\tau}, \\ \|\,\hat{\boldsymbol{\tau}}_{Aero}\| &\le \frac{\lambda_M(\mathbf{K}_d)}{\lambda_m(\mathbf{K}_d)}\tau_{dM} := \varrho_{4\tau}. \end{aligned} \tag{3-33}$$

Finally consider the Lyapunov function candidate

$$V_{sum} = V_2 + \tfrac{1}{2}\delta_1\boldsymbol{\tau}_{de}^{\mathrm{T}}\boldsymbol{\tau}_{de}. \tag{3-34}$$

Whose derivative along the solutions of (3-30) and (3-32) is:

$$\dot{V}_{sum} = -\boldsymbol{\eta}_{2e}^{\mathrm{T}}\mathbf{K}_1\boldsymbol{\eta}_{2e} - \boldsymbol{\omega}_e^{\mathrm{T}}\mathbf{K}_2\boldsymbol{\omega}_e + \boldsymbol{\omega}_e^{\mathrm{T}}\mathbf{I}_{\boldsymbol{H}}^{-1}\boldsymbol{\tau}_{de} - \delta_1\boldsymbol{\tau}_{de}^{\mathrm{T}}\mathbf{K}_{2d}\boldsymbol{\tau}_{de} + \delta_1\boldsymbol{\tau}_{de}^{\mathrm{T}}\dot{\boldsymbol{\tau}}_{Aero}. \tag{3-35}$$

With the application of Young's inequality, it can be stated:

$$\boldsymbol{\omega}_e^{\mathrm{T}}\mathbf{I}_{\boldsymbol{H}}^{-1}\boldsymbol{\tau}_{de} \le \frac{\varepsilon_1}{2}\|\boldsymbol{\omega}_e\|^2 + \frac{\left\|\mathbf{I}_{\boldsymbol{H}}^{-1}\right\|^2}{2\varepsilon_1}\|\boldsymbol{\tau}_{de}\|^2, \tag{3-36}$$

$$\delta_1\boldsymbol{\tau}_{de}^{\mathrm{T}}\dot{\boldsymbol{\tau}}_{Aero} \le \delta_1\frac{\varepsilon_2}{2}\|\boldsymbol{\tau}_{de}\|^2 + \frac{\delta_1}{2\varepsilon_2}\varrho_{1\tau}^2, \tag{3-37}$$

where $\varrho_{1\tau}$ is defined in (3-13) thus we can state that:

$$\dot{V}_{sum} \le -\boldsymbol{\eta}_{2e}^{\mathrm{T}}\mathbf{K_1}\boldsymbol{\eta}_{2e} - \left(\lambda_m(\mathbf{K}_2) - \frac{\varepsilon_1}{2}\right)\|\boldsymbol{\omega}_e\|^2 - \delta_1\left(\lambda_m(\mathbf{K}_{2d}) - \frac{\left\|\mathbf{I}_{\boldsymbol{H}}^{-1}\right\|^2}{2\delta_1\varepsilon_1} - \frac{\varepsilon_2}{2}\right)\|\boldsymbol{\tau}_{de}\|^2 + \frac{\delta_1}{2\varepsilon_2}\varrho_{1\tau}^2. \tag{3-38}$$

we choose δ_1, ε_1 and ε_2 such:

$$\lambda_m(\mathbf{K}_2) - \frac{\varepsilon_1}{2} > 0, \tag{3-39}$$

$$\lambda_m(\mathbf{K}_{2d}) - \frac{\left\|\mathbf{I}_{\boldsymbol{H}}^{-1}\right\|^2}{2\delta_1\varepsilon_1} - \frac{\varepsilon_2}{2} > 0. \tag{3-40}$$

3.1.3 Stability Analysis and Initial Conditions and Control Gain Selection

From (3-34) and (3-38) we can state

$$V_{sum} = \tfrac{1}{2}\boldsymbol{\eta}_{2e}^{\mathrm{T}}\boldsymbol{\eta}_{2e} + \tfrac{1}{2}\boldsymbol{\omega}_{e}^{\mathrm{T}}\boldsymbol{\omega}_{e} + \tfrac{1}{2}\delta_1\boldsymbol{\tau}_{de}^{\mathrm{T}}\boldsymbol{\tau}_{de}, \tag{3-41}$$

$$\dot{V}_{sum} \le -\boldsymbol{\eta}_{2e}^{\mathrm{T}}\mathbf{K_1}\boldsymbol{\eta}_{2e} - \left(\lambda_m(\mathbf{K}_2) - \frac{\varepsilon_1}{2}\right)\|\boldsymbol{\omega}_e\|^2 - \delta_1\left(\lambda_m(\mathbf{K}_{2d}) - \frac{\left\|\mathbf{I}_H^{-1}\right\|^2}{2\delta_1\varepsilon_1} - \frac{\varepsilon_2}{2}\right)\|\boldsymbol{\tau}_{de}\|^2 + \frac{\delta_1}{2\varepsilon_2}\varrho_{1\tau}^2 \tag{3-42}$$

Where we have chosen:

$$\dot{\boldsymbol{\eta}}_{2e} = -\mathbf{K}_3\boldsymbol{\eta}_{2e} - \dot{\boldsymbol{\eta}}_{2d} + \mathbf{T}(\boldsymbol{\eta}_2)\boldsymbol{\omega}_e, \tag{3-43}$$
$$\dot{\boldsymbol{\omega}}_e = -\mathbf{T}(\boldsymbol{\eta}_2)\boldsymbol{\eta}_{2e} - \mathbf{K}_2\boldsymbol{\omega}_e + \mathbf{I}_{\mathrm{H}}^{-1}\boldsymbol{\tau}_{de}, \tag{3-44}$$
$$\boldsymbol{\alpha}_\omega = \mathbf{T}^{-1}(\boldsymbol{\eta}_2)(\dot{\boldsymbol{\eta}}_{2d} - \mathbf{K}_3\boldsymbol{\eta}_{2e}), \tag{3-45}$$
$$\dot{\boldsymbol{\alpha}}_\omega = \dot{\mathbf{T}}^{-1}(\boldsymbol{\eta}_2)(\dot{\boldsymbol{\eta}}_{2d} - \mathbf{K}_3\boldsymbol{\eta}_{2e}) + \mathbf{T}^{-1}(\boldsymbol{\eta}_2)\big(\ddot{\boldsymbol{\eta}}_{2d} - \mathbf{K}_3(-\mathbf{K}_3\boldsymbol{\eta}_{2e} - \dot{\boldsymbol{\eta}}_{2d} + \mathbf{T}(\boldsymbol{\eta}_2)\boldsymbol{\omega}_e\,)\big). \tag{3-46}$$

Furthermore, we can state that:

$$\dot{V}_{sum} \le -c(V_{sum}) + \lambda, \tag{3-47}$$

where:

$$c = \frac{\min\left(\lambda_m(\mathbf{K_1}), \left(\lambda_m(\mathbf{K}_2) - \frac{\varepsilon_1}{2}\right), \delta_1\left(\lambda_m(\mathbf{K}_{2d}) - \frac{\left\|\mathbf{I}_H^{-1}\right\|^2}{2\delta_1\varepsilon_1} - \frac{\varepsilon_2}{2}\right)\right)}{\max\left(\frac{1}{2}, \frac{1}{2}\delta_1\right)},$$
$$\lambda = \frac{\delta_1}{2\varepsilon_2}\frac{\delta_1}{2\varepsilon_2}\varrho_{1\tau}^2. \tag{3-48}$$

Hence we can now say:

$$V_{sum}(t) \le \left(V_{sum}(t_0) - \frac{\lambda}{c}\right)\mathrm{e}^{-c(t-t_0)} + \frac{\lambda}{c}. \tag{3-49}$$

Furthermore, in accordance with Theorem 2.1 we can conclude that the system is locally asymptotically stable, due to the singularity induced by the use of Euler angles global asymptotic stability is not achieved as per the conditions for global asymptotic stability per Theorem 2.2 are not met.

Moreover, in accordance with Theorem 2.3 we can conclude that the system is locally exponentially stable.

Therefore, we can state:

$$\sqrt{\|\boldsymbol{\eta}_{2e}(t)\|^2 + \|\boldsymbol{\omega}_e(t)\|^2 + \delta_1\|\boldsymbol{\tau}_{de}(t)\|^2} \le \sqrt{\|\boldsymbol{\eta}_{2e}(t_0)\|^2 + \|\boldsymbol{\omega}_e(t_0)\|^2 + \delta_1\|\boldsymbol{\tau}_{de}(t_0)\|^2} + \sqrt{\frac{\lambda}{c}} := \sqrt{\Omega_0}, \tag{3-50}$$

this implies that:

$$\|\boldsymbol{\eta}_{2e}(t)\| \leq \sqrt{\Omega_0}, \qquad \|\boldsymbol{\omega}_e(t)\| \leq \sqrt{\Omega_0},, \qquad \|\boldsymbol{\tau}_{de}(t)\| \sqrt{\frac{\Omega_0}{\delta_2}}, \tag{3-51}$$

we can state that the attitude errors obey the following:

$$\|\phi_e(t)\| \leq \|\boldsymbol{\eta}_{2e}(t)\|, \qquad \|\phi_e(t)\| < \frac{\pi}{2}, \tag{3-52}$$
$$\|\theta_e(t)\| \leq \|\boldsymbol{\eta}_{2e}(t)\|, \qquad \|\theta_e(t)\| < \frac{\pi}{2}, \tag{3-53}$$
$$\|\psi_e(t)\| \leq \|\boldsymbol{\eta}_{2e}(t)\|, \qquad \|\psi_e(t)\| < \pi. \tag{3-54}$$

Furthermore, the total angular position error obeys:

$$\|\boldsymbol{\eta}_{2e}(t)\| \leq \|\boldsymbol{\eta}_{2e}(t), \boldsymbol{\omega}_e(t), \delta_1\boldsymbol{\tau}_{de}(t)\| + \sqrt{\frac{\lambda}{c}}. \tag{3-55}$$

Furthermore, we can state:

$$\alpha_p = \dot{\phi}_d(t) - \mathbf{K}_{1,11}\phi_e(t) - \mathrm{s}_\theta\left(\dot{\psi}_d(t) - \mathbf{K}_{1,33}\psi_e(t)\right), \tag{3-56}$$
$$\alpha_q = \mathrm{c}_\phi\left(\dot{\theta}_d(t) - \mathbf{K}_{1,22}\theta_e(t)\right) + \mathrm{c}_\theta\mathrm{s}_\phi\left(\dot{\psi}_d(t) - \mathbf{K}_{1,33}\psi_e(t)\right), \tag{3-57}$$
$$\alpha_r = -\mathrm{s}_\phi\left(\dot{\theta}_d(t) - \mathbf{K}_{1,22}\theta_e(t)\right) + \mathrm{c}_\theta\mathrm{c}_\phi\left(\dot{\psi}_d(t) - \mathbf{K}_{1,33}\psi_e(t)\right) \tag{3-58}$$

Since $\|\mathrm{c}_\phi\| \leq 1$, $\|\mathrm{s}_\phi\| \leq 1$, $\|\mathrm{c}_\theta\| \leq 1$ and $\|\mathrm{s}_\theta\| \leq 1$ then (3-56), (3-57) and (3-58) become:

$$\alpha_p \leq \dot{\phi}_d(t) + \mathbf{K}_{1,11}\phi_e(t) + \left(\dot{\psi}_d(t) + \mathbf{K}_{1,33}\psi_e(t)\right), \tag{3-59}$$
$$\alpha_q \leq \left(\dot{\theta}_d(t) + \mathbf{K}_{1,22}\theta_e(t)\right) + \left(\dot{\psi}_d(t) + \mathbf{K}_{1,33}\psi_e(t)\right), \tag{3-60}$$
$$\alpha_r \leq \left(\dot{\theta}_d(t) + \mathbf{K}_{1,22}\theta_e(t)\right) + \left(\dot{\psi}_d(t) + \mathbf{K}_{1,33}\psi_e(t)\right). \tag{3-61}$$

By using (3-50) - (3-55) we can write the following:

$$\left\|\begin{matrix} p(t_0) - \left(\dot{\phi}_d(t_0) - \mathbf{K}_{1,11}\phi_e(t_0) - \mathrm{s}_{\theta(t_0)}\left(\dot{\psi}_d(t_0) - \mathbf{K}_{1,33}\psi_e(t_0)\right)\right) \\ q(t_0) - \left(\mathrm{c}_{\phi(t_0)}\left(\dot{\theta}_d(t_0) - \mathbf{K}_{1,22}\theta_e(t_0)\right) + \mathrm{c}_{\theta(t_0)}\mathrm{s}_{\phi(t_0)}\left(\dot{\psi}_d(t_0) - \mathbf{K}_{1,33}\psi_e(t_0)\right)\right) \\ r(t_0) - \left(-\mathrm{s}_{\phi(t_0)}\left(\dot{\theta}_d(t_0) - \mathbf{K}_{1,22}\theta_e(t_0)\right) + \mathrm{c}_{\theta(t_0)}\mathrm{c}_{\phi(t_0)}\left(\dot{\psi}_d(t_0) - \mathbf{K}_{1,33}\psi_e(t_0)\right)\right)\end{matrix}\right\|^2 < \left(\left\|\begin{matrix}\frac{\pi}{2} \\ \frac{\pi}{2} \\ \pi\end{matrix}\right\| - \sqrt{\frac{\lambda}{c}}\right)^2 - \left\|\begin{matrix}\phi_e(t_0) \\ \theta_e(t_0) \\ \psi_e(t_0)\end{matrix}\right\|^2 - \delta_1\left\|\begin{matrix}\boldsymbol{\tau}_{de,p}(t_0) \\ \boldsymbol{\tau}_{de,q}(t_0) \\ \boldsymbol{\tau}_{de,r}(t_0)\end{matrix}\right\|^2, \tag{3-62}$$

hence we can state:

$$\left\|p(t_0)-\left(\dot{\phi}_d(t_0)-\mathbf{K}_{1,11}\phi_e(t_0)-s_{\theta(t_0)}\left(\dot{\psi}_d(t_0)-\mathbf{K}_{1,33}\psi_e(t_0)\right)\right)\right\|^2 \\ +\left\|q(t_0)-\left(c_{\phi(t_0)}\left(\dot{\theta}_d(t_0)-\mathbf{K}_{1,22}\theta_e(t_0)\right)+c_{\theta(t_0)}s_{\phi(t_0)}\left(\dot{\psi}_d(t_0)-\mathbf{K}_{1,33}\psi_e(t_0)\right)\right)\right\|^2 \\ +\left\|r(t_0)-\left(-s_{\phi(t_0)}\left(\dot{\theta}_d(t_0)-\mathbf{K}_{1,22}\theta_e(t_0)\right)+c_{\theta(t_0)}c_{\phi(t_0)}\left(\dot{\psi}_d(t_0)-\mathbf{K}_{1,33}\psi_e(t_0)\right)\right)\right\|^2 \\ <\left(\frac{\pi}{2}-\sqrt{\frac{\lambda}{c}}\right)^2-\|\phi_e(t_0)\|^2-\|\theta_e(t_0)\|-\|\psi_e(t_0)\|^2-\delta_1\left\|\boldsymbol{\tau}_{de,p}(t_0)\right\|^2-\delta_1\left\|\boldsymbol{\tau}_{de,q}(t_0)\right\|^2 \\ -\delta_1\left\|\boldsymbol{\tau}_{de,r}(t_0)\right\|^2, \tag{3-63}$$

where $\boldsymbol{\tau}_{de,p}$, $\boldsymbol{\tau}_{de,p}$ and $\boldsymbol{\tau}_{de,p}$ repreesnet the "aerodynamic disturbance torque errors" about the aircraft's body fixed *frame.* From (3-62) we are able to obtain the following:

$$\left\|p(t_0)-\left(\dot{\phi}_d(t_0)-\mathbf{K}_{1,11}\phi_e(t_0)-s_{\theta(t_0)}\left(\dot{\psi}_d(t_0)-\mathbf{K}_{1,33}\psi_e(t_0)\right)\right)\right\| \\ <\sqrt{\left(\frac{\pi}{2}-\sqrt{\frac{\lambda}{c}}\right)^2-\|\phi_e(t_0)\|^2-\delta_1\left\|\boldsymbol{\tau}_{de,p}(t_0)\right\|^2}, \tag{3-64}$$

$$\left\|q(t_0)-\left(c_{\phi(t_0)}\left(\dot{\theta}_d(t_0)-\mathbf{K}_{1,22}\theta_e(t_0)\right)+c_{\theta(t_0)}s_{\phi(t_0)}\left(\dot{\psi}_d(t_0)-\mathbf{K}_{1,33}\psi_e(t_0)\right)\right)\right\| \\ <\sqrt{\left(\frac{\pi}{2}-\sqrt{\frac{\lambda}{c}}\right)^2-\|\theta_e(t_0)\|^2-\delta_1\left\|\boldsymbol{\tau}_{de,q}(t_0)\right\|^2}, \tag{3-65}$$

$$\left\|r(t_0)-\left(-s_{\phi(t_0)}\left(\dot{\theta}_d(t_0)-\mathbf{K}_{1,22}\theta_e(t_0)\right)+c_{\theta(t_0)}c_{\phi(t_0)}\left(\dot{\psi}_d(t_0)-\mathbf{K}_{1,33}\psi_e(t_0)\right)\right)\right\| \\ <\sqrt{\left(\frac{\pi}{2}-\sqrt{\frac{\lambda}{c}}\right)^2-\|\psi_e(t_0)\|^2-\delta_1\left\|\boldsymbol{\tau}_{de,r}(t_0)\right\|^2}. \tag{3-66}$$

By rearranging equations (3-64), (3-65) and (3-66) we obtain the following:

$$\left\|\mathbf{K}_{1,11}\phi_e(t_0)-s_{\theta(t_0)}\mathbf{K}_{1,33}\psi_e(t_0)\right\| <\||p(t_0)-\dot{\phi}_d(t_0)+s_{\theta(t_0)}\dot{\psi}_d(t_0)| \\ +\sqrt{\left(\frac{\pi}{2}-\sqrt{\frac{\lambda}{c}}\right)^2-\|\phi_e(t_0)\|^2-\delta_1\left\|\boldsymbol{\tau}_{de,p}(t_0)\right\|^2}, \tag{3-67}$$

$$\left\|c_{\phi(t_0)}\mathbf{K}_{1,22}\theta_e(t_0)+c_{\theta(t_0)}s_{\phi(t_0)}\mathbf{K}_{1,33}\psi_e(t_0)\right\| \\ <\left\|q(t_0)-c_{\phi(t_0)}\dot{\theta}_d(t_0)-c_{\theta(t_0)}s_{\phi(t_0)}\dot{\psi}_d(t_0)\right\| \\ +\sqrt{\left(\frac{\pi}{2}-\sqrt{\frac{\lambda}{c}}\right)^2-\|\theta_e(t_0)\|^2-\delta_1\left\|\boldsymbol{\tau}_{de,q}(t_0)\right\|^2}, \tag{3-68}$$

$$\left\|-s_{\phi(t_0)}\mathbf{K}_{1,22}\theta_e(t_0)+c_{\theta(t_0)}c_{\phi(t_0)}\mathbf{K}_{1,33}\psi_e(t_0)\right\| \\ <\left\|r(t_0)+s_{\phi(t_0)}\dot{\theta}_d(t_0)+c_{\theta(t_0)}c_{\phi(t_0)}\dot{\psi}_d(t_0)\right\| \\ +\sqrt{\left(\frac{\pi}{2}-\sqrt{\frac{\lambda}{c}}\right)^2-\|\psi_e(t_0)\|^2-\delta_1\left\|\boldsymbol{\tau}_{de,r}(t_0)\right\|^2}. \tag{3-69}$$

To ensure that (3-67), (3-68) and (3-69) are well defined we need to choose the initial pitch, roll and yaw, errors as follows:

$$\|\phi_e(t_0)\| \leq \sqrt{\left(\frac{\pi}{2} - \sqrt{\frac{\lambda}{c}}\right)^2 - \delta_1 \left\|\boldsymbol{\tau}_{de,p}(t_0)\right\|^2}, \tag{3-70}$$

$$\|\theta_e(t_0)\| \leq \sqrt{\left(\frac{\pi}{2} - \sqrt{\frac{\lambda}{c}}\right)^2 - \delta_1 \left\|\boldsymbol{\tau}_{de,q}(t_0)\right\|^2}, \tag{3-71}$$

$$\|\psi_e(t_0)\| \leq \sqrt{\left(\frac{\pi}{2} - \sqrt{\frac{\lambda}{c}}\right)^2 - \delta_1 \left\|\boldsymbol{\tau}_{de,r}(t_0)\right\|^2}, \tag{3-72}$$

By rearranging (3-67), (3-68) and (3-69) we can explicitly obtain the bounds for the non-zero elements of the gain matrix $\mathbf{K}_1$ as follows:

$$\begin{aligned}\mathbf{K}_{1,22} < \Bigg(&\frac{1}{\left(\left\|\mathrm{c}_{\phi(t_0)}\theta_e(t_0)\right\|\left\|\mathrm{c}_{\theta(t_0)}\mathrm{c}_{\phi(t_0)}\psi_e(t_0)\right\| - \left\|\mathrm{s}_{\phi(t_0)}\theta_e(t_0)\right\|\left\|\mathrm{c}_{\theta(t_0)}\mathrm{s}_{\phi(t_0)}\psi_e(t_0)\right\|\right)}\Bigg) \\ &* \Bigg(\Bigg(\left\|r(t_0) + \mathrm{s}_{\phi(t_0)}\dot{\theta}_d(t_0) + \mathrm{c}_{\theta(t_0)}\mathrm{c}_{\phi(t_0)}\dot{\psi}_d(t_0)\right\| \\ &+ \sqrt{\left(\frac{\pi}{2} - \sqrt{\frac{\lambda}{c}}\right)^2 - \|\psi_e(t_0)\|^2 - \delta_1\left\|\boldsymbol{\tau}_{de,r}(t_0)\right\|^2}\Bigg)\left\|\mathrm{c}_{\theta(t_0)}\mathrm{s}_{\phi(t_0)}\psi_e(t_0)\right\| \\ &+ \left\|\mathrm{c}_{\theta(t_0)}\mathrm{c}_{\phi(t_0)}\psi_e(t_0)\right\|\left\|q(t_0) - \mathrm{c}_{\phi(t_0)}\dot{\theta}_d(t_0) - \mathrm{c}_{\theta(t_0)}\mathrm{s}_{\phi(t_0)}\dot{\psi}_d(t_0)\right\| \\ &+ \left\|\mathrm{c}_{\theta(t_0)}\mathrm{c}_{\phi(t_0)}\psi_e(t_0)\right\|\sqrt{\left(\frac{\pi}{2} - \sqrt{\frac{\lambda}{c}}\right)^2 - \|\theta_e(t_0)\|^2 - \delta_1\left\|\boldsymbol{\tau}_{de,q}(t_0)\right\|^2}\Bigg),\end{aligned} \tag{3-73}$$

$$\begin{aligned}\mathbf{K}_{1,33} < (\mathbf{K}_{1,22})&\frac{\left\|\mathrm{s}_{\phi(t_0)}\theta_e(t_0)\right\|}{\left\|\mathrm{c}_{\theta(t_0)}\mathrm{c}_{\phi(t_0)}\psi_e(t_0)\right\|} + \frac{\left\|r(t_0) + \mathrm{s}_{\phi(t_0)}\dot{\theta}_d(t_0) + \mathrm{c}_{\theta(t_0)}\mathrm{c}_{\phi(t_0)}\dot{\psi}_d(t_0)\right\|}{\left\|\mathrm{c}_{\theta(t_0)}\mathrm{c}_{\phi(t_0)}\psi_e(t_0)\right\|} \\ &+ \frac{\sqrt{\left(\frac{\pi}{2} - \sqrt{\frac{\lambda}{c}}\right)^2 - \|\psi_e(t_0)\|^2 - \delta_1\left\|\boldsymbol{\tau}_{de,r}(t_0)\right\|^2}}{\left\|\mathrm{c}_{\theta(t_0)}\mathrm{c}_{\phi(t_0)}\psi_e(t_0)\right\|},\end{aligned} \tag{3-74}$$

$$\begin{aligned}\mathbf{K}_{1,11} < (\mathbf{K}_{1,33})&\frac{\left\|\mathrm{s}_{\theta(t_0)}\psi_e(t_0)\right\|}{\|\phi_e(t_0)\|} + \frac{\left\|p(t_0) - \dot{\phi}_d(t_0) + \mathrm{s}_{\theta(t_0)}\dot{\psi}_d(t_0)\right\|}{\|\phi_e(t_0)\|} \\ &+ \frac{\sqrt{\left(\frac{\pi}{2} - \sqrt{\frac{\lambda}{c}}\right)^2 - \|\phi_e(t_0)\|^2 - \delta_1\left\|\boldsymbol{\tau}_{de,p}(t_0)\right\|^2}}{\|\phi_e(t_0)\|}.\end{aligned} \tag{3-75}$$

Furthermore, to ensure that actuator saturation does not occur, we will consider the maximum norm of (3-28):

$$\boldsymbol{\tau} = \left\| \boldsymbol{I}_H \left(\boldsymbol{I}_H^{-1} \boldsymbol{\omega} \times \boldsymbol{I}_H \boldsymbol{\omega} - \boldsymbol{I}_H^{-1} \hat{\boldsymbol{\tau}}_{Aero} - \mathbf{T}(\boldsymbol{\eta}_2) \boldsymbol{\eta}_{2e} + \dot{\boldsymbol{\alpha}}_\omega - \mathbf{K}_2 \boldsymbol{\omega}_e \right) \right\| \leq \left\| \begin{bmatrix} U_2^M \\ U_3^M \\ U_4^M \end{bmatrix} \right\|, \tag{3-76}$$

Taking the derivatives of both sides of (3-56), (3-57) and (3-58) we obtain:

$$\begin{aligned} \dot{\alpha}_p = & -\mathrm{c}_\theta \dot{\theta}(t) \left(\dot{\psi}_d(t) - \mathbf{K}_{1,33} \psi_e(t) \right) \\ & + \Big(\ddot{\phi}_d(t) - \mathbf{K}_{1,11} \left(-\mathbf{K}_{1,11} \phi_e(t) - \dot{\phi}_d(t) + p_e(t) \right) \\ & - \mathrm{s}_\theta \left(\ddot{\psi}_d(t) - \mathbf{K}_{1,33} \left(-\mathbf{K}_{1,33} \psi_e(t) - \dot{\psi}_d(t) \right) \right) \\ & + \left(-\mathbf{K}_{1,11} + \mathbf{K}_{1,33} \right) \left(\mathrm{s}_\phi \mathrm{t}_\theta q_e(t) + \mathrm{c}_\phi \mathrm{t}_\theta r_e(t) \right) \Big), \end{aligned} \tag{3-77}$$

$$\begin{aligned} \dot{\alpha}_q = & -\mathrm{s}_\phi \dot{\phi}(t) \left(\dot{\theta}_d(t) - \mathbf{K}_{1,22} \theta_e(t) \right) \\ & + \left(\mathrm{c}_\theta \mathrm{c}_\phi \dot{\phi}(t) - s_\theta \mathrm{s}_\phi \dot{\theta}(t) \right) \left(\dot{\psi}_d(t) - \mathbf{K}_{1,33} \psi_e(t) \right) \\ & + \Big(\mathrm{c}_\phi \left(\ddot{\theta}_d(t) - \mathbf{K}_{1,22} \left(-\mathbf{K}_{1,22} \theta_e(t) - \dot{\theta}_d(t) \right) \right) \\ & + \mathrm{c}_\theta \mathrm{s}_\phi \left(\ddot{\psi}_d(t) - \mathbf{K}_{1,33} \left(-\mathbf{K}_{1,33} \psi_e(t) - \dot{\psi}_d(t) \right) \right) \\ & - (\mathbf{K}_{1,22} \mathrm{c}_\phi^2 + \mathbf{K}_{1,33} \mathrm{s}_\phi^2)(q_e(t)) + \left(-\mathbf{K}_{1,22} + \mathbf{K}_{1,33} \right) \left(\mathrm{s}_\phi \mathrm{c}_\phi r_e(t) \right) \Big), \end{aligned} \tag{3-78}$$

$$\begin{aligned} \dot{\alpha}_r = & -c_\phi \dot{\phi}(t) \left(\dot{\theta}_d(t) - \mathbf{K}_{1,22} \theta_e(t) \right) \\ & - \left(\mathrm{c}_\theta \mathrm{s}_\phi \dot{\phi}(t) + s_\theta \mathrm{c}_\phi \dot{\theta}(t) \right) \left(\dot{\psi}_d(t) - \mathbf{K}_{1,33} \psi_e(t) \right) \\ & + \Big(-\mathrm{s}_\phi \left(\ddot{\theta}_d(t) - \mathbf{K}_{1,22} \left(-\mathbf{K}_{1,22} \theta_e(t) - \dot{\theta}_d(t) \right) \right) \\ & + \mathrm{c}_\theta \mathrm{c}_\phi \left(\ddot{\psi}_d(t) - \mathbf{K}_{1,33} \left(-\mathbf{K}_{1,33} \psi_e(t) - \dot{\psi}_d(t) \right) \right) \\ & + \left(\mathbf{K}_{1,22} - \mathbf{K}_{1,33} \right) \left(\mathrm{s}_\phi \mathrm{c}_\phi q_e(t) \right) - \left(\mathbf{K}_{1,22} \mathrm{s}_\phi^2 + \mathbf{K}_{1,33} \mathrm{c}_\phi^2 \right) r_e(t) \Big). \end{aligned} \tag{3-79}$$

Combing (3-77), (3-78), (3-79) with (3-76) gives:

$$\begin{aligned} I_x \Bigg(& \left\| \frac{I_z - I_y}{I_x} \right\| q^{\boldsymbol{M}} r^M + \frac{1}{I_x} \hat{\boldsymbol{\tau}}_p^M + \left(\phi_e(t) + \mathrm{s}_\phi \mathrm{t}_\theta \theta_e(t) + \mathrm{c}_\phi \mathrm{t}_\theta \psi_e(t) \right) \\ & - \mathrm{c}_\theta \dot{\theta}(t) \left(\dot{\psi}_d(t) - \mathbf{K}_{1,33} \psi_e(t) \right) \\ & + \Big(\ddot{\phi}_d(t) - \mathbf{K}_{1,11} \left(-\mathbf{K}_{1,11} \phi_e(t) - \dot{\phi}_d(t) + p_e(t) \right) \\ & - \mathrm{s}_\theta \left(\ddot{\psi}_d(t) - \mathbf{K}_{1,33} \left(-\mathbf{K}_{1,33} \psi_e(t) - \dot{\psi}_d(t) \right) \right) \\ & + \left(-\mathbf{K}_{1,11} + \mathbf{K}_{1,33} \right) \left(\mathrm{s}_\phi \mathrm{t}_\theta q_e(t) + \mathrm{c}_\phi \mathrm{t}_\theta r_e(t) \right) \Big) + \mathbf{K}_{2,11} p_e^M \Bigg) \leq U_2^M, \end{aligned} \tag{3-80}$$

$$I_y\Bigg(\left\|\frac{I_x - I_z}{I_y}\right\| r^M p^M + \frac{1}{I_y}\hat{\boldsymbol{\tau}}_q^M + \big(\mathrm{c}_\phi\theta_e(t) - \mathrm{s}_\phi\psi_e(t)\big) - \mathrm{s}_\phi\dot{\phi}(t)\left(\dot{\theta}_d(t) - \mathbf{K}_{1,22}\theta_e(t)\right) \\ + \left(\mathrm{c}_\theta\mathrm{c}_\phi\dot{\phi}(t) - s_\theta\mathrm{s}_\phi\dot{\theta}(t)\right)\left(\dot{\psi}_d(t) - \mathbf{K}_{1,33}\psi_e(t)\right) \\ + \Bigg(\mathrm{c}_\phi\left(\ddot{\theta}_d(t) - \mathbf{K}_{1,22}\left(-\mathbf{K}_{1,22}\theta_e(t) - \dot{\theta}_d(t)\right)\right) \\ + \mathrm{c}_\theta\mathrm{s}_\phi\left(\ddot{\psi}_d(t) - \mathbf{K}_{1,33}\left(-\mathbf{K}_{1,33}\psi_e(t) - \dot{\psi}_d(t)\right)\right) \\ - \left(\mathbf{K}_{1,22}\mathrm{c}_\phi^2 + \mathbf{K}_{1,33}\mathrm{s}_\phi^2\right)\left(q_e(t)\right) + \left(-\mathbf{K}_{1,22} + \mathbf{K}_{1,33}\right)\left(\mathrm{s}_\phi\mathrm{c}_\phi r_e(t)\right)\Bigg) \\ + \mathbf{K}_{2,22}q_e^M\Bigg) \le U_3^M, \tag{3-81}$$

$$I_z\Bigg(\left\|\frac{I_y I_x -}{I_z}\right\| p^M q^M + \frac{1}{I_z}\hat{\boldsymbol{\tau}}_r^M + \left(\frac{\mathrm{s}_\phi}{\mathrm{c}_\theta}\theta_e(t) + \frac{\mathrm{c}\phi}{\mathrm{c}_\theta}\psi_e(t)\right) \\ - c_\phi\dot{\phi}(t)\left(\dot{\theta}_d(t) - \mathbf{K}_{1,22}\theta_e(t)\right) \\ - \left(\mathrm{c}_\theta\mathrm{s}_\phi\dot{\phi}(t) + s_\theta\mathrm{c}_\phi\dot{\theta}(t)\right)\left(\dot{\psi}_d(t) - \mathbf{K}_{1,33}\psi_e(t)\right) \\ + \Bigg(-\mathrm{s}_\phi\left(\ddot{\theta}_d(t) - \mathbf{K}_{1,22}\left(-\mathbf{K}_{1,22}\theta_e(t) - \dot{\theta}_d(t)\right)\right) \\ + \mathrm{c}_\theta\mathrm{c}_\phi\left(\ddot{\psi}_d(t) - \mathbf{K}_{1,33}\left(-\mathbf{K}_{1,33}\psi_e(t) - \dot{\psi}_d(t)\right)\right) \\ + \left(\mathbf{K}_{1,22} - \mathbf{K}_{1,33}\right)\left(\mathrm{s}_\phi\mathrm{c}_\phi q_e(t)\right) - \left(\mathbf{K}_{1,22}\mathrm{s}_\phi^2 + \mathbf{K}_{1,33}\mathrm{c}_\phi^2\right)r_e(t)\Bigg) \\ + \mathbf{K}_{2,33}r_e^M\Bigg) \le U_4^M. \tag{3-82}$$

Rearranging (3-80), (3-81) and (3-82) gives:

$$\mathbf{K}_{2,11} \le \frac{\frac{U_2^M}{I_x} - \left(\left\|\frac{I_z - I_y}{I_x}\right\| q^M r^M + \frac{1}{I_x}\hat{\boldsymbol{\tau}}_p^M\right)}{p_e^M} \\ - \frac{1}{p_e^M}\Bigg(-\left(\phi_e^M + \mathrm{s}_\phi\mathrm{t}_\theta\theta_e^M + \mathrm{c}_\phi\mathrm{t}_\theta\psi_e^M\right) + \mathrm{c}_\theta\dot{\theta}^M * \left(\dot{\psi}_d^M + \mathbf{K}_{1,33}\psi_{ee}^M\right) \\ + \Big(\ddot{\phi}_d^M + \mathbf{K}_{1,11}\left(+\mathbf{K}_{1,11}\phi_e^M + \dot{\phi}_d^M + p_e^M\right) \\ + \mathrm{s}_\theta\left(\ddot{\psi}_d^M + \mathbf{K}_{1,33}\left(+\mathbf{K}_{1,33}\psi_{ee}^M + \dot{\psi}_d^M\right)\right) + \left\|-\mathbf{K}_{1,11} + \mathbf{K}_{1,33}\right\| \\ * \left(\mathrm{s}_\phi\mathrm{t}_\theta q_e^M + \mathrm{c}_\phi\mathrm{t}_\theta r_e^M\right)\Big)\Bigg), \tag{3-83}$$

$$\mathbf{K}_{2,22} \le \frac{\frac{U_3^M}{I_y} - \left(\left\|\frac{I_x - I_z}{I_y}\right\| r^M p^M + \frac{1}{I_y}\hat{\boldsymbol{\tau}}_q^M\right)}{q_e^M} - \frac{1}{q_e^M}\Bigg(-\left(\mathrm{c}_\phi \theta_e^M - \mathrm{s}_\phi \psi_e^M\right) + \mathrm{s}_\phi \dot{\phi}^M\left(\dot{\theta}_d^M + \mathbf{K}_{1,22}\theta_e^M\right) + \left(\mathrm{c}_\theta \mathrm{c}_\phi \dot{\phi}^M + s_\theta \mathrm{s}_\phi \dot{\theta}^M\right) * \left(\dot{\psi}_d^M + \mathbf{K}_{1,33}\psi_e^M\right) + \Big(\mathrm{c}_\phi\left(\ddot{\theta}_d^M + \mathbf{K}_{1,22}\left(+\mathbf{K}_{1,22}\theta_e^M + \dot{\theta}_d^M\right)\right) + \mathrm{c}_\theta \mathrm{s}_\phi\left(\ddot{\psi}_d^M + \mathbf{K}_{1,33}\left(+\mathbf{K}_{1,33}\psi_e^M + \dot{\psi}_d^M\right)\right)\Big) + \left(\mathbf{K}_{1,22}\mathrm{c}_\phi^2 + \mathbf{K}_{1,33}\mathrm{s}_\phi^2\right) * \left(q_e^M\right) + \left\|-\mathbf{K}_{1,22} + \mathbf{K}_{1,33}\right\| * \left(\mathrm{s}_\phi \mathrm{c}_\phi r_e^M\right)\Bigg), \tag{3-84}$$

$$\mathbf{K}_{2,33} \le \frac{\frac{U_4^M}{I_z} - \left(\left\|\frac{I_y - I_x}{I_z}\right\| p^M q^M + \frac{1}{I_z}\hat{\boldsymbol{\tau}}_r^M\right)}{r_e^M} - \frac{1}{r_e^M}\Bigg(-\left(\frac{\mathrm{s}_\phi}{\mathrm{c}_\theta}\theta_e^M + \frac{\mathrm{c}\phi}{\mathrm{c}_\theta}\psi_e^M\right) + c_\phi \dot{\phi}^M * \left(\dot{\theta}_d^M + \mathbf{K}_{1,22}\theta_e^M\right) + \left(\mathrm{c}_\theta \mathrm{s}_\phi \dot{\phi}^M + s_\theta \mathrm{c}_\phi \dot{\theta}^M\right) * \left(\dot{\psi}_d^M + \mathbf{K}_{1,33}\psi_e^M\right) + \Big(+\mathrm{s}_\phi\left(\ddot{\theta}_d^M + \mathbf{K}_{1,22}\left(+\mathbf{K}_{1,22}\theta_{ee}^M + \dot{\theta}_d^M\right)\right) + \mathrm{c}_\theta \mathrm{c}_\phi\left(\ddot{\psi}_d^M + \mathbf{K}_{1,33}\left(+\mathbf{K}_{1,33}\psi_{ee}^M + \dot{\psi}_d^M\right)\right)\Big) + \left(\mathbf{K}_{1,22} - \mathbf{K}_{1,33}\right) * \left(\mathrm{s}_\phi \mathrm{c}_\phi q_e^M\right) + \left(\mathbf{K}_{1,22}\mathrm{s}_\phi^2 + \mathbf{K}_{1,33}\mathrm{c}_\phi^2\right) r_e^M\Bigg), \tag{3-85}$$

we define $\dot{\phi}^M$ and $\dot{\theta}^M$ as:

$$\dot{\phi}^M \le p^M + \mathrm{s}_{\phi^M}\mathrm{t}_{\theta^M} q^M + \mathrm{t}_{\theta^M} r^M, \tag{3-86}$$

$$\dot{\theta}^M \le q^M + \mathrm{s}_{\phi^M} r^M\,. \tag{3-87}$$

Therefore, the control objective 3.1 has been met.

3.2 Stochastic Attitude System

This section presents the analysis and control of the attitude system of the quadrotor, subject to stochastic loads as per the system described in (2-120). For convenience the equations are restated here:

$$\dot{\boldsymbol{q}} = \mathbf{R}_2(\boldsymbol{q})\mathbf{T}(\boldsymbol{\eta}_2)^{-1}\dot{\boldsymbol{\eta}}_2, \tag{3-88}$$
$$\dot{\boldsymbol{q}} = \mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}, \tag{3-89}$$
$$d\boldsymbol{\omega} = (\mathbf{I}_A + \mathbf{I}_H)^{-1}(-\mathbf{D}_2\boldsymbol{\omega} - \mathbf{C}_B(\boldsymbol{\omega})\boldsymbol{\omega} - \mathbf{C}_{A2}(\boldsymbol{\omega})\boldsymbol{\omega} + \boldsymbol{\tau})dt + (\mathbf{I}_A + \mathbf{I}_H)^{-\mathbf{1}}\boldsymbol{\theta}\Delta_{\mathbf{1}}(t)d\boldsymbol{w}_1 + (\mathbf{I}_A + \mathbf{I}_H)^{-\mathbf{1}}\Delta_{\mathbf{2}}(t)d\boldsymbol{w}_2. \tag{3-90}$$

where $\boldsymbol{\Delta}_1(t)$ and $\boldsymbol{\Delta}_2(t)$ the (time-varying) covariance matrices, $\boldsymbol{w}_1$ denotes the 3-dimensional Wiener standard process vector and $\boldsymbol{\theta}$ is deffiend as:

$$\boldsymbol{\theta} = (\mathbf{C}_{A2}(\boldsymbol{\omega}) + \boldsymbol{S}(\boldsymbol{\omega})\mathbf{I}_A + \mathbf{D}_2)\Delta_{\mathbf{2}}(t). \tag{3-91}$$

Recall from section 2.6 the definition:

$$\mathbf{T}(\boldsymbol{\eta}_2) = \begin{bmatrix} 1 & s_\phi t_\theta & c_\phi t_\theta \\ 0 & c_\phi & -s_\phi \\ 0 & \frac{s_\phi}{c_\theta} & \frac{c\phi}{c_\theta} \end{bmatrix}, \quad \mathbf{T}(\boldsymbol{\eta}_2)^{-1} = \begin{bmatrix} 1 & 0 & -s_\theta \\ 0 & c_\phi & c_\theta s_\phi \\ 0 & -s_\phi & c_\theta c_\phi \end{bmatrix}, \tag{3-92}$$

3.2.1 Model Assumptions

Assumption 3.4

Assume that the reference attitude trajectory $\boldsymbol{\eta}_{2d}(t) = [\phi_d(t) \quad \theta_d(t) \quad \psi_d(t)]^{\mathrm{T}}$ is sufficiently smooth, i.e., the first two derivatives exist and are bounded, that is there exists non-negative constants ϱ_1 and ϱ_2 such that:

$$\sup_{t\in R^n} \|\dot{\boldsymbol{\eta}}_{2d}(t)\| \le \varrho_1, \qquad \sup_{t\in R^n} \|\ddot{\boldsymbol{\eta}}_{2d}(t)\| \le \varrho_2. \tag{3-93}$$

Assumption 3.5

Assume that reference pitch, roll and yaw trajectory is bounded by a positive constant ϱ_3, between $\pm 2\pi$ such that

$$\sup_{t\in R^n} \|\phi_d(t)\| \le \varrho_3, \qquad \sup_{t\in R^n} \|\theta_d(t)\| \le \varrho_3, \qquad \sup_{t\in R^n} \|\psi_d(t)\| \le \varrho_3 \;\; \varrho_3 < 2\pi. \tag{3-94}$$

Assumption 3.6

Assume that the aerodynamic torque disturbance vector $\boldsymbol{\tau}_{Aero}$ and its derivative are bounded, i.e., there exist nonnegative constants such that $\|\boldsymbol{\tau}_{Aero}(t)\| \le \varrho_\tau$ and $\left\|\dot{\boldsymbol{\tau}}_{Aero,\boldsymbol{\eta}}(t)\right\| = 0, \forall\; t \ge t_0 \ge 0$. Furthermore, there exists constants $\varrho_{\tau 1}$, $\varrho_{\tau 2}$ and $\varrho_{\tau 3}$ such that:

$$\sup_{t\in R^n} \left\|\boldsymbol{\tau}_{Aero,\boldsymbol{p}}(t)\right\| \le \varrho_{\tau 1}, \qquad \sup_{t\in R^n} \left\|\boldsymbol{\tau}_{Aero,\boldsymbol{q}}(t)\right\| \le \varrho_{\tau 2}, \qquad \sup_{t\in R^n} \left\|\boldsymbol{\tau}_{Aero,\boldsymbol{r}}(t)\right\| \le \varrho_{\tau 3}. \tag{3-95}$$

Assumption 3.7
The unknown parameter vector $\bar{\boldsymbol{\tau}}_{Aero}$ and unknown covariance matrix $\Delta(t)$ is bounded, i.e., there exist $\bar{\boldsymbol{\tau}}_{Aero_M}$ and Δ_M such that:

$$\sup_{t \in R^n} \|\bar{\boldsymbol{\tau}}_{Aero}(t)\| \leq \varrho_4, \qquad \sup_{t \in R^n} \| \|\Delta(t)\Delta^T(t)\|_\infty \| \leq \varrho_5. \tag{3-96}$$

Under the above assumptions, it is possible to design the control input vector $\boldsymbol{\tau} = [U_2 \quad U_3 \quad U_4]$ in such a manner that the angular position $\boldsymbol{\eta}_2(t)$ of the aircraft asymptotically track the reference trajectories $\boldsymbol{\eta}_{1d}(t)$, i.e.,

$$\lim_{t \to \infty} \left(\boldsymbol{\eta}_2(t) - \boldsymbol{\eta}_{1d}(t)\right) = 0, \tag{3-97}$$

<u>Control objective 3.2</u>

Based on the above assumptions there are two objectives to achieve:

1. Design the control input vector $\boldsymbol{\tau} = [U_2 \quad U_3 \quad U_4]$ and estimate laws for $\bar{\boldsymbol{\tau}}_{Aero}$ and $\|\Delta(t)\Delta^T(t)\|_\infty$ in such a manner to force the aircraft's attitude vectors $\boldsymbol{\eta}_2$ to stochastically asymptotically track their reference trajectory vector $\boldsymbol{\eta}_{2d}$ iwith a wider operation envelope than that obtained in section 3.1.
2. Keep the aircraft's attitude between the range of $\pm 2\pi$.

3.3 Stabilisation of Attitude System

Step 1: Define the following tracking errors:

$$\boldsymbol{\eta}_{2e} = \boldsymbol{\eta}_2 - \boldsymbol{\eta}_{2d}, \tag{3-98}$$
$$\boldsymbol{q}_e = \boldsymbol{q} - \boldsymbol{q}_d, \tag{3-99}$$
$$\boldsymbol{\omega}_e = \boldsymbol{\omega} - \boldsymbol{\alpha}_\omega, \tag{3-100}$$

where $\boldsymbol{\omega}_e$ is the tracking error of the angular velocity and $\boldsymbol{\alpha}_{\boldsymbol{\omega}}$ the virtual control for the angular velocity. In the first step, we design $\boldsymbol{\alpha}_\omega$ to asymptotically stabilize the tracking error $\boldsymbol{q}_e$ and in turn $\boldsymbol{\eta}_{2e}$ at the origin. In the second step, we design the torque vector $\boldsymbol{\tau}$ to stochastically asymptotically stabilize the tracking error $\boldsymbol{\omega}_e$ at the origin. Differentiating both sides of (3-99) along the solutions of (3-88) and (3-89) yields:

$$\dot{\boldsymbol{q}}_e = \mathbf{R}_2(\boldsymbol{q})\boldsymbol{\alpha}_{\boldsymbol{\omega}} - \mathbf{R}_2(\boldsymbol{q}_d)\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}\dot{\boldsymbol{\eta}}_{2d} + \mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e. \tag{3-101}$$

To design the virtual controls $\boldsymbol{\alpha}_{\boldsymbol{\omega}}$ to stabilize $\boldsymbol{\eta}_{2e}$ at the origin, we consider the following Lyapunov function candidate:

$$V_1 = \frac{\gamma_1}{2}\|\boldsymbol{q}_{2e}\|^2 . \tag{3-102}$$

Differentiating both sides of (3-102) along the solutions of (3-101) results in the following infinite generator:

$$\mathcal{L}V_1 = \gamma_1 \boldsymbol{q}_e^T(\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\alpha}_{\boldsymbol{\omega}} - \mathbf{R}_2(\boldsymbol{\alpha}_2)\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}\dot{\boldsymbol{\eta}}_{2d} + \mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e). \tag{3-103}$$

This suggests choosing the virtual control $\boldsymbol{\alpha}_\omega$ as follows:

$$\boldsymbol{\alpha}_\omega = \mathbf{R}_2^{-1}(\boldsymbol{q})(\mathbf{R}_2(\boldsymbol{\alpha}_2)\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}\dot{\boldsymbol{\eta}}_{2d} - \mathbf{K}_1\boldsymbol{q}_e). \quad \text{(3-104)}$$

Substituting (3-104) into (3-101) yields:

$$\dot{\boldsymbol{q}}_e = -\mathbf{K}_1\boldsymbol{q}_e + \mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e. \quad \text{(3-105)}$$

To prepare for the next step, we calculate $d\boldsymbol{\omega}_e$ using (3-100) together with (3-90) and (3-101) as follows:

$$d\boldsymbol{\omega}_e = d\boldsymbol{\omega} - \dot{\boldsymbol{\alpha}}_\omega dt, \quad \text{(3-106)}$$

$$d\boldsymbol{\omega}_e = \left((\mathbf{I}_A + \mathbf{I}_H)^{-1}(\boldsymbol{\tau} + \mathbf{D}_1\boldsymbol{\omega} + \boldsymbol{\omega}\times(\mathbf{I}_A + \mathbf{I}_H)\boldsymbol{\omega} - \dot{\boldsymbol{\alpha}}_{\boldsymbol{\omega}})\right)dt + (\mathbf{I}_A + \mathbf{I}_H)^{-1}(\mathbf{C}_{A2}(\boldsymbol{\omega}) + \boldsymbol{S}(\boldsymbol{\omega})\mathbf{I}_A + \mathbf{D}_2)\Delta(t)d\boldsymbol{w}. \quad \text{(3-107)}$$

The term $\dot{\boldsymbol{\alpha}}_\omega$ we define by differentiating both sides of (3-104) along the solution of (3-101) we obtain:

$$\dot{\boldsymbol{\alpha}}_{\boldsymbol{\omega}} = \mathbf{R}_2^{-1}(\boldsymbol{q})\big(\dot{\mathbf{R}}_2(\boldsymbol{\alpha}_2)\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}\dot{\boldsymbol{\eta}}_{2d} + \mathbf{R}_2(\boldsymbol{\alpha}_2)\dot{\mathbf{T}}(\boldsymbol{\eta}_{2d})^{-1}\dot{\boldsymbol{\eta}}_{2d} + \mathbf{R}_2(\boldsymbol{\alpha}_2)\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}\ddot{\boldsymbol{\eta}}_{2d} - \mathbf{K}_1\dot{\boldsymbol{q}}_e\big) + \dot{\mathbf{T}}^{-1}(\boldsymbol{\eta}_2)(\dot{\boldsymbol{\eta}}_{2d} - \mathbf{K}_1\boldsymbol{\eta}_{2e}) \quad \text{(3-108)}$$

Step 2: To design the control torque input $\boldsymbol{\tau}$ that will ensure robust adaptive control, we could consider the following Lyapunov function candidate:

$$V_2 = V_1 + \frac{\gamma_2}{2}\sqrt[2]{1 + \|\boldsymbol{\omega}_e\|^4} + \frac{1}{2}\bar{\boldsymbol{\tau}}_{de,Aero}^{\mathrm{T}}\boldsymbol{\Gamma}^{-1}\bar{\boldsymbol{\tau}}_{de,Aero} + \frac{1}{2\mu_1}\tilde{\delta}_1^2 + \frac{1}{2\mu_2}\tilde{\delta}_2^2 - \frac{\gamma_2}{2}, \quad \text{(3-109)}$$

where, the symbol $\boldsymbol{\Gamma}$ is a symmetric and positive definite matrix, μ and γ_2 are positive constants, and we define the following uncertainty errors:

$$\begin{aligned} \bar{\boldsymbol{\tau}}_{de,Aero} &= \bar{\boldsymbol{\tau}}_{Aero} - \hat{\boldsymbol{\tau}}_{Aero}, \\ \tilde{\delta}_1 &= \left\|\Delta_1(t)\Delta_1^{\mathrm{T}}(t)\right\|_\infty^2 - \hat{\delta}_1, \\ \tilde{\delta}_2 &= \left\|\Delta_2(t)\Delta_2^{\mathrm{T}}(t)\right\|_\infty^2 - \hat{\delta}_2. \end{aligned} \quad \text{(3-110)}$$

The infinite generator of (3-109) along the solutions of (3-103) and (3-110) is:

$$\begin{aligned} \mathcal{L}V_2 = {} & \gamma_1\boldsymbol{q}_e^T(-\mathbf{K}_1\boldsymbol{q}_e + \mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e) + \frac{\gamma_2\|\boldsymbol{\omega}_e\|^2\boldsymbol{\omega}_e^T}{(1 + \|\boldsymbol{\omega}_e\|)^{0.5}}\left(\mathbf{I}_{\boldsymbol{H}}^{-1}(\boldsymbol{\tau} - \boldsymbol{\omega}\times\mathbf{I}_{\boldsymbol{H}}\boldsymbol{\omega} + \hat{\bar{\boldsymbol{\tau}}}_{Aero}) - \dot{\boldsymbol{\alpha}}_{\boldsymbol{\omega}}\right) \\ & + \frac{1}{2}\mathrm{Tr}\left\{\Delta_1^{\mathrm{T}}(t)\boldsymbol{\theta}^{\mathrm{T}}(\mathbf{I}_A + \mathbf{I}_H)^{-\mathrm{T}}\frac{\partial^2 V_2}{\partial\boldsymbol{\omega}_e^2}(\mathbf{I}_A + \mathbf{I}_H)^{-1}\boldsymbol{\theta}\Delta_1(t)\right\} \\ & + \frac{1}{2}\mathrm{Tr}\left\{\Delta_2^{\mathrm{T}}(t)(\mathbf{I}_A + \mathbf{I}_H)^{-\mathrm{T}}\frac{\partial^2 V_2}{\partial\boldsymbol{\omega}_e^2}(\mathbf{I}_A + \mathbf{I}_H)^{-1}\Delta_2(t)\right\} - \bar{\boldsymbol{\tau}}_{de,Aero}^{\mathrm{T}}\boldsymbol{\Gamma}^{-1}\dot{\hat{\boldsymbol{\tau}}}_{Aero} - \frac{1}{\mu_1}\tilde{\delta}_1\dot{\hat{\delta}}_1 \\ & - \frac{1}{\mu_2}\tilde{\delta}_2\dot{\hat{\delta}}_2, \end{aligned} \quad \text{(3-111)}$$

where we define:

$$\frac{\partial^2 V_2}{\partial \boldsymbol{\omega}_e^2} = \frac{\mathbf{I}_{3\times3}\|\boldsymbol{\omega}_e\|^2 + 2\boldsymbol{\omega}_e\boldsymbol{\omega}_e^T}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}} - 2\frac{\|\boldsymbol{\omega}_e\|^4\boldsymbol{\omega}_e\boldsymbol{\omega}_e^T}{(1+\|\boldsymbol{\omega}_e\|^4)^{1.7}}. \tag{3-112}$$

Applying Young's inequality to (3-111) yields:

$$\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e \leq \varepsilon_1\|\boldsymbol{q}_e\|^2 + \varepsilon_2\left(\frac{1}{4\varepsilon_5}\right)^2\frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^4} + \frac{\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)} + \frac{1}{4\varepsilon_2}, \tag{3-113}$$

$$\begin{aligned}\mathrm{Tr}&\left\{\Delta_1^{\mathrm{T}}(t)\boldsymbol{\theta}^{\mathrm{T}}(\mathbf{I}_A+\mathbf{I}_H)^{-\mathrm{T}}\frac{\partial^2 V_2}{\partial \boldsymbol{\omega}_e^2}(\mathbf{I}_A+\mathbf{I}_H)^{-1}\boldsymbol{\theta}\Delta_1(t)\right\}\\ &\leq \frac{9\gamma_2}{4}\varepsilon_3\frac{\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{1}}\boldsymbol{\theta}\right\|^4\|\boldsymbol{\omega}_e\|^4\delta_1}{(1+\|\boldsymbol{\omega}_e\|^4)^1} + \frac{1}{4\gamma_2\varepsilon_3},\end{aligned} \tag{3-114}$$

$$\mathrm{Tr}\left\{\Delta_2^{\mathrm{T}}(t)(\mathbf{I}_A+\mathbf{I}_H)^{-\mathrm{T}}\frac{\partial^2 V_2}{\partial \boldsymbol{\omega}_e^2}(\mathbf{I}_A+\mathbf{I}_H)^{-1}\Delta_2(t)\right\} \leq \frac{9\gamma_2}{4}\varepsilon_4\frac{\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{1}}\right\|^4\|\boldsymbol{\omega}_e\|^4\delta_2}{(1+\|\boldsymbol{\omega}_e\|^4)^1} + \frac{1}{4\gamma_2\varepsilon_4}, \tag{3-115}$$

where ε_1, ε_2, and ε_4 are positive constants using (3-113), (3-114), (3-115) and (3-110), we rewrite (3-111) as:

$$\begin{aligned}\mathcal{L}V_2 \leq& -\gamma_1\|\boldsymbol{q}_e\|^2\left(\mathbf{K}_1 - \frac{\varepsilon_1}{\gamma_1}\mathbf{I}_{3\times3}\right)\\ &+\gamma_2\frac{\|\boldsymbol{\omega}_e\|^2\boldsymbol{\omega}_e^T}{(1+\|\boldsymbol{\omega}_e\|^4)^{.75}}\Bigg[\Bigg(\frac{\varepsilon_2}{\gamma_2}\left(\frac{1}{4\varepsilon_5}\right)^2\frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{3.5}} + \frac{1}{\gamma_2}\frac{(\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e)}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}\\ &+\frac{9}{4}\gamma_2\varepsilon_3\frac{\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{1}}\boldsymbol{\theta}\right\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}\hat{\delta}_1 + \frac{9}{4}\gamma_2\varepsilon_4\frac{\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{1}}\right\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}\hat{\delta}_2\Bigg)\boldsymbol{\omega}_e\\ &+(\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{1}}\boldsymbol{\tau} - (\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{1}}(\boldsymbol{\omega}\times(\mathbf{I}_A+\mathbf{I}_H)\boldsymbol{\omega}) + (\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{1}}(\hat{\bar{\boldsymbol{\tau}}}_{Aero}) - \dot{\boldsymbol{\alpha}}_{\boldsymbol{\omega}}\Bigg]\\ &+\tilde{\delta}_1\left(\gamma_2^2\frac{9}{4}\varepsilon_3\frac{\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{1}}\boldsymbol{\theta}\right\|^4\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^1} - \frac{1}{\mu_1}\dot{\hat{\delta}}_1\right)\\ &+\tilde{\delta}_2\left(\gamma_2^2\frac{9}{4}\varepsilon_4\frac{\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{1}}\right\|^4\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^1} - \frac{1}{\mu_2}\dot{\hat{\delta}}_2\right)\\ &+\tilde{\bar{\boldsymbol{\tau}}}_{de,Aero}^{\mathrm{T}}\left(\gamma_2\frac{\|\boldsymbol{\omega}_e\|^2(\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{T}}\boldsymbol{\omega}_e}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}} - \boldsymbol{\Gamma}^{-1}\dot{\hat{\bar{\boldsymbol{\tau}}}}_{Aero}\right) + \sum_{i=2}^{4}\frac{1}{4\varepsilon_i}.\end{aligned} \tag{3-116}$$

The time varying disturbance $\Delta_1(t)$ and $\Delta_2(t)$ results in the generation of the term $\frac{1}{4\varepsilon_3}, \frac{1}{4\varepsilon_4}$, to overcome the effects of the disturbance on the system the control law for $\boldsymbol{\tau}$ and update laws for $\hat{\boldsymbol{\tau}}_{Aero}, \hat{\delta}_1$ and $\hat{\delta}_2$ are designed using the projection algorithm presented in section 2.7.3 as follows:

$$\boldsymbol{\tau} = (\mathbf{I}_A + \mathbf{I}_H)\left[\mathbf{I}_{\boldsymbol{H}}^{-1}(\boldsymbol{\omega}\times\mathbf{I}_{\boldsymbol{H}}\boldsymbol{\omega}) + \dot{\boldsymbol{\alpha}}_{\boldsymbol{\omega}} - \mathbf{K}_2\boldsymbol{\omega}_e \right. \\ - \left(\frac{\varepsilon_2}{\gamma_2}\left(\frac{1}{4\varepsilon_5}\right)^2 \frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{3.5}} + \frac{1}{\gamma_2}\frac{(\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e)}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}} \right. \\ \left. \left. + \frac{9}{4}\gamma_2\left(\varepsilon_3\frac{\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{1}}\boldsymbol{\theta}\right\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}\hat{\delta}_1 + \varepsilon_4\frac{\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{1}}\right\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}\hat{\delta}_2\right)\right)\boldsymbol{\omega}_e - \mathbf{I}_{\boldsymbol{H}}^{-1}\hat{\boldsymbol{\tau}}_{Aero}\right],$$

$$\dot{\hat{\boldsymbol{\tau}}}_{Aero} = \boldsymbol{\Gamma}\mathrm{proj}\left(\gamma_2\frac{\|\boldsymbol{\omega}_e\|^2(\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{T}}\boldsymbol{\omega}_e}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}, \hat{\boldsymbol{\tau}}_{Aero}\right),$$

$$\dot{\hat{\delta}}_1 = \mu_1\mathrm{proj}\left(\gamma_2\frac{9}{4}\varepsilon_3\frac{\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\boldsymbol{\theta}\|^4\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^1}, \hat{\delta}_1\right),$$

$$\dot{\hat{\delta}}_2 = \mu_2\mathrm{proj}\left(\gamma_2\frac{9}{4}\varepsilon_4\frac{\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\|^4\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^1}, \hat{\delta}_2\right),$$

(3-117)

where, $\mathbf{K}_2$ is a positive definite matrix.

Note that unlike the deterministic case, we have not used the disturbance observer presented in section 2.7.1. this is because the disturbance observer cannot handle the stochastic component of the disturbance, only the deterministic portion. This can be seen when viewing equations (2-124) and (2-127). Moreover, substituting (3-117) into (3-114) and using property d.) of the projection algorithm (2-168) we obtain:

$$\mathcal{L}V_2 \le -\gamma_1\|\boldsymbol{q}_e\|^2\left(\lambda_m(\mathbf{K}_1) - \frac{\varepsilon_1}{\gamma_1}\right) - \gamma_2\lambda_m(\mathbf{K}_2)\frac{\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.75}} + \sum_{i=2}^{4}\frac{1}{4\varepsilon_i}. \tag{3-118}$$

Furthermore, substituting (3-117) into (3-107) yields:

$$d\boldsymbol{\omega}_e = \left(-\mathbf{K}_2\boldsymbol{\omega}_e - \left(\frac{\varepsilon_2}{\gamma_2}\left(\frac{1}{4\varepsilon_5}\right)^2\frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{3.5}} + \frac{1}{\gamma_2}\frac{(\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e)}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}} \right.\right. \\ \left. + \frac{9}{4}\gamma_2\varepsilon_3\frac{\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\boldsymbol{\theta}\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}\hat{\delta}_1 + \frac{9}{4}\gamma_2\varepsilon_4\frac{\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}\hat{\delta}_2\right)\boldsymbol{\omega}_e \\ \left. + (\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{1}}\tilde{\boldsymbol{\tau}}_{Aero}\right)dt + (\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{1}}\boldsymbol{\theta}\Delta_{\mathbf{1}}(t)d\boldsymbol{w}_1 + (\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{1}}\Delta_{\mathbf{2}}(t)d\boldsymbol{w}_2 \tag{3-119}$$

The control design is complete. In the next sub section, we will prove that the control and estimate update laws presented ensure that the system is stochastically asymptotically stable.

3.4 Stability Analysis

Recall from (3-109) and (3-118):

$$V_2 = \frac{\gamma_1}{2}\|\boldsymbol{q}_e\|^2 + \frac{\gamma_2}{2}\sqrt[2]{1+\|\boldsymbol{\omega}_e\|^4} + \frac{1}{2}\bar{\boldsymbol{\tau}}_{de,Aero}^{\mathrm{T}}\boldsymbol{\Gamma}^{-1}\bar{\boldsymbol{\tau}}_{de,Aero} + \frac{1}{2\mu_1}\tilde{\delta}_1^2 + \frac{1}{2\mu_2}\tilde{\delta}_2^2 - \frac{\gamma_2}{2}, \tag{3-120}$$

$$\mathcal{L}V_2 \le -\gamma_1\|\boldsymbol{q}_e\|^2\left(\lambda_m(\mathbf{K}_1) - \frac{\varepsilon_1}{\gamma_1}\right) - \gamma_2\lambda_m(\mathbf{K}_2)\frac{\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}} + \frac{1}{4\varepsilon_2} + \frac{\gamma_2}{4\varepsilon_3}, \tag{3-121}$$

we can therefore state:

$$\mathcal{L}V_2 \le -W(\mathbf{X}_e) + \sum_{i=2}^{4}\frac{1}{4\varepsilon_i}, \tag{3-122}$$

where we define:

$$\mathbf{X}_e = \mathrm{col}(\boldsymbol{\eta}_{2e}, \boldsymbol{\omega}_e), \tag{3-123}$$

$$W(\mathbf{X}_e) = -\gamma_1\|\boldsymbol{q}_e\|^2\left(\lambda_m(\mathbf{K}_1) - \frac{\varepsilon_1}{\gamma_1}\right) - \gamma_2\lambda_m(\mathbf{K}_2)\frac{\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}, \tag{3-124}$$

where we choose the positive definite matrix $\mathbf{K}_1$ and positive constants ε_1 and γ_1 such that:

$$\lambda_m(\mathbf{K}_1) - \frac{\varepsilon_1}{\gamma_1} > 0. \tag{3-125}$$

Method 1

Similar to the deterministic case presented in section 3.1 we wish to obtain an inequality of the infinite generator of the form:

$$\mathcal{L}V_2 \le -cV_2 + \lambda. \tag{3-126}$$

Adding $\frac{1}{2}\tilde{\boldsymbol{\tau}}_{Aero}^{\mathrm{T}}\boldsymbol{\Gamma}^{-1}\tilde{\boldsymbol{\tau}}_{Aero} + \frac{1}{2\mu}\tilde{\delta}^2$ to the right hand side of (3-118) results in:

$$c = \frac{\min\left(\gamma_1\left(\lambda_m(\mathbf{K}_1) - \frac{\varepsilon_1}{\gamma_1}\right), \gamma_2\lambda_m(\mathbf{K}_2), \lambda_m(\boldsymbol{\Gamma}^{-1}), \frac{1}{\mu_1}, \frac{1}{\mu_2}\right)}{\max\left(\frac{\gamma_1}{2}, \frac{\gamma_2}{2}, \frac{1}{2}\|\boldsymbol{\Gamma}^{-1}\|, \frac{1}{2\mu_1}, \frac{1}{2\mu_2}\right)}, \tag{3-127}$$

$$\lambda = \frac{1}{4\varepsilon_2} + \frac{\gamma_2}{4\varepsilon_3} + \frac{1}{2}\lambda_m(\boldsymbol{\Gamma}^{-1})\left(2\boldsymbol{\tau}_{Aero_M} + \xi_{\boldsymbol{\tau}_{Aero}}\right)^2 + \sum_{i=1}^{2}\frac{1}{2\mu_i}\left(2\delta_{\mathrm{i,M}} + \xi_{i,\delta}\right)^2.$$

As mentioned previously $\lambda_m(\bullet)$ denotes the minimum eigenvalue of the matrix $(\bullet)$. By applying point a.) from the projection algorithm, we are able to obtain the following inequality:

$$\frac{1}{2}\bar{\boldsymbol{\tau}}_{de,Aero}^{\mathrm{T}}\boldsymbol{\Gamma}^{-1}\bar{\boldsymbol{\tau}}_{de,Aero} + \sum_{i=1}^{2}\frac{\tilde{\delta}_i^2}{2\mu_i} \le \lambda_m(\boldsymbol{\Gamma}^{-1})\left(2\bar{\boldsymbol{\tau}}_{Aero_M} + \xi_{\boldsymbol{\tau}_{Aero}}\right)^2 + \sum_{i=1}^{2}\frac{1}{\mu_i}\left(2\delta_{\mathrm{i,M}} + \xi_{i,\delta}\right)^2, \tag{3-128}$$

with ξ being an arbitrary constant defined in the projection algorithm.

Applying Theorem 2.7 to (3-126) results in the fact that the closed loop system consisting of (3-105), (3-119) and the last two equations of (3-117) with Assumption 3.4 to Assumption 3.7 has a unique strong solution. Furthermore, the system realises exponential convergence of the expectation of the tracking error to $\frac{\lambda}{c}$. However, $\frac{\lambda}{c}$ cannot be made arbitrarily small

Method 2

In order to obtain an arbitrary small value to which the tracking error converges in probability we will apply Theorem 2.6.Firstly we will substitute (3-102) in to (3-109), rearranging we obtain:

$$V_2 = W_2(\mathbf{X}_e) + \frac{1}{2}\bar{\boldsymbol{\tau}}_{de,Aero}^{\mathrm{T}}\boldsymbol{\Gamma}^{-1}\bar{\boldsymbol{\tau}}_{de,Aero} + \frac{1}{2\mu_1}\tilde{\delta}_1^2 + \frac{1}{2\mu_2}\tilde{\delta}_2^2, \tag{3-129}$$

where:

$$W_2(\mathbf{X}_e) = \tfrac{\gamma_1}{2}\|\boldsymbol{q}_e\|^2 + \frac{\gamma_2}{2}\sqrt[2]{1+\|\boldsymbol{\omega}_e\|^4} - \frac{\gamma_2}{2}, \tag{3-130}$$

$$\mathbf{X}_e = \mathrm{col}(\boldsymbol{q}_e, \boldsymbol{\omega}_e). \tag{3-131}$$

Clearly $W_3(\mathbf{X}_e)$ is a class $\mathcal{K}_\infty$ function of $\|\mathbf{X}_e\|$, it is clear that $W(\mathbf{X}_e)$ is an increasing function of $\|\mathbf{X}_e\|$ because the control gains are strictly positive. Therefore, there exist class $\mathcal{K}_\infty$ function of $\overline{W}(\overline{\mathbf{X}}_e)$ and of $\overline{W}_3(\overline{\mathbf{X}}_e)$ of a vector $\overline{\mathbf{X}}_e$ such that:

$$\overline{W}(\overline{\mathbf{X}}_e) = W(\mathbf{X}_e) - \frac{1}{4\varepsilon_2} - \frac{1}{4\varepsilon_3} - \frac{1}{4\varepsilon_4}, \tag{3-132}$$

$$\mathcal{L}\bar{V}_2 \le -cV_2 + \lambda, \tag{3-133}$$

where:

$$\bar{V}_2 = \overline{W}_2(\overline{\mathbf{X}}_e) + \frac{1}{2}\bar{\boldsymbol{\tau}}_{de,Aero}^{\mathrm{T}}\boldsymbol{\Gamma}^{-1}\bar{\boldsymbol{\tau}}_{de,Aero} + \frac{1}{2\mu_1}\tilde{\delta}_1^2 + \frac{1}{2\mu_2}\tilde{\delta}_2^2, \tag{3-134}$$

taking:

$$\mathbf{X}_e = \mathrm{col}(\boldsymbol{\eta}_{2e}, \overline{\boldsymbol{\omega}}_e), \quad \gamma_2\sqrt[4]{1+\|\overline{\boldsymbol{\omega}}_e\|^4} = \gamma_2\sqrt[4]{1+\|\boldsymbol{\omega}_e\|^4} - \frac{1}{\lambda_m(\mathbf{K}_1)}\left(\sum_{i=2}^{4}\frac{1}{4\varepsilon_i}\right), \tag{3-135}$$

then:

$$W_2(\overline{\mathbf{X}}_e) = \tfrac{\gamma_1}{2}\|\boldsymbol{q}_e\|^2 + \frac{\gamma_2}{2}\sqrt[2]{1+\|\overline{\boldsymbol{\omega}}_e\|^4} - \frac{\gamma_2}{2}, \tag{3-136}$$

$$\overline{W}(\overline{\mathbf{X}}_e) = -\gamma_1\|\boldsymbol{q}_e\|^2\left(\lambda_m(\mathbf{K}_1) - \frac{\varepsilon_1}{\gamma_1}\right) - \gamma_2\lambda_m(\mathbf{K}_2)\frac{\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.75}}, \tag{3-137}$$

since:

$$\mathcal{L}\bar{V}_2 \le -W(\overline{\mathbf{X}}_e) + \sum_{i=2}^{4}\frac{1}{4\varepsilon_i}, \tag{3-138}$$

from (3-132) we have

$$\mathcal{L}\bar{V}_2 \leq -\bar{W}(\bar{\mathbf{X}}_e) + \sum_{i=2}^{4} \frac{1}{4\varepsilon_i}, \tag{3-139}$$

from Theorem 2.6 we have:

$$P\left\{\lim_{t\to\infty} \bar{W}\big(\bar{\mathbf{X}}_e(t)\big) = 0\right\} = 1, \forall \boldsymbol{x}(t_0) \in \mathbb{R}^n, \tag{3-140}$$

since:

$$\bar{W}(\bar{\mathbf{X}}_e) = W(\mathbf{X}_e) - \sum_{i=2}^{4} \frac{1}{4\varepsilon_i}. \tag{3-141}$$

Thus, it can be stated that the closed loop system consisting of (3-105), (3-119) and the last two equations of (3-117) has a unique strong solution. In addition, the tracking error converges to a ball centred at the origin in probability.

3.4.1 Initial Condition and Control Gain Selection

It should be noted that under the conditions prescribed method 2 in the section 3.4 the selection of initial conditions and gains $\mathbf{K}_1, \mathbf{K}_2, \mathbf{\Gamma}_\tau^{-1}, \mu_1$ and μ_2 must be chosen through trial and error. However, it is possible to prove local exponential stability to a greater ball centred at the solution and in this case provide explicit conditions on all control gains and initial conditions. To do this we must first note that the use of the Modified Rodriquez Parameter implies that a singularity occurs at an angular position of $\pm n2\pi,\ n = 1,2,3 \ldots$ To address this we must recall that:

$$\boldsymbol{q}(\boldsymbol{\eta}_2) = \begin{bmatrix} \dfrac{\sin\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) - \cos\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right)}{1 + \cos\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) + \sin\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right)} \\ \dfrac{\cos\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) + \sin\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right)}{1 + \cos\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) + \sin\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right)} \\ \dfrac{\cos\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right) - \sin\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right)}{1 + \cos\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) + \sin\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right)} \end{bmatrix}. \quad (3\text{-}142)$$

The MRP experiences a singularity at the following conditions:

Euler Angle			MRP		
ϕ	θ	ψ	q_1	q_2	q_3
0	0	2π	Singularity	Singularity	Singularity
0	0	-2π	Singularity	Singularity	Singularity
0	2π	0	Singularity	Singularity	Singularity
0	-2π	0	Singularity	Singularity	Singularity
2π	0	0	Singularity	Singularity	Singularity
-2π	0	0	Singularity	Singularity	Singularity
2π	2π	2π	Singularity	Singularity	Singularity
2π	2π	-2π	Singularity	Singularity	Singularity
2π	-2π	2π	Singularity	Singularity	Singularity
-2π	2π	2π	Singularity	Singularity	Singularity
2π	-2π	-2π	Singularity	Singularity	Singularity
-2π	2π	-2π	Singularity	Singularity	Singularity
-2π	-2π	2π	Singularity	Singularity	Singularity
-2π	-2π	-2π	Singularity	Singularity	Singularity
π	π	$-\pi$	Singularity	Singularity	Singularity
π	$-\pi$	π	Singularity	Singularity	Singularity
$-\pi$	π	π	Singularity	Singularity	Singularity
$-\pi$	$-\pi$	$-\pi$	Singularity	Singularity	Singularity

Table 3-1: MRP Regions of Singularity.

Therefore, it can be stated:

$$|\boldsymbol{\eta}_2| \leq b_1 < 2\pi, \quad (3\text{-}143)$$
$$|\boldsymbol{q}(\boldsymbol{\eta}_2)| \leq b_2, \quad (3\text{-}144)$$
$$|\boldsymbol{q}| \leq b_2, \quad (3\text{-}145)$$
$$|\boldsymbol{q}_e| \leq b_2, \quad (3\text{-}146)$$

hence we wish to obtain:

$$V_2 \le b_3 \le b_2, \tag{3-147}$$

recall from (3-109) the Lyapunov function:

$$V_2 = \frac{\gamma_1}{2}\|\boldsymbol{q}_e\|^2 + \frac{\gamma_2}{2}\sqrt[2]{1+\|\boldsymbol{\omega}_e\|^4} + \frac{1}{2}\bar{\boldsymbol{\tau}}_{de,Aero}^{\mathrm{T}}\boldsymbol{\Gamma}^{-1}\bar{\boldsymbol{\tau}}_{de,Aero} + \frac{1}{2\mu_1}\tilde{\delta}_1^2 + \frac{1}{2\mu_2}\tilde{\delta}_2^2 - \frac{\gamma_2}{2}. \tag{3-148}$$

We can now state:

$$\mathcal{L}V_2 \le -c\left(\|\boldsymbol{q}_e\|^2 + \|\boldsymbol{\omega}_e\|^2 + \left\|\bar{\boldsymbol{\tau}}_{de,Aero}(t)\right\|^2 + \left\|\tilde{\delta}_1(t)\right\|^2 + \left\|\tilde{\delta}_2(t)\right\|^2\right) + \sum_{i=2}^{4}\frac{1}{4\varepsilon_i}, \tag{3-149}$$

recall from (3-126) and (3-127)

$$\mathcal{L}V_2 \le -cV_2 + \lambda, \tag{3-150}$$

therefore, we can state:

$$V_2(t) \le V_2(t_0) + \frac{\lambda}{c} := b_3. \tag{3-151}$$

This implies that:

$$\|\boldsymbol{q}_e(t)\| \le b_3, \qquad \|\boldsymbol{\omega}_e(t)\| \le b_3. \tag{3-152}$$

Furthermore, the total angular position error obeys:

$$\|\boldsymbol{q}_e(t)\| \le \left\|\boldsymbol{q}_e(t), \boldsymbol{\omega}_e(t), \bar{\boldsymbol{\tau}}_{de}(t), \tilde{\delta}(t)\right\| + \sqrt{\frac{\lambda}{c}} \tag{3-153}$$

Furthermore, it can be stated:

$$\alpha_p = \begin{bmatrix}1\\0\\0\end{bmatrix}^{\mathrm{T}} \mathbf{R}_2^{-1}(\boldsymbol{q})\left(\mathbf{R}_2(\boldsymbol{q}_d)\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}\dot{\boldsymbol{\eta}}_{2d} - \mathbf{K}_3(\boldsymbol{q}-\boldsymbol{q}_d)\right), \tag{3-154}$$

$$\alpha_q = \begin{bmatrix}0\\1\\0\end{bmatrix}^{\mathrm{T}} \mathbf{R}_2^{-1}(\boldsymbol{q})\left(\mathbf{R}_2(\boldsymbol{q}_d)\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}\dot{\boldsymbol{\eta}}_{2d} - \mathbf{K}_3(\boldsymbol{q}-\boldsymbol{q}_d)\right), \tag{3-155}$$

$$\alpha_r = \begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}} \mathbf{R}_2^{-1}(\boldsymbol{q})\left(\mathbf{R}_2(\boldsymbol{q}_d)\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}\dot{\boldsymbol{\eta}}_{2d} - \mathbf{K}_3(\boldsymbol{q}-\boldsymbol{q}_d)\right), \tag{3-156}$$

and therefore state that:

$$\|\alpha_p\| \le \alpha_p^{\mathrm{M}}, \qquad \|\alpha_q\| \le \alpha_q^{\mathrm{M}}, \qquad \|\alpha_r\| \le \alpha_r^{\mathrm{M}}, \qquad \|\boldsymbol{\alpha}_{\boldsymbol{\omega}}\| \le \alpha_{\boldsymbol{\omega}}^{\mathrm{M}}. \tag{3-157}$$

Taking the derivatives of both sides of (3-154), (3-155) and (3-156) we obtain:

$$\dot{\boldsymbol{\alpha}}_\omega = \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\eta}_2}\dot{\boldsymbol{\eta}}_{2d} + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\eta}_{2d}}\dot{\boldsymbol{\eta}}_{2d} + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \dot{\boldsymbol{\eta}}_{2d}}\ddot{\boldsymbol{\eta}}_{2d}, \tag{3-158}$$

where we define:

$$\frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{q}} = \frac{64\boldsymbol{\alpha}_\omega \boldsymbol{q}^{\mathrm{T}}}{(1+\|\boldsymbol{q}\|^2)^3} - \mathbf{R}_2^{-1}(\boldsymbol{q})\mathbf{K}_3 + \frac{8}{(1+\|\boldsymbol{q}\|^2)^2} * \Bigg(\boldsymbol{q}^{\mathrm{T}}\big(\mathbf{R}_2(\boldsymbol{q}_d)\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}\dot{\boldsymbol{\eta}}_{2d} - \mathbf{K}_3(\boldsymbol{q}-\boldsymbol{q}_d)\big)\mathbf{I}_{3\times 3} + \boldsymbol{q}\big(\mathbf{R}_2(\boldsymbol{q}_d)\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}\dot{\boldsymbol{\eta}}_{2d} - \mathbf{K}_3(\boldsymbol{q}-\boldsymbol{q}_d)\big)^{\mathrm{T}} - \big(\mathbf{R}_2(\boldsymbol{q}_d)\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}\dot{\boldsymbol{\eta}}_{2d} - \mathbf{K}_3(\boldsymbol{q}-\boldsymbol{q}_d)\big)\boldsymbol{q}^{\mathrm{T}} - \Bigg(\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \left(\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}^T \big(\mathbf{R}_2(\boldsymbol{q}_d)\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}\dot{\boldsymbol{\eta}}_{2d} - \mathbf{K}_3(\boldsymbol{q}-\boldsymbol{q}_d)\big)\right) + \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \left(\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}^T \big(\mathbf{R}_2(\boldsymbol{q}_d)\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}\dot{\boldsymbol{\eta}}_{2d} - \mathbf{K}_3(\boldsymbol{q}-\boldsymbol{q}_d)\big)\right) + \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \left(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^T \big(\mathbf{R}_2(\boldsymbol{q}_d)\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}\dot{\boldsymbol{\eta}}_{2d} - \mathbf{K}_3(\boldsymbol{q}-\boldsymbol{q}_d)\big)\right)\Bigg)\Bigg), \tag{3-159}$$

$$\frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\eta}_2} = \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{q}}\mathbf{T}(\boldsymbol{\eta}_2)^{-1}, \tag{3-160}$$

$$\frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{q}_d} = \mathbf{R}_2^{-1}(\boldsymbol{q})\Bigg(\Bigg(\boldsymbol{q}_d^{\mathrm{T}}\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}\dot{\boldsymbol{\eta}}_{2d}\mathbf{I}_{3\times 3} + \boldsymbol{q}_d(\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}\dot{\boldsymbol{\eta}}_{2d})^{\mathrm{T}} - \mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}\dot{\boldsymbol{\eta}}_{2d}\boldsymbol{q}_d^{\mathrm{T}} + \mathbf{K}_3 + \left(\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \left(\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}^T (\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}\dot{\boldsymbol{\eta}}_{2d})\right) + \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} * \left(\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}^T (\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}\dot{\boldsymbol{\eta}}_{2d})\right) + \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \left(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^T (\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}\dot{\boldsymbol{\eta}}_{2d})\right)\right)\Bigg)\Bigg), \tag{3-161}$$

$$\frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\eta}_{2d}} = \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{q}_d}\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1} + \mathbf{R}_2^{-1}(\boldsymbol{q})\mathbf{R}_2(\boldsymbol{\alpha}_q)\left(\begin{bmatrix} 0 & 0 & 0 \\ 0 & -\sin(\phi_d) & \cos(\theta_d)\cos(\phi_d) \\ 0 & -\cos(\phi_d) & -\cos(\theta_d)\sin(\phi_d)\end{bmatrix}\dot{\boldsymbol{\eta}}_{2d}\begin{bmatrix}1\\0\\0\end{bmatrix}^{\mathrm{T}} + \begin{bmatrix} 0 & 0 & -\cos(\theta_d) \\ 0 & 0 & -\sin(\theta_d)\sin(\phi_d) \\ 0 & 0 & -\sin(\theta_d)\cos(\phi_d)\end{bmatrix}\dot{\boldsymbol{\eta}}_{2d}\begin{bmatrix}0\\1\\0\end{bmatrix}^{\mathrm{T}}\right), \tag{3-162}$$

$$\frac{\partial \boldsymbol{\alpha}_\omega}{\partial \dot{\boldsymbol{\eta}}_{2d}} = \mathbf{R}_2^{-1}(\boldsymbol{q})\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\eta}_{2d})^{-1}. \tag{3-163}$$

in conjunction with which Assumption 3.4 and Assumption 3.5, it can be said that:

$$\sup_{t\in R^n}\left\|\frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \boldsymbol{q}(t)}\right\| \le \varrho_6, \quad \sup_{t\in R^n}\left\|\frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \boldsymbol{\eta}_2(t)}\right\| \le \varrho_7, \quad \sup_{t\in R^n}\left\|\frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \boldsymbol{q}_d(t)}\right\| \le \varrho_8,$$
$$\sup_{t\in R^n}\left\|\frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \boldsymbol{\eta}_{2d}(t)}\right\| \le \varrho_9, \quad \sup_{t\in R^n}\left\|\frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \dot{\boldsymbol{\eta}}_{2d}(t)}\right\| \le \varrho_{10}. \tag{3-164}$$

Therefore, it can be said that:

$$\sup_{t\in R^n}\|\dot{\boldsymbol{\alpha}}_\omega(t)\| \le \dot{\boldsymbol{\alpha}}_\omega^{\mathrm{M}}, \tag{3-165}$$

hence we can now state:

$$\dot{\alpha}_p = \begin{bmatrix}1\\0\\0\end{bmatrix}^T \dot{\boldsymbol{\alpha}}_\omega, \quad \sup_{t\in R^n}\|\dot{\alpha}_p\| \le \dot{\alpha}_p^{\mathrm{M}}, \tag{3-166}$$

$$\dot{\alpha}_q = \begin{bmatrix}0\\1\\0\end{bmatrix}^T \dot{\boldsymbol{\alpha}}_\omega, \quad \sup_{t\in R^n}\|\dot{\alpha}_q\| \le \dot{\alpha}_q^{\mathrm{M}}, \tag{3-167}$$

$$\dot{\alpha}_r = \begin{bmatrix}0\\0\\1\end{bmatrix}^T \dot{\boldsymbol{\alpha}}_\omega, \quad \sup_{t\in R^n}\|\dot{\alpha}_r\| \le \dot{\alpha}_r^{\mathrm{M}}. \tag{3-168}$$

By using (3-166) to (3-168) we can write the following:

$$\left\|\boldsymbol{\omega}(t_0) - \mathbf{R}_2^{-1}(\boldsymbol{q}(t_0))\left(\mathbf{R}_2\left(\boldsymbol{\alpha}_q(t_0)\right)\mathbf{T}(\boldsymbol{\eta}_{2d}(t_0))^{-1}\dot{\boldsymbol{\eta}}_{2d}(t_0) - \mathbf{K}_3(\boldsymbol{q}(t_0) - \boldsymbol{q}_d(t_0))\right)\right\| < \sqrt[4]{\left(b_3 - \frac{\lambda}{c} - \|\boldsymbol{q}_e(t_0)\|^2 - \|\bar{\boldsymbol{\tau}}_{de,Aero}(t)\|^2 - \|\tilde{\delta}_1(t)\|^2 - \|\tilde{\delta}_2(t)\|^2 + 1\right)^2 - 1}, \tag{3-169}$$

where $\bar{\boldsymbol{\tau}}_{de,p}$, $\bar{\boldsymbol{\tau}}_{de,p}$ and $\bar{\boldsymbol{\tau}}_{de,p}$ represent the deterministic aerodynamic disturbance torque errors about the aircraft's body-fixed *frame*.

To ensure that (3-169) is well defined we need to choose the initial pitch, roll and yaw, errors as follows:

$$\|\boldsymbol{q}_e(t_0)\| \le \sqrt[2]{1 + b_3 - \frac{\lambda}{c}}, \tag{3-170}$$

$$\|\mathbf{K}_3\| < \left(\sqrt[4]{\left(b_3 - \frac{\lambda}{c} - \|\boldsymbol{q}_e(t_0)\|^2\right)^2 - \left\|\bar{\boldsymbol{\tau}}_{de,Aero}(t_0)\right\|^2 - \sum_{i=1}^{2}\left\|\tilde{\delta}_i(t_0)\right\|^2 - 1 - \|\boldsymbol{\omega}(t_0)\|} - \left\|\mathbf{R}_2^{-1}(\boldsymbol{q}(t_0))\left(\mathbf{R}_2\left(\boldsymbol{\alpha}_q(t_0)\right)\mathbf{T}(\boldsymbol{\eta}_{2d}(t_0))^{-1}\dot{\boldsymbol{\eta}}_{2d}(t_0)\right)\right\|\right)\frac{1}{\left\|\mathbf{R}_2^{-1}(\boldsymbol{q}(t_0))\right\|\|\boldsymbol{q}_e(t_0)\|} \tag{3-171}$$

Thus, we have a way of choosing the initial conditions and control gains. Furthermore, to ensure that actuator saturation does not occur we consider the maximum norm of (3-117).

$$\boldsymbol{\tau} = \left\|\mathbf{I}_{\mathrm{H}}\left[\mathbf{I}_H^{-1}(\boldsymbol{\omega}\times\mathbf{I}_H\boldsymbol{\omega}) + \dot{\boldsymbol{\alpha}}_{\omega} - \mathbf{K}_2\boldsymbol{\omega}_e - \left(\frac{\varepsilon_2}{\gamma_2}\left(\frac{1}{4\varepsilon_5}\right)^2\frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{3.5}} + \frac{1}{\gamma_2}\frac{(\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e)}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}} + \frac{9}{4}\gamma_2\left(\varepsilon_3\frac{\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\boldsymbol{\theta}\right\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}\hat{\delta}_1^M + \varepsilon_4\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\right\|\hat{\delta}_2^M\right)\right)\boldsymbol{\omega}_e - \mathbf{I}_H^{-1}\hat{\bar{\boldsymbol{\tau}}}_{Aero}\right]\right\| \le \left\|\begin{bmatrix}U_2^M\\U_3^M\\U_4^M\end{bmatrix}\right\|. \tag{3-172}$$

This gives:

$$\left\|\frac{I_z - I_y}{I_x}\right\| q^M r^M + \frac{1}{I_x}\hat{\bar{\boldsymbol{\tau}}}_p^M + \dot{\alpha}_p^{\mathrm{M}} + \frac{1}{\gamma_2}\|\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{q})\| + \left(\left(\mathbf{K}_{2,11} - \frac{\varepsilon_2}{\gamma_2}\left(\frac{1}{4\varepsilon_5}\right)^2\frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1+\|r_e^M\|^4)^{3.5}} - \frac{9}{4}\gamma_2\left(\varepsilon_3\frac{\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\boldsymbol{\theta}\right\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}\hat{\delta}_1^M + \varepsilon_4\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\right\|\hat{\delta}_2^M\right)\right)p_e^M \le \frac{U_2^M}{I_x}, \tag{3-173}$$

$$\left\|\frac{I_x - I_z}{I_y}\right\| r^M p^M + \frac{1}{I_y}\hat{\bar{\boldsymbol{\tau}}}_q^M + \dot{\alpha}_q^{\mathrm{M}} + \frac{1}{\gamma_2}\|\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{q})\| + \left(\mathbf{K}_{2,22} - \frac{\varepsilon_2}{\gamma_2}\left(\frac{1}{4\varepsilon_5}\right)^2\frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1+\|r_e^M\|^4)^{3.5}} - \frac{9}{4}\gamma_2\left(\varepsilon_3\frac{\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\boldsymbol{\theta}\right\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}\hat{\delta}_1^M + \varepsilon_4\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\right\|\hat{\delta}_2^M\right)\right)q_e^M \le \frac{U_3^M}{I_y}, \tag{3-174}$$

$$\left\|\frac{I_y I_x -}{I_z}\right\| p^M q^M + \frac{1}{I_z}\hat{\bar{\boldsymbol{\tau}}}_r^M + \dot{\alpha}_r^{\mathrm{M}} + \frac{1}{\gamma_2}\|\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{q})\| + \left(\mathbf{K}_{2,33} - \frac{\varepsilon_2}{\gamma_2}\left(\frac{1}{4\varepsilon_5}\right)^2\frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1+\|r_e^M\|^4)^{3.5}} - \frac{9}{4}\gamma_2\left(\varepsilon_3\frac{\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\boldsymbol{\theta}\right\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}\hat{\delta}_1^M + \varepsilon_4\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\right\|\hat{\delta}_2^M\right)\right)r_e^M \le \frac{U_4^M}{I_z}. \tag{3-175}$$

Rearranging (3-173), (3-174) and (3-175) gives:

$$0 < \mathbf{K}_{2,11} \leq \frac{\frac{U_2^M}{I_x} - \left(\left\|\frac{I_z - I_y}{I_x}\right\| q^M r^M + \frac{1}{I_x}\hat{\boldsymbol{\tau}}_p^M\right)}{p_e^M} - \frac{1}{p_e^M}\left(\dot{\alpha}_p^M + \frac{1}{\gamma_2}\|\boldsymbol{q}_e^T \mathbf{R}_2(\boldsymbol{q})\| p_e^M - \frac{\varepsilon_2}{\gamma_2}\left(\frac{1}{4\varepsilon_5}\right)^2 \frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1 + \|r_e^M\|^4)^{3.5}} - \frac{9}{4}\gamma_2\left(\varepsilon_3 \frac{\left\|(\mathbf{I}_A + \mathbf{I}_H)^{-1}\boldsymbol{\theta}\right\|^4}{(1 + \|\boldsymbol{\omega}_e\|^4)^{0.5}}\hat{\delta}_1^M + \varepsilon_4\left\|(\mathbf{I}_A + \mathbf{I}_H)^{-1}\right\|\hat{\delta}_2^M\right) p_e^M\right), \tag{3-176}$$

$$0 < \mathbf{K}_{2,22} \leq \frac{\frac{U_3^M}{I_y} - \left(\left\|\frac{I_x - I_z}{I_y}\right\| r^M p^M + \frac{1}{I_y}\hat{\boldsymbol{\tau}}_q^M\right)}{q_e^M} - \frac{1}{q_e^M}\left(\dot{\alpha}_q^M + \frac{1}{\gamma_2}\|\boldsymbol{q}_e^T \mathbf{R}_2(\boldsymbol{q})\| q_e^M - \frac{\varepsilon_2}{\gamma_2}\left(\frac{1}{4\varepsilon_5}\right)^2 \frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1 + \|r_e^M\|^4)^{3.5}} - \frac{9}{4}\gamma_2\left(\varepsilon_3 \frac{\left\|(\mathbf{I}_A + \mathbf{I}_H)^{-1}\boldsymbol{\theta}\right\|^4}{(1 + \|\boldsymbol{\omega}_e\|^4)^{0.5}}\hat{\delta}_1^M + \varepsilon_4\left\|(\mathbf{I}_A + \mathbf{I}_H)^{-1}\right\|\hat{\delta}_2^M\right) q_e^M\right), \tag{3-177}$$

$$0 < \mathbf{K}_{2,33} \leq \frac{\frac{U_4^M}{I_z} - \left(\left\|\frac{I_y - I_x}{I_z}\right\| p^M q^M + \frac{1}{I_z}\hat{\boldsymbol{\tau}}_r^M\right)}{r_e^M} - \frac{1}{r_e^M}\left(\dot{\alpha}_r^M + \frac{1}{\gamma_2}\|\boldsymbol{q}_e^T \mathbf{R}_2(\boldsymbol{q})\| q_e^M - \frac{\varepsilon_2}{\gamma_2}\left(\frac{1}{4\varepsilon_5}\right)^2 \frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1 + \|r_e^M\|^4)^{3.5}} - \frac{9}{4}\gamma_2\left(\varepsilon_3 \frac{\left\|(\mathbf{I}_A + \mathbf{I}_H)^{-1}\boldsymbol{\theta}\right\|^4}{(1 + \|\boldsymbol{\omega}_e\|^4)^{0.5}}\hat{\delta}_1^M + \varepsilon_4\left\|(\mathbf{I}_A + \mathbf{I}_H)^{-1}\right\|^4\hat{\delta}_2^M\right) r_e^M\right), \tag{3-178}$$

Therefore, the control objective 3.2 has been met. In addition we have meet the second and third point of the project aims depicted in section 1.3.

4 Multi-rotor Control

In this chapter, the control design and stability analysis of the quadrotor aircraft is presented. This chapter is split up into two separate sections Firstly we will consider a deterministic system and design the controls to stabilise the aircraft accordingly. Second a stochastic system is considered and a set of control algorithms are designed to stochastically practically asymptotically stabilise the aircraft.

The first section is broken up into three sub-sections. Firstly, the model assumptions for the deterministic system are presented. Secondly, the design of a set of controls to stabilise the linear position system is presented. These controls are then used to design the control laws to stabilise the attitude system. Finally, stability analysis and specification of the initial conditions and control gains are presented for the completed control system.

The Second section is also broken up into three sub-sections in this section the new stochastic controller will be presented. Firstly, the model assumptions for the stochastic system are presented. Secondly, the design of a set of controls to stabilise the linear position system with the aid of output state feedback is presented. These controls are then used to design the control laws to stabilise the attitude system. Finally, stability analysis and specification of the initial conditions and control gains are presented for the completed control system.

4.1 Deterministic State Feedback Control Design.

In this section the design of a control law to stabilise a quadrotor aircraft operating in a purely deterministic environment is presented and analysed.

4.1.1 Model Assumptions

Assumption 4.1
Assume that the aeronautic force disturbance vector $\boldsymbol{f}_{Aero}$ and its derivative are bounded, i.e., there exist nonnegative constants such that:

$$\sup_{t\in R^n}\|\boldsymbol{f}_{Aero}(t)\| \leq \varrho_f, \qquad \sup_{t\in R^n}\left\|\dot{\boldsymbol{f}}_{Aero\boldsymbol{\eta}}(t)\right\| \leq \varrho_{1f}, \forall t \geq t_0 \geq 0. \tag{4-1}$$

Furthermore, there exists constants ϱ_{f1}, ϱ_{f2} and ϱ_{f3} such that:

$$\sup_{t\in R^n}\left\|\boldsymbol{f}_{Aero,x}(t)\right\| \leq \varrho_{f1}, \sup_{t\in R^n}\left\|\boldsymbol{f}_{Aero,y}(t)\right\| \leq \varrho_{f2} \sup_{t\in R^n}\left\|\boldsymbol{f}_{Aero,z}(t)\right\| \leq \varrho_{f3}, \tag{4-2}$$

and constants ϱ_{f4}, ϱ_{f5} and ϱ_{f6} such that

$$\sup_{t\in R^n}\left\|\dot{\boldsymbol{f}}_{Aero,\boldsymbol{x}}(t)\right\| \leq \varrho_{f4}, \sup_{t\in R^n}\left\|\dot{\boldsymbol{f}}_{Aero,y}(t)\right\| \leq \varrho_{f5}, \sup_{t\in R^n}\left\|\dot{\boldsymbol{f}}_{Aero,\boldsymbol{z}}(t)\right\| \leq \varrho_{f6}. \tag{4-3}$$

Assumption 4.2
Assume that the aerodynamic torque disturbance vector $\boldsymbol{\tau}_{Aero}$ and its derivative are bounded, i.e., there exist nonnegative constants such that:

$$\sup_{t\in R^n}\|\boldsymbol{\tau}_{Aero}(t)\| \leq \varrho_\tau, \qquad \sup_{t\in R^n}\left\|\dot{\boldsymbol{\tau}}_{Aero,\boldsymbol{\eta}}(t)\right\| \leq \varrho_{1\tau}, \forall t \geq t_0 \geq 0. \tag{4-4}$$

Furthermore, there exists constants $\varrho_{\tau1}$, $\varrho_{\tau2}$ and $\varrho_{\tau3}$ such that:

$$\sup_{t\in R^n}\left\|\boldsymbol{\tau}_{Aero,\boldsymbol{p}}(t)\right\| \leq \varrho_{\tau1}, \sup_{t\in R^n}\left\|\boldsymbol{\tau}_{Aero,\boldsymbol{q}}(t)\right\| \leq \varrho_{\tau2} \sup_{t\in R^n}\left\|\boldsymbol{\tau}_{Aero,\boldsymbol{r}}(t)\right\| \leq \varrho_{\tau3}, \tag{4-5}$$

and constants $\varrho_{\tau4}$, $\varrho_{\tau5}$ and $\varrho_{\tau6}$ such that

$$\sup_{t\in R^n}\left\|\dot{\boldsymbol{\tau}}_{Aero,\boldsymbol{p}}(t)\right\| \leq \varrho_{\tau4}, \sup_{t\in R^n}\left\|\dot{\boldsymbol{\tau}}_{Aero,\boldsymbol{q}}(t)\right\| \leq \varrho_{\tau5}, \sup_{t\in R^n}\left\|\dot{\boldsymbol{\tau}}_{Aero,\boldsymbol{r}}(t)\right\| \leq \varrho_{\tau6}. \tag{4-6}$$

Assumption 4.3

Assume that the reference position trajectory $\boldsymbol{\eta}_{1d}(t) = [\boldsymbol{\eta}_d(t) \quad y_d(t) \quad \boldsymbol{z}_d(t)]^{\mathrm{T}}$ *is sufficiently smooth, i.e., its first four derivatives exist and are bounded, that is there exist non-negative constants* $\varrho_1,\ \varrho_2,\ \varrho_3$ *and* ϱ_4 *such that:*

$$\sup_{t\in R^n}\|\dot{\boldsymbol{\eta}}_{1d}(t)\| \le \varrho_1, \sup_{t\in R^n}\|\ddot{\boldsymbol{\eta}}_{1d}(t)\| \le \varrho_2, \sup_{t\in R^n}\|\dddot{\boldsymbol{\eta}}_{1d}(t)\| \le \varrho_3, \sup_{t\in R^n}\left\|\ddddot{\boldsymbol{\eta}}_{1d}(t)\right\| \le \varrho_4. \tag{4-7}$$

Furthermore, the second derivative of $\boldsymbol{z}_d(t)$ *is assumed strictly less than* g *and the aerodynamic disturbance force acting along the vertical axis of the E-frame:*

$$\sup_{t\in R^n}|\ddot{\boldsymbol{z}}_d(t)| \le g - \varrho_5 - \varrho_{f3}, \tag{4-8}$$

where ϱ_5 *is a strictly positive constant as to* ϱ_{f3} *which is defined in* (4-2)*, the reference yaw angle* $\psi_d(t)$ *is also assumed as being sufficiently smooth. That is the first two derivatives exist and are bounded, i.e. there are non-negative constants* ϱ_6 *and* ϱ_7 *exist such that:*

$$\sup_{t\in R^n}\left\|\dot{\psi}_d(t)\right\| \le \varrho_6, \sup_{t\in R^n}\left\|\ddot{\psi}_d(t)\right\| \le \varrho_7, \tag{4-9}$$

Assumption 4.4

Assume that reference yaw trajectory is bounded by a positive constant ϱ_8*, between* $\pm 2\pi$ *such that*

$$\sup_{t\in R^n}|\psi_d(t)| \le \varrho_8, \qquad \varrho_8 < 2\pi. \tag{4-10}$$

Under the above assumptions, it is possible to design the control inputs U_1, U_2, U_3 and U_4 in such a manner that the linear position $\boldsymbol{\eta}_1(t)$ and yaw angle $\psi(t)$ of the aircraft asymptotically track their reference trajectories $\boldsymbol{\eta}_{1d}(t)$ and $\psi_d(t)$, i.e.,

$$\lim_{t\to\infty}\big(\boldsymbol{\eta}_1(t) - \boldsymbol{\eta}_{1d}(t)\big) = 0, \qquad \lim_{t\to\infty}\big(\psi_1(t) - \psi_{1d}(t)\big) = 0. \tag{4-11}$$

While keeping all other states of the aircraft dynamics bounded for all initial conditions $\boldsymbol{\eta}_1(t_0) \in R^n$, $\boldsymbol{v}_1(t_0) \in R^n$, $\boldsymbol{\eta}_2(t_0) \in R^n$, $\boldsymbol{q}(t_0) \in R^n$ and $\boldsymbol{\omega}(t_0) \in R^n$. The condition (4-8) implies that the aircraft is not allowed to descend faster than it freely falls under the gravitational force. In a practical sense, this implies that the aircraft's propellers cannot spin in the opposing direction to that which provides lift nor can the quadrotor fly in an inverted manner. This condition is essential when designing the control scheme for the under actuated quadrotor. The system states are defined as shown below:

$$\boldsymbol{\eta}_1 = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \ \boldsymbol{v}_1 = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}, \ \boldsymbol{\eta}_2 = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}, \ \boldsymbol{q} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix}, \ \boldsymbol{\omega} = \begin{bmatrix} p \\ q \\ r \end{bmatrix}, \ \boldsymbol{f} = [U_1], \ \boldsymbol{\tau} = \begin{bmatrix} U_2 \\ U_3 \\ U_4 \end{bmatrix}, \tag{4-12}$$

where $\boldsymbol{\eta}_1$ denotes the linear position of the system, while $\boldsymbol{v}_1$ denotes the linear velocity of the system $\boldsymbol{\eta}_2$ denotes the attitude of the quadrotor through the use of Euler angles, while $\boldsymbol{q}$ denotes the attitude of the aircraft through the use of MRP and $\boldsymbol{\omega}$ denotes the angular velocity of the quadrotor with respect to the predefined body frame of reference. Referring back to (2-114) and (2-115) the system of equations governing the aircraft are as follows:

$$s_1 = \begin{cases} \dot{\boldsymbol{\eta}}_1 = \boldsymbol{\nu}_1, \\ \dot{\boldsymbol{\nu}}_1 = -\mathbf{D}_1\boldsymbol{\nu}_1 - g\boldsymbol{e}_3 + \frac{1}{m} f\mathbf{R}_1(\boldsymbol{q}) + \frac{1}{m}\bar{\boldsymbol{f}}_{Aero}, \end{cases} \quad (4\text{-}13)$$

$$s_2 = \begin{cases} \dot{\boldsymbol{q}} = \mathbf{R}_2(\boldsymbol{q})\mathbf{T}(\boldsymbol{\eta}_2)^{-1}\dot{\boldsymbol{\eta}}_2, \\ \dot{\boldsymbol{q}} = \mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}, \\ \dot{\boldsymbol{\omega}} = (\mathbf{I}_A + \mathbf{I}_H)^{-1}\left(-\mathbf{D}_2\boldsymbol{\omega} - \mathbf{C}_{A21}(\boldsymbol{v}^b)\boldsymbol{v}^b - \mathbf{C}_{B22}(\boldsymbol{\omega})\boldsymbol{\omega} - \mathbf{C}_{A22}(\boldsymbol{\omega})\boldsymbol{\omega} + \boldsymbol{\tau} + \bar{\boldsymbol{\tau}}_{Aero}\right). \end{cases} \quad (4\text{-}14)$$

Listed below is the system inertia matrix:

$$\mathbf{I}_H = \begin{bmatrix} I_X & 0 & 0 \\ 0 & I_Y & 0 \\ 0 & 0 & I_Z \end{bmatrix} \quad (4\text{-}15)$$

The transformation between the B frame and E frame angular velocities is as shown below:

$$\dot{\boldsymbol{\eta}}_2 \mathbf{T}(\boldsymbol{\eta}_2)\, \boldsymbol{\omega}, \quad (4\text{-}16)$$

$$\mathbf{T}(\boldsymbol{\eta}_2) = \begin{bmatrix} 1 & s_\phi t_\theta & c_\phi t_\theta \\ 0 & c_\phi & -s_\phi \\ 0 & \frac{s_\phi}{c_\theta} & \frac{c\phi}{c_\theta} \end{bmatrix}, \quad (4\text{-}17)$$

$$\mathbf{T}(\boldsymbol{\eta}_2)^{-1} = \begin{bmatrix} 1 & 0 & -s_\theta \\ 0 & c_\phi & c_\theta s_\phi \\ 0 & -s_\phi & c_\theta c_\phi \end{bmatrix}, \quad (4\text{-}18)$$

$$\mathbf{R}_1(\boldsymbol{q}) = \mathbf{R}(\boldsymbol{\eta}_2) = \begin{bmatrix} c_\psi c_\theta & -s_\psi c_{v_\phi} + c_\psi s_\theta s_\phi & s_\psi s_\phi + c_\psi s_\theta c_\phi \\ s_\psi c_\theta & c_\psi c_{v_\phi} + s_\psi s_\theta s_\phi & -c_\psi s_\phi + s_\psi s_\theta c_\phi \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{bmatrix}, \quad (4\text{-}19)$$

The control design consists of two stages. In the first stage, the two equations of (4-13)will be considered, using the newly presented one-step ahead backstepping method [41]. During this stage the design of the total thrust f and virtual controls for the roll and pitch angles is presented. Moreover, they are designed to ensure asymptotic stabilisation of the tracking error vector $\boldsymbol{\eta}_1(t) - \boldsymbol{\eta}_{1d}(t)$ at the origin in the presence of matched external disturbances. In the second stage, the last two equations of (4-14)along with the standard backstepping method [27] will be used to design the control moment vector $\boldsymbol{\tau}$. By applying the standard backstepping method, the control vector $\boldsymbol{\tau}$ will ensure boundedness in the presence of external disturbances for stabilizing the attitude and tracking error $\psi_1(t) - \psi_{1d}(t)$ and the errors between the virtual controls of the roll and pitch angles and system attitude values at the origin.

Control objective 4.1

Based on the above assumptions there is two objectives to achieve:

1. Under Assumption 4.1 to Assumption 4.4, design the control inputs f and $\boldsymbol{\tau} = [U_2 \quad U_3 \quad U_4]$ along with estimates for the disturbances $\boldsymbol{f}_{Aero}$ and $\boldsymbol{\tau}_{Aero}$ to force the position vector $\boldsymbol{\eta}_1$ and yaw angle ψ of the aircraft to asymptotically track $\boldsymbol{\eta}_{1d}$ d and ψ_d in probability while keep all other states of the aircraft bounded.
2. Keep the aircraft's attitude between the range of $\pm 2\pi$.

4.1.2 Linear Position Tracking.

In this stage, we are concerned with the two equations of (4-13) because we are considering an output feedback system were the liner position is known and the liner velocity is unknown we will define the error of the state and state estimate as:

$$\tilde{\boldsymbol{\eta}}_1 = \boldsymbol{\eta}_1 - \hat{\boldsymbol{\eta}}_1, \tag{4-20}$$
$$\tilde{\boldsymbol{\nu}}_1 = \boldsymbol{\nu}_1 - \hat{\boldsymbol{\nu}}_1. \tag{4-21}$$

We will define the derivative of the linear position estimate as follows:

$$\dot{\hat{\boldsymbol{\eta}}}_1 = \hat{\boldsymbol{\nu}}_1 + \mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1 + \boldsymbol{h}_1. \tag{4-22}$$

We will define the derivative of the linear velocity estimate latter. In the meantime, we will define the tracking error as:

$$\begin{aligned} \boldsymbol{\eta}_{1e} &= \boldsymbol{\eta}_1 - \boldsymbol{\eta}_{1d}, \\ \boldsymbol{\nu}_{1e} &= \hat{\boldsymbol{\nu}}_1 - \boldsymbol{\alpha}_1. \end{aligned} \tag{4-23}$$

Where $\boldsymbol{\eta}_{1e}$ denotes the tracking error between the desired position $\boldsymbol{\eta}_{1d}$ and current position $\boldsymbol{\eta}_1$ in the second equation $\boldsymbol{\nu}_{1e}$ denotes the tracking error between the estimate of the systems linear velocity and virtual control signal $\boldsymbol{\alpha}_1$ of the velocity. The first step of the design process is to design $\boldsymbol{\alpha}_1$ to stabilise the tracking error of $\boldsymbol{\eta}_{1e}(t)$ and $\boldsymbol{\nu}_{1e}(t)$ at the origin. By substituting (4-23) into the first equation of (4-13) we obtain:

$$\dot{\boldsymbol{\eta}}_{1e} = \tilde{\boldsymbol{\nu}}_1 + \boldsymbol{\nu}_{1e} + \boldsymbol{\alpha}_1 - \dot{\boldsymbol{\eta}}_{1d}. \tag{4-24}$$

Therefore, we will choose the derivative of the estimate of the liner velocity:

$$\dot{\hat{\boldsymbol{\nu}}}_1 = -\mathbf{D}_1\hat{\boldsymbol{\nu}}_1 - g\boldsymbol{e}_3 + \frac{f}{m}\mathbf{R}(\boldsymbol{\eta}_2)\boldsymbol{e}_3 + \mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1 + \boldsymbol{h}_2 + \boldsymbol{h}_3 + \boldsymbol{h}_4 + \boldsymbol{h}_5, \tag{4-25}$$

and hence define the derivative of the error of the estimate of the linear velocity as follows:

$$\dot{\tilde{\boldsymbol{\nu}}}_1 = \left(-\mathbf{D}_1\tilde{\boldsymbol{\nu}}_1 + \mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1 + \frac{1}{m}\bar{\boldsymbol{f}}_{Aero} + \boldsymbol{h}_2 + \boldsymbol{h}_3 + \boldsymbol{h}_4 + \boldsymbol{h}_5\right). \tag{4-26}$$

Considering the positive definite Lyapunov function candidate:

$$V_1 = \int_0^{\boldsymbol{\eta}_{1e}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s}. \tag{4-27}$$

Taking the derivative of both sides of (4-27) along the solution of (4-24) results in the following derivative:

$$\dot{V}_1 = \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})(\boldsymbol{\alpha}_1 - \dot{\boldsymbol{\eta}}_{1d}) + \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\boldsymbol{\nu}_{1e} + \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\tilde{\boldsymbol{\nu}}_1. \tag{4-28}$$

To stabilise $\boldsymbol{\eta}_{1e}$ we will implement the one-step ahead backstepping method to design the virtual control signal $\boldsymbol{\alpha}_1$ as follows:

$$\boldsymbol{\alpha}_1 = -\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \dot{\boldsymbol{\eta}}_{1d}. \tag{4-29}$$

Defining the matrix $\mathbf{K}_1 \in \mathbb{R}^{3\times3}$, $\mathbf{K}_1 = diag(k_{1,1}, k_{1,2}, k_{1,3})$ where $k_{1,1}$, $k_{1,2}$ and $k_{1,3}$ are positive constants and $\sigma(\boldsymbol{s})$ and $\Delta_1(\boldsymbol{s})$ are defined as follows:

$$\sigma(s) = \frac{s}{\sqrt{1+\|s\|^2}}, \quad \Delta_1(\boldsymbol{s}) = 1 + \frac{1}{2}\|\boldsymbol{s}\|^2. \tag{4-30}$$

Substituting (4-29) into (4-28) yields:

$$\dot{V}_1 \leq -\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} + \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\boldsymbol{v}_{1e} + \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\tilde{\boldsymbol{v}}_1. \tag{4-31}$$

Substituting (4-29) into (4-24) yields:

$$\dot{\boldsymbol{\eta}}_{1e} = -\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} + \tilde{\boldsymbol{v}}_1 + \boldsymbol{v}_{1e}. \tag{4-32}$$

The derivative of (4-29) along the solutions of (4-24) and (4-26) yields:

$$\dot{\boldsymbol{\alpha}}_1 = \mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\dot{\hat{\boldsymbol{v}}}_1 - \mathbf{K}_1 \frac{\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\left(-\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} + \tilde{\boldsymbol{v}}_1 + \boldsymbol{v}_{1e}\right) + \ddot{\boldsymbol{\eta}}_{1d}. \tag{4-33}$$

To prepare for the next step, we calculate $\dot{\boldsymbol{v}}_{1e}$, by taking the derivative of the second equation of (4-23) along the solution of (4-24) and (4-33) as follows:

$$\dot{\boldsymbol{v}}_{1e} = \mathbf{G}_1\left(-\mathbf{D}_1\hat{\boldsymbol{v}}_1 - g\boldsymbol{e}_3 + \frac{f}{m}\mathbf{R}_1(\boldsymbol{q})\boldsymbol{e}_3 + \mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1 + \sum_{i=2}^{5}\boldsymbol{h}_i\right) + \boldsymbol{F}_1 + \boldsymbol{E}_2 - \ddot{\boldsymbol{\eta}}_{1d}, \tag{4-34}$$

Where the terms $\mathbf{G}_1, \boldsymbol{F}_1$ and $\boldsymbol{E}_2$ are defined as follows:

$$\mathbf{G}_1 = \left(\mathbf{I}_{3\times3} - \mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\right). \tag{4-35}$$

$$\boldsymbol{F}_1 = \left(\frac{\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\sigma'(\boldsymbol{\eta}_{1e})\left(-\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} + \boldsymbol{v}_{1e}\right), \quad \sigma'(\boldsymbol{\eta}_{1e}) = \frac{\partial\sigma(\boldsymbol{\eta}_{1e})}{\partial\boldsymbol{\eta}_{1e}}. \tag{4-36}$$

$$\boldsymbol{E}_2 = \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\tilde{\boldsymbol{v}}_1. \tag{4-37}$$

We need to consider the conditions for which $\mathbf{G}_1$ is invertible, clearly $\det(\mathbf{G}_1) \geq 1 - k_{1,1} - k_{1,2} - k_{1,3}$, It should be noted that every element in $\mathbf{G}_1$ is bounded between -1 and 1. This implies that the determinant of $\mathbf{G}_1$ is bounded between -2 and 4. Furthermore the constants $k_{1,1}$, $k_{1,2}$ and $k_{1,3}$ should be chosen such that:

$$1 - 0.28 * (k_{1,1} - k_{1,2} - k_{1,3}) > 0. \tag{4-38}$$

The above is based on the determinant equation for a 3x3 matrix. The matrix $\mathbf{G}_1$ obeys:

$$\|\mathbf{G}_1\| \leq 1 + \|\mathbf{K}_1\|. \tag{4-39}$$

The vector $\boldsymbol{F}_1$ is bounded by:

$$\|\boldsymbol{F}_1\| \leq \lambda_M(\mathbf{K}_1)(2+\varrho_1). \tag{4-40}$$

Where $\lambda_M(\mathbf{K}_1)$ denotes the maximum Eigen value of $\mathbf{K}_1$.

Step 2: We define the coordinate transformation representing the error between the current system angular position and the virtual control signal $\boldsymbol{\alpha}_q$ as:

$$\boldsymbol{q}_e = \boldsymbol{q} - \boldsymbol{\alpha}_q. \tag{4-41}$$

Where $\boldsymbol{q}_e$ is the tracking error of the attitude and $\boldsymbol{\alpha}_q = [\alpha_{q,1} \quad \alpha_{q,2} \quad \alpha_{q,3}]^{\mathrm{T}}$, the MRP virtual control signal for the attitude. Furthermore, substituting (4-41) into (2-90) yields:

$$\mathbf{R_1}(\boldsymbol{q}) = \mathbf{H}(\boldsymbol{q}_e, \boldsymbol{\alpha}_q) + \mathbf{R_1}(\boldsymbol{\alpha}_q), \tag{4-42}$$

where

$$\mathbf{H}(\boldsymbol{q}_e, \boldsymbol{\alpha}_q) = \mathbf{R}_2(\boldsymbol{q}_e + \boldsymbol{\alpha}_q) - \mathbf{R_1}(\boldsymbol{\alpha}_q), \tag{4-43}$$

$$\begin{aligned}\mathbf{H}(\boldsymbol{q}_e, \boldsymbol{\alpha}_q) = 8\frac{\boldsymbol{S}(\boldsymbol{q}_e)(\mathbf{S}(\boldsymbol{q}) - \mathbf{I}_{3\times3}) + \mathbf{S}(\boldsymbol{\alpha}_q)\boldsymbol{S}(\boldsymbol{q}_e)}{(1+\|\boldsymbol{q}\|^2)^2} + \mathbf{S}(\boldsymbol{\alpha}_q)(\mathbf{S}(\boldsymbol{\alpha}_q) - \mathbf{I}_{3\times3}) \\ * \left(\frac{8}{\left(1+\|\boldsymbol{q_e} + \boldsymbol{\alpha}_q\|^2\right)^2} - \frac{8}{\left(1+\|\boldsymbol{\alpha}_q\|^2\right)^2}\right) \\ + 4\mathbf{S}(\boldsymbol{\alpha}_q)\left(\frac{1}{1+\|\boldsymbol{q_e} + \boldsymbol{\alpha}_q\|^2} - \frac{1}{1+\|\boldsymbol{\alpha}_q\|^2}\right).\end{aligned} \tag{4-44}$$

It is difficult to design the virtual control signal $\boldsymbol{\alpha}_q$ to stabilise the error vector $\boldsymbol{\nu}_{1e}$ at the origin. To get around this and stabilise $\boldsymbol{\eta}_{1e}$ and $\boldsymbol{\nu}_{1e}$ we will design the virtual control signal $\boldsymbol{\alpha}_2$ and then calculate $\boldsymbol{\alpha}_q$ after. The Euler virtual control for the attitude:

$$\boldsymbol{\alpha}_2 = [\alpha_\phi \quad \alpha_\theta \quad \alpha_\psi]^{\mathrm{T}}, \tag{4-45}$$

where, α_ϕ denotes the virtual control for the pitch, α_θ the virtual control for the roll and α_ψ virtual control for the yaw angle:

$$\alpha_\psi = \psi_d. \tag{4-46}$$

We will now define each component of $\boldsymbol{\alpha}_q$ as follows:

$$\alpha_{q_1} = \frac{\sin\left(\frac{\alpha_\phi}{2}\right)\cos\left(\frac{\alpha_\theta}{2}\right)\cos\left(\frac{\alpha_\psi}{2}\right) - \cos\left(\frac{\alpha_\phi}{2}\right)\sin\left(\frac{\alpha_\theta}{2}\right)\sin\left(\frac{\alpha_\psi}{2}\right)}{1+\cos\left(\frac{\alpha_\phi}{2}\right)\cos\left(\frac{\alpha_\theta}{2}\right)\cos\left(\frac{\alpha_\psi}{2}\right) + \sin\left(\frac{\alpha_\phi}{2}\right)\sin\left(\frac{\alpha_\theta}{2}\right)\sin\left(\frac{\alpha_\psi}{2}\right)}, \tag{4-47}$$

$$\alpha_{q_2} = \frac{\cos\left(\frac{\alpha_\phi}{2}\right)\sin\left(\frac{\alpha_\theta}{2}\right)\cos\left(\frac{\alpha_\psi}{2}\right) + \sin\left(\frac{\alpha_\phi}{2}\right)\cos\left(\frac{\alpha_\theta}{2}\right)\sin\left(\frac{\alpha_\psi}{2}\right)}{1+\cos\left(\frac{\alpha_\phi}{2}\right)\cos\left(\frac{\alpha_\theta}{2}\right)\cos\left(\frac{\alpha_\psi}{2}\right) + \sin\left(\frac{\phi_d}{2}\right)\sin\left(\frac{\alpha_\theta}{2}\right)\sin\left(\frac{\alpha_\psi}{2}\right)}, \tag{4-48}$$

$$\alpha_{q_3} = \frac{\cos\left(\frac{\alpha_\phi}{2}\right)\cos\left(\frac{\alpha_\theta}{2}\right)\sin\left(\frac{\alpha_\psi}{2}\right) - \sin\left(\frac{\alpha_\phi}{2}\right)\sin\left(\frac{\alpha_\theta}{2}\right)\cos\left(\frac{\alpha_\psi}{2}\right)}{1+\cos\left(\frac{\alpha_\phi}{2}\right)\cos\left(\frac{\alpha_\theta}{2}\right)\cos\left(\frac{\alpha_\psi}{2}\right) + \sin\left(\frac{\alpha_\phi}{2}\right)\sin\left(\frac{\alpha_\theta}{2}\right)\sin\left(\frac{\alpha_\psi}{2}\right)}. \tag{4-49}$$

we can write:

$$\mathbf{R}_1(\boldsymbol{\alpha}_q) = \mathbf{R}_1(\boldsymbol{\alpha}_2). \tag{4-50}$$

To stabilise $\boldsymbol{\nu}_{1e}$ and design the linear velocity $\hat{\boldsymbol{\nu}}_1$ estimate we will consider the Lyapunov function candidate:

$$V_2 = \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} V_1 + \tfrac{1}{2}\left(4\gamma_1\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2} + \|\widetilde{\boldsymbol{\eta}}_1\|^2 + \|\widetilde{\boldsymbol{\nu}}_1\|^2\right) - 2\gamma_1. \tag{4-51}$$

To stabilise the error vector $\boldsymbol{\nu}_{1e}$ at the origin we will design the virtual control $\boldsymbol{\alpha}_2$. Only components α_ϕ and α_θ are considered as α_ψ has already been defined in (4-46). To design the control f and the virtual control $\boldsymbol{\alpha}_2$ we calculate the derivative of along the solution of(4-31), (4-32) and the derivative of (4-21) gives:

$$\begin{aligned}
\dot{V}_2 \leq & -\gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \gamma_1 \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\boldsymbol{\nu}_{1e} \\
& + \frac{2\gamma_1 \boldsymbol{\nu}_{1e}^{\mathrm{T}}}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}} \Bigg(\mathbf{G}_1 \Bigg(-g\boldsymbol{e}_3 + \mathbf{K}_{02}\widetilde{\boldsymbol{\eta}}_1 + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \sigma(\boldsymbol{\eta}_{1e}) + \frac{f}{m}\mathbf{R}_1(\boldsymbol{\alpha}_q)\boldsymbol{e}_3 \\
& + \boldsymbol{h}_3 + \boldsymbol{h}_4 + \boldsymbol{h}_5\Bigg) + \boldsymbol{F}_1 + \boldsymbol{E}_2 - \ddot{\boldsymbol{\eta}}_{1d} + \mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^T}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\mathbf{D}_1\hat{\boldsymbol{\nu}}_1 - \mathbf{D}_1\boldsymbol{\nu}_{1e} - \mathbf{D}_1\boldsymbol{\alpha}_1 \\
& + \mathbf{G}_1\frac{f}{m}\mathrm{H}(\boldsymbol{q}_e, \boldsymbol{\alpha}_q)\boldsymbol{e}_3\Bigg) - \widetilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\mathbf{K}_{01}\widetilde{\boldsymbol{\eta}}_1 + \widetilde{\boldsymbol{\eta}}_1^{\mathrm{T}}(-\boldsymbol{h}_1) - \widetilde{\boldsymbol{\nu}}_1^{\mathrm{T}}\mathbf{D}_1\widetilde{\boldsymbol{\nu}}_1 \\
& + \widetilde{\boldsymbol{\nu}}_1^{\mathrm{T}}(-\boldsymbol{h}_2 - \boldsymbol{h}_3 - \boldsymbol{h}_4 - \boldsymbol{h}_5) + \widetilde{\boldsymbol{\nu}}_1^{\mathrm{T}}(\mathbf{I}_{3\times3} - \mathbf{K}_{02})\widetilde{\boldsymbol{\eta}}_1 + \tfrac{1}{m}\widetilde{\boldsymbol{\nu}}_1^{\mathrm{T}}\bar{\boldsymbol{f}}_{Aero}.
\end{aligned} \tag{4-52}$$

We will choose the interlace term $\boldsymbol{h}_2$ and $\boldsymbol{h}_3$ as:

$$\boldsymbol{h}_2 = \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \sigma(\boldsymbol{\eta}_{1e}), \tag{4-53}$$

$$\boldsymbol{h}_3 = \frac{2\gamma_1}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}} \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \boldsymbol{\nu}_{1e}. \tag{4-54}$$

We now use Young's inequality:

$$\tfrac{1}{m}\widetilde{\boldsymbol{\nu}}_1^{\mathrm{T}}\bar{\boldsymbol{f}}_{Aero} \leq \lambda_m(\mathbf{D}_1)\frac{\varepsilon_1}{2}\|\widetilde{\boldsymbol{\nu}}_1\|^2 + \frac{\left\|\bar{\boldsymbol{f}}_{Aero}\right\|^2}{\varepsilon_1 2m^2}, \tag{4-55}$$

$$\gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \boldsymbol{\nu}_{1e}^{\mathrm{T}}\sigma(\boldsymbol{\eta}_{1e}) \leq \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{4}\left(\varepsilon_2 \frac{\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}} + \frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\varepsilon_2} \frac{1 + \|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}\right). \tag{4-56}$$

Moreover,

$$-1 \leq \frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}}\sigma(\boldsymbol{\eta}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \leq 1, \tag{4-57}$$

$$\frac{2\gamma_1^2\boldsymbol{\nu}_{1e}^{\mathrm{T}}\sigma(\boldsymbol{\eta}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\frac{\lambda_m(\mathbf{D}_1)}{2} \leq 2\gamma_1^2\frac{\lambda_m(\mathbf{D}_1)}{2}\left(\frac{\|\boldsymbol{\nu}_{1e}\|^2+2\boldsymbol{\nu}_{1e}^{\mathrm{T}}\boldsymbol{\alpha}_1}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}+\frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}}\sigma(\boldsymbol{\eta}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\frac{2+\|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right), \tag{4-58}$$

this is true because:

$$\left\|\frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}}\sigma(\boldsymbol{\eta}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right\| \leq 1, \tag{4-59}$$

$$\frac{\Delta_1(\hat{\boldsymbol{\nu}}_1)}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} = \frac{2+\|\boldsymbol{\alpha}_1\|^2+\|\boldsymbol{\nu}_{1e}\|^2+2\boldsymbol{\nu}_{1e}^{\mathrm{T}}\boldsymbol{\alpha}_1}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} = 1. \tag{4-60}$$

By substituting (4-53) to (4-56) and (4-58) back into (4-52) gives:

$$\begin{aligned}
\mathcal{L}V_2 \leq{}& -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}+\frac{2\gamma_1\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \\
&* \Bigg(\mathbf{G}_1\left(-g\boldsymbol{e}_3+\mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1+\frac{f}{m}\mathbf{R}_1(\boldsymbol{\alpha}_q)\boldsymbol{e}_3+\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\frac{2\gamma_1\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) \\
&-\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}-\mathbf{D}_1\boldsymbol{\alpha}_1-\mathbf{D}_1\boldsymbol{\nu}_{1e}-\ddot{\boldsymbol{\eta}}_{1d}+\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\mathbf{D}_1\hat{\boldsymbol{\nu}}_1 \\
&-\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\sigma(\boldsymbol{\eta}_{1e})+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{\boldsymbol{\nu}_{1e}+2\boldsymbol{\alpha}_1}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2} \\
&+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma(\boldsymbol{\eta}_{1e})\frac{2+\|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\Bigg) \\
&+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{4}\left(\varepsilon_2\frac{\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}+\frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\varepsilon_2}\frac{1+\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) \\
&+\frac{2\gamma_1\|\boldsymbol{\nu}_{1e}\|^2}{(1+\|\boldsymbol{\nu}_{1e}\|^2)}\left(\frac{\|\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\|}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)+\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{G}_1(\boldsymbol{h}_4+\boldsymbol{h}_5) \\
&+\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{G}_1\frac{f}{m}\mathbf{H}(\boldsymbol{q}_e,\boldsymbol{\alpha}_q)\boldsymbol{e}_3-\tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1+\tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\left(-\boldsymbol{h}_1+\mathbf{K}_{02}^{\mathrm{T}}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}\right)-\tilde{\boldsymbol{\nu}}_1^{\mathrm{T}}\mathbf{D}_1\tilde{\boldsymbol{\nu}}_1 \\
&+\lambda_m(\mathbf{D}_1)\frac{\varepsilon_1}{2}\|\tilde{\boldsymbol{\nu}}_1\|^2+\tilde{\boldsymbol{\nu}}_1^{\mathrm{T}}(\mathbf{I}_{3\times3}-\mathbf{K}_{02})\tilde{\boldsymbol{\eta}}_1+\tilde{\boldsymbol{\nu}}_1^{\mathrm{T}}(-\boldsymbol{h}_4-\boldsymbol{h}_5) \\
&+\frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{\varepsilon_1 2m^2\lambda_m(\mathbf{D}_1)}.
\end{aligned} \tag{4-61}$$

This suggests the choosing of:

$$\begin{aligned}
&\mathbf{G}_1\left(-g\boldsymbol{e}_3+\frac{f}{m}\mathbf{R}_1(\boldsymbol{\alpha}_q)\boldsymbol{e}_3+\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\frac{2\gamma_1\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)-\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\sigma(\boldsymbol{\eta}_{1e}) \\
&-\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}-\mathbf{D}_1\boldsymbol{\alpha}_1+\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\mathbf{D}_1\hat{\boldsymbol{\nu}}_1+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{\boldsymbol{\nu}_{1e}+2\boldsymbol{\alpha}_1}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2} \\
&+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma(\boldsymbol{\eta}_{1e})\frac{2+\|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}-\ddot{\boldsymbol{\eta}}_{1d} = -\mathbf{K}_2\sigma(\boldsymbol{\nu}_{1e})-c\frac{2\boldsymbol{\alpha}_1+\boldsymbol{\nu}_{1e}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2},
\end{aligned} \tag{4-62}$$

where the matrix $\mathbf{K}_2 \in \mathbb{R}^{3\times 3}$, $\mathbf{K}_2 = diag(k_{2,1}, k_{2,2}, k_{2,3})$ where $k_{2,1}$, $k_{2,2}$ and $k_{2,3}$ are positive constants to be defined later.

Substituting(4-32)back into (4-61) gives:

$$\begin{aligned}\dot{V}_2 \leq & -\gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e}) \mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} - \frac{2\gamma_1 \boldsymbol{\nu}_{1e}^{\mathrm{T}} \mathbf{K}_2 \sigma(\boldsymbol{\nu}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} - \frac{2\gamma_1 \boldsymbol{\nu}_{1e}^{\mathrm{T}} \mathbf{D}_1 \boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \\ & -2\gamma_1 c \boldsymbol{\nu}_{1e}^{\mathrm{T}} \left(\frac{2\boldsymbol{\alpha}_1 + \boldsymbol{\nu}_{1e}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right) + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{4} \left(\varepsilon_2 \frac{\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} + \frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\varepsilon_2} \frac{1+\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) \\ & + \frac{2\gamma_1 \|\boldsymbol{\nu}_{1e}\|^2}{(1+\|\boldsymbol{\nu}_{1e}\|^2)} \left(\frac{\|\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})\|}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}\right) + \frac{2\gamma_1 \boldsymbol{\nu}_{1e}^{\mathrm{T}}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \mathbf{G}_1 (\boldsymbol{h}_4 + \boldsymbol{h}_5) \\ & -\tilde{\boldsymbol{\eta}}_1^{\mathrm{T}} \mathbf{K}_{01} \tilde{\boldsymbol{\eta}}_1 + \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}} \left(-\boldsymbol{h}_1 + \frac{2\gamma_1 \mathbf{K}_{02}^{\mathrm{T}} \mathbf{G}_1^{\mathrm{T}} \boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) - \lambda_m(\mathbf{D}_1)\left(1 - \frac{\varepsilon_1}{2}\right) \|\tilde{\boldsymbol{\nu}}_1\|^2 \\ & + \tilde{\boldsymbol{\nu}}_1^{\mathrm{T}} (\mathbf{I}_{3\times 3} - \mathbf{K}_{02}) \tilde{\boldsymbol{\eta}}_1 - \tilde{\boldsymbol{\nu}}_1^{\mathrm{T}} (\boldsymbol{h}_4 + \boldsymbol{h}_5) + \frac{\varepsilon_2 \lambda_m(\mathbf{D}_1)}{2} \|\tilde{\boldsymbol{\nu}}_1\|^2 + \frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{2\varepsilon_1 \lambda_m(\mathbf{D}_1) m^2} \\ & + \frac{2\gamma_1 \boldsymbol{\nu}_{1e}^{\mathrm{T}}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \mathbf{G}_1 \frac{f}{m} \mathbf{H}(\boldsymbol{q}_e, \boldsymbol{\alpha}_q) \boldsymbol{e}_3. \end{aligned} \tag{4-63}$$

The last stage of (4-63) will be considered in the second stage of the control design for the time being we do not consider it for the design of $f\mathbf{R}(\boldsymbol{\alpha}_q)\boldsymbol{e}_3$. To achieve stability, we choose:

$$\begin{aligned} f\mathbf{R}_1(\boldsymbol{\alpha}_q)\boldsymbol{e}_3 = & m\mathbf{G}_1^{-1} \Bigg(\mathbf{D}_1 \boldsymbol{\alpha}_1 - \mathbf{K}_2 \sigma(\boldsymbol{\nu}_{1e}) - \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \sigma(\boldsymbol{\eta}_{1e}) \frac{2+\|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} \\ & + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e}) \hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2} \sigma(\boldsymbol{\eta}_{1e}) - \frac{\mathbf{K}_1 \sigma(\boldsymbol{\eta}_{1e}) \hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2} \mathbf{D}_1 \hat{\boldsymbol{\nu}}_1 \\ & + \frac{\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \frac{\mathbf{K}_1 \sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} - \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{2\boldsymbol{\alpha}_1 + \boldsymbol{\nu}_{1e}}{8\Delta_1(\hat{\boldsymbol{\nu}}_1)} \sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2} \\ & + \ddot{\boldsymbol{\eta}}_{1d}\Bigg) + m\left(g\boldsymbol{e}_3 - \frac{\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \frac{2\gamma_1 \boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) \quad \stackrel{\text{def}}{=} \boldsymbol{\Omega}. \end{aligned} \tag{4-64}$$

Because:

$$\mathbf{R}(\boldsymbol{\alpha}_2) = \mathbf{R}(\boldsymbol{\alpha}_q), \tag{4-65}$$

equation (4-64) can now be written as:

$$f = \mathbf{R}_1^{-1}(\boldsymbol{\alpha}_2)\boldsymbol{\Omega}. \tag{4-66}$$

Let the components of $\boldsymbol{\Omega}$ be Ω_1, Ω_2 and Ω_3 i.e. $\boldsymbol{\Omega} = [\Omega_1 \quad \Omega_2 \quad \Omega_3]^{\mathrm{T}}$. Taking the norms of both sides of (4-66) gives:

$$\sqrt{ff\boldsymbol{e}_3^{\mathrm{T}}\boldsymbol{e}_3} = \sqrt{\mathbf{R}_1^{-\mathrm{T}}(\boldsymbol{\alpha}_2)\mathbf{R}_1^{-1}(\boldsymbol{\alpha}_2)\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}, \tag{4-67}$$

$$\sqrt{ff} = \sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}, \tag{4-68}$$

$$f = \sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}. \tag{4-69}$$

Where $\mathbf{R}_1^{-\mathrm{T}}(\boldsymbol{\alpha}_2)\mathbf{R}_1^{-1}(\boldsymbol{\alpha}_2) = \mathbf{I}_{3\times3}$. By expanding out (4-66) and using the fact that $\mathbf{R}_1^{-1}(\boldsymbol{\alpha}_2) = \mathbf{R}_1^{\mathrm{T}}(\boldsymbol{\alpha}_2)$ we obtain:

$$\begin{aligned}\left(\mathrm{c}_{\alpha_\psi}\mathrm{c}_{\alpha_\theta}\right)\Omega_1 + \left(\mathrm{s}_{\alpha_\psi}\mathrm{c}_{\alpha_\theta}\right)\Omega_2 - (\mathrm{s}_{\alpha_\theta})\Omega_3 = 0,\\ \left(-\mathrm{s}_{\alpha_\psi}\mathrm{c}_{\alpha_\phi} + \mathrm{c}_{\alpha_\psi}\mathrm{s}_{\alpha_\theta}\mathrm{s}_{\alpha_\phi}\right)\Omega_1 + \left(\mathrm{c}_{\alpha_\psi}\mathrm{c}_{\alpha_\phi} + \mathrm{s}_{\alpha_\psi}\mathrm{s}_{\alpha_\theta}\mathrm{s}_{\alpha_\phi}\right)\Omega_2 + \left(\mathrm{c}_{\alpha_\theta}\mathrm{s}_{\alpha_\phi}\right)\Omega_3 = 0,\\ \left(\mathrm{s}_{\alpha_\psi}\mathrm{s}_{\alpha_\phi} + \mathrm{c}_{\alpha_\psi}\mathrm{s}_{\alpha_\theta}\mathrm{c}_{\alpha_\phi}\right)\Omega_1 + \left(-\mathrm{c}_{\alpha_\psi}\mathrm{s}_{\alpha_\phi} + \mathrm{s}_{\alpha_\psi}\mathrm{s}_{\alpha_\theta}\mathrm{c}_{\alpha_\phi}\right)\Omega_2 + \left(\mathrm{c}_{\alpha_\theta}\mathrm{c}_{\alpha_\phi}\right)\Omega_3 = f.\end{aligned} \tag{4-70}$$

Multiplying the second equation of (4-70) through by $-\mathrm{c}_{\alpha_\phi}$ and multiplying the third equation of (4-70) by s_{α_ϕ} and adding the resultant together yields:

$$\alpha_\phi = \sin^{-1}\left(\frac{\mathrm{s}_{\alpha_\psi}\Omega_1 - \mathrm{c}_{\alpha_\psi}\Omega_2}{\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}}\right). \tag{4-71}$$

Which is well defined since $\left|\mathrm{s}_{\alpha_\psi}\Omega_1 - \mathrm{c}_{\alpha_\psi}\Omega_2\right| \le \sqrt{\Omega_1^2 + \Omega_2^2} \le \sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}} > 0$. Furthermore, from the first equation of (4-70) we can obtain:

$$\alpha_\theta = \tan^{-1}\left(\frac{\mathrm{c}_{\alpha_\psi}\Omega_1 + \mathrm{s}_{\alpha_\psi}\Omega_2}{\Omega_3}\right). \tag{4-72}$$

With:

$$\begin{aligned}\boldsymbol{\Omega} = m\mathbf{G}_1^{-1}\Bigg(&\mathbf{D}_1\boldsymbol{\alpha}_1 - \mathbf{K}_2\sigma(\boldsymbol{\nu}_{1e}) - \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma(\boldsymbol{\eta}_{1e})\frac{2 + \|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\\ &+ \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\sigma(\boldsymbol{\eta}_{1e}) - \frac{\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}\mathbf{D}_1\hat{\boldsymbol{\nu}}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})\\ &+ \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e}) - \left(c + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{2\boldsymbol{\alpha}_1 + \boldsymbol{\nu}_{1e}}{8\Delta_1(\hat{\boldsymbol{\nu}}_1)}\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}\\ &+ \ddot{\boldsymbol{\eta}}_{1d}\Bigg) + m\left(g\boldsymbol{e}_3 - \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\frac{2\gamma_1\boldsymbol{\nu}_{1e}}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}\right).\end{aligned} \tag{4-73}$$

To ensure that (4-73) is well defined, we will show that $\boldsymbol{\Omega}$ is bounded and there are initial values of the states and control gains such that $\Omega_3(t) > 0$. We state:

$$\begin{aligned}\|\boldsymbol{\Omega}\| \le m\|\varrho_0\|\Bigg(&\|\mathbf{D}_1\|(\|\mathbf{K}_1\| + \varrho_1) + \|\mathbf{K}_2\| + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2 + (\|\mathbf{K}_1\| + \varrho_1)}{2}\\ &+ 0.8\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\|\mathbf{K}_1\| + 0.5\|\mathbf{K}_1\|\|\mathbf{D}_1\| + \|\mathbf{K}_1\|^2\\ &+ \left(\frac{c}{8} + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{16}\right)\Big(\sqrt{0.5 + (0.46\|\mathbf{K}_1\| + 0.7\varrho_1)^2 + 1.3\|\mathbf{K}_1\| + 1.6\varrho_1 + 4}\\ &+ (\|\mathbf{K}_1\| + \varrho_1)\sqrt{1 + (\|\mathbf{K}_1\| + \varrho_1)^2 + 1.3\|\mathbf{K}_1\| + 1.6\varrho_1 + 0.5}\Big) + \varrho_2\Bigg)\\ &+ m(g + 2\gamma_1\|\mathbf{K}_1\|).\end{aligned} \tag{4-74}$$

In addition, ϱ_0 is defined as shown below:

$$\varrho_0 = 1 - 0.27\big(\mathbf{K}_{1,1} + \mathbf{K}_{1,2} + \mathbf{K}_{1,3}\big) > 0. \tag{4-75}$$

We can list the upper bounds for Ω_1, Ω_2 and Ω_3 as follows:

$$\begin{aligned}
\lfloor \Omega_1 \rfloor \le \Omega_{1\mathrm{M}} = \frac{m}{\varrho_0}\Bigg(&\big(1 - k_{1,2} - k_{1,3}\big)\Bigg(\mathbf{D}_{1,1}\big(k_{1,1} + \varrho_{1,1}\big) + k_{2,1} \\
&+ \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2 + \|(\|\mathbf{K}_1\| + \varrho_1)\|^2}{2} + 0.46\gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} k_{1,1} + 0.5\lambda_M(\mathbf{D}_1)k_{1,1} \\
&+ k_{1,1}^2 + \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \\
&* \left(\big(k_{1,1} + \varrho_{1,1}\big)\frac{1 + k_{1,1} + \varrho_{1,1}}{2} + \sqrt{(1 + \lambda_M(\mathbf{K}_1) + \varrho_1)/2}\right) + +\varrho_{2,1}\Bigg) \\
&+ \Big(k_{1,1}\big(2k_{1,3} + 1\big)\Big)\Bigg(\mathbf{D}_{2,2}\big(k_{1,2} + \varrho_{1,2}\big) + k_{2,2} \\
&+ \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2 + \|(\|\mathbf{K}_1\| + \varrho_1)\|^2}{2} + +0.46\gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} k_{1,2} \\
&+ 0.5\lambda_M(\mathbf{D}_1)k_{1,2} + k_{1,2}^2 + \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \\
&* \left(\big(k_{1,2} + \varrho_{1,2}\big)\frac{1 + k_{1,2} + \varrho_{1,2}}{2} + \sqrt{(1 + \lambda_M(\mathbf{K}_1) + \varrho_1)/2}\right) + \varrho_{2,2}\Bigg) \\
&+ \Big(k_{1,1}\big(2k_{1,2} + 1\big)\Big)\Bigg(\mathbf{D}_{2,2}\big(k_{1,3} + \varrho_{1,3}\big) + k_{2,3} \\
&+ \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2 + \|(\|\mathbf{K}_1\| + \varrho_1)\|^2}{2} + +0.46\gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} k_{1,3} \\
&+ 0.5\lambda_M(\mathbf{D}_1)k_{1,3} + k_{1,3}^2 + \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \\
&* \left(\big(k_{1,3} + \varrho_{1,3}\big)\frac{1 + k_{1,3} + \varrho_{1,3}}{2} + \sqrt{(1 + \lambda_M(\mathbf{K}_1) + \varrho_1)/2}\right) + \varrho_{2,3}\Bigg)\Bigg) \\
&+ m\big(-k_{1,1} + 2\gamma_1\big),
\end{aligned} \tag{4-76}$$

$$\lfloor \Omega_2 \rfloor \leq \Omega_{2M} = \frac{m}{\varrho_0}\Bigg(\Big(k_{1,2}(2k_{1,3}+1)\Big)\Bigg(\mathbf{D}_{1,1}(k_{1,1}+\varrho_{1,1})+k_{2,1}$$
$$+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2}+0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,1}+0.5\lambda_M(\mathbf{D}_1)k_{1,1}$$
$$+k_{1,1}^2+\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)$$
$$*\left(\left(k_{1,1}+\varrho_{1,1}\right)\frac{1+k_{1,1}+\varrho_{1,1}}{2}+\sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right)++\varrho_{2,1}\Bigg)$$
$$+\left(1-k_{1,1}-k_{1,3}\right)\Bigg(\mathbf{D}_{2,2}(k_{1,2}+\varrho_{1,2})+k_{2,2}$$
$$+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2}++0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,2}$$
$$+0.5\lambda_M(\mathbf{D}_1)k_{1,2}+k_{1,2}^2+\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)$$
$$*\left(\left(k_{1,2}+\varrho_{1,2}\right)\frac{1+k_{1,2}+\varrho_{1,2}}{2}+\sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right)+\varrho_{2,2}\Bigg)$$
$$+\Big(k_{1,2}(2kk_{1,1}+1)\Big)\Bigg(\mathbf{D}_{2,2}(k_{1,3}+\varrho_{1,3})+k_{2,3}$$
$$+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2}++0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,3}$$
$$+0.5\lambda_M(\mathbf{D}_1)k_{1,3}+k_{1,3}^2+\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)$$
$$*\left(\left(k_{1,3}+\varrho_{1,3}\right)\frac{1+k_{1,3}+\varrho_{1,3}}{2}+\sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right)+\varrho_{2,3}\Bigg)\Bigg)$$
$$+mk_{1,2}(k_{1,2}+\varrho_{1,2})+2\varepsilon_2\gamma_1^2\frac{1+k_{1,1}+k_{1,2}+k_{1,3}}{\lambda_m(\mathbf{D}_1)}, \tag{4-77}$$

$$\lfloor \Omega_3 \rfloor \le \Omega_{3\mathrm{M}} = \frac{m}{\varrho_0}\Bigg(\Big(k_{1,3}\big(2k_{1,2}+1\big)\Big)\Bigg(\mathbf{D}_{1,1}\big(k_{1,1}+\varrho_{1,1}\big)+k_{2,1}$$
$$+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2}+0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,1}+0.5\lambda_M(\mathbf{D}_1)k_{1,1}$$
$$+k_{1,1}^2+\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)$$
$$*\left(\big(k_{1,1}+\varrho_{1,1}\big)\frac{1+k_{1,1}+\varrho_{1,1}}{2}+\sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right)++\varrho_{2,1}\Bigg)$$
$$+\Big(k_{1,3}\big(2\mathrm{i}k_{1,1}+1\big)\Big)\Bigg(\mathbf{D}_{2,2}\big(k_{1,2}+\varrho_{1,2}\big)+k_{2,2}$$
$$+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2}++0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,2}$$
$$+0.5\lambda_M(\mathbf{D}_1)k_{1,2}+k_{1,2}^2+\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)$$
$$*\left(\big(k_{1,2}+\varrho_{1,2}\big)\frac{1+k_{1,2}+\varrho_{1,2}}{2}+\sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right)+\varrho_{2,2}\Bigg)$$
$$+\big(1-k_{1,1}-k_{1,2}\big)\Bigg(\mathbf{D}_{2,2}\big(k_{1,3}+\varrho_{1,3}\big)+k_{2,3}$$
$$+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2}++0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,3}$$
$$+0.5\lambda_M(\mathbf{D}_1)k_{1,3}+k_{1,3}^2+\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)$$
$$*\left(\big(k_{1,3}+\varrho_{1,3}\big)\frac{1+k_{1,3}+\varrho_{1,3}}{2}+\sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right)+\varrho_{2,3}\Bigg)\Bigg)+mg$$
$$+mk_{1,3}\big(k_{1,3}+\varrho_{1,3}\big)+2\varepsilon_2\gamma_1^2\frac{1+k_{1,1}+k_{1,2}+k_{1,3}}{\lambda_m(\mathbf{D}_1)}. \tag{4-78}$$

where $\Omega_{1\mathrm{M}}$, $\Omega_{2\mathrm{M}}$ and $\Omega_{3\mathrm{M}}$ are the upper bounds of the control signals Ω_1, Ω_2 and Ω_3 of Referring back to (4-8) where we specified that the aircraft cannot descend faster than it freely falls due to ϱ_5 which is a strictly positive constant. The control gains γ_1, $\mathbf{K}_1$, $\mathbf{K}_2$ and $\mathbf{K}_{1\mathrm{d},3}$ need to be selected so that the following holds true:

$$\begin{aligned}
\Omega_3 \geq -\frac{m}{\varrho_0}\Bigg(& k_{1,3}(2k_{1,2}+1)\Bigg(\mathbf{D}_{1,1}(k_{1,1}+\varrho_{1,1}) + k_{2,1} + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2} \\
& + 0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,1} + 0.5\lambda_M(\mathbf{D}_1)k_{1,1} + k_{1,1}^2 + \left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right) \\
& * \left((k_{1,1}+\varrho_{1,1})\frac{1+k_{1,1}+\varrho_{1,1}}{2} + \sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right) + +\varrho_{2,1}\Bigg) \\
& + k_{1,3}(2k_{1,1}+1) \\
& * \Bigg(\mathbf{D}_{2,2}(k_{1,2}+\varrho_{1,2}) + k_{2,2} + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2} \\
& + +0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,2} + 0.5\lambda_M(\mathbf{D}_1)k_{1,2} + k_{1,2}^2 + \left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right) \\
& * \left((k_{1,2}+\varrho_{1,2})\frac{1+k_{1,2}+\varrho_{1,2}}{2} + \sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right) + \varrho_{2,2}\Bigg) \\
& + (1-k_{1,1}-k_{1,2}) \\
& * \Bigg(\mathbf{D}_{2,2}(k_{1,3}+\varrho_{1,3}) + k_{2,3} + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2} \\
& + +0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,3} + 0.5\lambda_M(\mathbf{D}_1)k_{1,3} + k_{1,3}^2 + \left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right) \\
& * \left((k_{1,3}+\varrho_{1,3})\frac{1+k_{1,3}+\varrho_{1,3}}{2} + \sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right) + \varrho_{2,3}\Bigg)\Bigg) + mg \\
& - mk_{1,3}(k_{1,3}+\varrho_{1,3}) - 2\varepsilon_2\gamma_1^2\frac{1+k_{1,1}+k_{1,2}+k_{1,3}}{\lambda_m(\mathbf{D}_1)} \geq \Omega_3^* > 0.
\end{aligned} \tag{4-79}$$

Where Ω_3^* is the lower bound of Ω_3 which strictly greater than zero and is a product of the effects ϱ_5 and ϱ_{f3} have on(4-10), this implies that Ω_3 has an upper and lower bound. It can now be said that because $\Omega_3 > 0$, α_θ defined in (4-71) is well defined. Once the virtual control $\boldsymbol{\alpha}_2 = col(\alpha_\phi, \alpha_\theta, \alpha_\psi)$ is available from(4-71), (4-72) and (4-46), the virtual control $\boldsymbol{\alpha}_q$ is determined from (4-48) to (4-50).

4.1.3 Stabilization of Attitude System

In this stage, we consider the two equations of (4-14) to be able to use the backstepping method for this stage we define the coordinate transformation:

$$\boldsymbol{\omega}_e = \boldsymbol{\omega} - \boldsymbol{\alpha}_\omega. \tag{4-80}$$

Where $\boldsymbol{\omega}_e$ is the tracking error of the angular velocity and $\boldsymbol{\alpha}_\omega$ the virtual control for the angular velocity. In the first step, we design $\boldsymbol{\alpha}_\omega$ to stabilize in probability the tracking error $\boldsymbol{q}_e$ at the origin. In the second step, we design the torque vector $\boldsymbol{\tau}$ to stabilise the tracking error $\boldsymbol{\omega}_e$ at the origin.

Step 1: Before calculating $\dot{\boldsymbol{q}}_e$ let us first calculate $\dot{\boldsymbol{\alpha}}_2$ as:

$$\dot{\boldsymbol{\alpha}}_2 = [\dot{\alpha}_\phi \quad \dot{\alpha}_\theta \quad \dot{\alpha}_\psi]^{\mathrm{T}}. \tag{4-81}$$

Furthermore, from (4-71) and (4-72) we obtain:

$$\alpha_\phi = \sin^{-1}\left(\frac{s_{\alpha_\psi}\Omega_1 - c_{\alpha_\psi}\Omega_2}{\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}}\right), \qquad \alpha_\theta = \tan^{-1}\left(\frac{c_{\alpha_\psi}\Omega_1 + s_{\alpha_\psi}\Omega_2}{\Omega_3}\right). \tag{4-82}$$

We can state that:

$$\begin{aligned}\dot{\boldsymbol{\Omega}} = \frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\Bigg(&-\mathbf{D}_1\hat{\boldsymbol{v}}_1 - g\boldsymbol{e}_3 + \frac{\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}}{m}\mathbf{R}_1(\boldsymbol{q})\boldsymbol{e}_3 + \mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1 + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma(\boldsymbol{\eta}_{1e}) + \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\boldsymbol{v}_{1e} \\ &+ \boldsymbol{h}_4 + \boldsymbol{h}_5\Bigg) + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}(\tilde{\boldsymbol{v}}_1 + \hat{\boldsymbol{v}}_1 - \dot{\boldsymbol{\eta}}_{1d}) + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1d}}\dot{\boldsymbol{\eta}}_{1d} + \frac{\partial\boldsymbol{\Omega}}{\partial\dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial\boldsymbol{\Omega}}{\partial\ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d},\end{aligned} \tag{4-83}$$

$$\begin{aligned}\boldsymbol{\Omega}_d = \frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\Bigg(&-\mathbf{D}_1\hat{\boldsymbol{v}}_1 - g\boldsymbol{e}_3 + \frac{\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}}{m}\mathbf{R}_1(\boldsymbol{q})\boldsymbol{e}_3 + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma(\boldsymbol{\eta}_{1e}) + \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\boldsymbol{v}_{1e} + \boldsymbol{h}_4\Bigg) \\ &+ \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\left(-\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} + \boldsymbol{v}_{1e}\right) + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1d}}\dot{\boldsymbol{\eta}}_{1d} + \frac{\partial\boldsymbol{\Omega}}{\partial\dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial\boldsymbol{\Omega}}{\partial\ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d},\end{aligned} \tag{4-84}$$

where the partial derivatives of $\boldsymbol{\Omega}$ are defined in Appendix D. We will state:

$$\dot{\boldsymbol{\Omega}} = \boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\tilde{\boldsymbol{v}}_1 + \frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\boldsymbol{h}_5 + \frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1. \tag{4-85}$$

Therefore, we state:

$$\dot{\boldsymbol{\alpha}}_2 = \mathbf{A}_{2,d}\boldsymbol{\Omega}_d + \mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\tilde{\boldsymbol{v}}_1 + \mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\boldsymbol{h}_5 + \mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1 + \begin{bmatrix}0\\0\\1\end{bmatrix}\dot{\alpha}_\psi + \mathbf{A}_{1,d}\boldsymbol{\Omega}\dot{\alpha}_\psi, \tag{4-86}$$

$$\mathbf{A}_{1,d} = \begin{bmatrix} c_{\alpha_\psi}\dfrac{1}{\cos(\alpha_\phi)\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}} & s_{\alpha_\psi}\dfrac{1}{\cos(\alpha_\phi)\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}} & 0 \\ -s_{\alpha_\psi}\dfrac{\cos^2(\alpha_\theta)}{\Omega_3} & c_{\alpha_\psi}\dfrac{\cos^2(\alpha_\theta)}{\Omega_3} & 0 \\ 0 & 0 & 0 \end{bmatrix}, \tag{4-87}$$

$$\mathbf{A}_{2,d} = \begin{bmatrix} \frac{s_{\alpha_\psi}}{\cos(\alpha_\phi)\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}} & \frac{-c_{\alpha_\psi}}{\cos(\alpha_\phi)\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}} & 0 \\ \cos^2(\alpha_\theta)\frac{c_{\alpha_\psi}}{\Omega_3} & \cos^2(\alpha_\theta)\frac{s_{\alpha_\psi}}{\Omega_3} & -\cos^2(\alpha_\theta)\frac{\tan(\alpha_\theta)}{\Omega_3} \\ 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} \frac{\tan(\alpha_\phi)}{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}} \\ 0 \\ 0 \end{bmatrix} \boldsymbol{\Omega}^{\mathrm{T}}. \tag{4-88}$$

As mentioned earlier we will use the modified Rodrigues parameter as such we will state:

$$\dot{\boldsymbol{\alpha}}_q = \mathbf{R}_2(\boldsymbol{\alpha}_2)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\dot{\boldsymbol{\alpha}}_2. \tag{4-89}$$

Differentiating both sides of (4-41) along the solutions of (4-89) and (4-80) yields:

$$\dot{\boldsymbol{q}}_e = \mathbf{R}_2(\boldsymbol{q})(\boldsymbol{\alpha}_{\boldsymbol{\omega}} + \boldsymbol{\omega}_e) - \mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\dot{\boldsymbol{\alpha}}_2. \tag{4-90}$$

To analyse the stability of the origin of $\boldsymbol{q}_e$ we consider the Lyapunov function candidate as follows:

$$V_3 = V_2 + \frac{\gamma_2}{2}\|\boldsymbol{q}_e\|^2. \tag{4-91}$$

Taking the derivative of both sides of (4-91) along the solution of (4-63)and (4-90) gives:

$$\begin{aligned} \dot{V}_3 \leq & -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} - \frac{2\gamma_1\boldsymbol{v}_{1e}^{\mathrm{T}}\mathbf{K}_2\sigma(\boldsymbol{v}_{1e})}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}} - 2\gamma_1 c\boldsymbol{v}_{1e}^{\mathrm{T}}\left(\frac{2\boldsymbol{\alpha}_1 + \boldsymbol{v}_{1e}}{2\Delta_1(\hat{\boldsymbol{v}}_1)}\right) \\ & + \frac{2\gamma_1\|\boldsymbol{v}_{1e}\|^2}{(1+\|\boldsymbol{v}_{1e}\|^2)}\left(\frac{\|\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\|}{\Delta_1(\hat{\boldsymbol{v}}_1)}\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}\right) - \frac{2\gamma_1\lambda_m(\mathbf{D}_1)\|\boldsymbol{v}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}} \\ & * \left(1 - \frac{1}{8}\left(\varepsilon_2 + \frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\varepsilon_2}\right)\right) + \frac{2\gamma_1\boldsymbol{v}_{1e}^{\mathrm{T}}\mathbf{G}_1}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}(\boldsymbol{h}_4 + \boldsymbol{h}_5) - \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1 \\ & + \tilde{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{I}_{3\times 3} - \mathbf{K}_{02})\tilde{\boldsymbol{\eta}}_1 - \tilde{\boldsymbol{v}}_1^{\mathrm{T}}(\boldsymbol{h}_4 + \boldsymbol{h}_5) - \lambda_m(\mathbf{D}_1)\|\tilde{\boldsymbol{v}}_1\|^2\left(1 - \frac{\varepsilon_2}{2}\right) \\ & + \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\left(-\boldsymbol{h}_1 + \frac{2\gamma_1\mathbf{K}_{02}^{\mathrm{T}}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\right) + \frac{2\gamma_1\boldsymbol{v}_{1e}^{\mathrm{T}}\mathbf{G}_1\frac{\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}}{m}\mathbf{H}(\boldsymbol{q}_e, \boldsymbol{\alpha}_q)\boldsymbol{e}_3}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}} \\ & + \gamma_2\boldsymbol{q}_e^T\left(\mathbf{R}_2(\boldsymbol{q})(\boldsymbol{\alpha}_{\boldsymbol{\omega}} + \boldsymbol{\omega}_e) - \dot{\boldsymbol{\alpha}}_q\right) + \frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2} + \frac{\|\Delta_1(t)\Delta_1^{\mathrm{T}}(t)\|_\infty^1}{2m^2}. \end{aligned} \tag{4-92}$$

We will define the following interlace term:

$$-\gamma_2\boldsymbol{q}_e^{\mathrm{T}}\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\tilde{\boldsymbol{v}}_1 = \tilde{\boldsymbol{v}}_1^{\mathrm{T}}\boldsymbol{h}_4. \tag{4-93}$$

We will now define

$$\boldsymbol{q}_e^{\mathrm{T}}\mathbf{L} = \left(\mathbf{H}(\boldsymbol{q}_e, \boldsymbol{\alpha}_q)\boldsymbol{e}_3\right)^{\mathrm{T}}. \tag{4-94}$$

This implies that:

$$\mathbf{L}=\frac{8}{(1+\|\boldsymbol{q}\|^2)^2}\begin{bmatrix}\boldsymbol{q}_3 & 1 & -\boldsymbol{q}_1-\boldsymbol{\alpha}_{q,1}\\ -1 & \boldsymbol{q}_3 & -\boldsymbol{q}_2-\boldsymbol{\alpha}_{q,2}\\ \boldsymbol{\alpha}_{q,1} & \boldsymbol{\alpha}_{q,2} & 0\end{bmatrix}+\left(\frac{4\mathbf{I}_{3\times3}}{(1+\|\boldsymbol{q}\|^2)}-\frac{4(\boldsymbol{q}+\boldsymbol{\alpha}_q)\boldsymbol{\alpha}_q^{\mathrm{T}}}{(1+\|\boldsymbol{q}\|^2)\left(1+\|\boldsymbol{\alpha}_q\|^2\right)}\right)$$
$$\times\begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}-8\left(\frac{\boldsymbol{q}_e\left(\|\boldsymbol{q}_e\|^2+2+2\|\boldsymbol{\alpha}_q\|^2\right)+4\boldsymbol{\alpha}_q\left(1+\|\boldsymbol{q}\|^2-\boldsymbol{q}_e^{\mathrm{T}}\boldsymbol{\alpha}_q\right)}{(1+\|\boldsymbol{q}\|^2)^2\left(1+\|\boldsymbol{\alpha}_q\|^2\right)^2}\right)$$
$$\times\left(\begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}}\left(\boldsymbol{\alpha}_q\boldsymbol{\alpha}_q^{\mathrm{T}}-\|\boldsymbol{\alpha}_q\|^2\mathbf{I}_{3\times3}\right)+\boldsymbol{\alpha}_q^{\mathrm{T}}\begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\right). \tag{4-95}$$

Substituting (4-93)- (4-95)into (4-92) gives:

$$\dot{V}_3\le-\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}-\frac{2\gamma_1\boldsymbol{v}_{1e}^{\mathrm{T}}\mathbf{K}_2\sigma(\boldsymbol{v}_{1e})}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}-2\gamma_1c\boldsymbol{v}_{1e}^{\mathrm{T}}\left(\frac{2\boldsymbol{\alpha}_1+\boldsymbol{v}_{1e}}{2\Delta_1(\hat{\boldsymbol{v}}_1)}\right)$$
$$+\frac{2\gamma_1\|\boldsymbol{v}_{1e}\|^2}{(1+\|\boldsymbol{v}_{1e}\|^2)}\left(\frac{\|\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\|}{\Delta_1(\hat{\boldsymbol{v}}_1)}\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}\right)-\frac{2\gamma_1\lambda_m(\mathbf{D}_1)\|\boldsymbol{v}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}$$
$$*\left(1-\frac{\varepsilon_2}{8}-\frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{8\varepsilon_2}\right)+\frac{2\gamma_1\boldsymbol{v}_{1e}^{\mathrm{T}}\mathbf{G}_1}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}(\boldsymbol{h}_4+\boldsymbol{h}_5)-\tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1$$
$$+\tilde{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{I}_{3\times3}-\mathbf{K}_{02})\tilde{\boldsymbol{\eta}}_1-\tilde{\boldsymbol{v}}_1^{\mathrm{T}}\boldsymbol{h}_5-\lambda_m(\mathbf{D}_1)\|\tilde{\boldsymbol{v}}_1\|^2\left(1-\frac{\varepsilon_2}{2}\right)-\frac{2\gamma_1\boldsymbol{v}_{1e}^{\mathrm{T}}(\boldsymbol{h}_5)}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}$$
$$-\tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\left(-\boldsymbol{h}_1+\frac{2\gamma_1\mathbf{K}_{02}^{\mathrm{T}}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}+\mathbf{K}_{02}^{\mathrm{T}}\left(\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{\alpha}_2)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\right)^{\mathrm{T}}\right)$$
$$+\gamma_2\boldsymbol{q}_e^T\left(-\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}+\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\alpha}_{\boldsymbol{\omega}}-\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\right.$$
$$\left.*\mathbf{A}_{2,d}\boldsymbol{\Omega}_d-\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}(\mathbf{A}_{1,d}\boldsymbol{\Omega}+\boldsymbol{e}_3)\dot{\alpha}_\psi+\frac{2\gamma_1\|\boldsymbol{\Omega}\|\mathbf{L}}{\gamma_2m}\frac{\mathbf{G}_1^{\mathrm{T}}\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\right)$$
$$+\gamma_2\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e-\gamma_2\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\boldsymbol{h}_5+\frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2}. \tag{4-96}$$

To stabilise $\boldsymbol{q}_e$ and ensures that $\dot{V}_3$ is stable we will implement the standard backstopping method to design the virtual control signal $\boldsymbol{\alpha}_{\boldsymbol{\omega}}$ as follows:

$$-\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d+\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\right)+(\mathbf{A}_{1,d}\boldsymbol{\Omega}+\boldsymbol{e}_3)\dot{\alpha}_\psi\right)$$
$$+2\gamma_1\frac{\|\boldsymbol{\Omega}\|\mathbf{L}}{\gamma_2m}\frac{\mathbf{G}_1^{\mathrm{T}}\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}+\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\alpha}_{\boldsymbol{\omega}}=-\mathbf{K}_3\boldsymbol{q}_e. \tag{4-97}$$

Where the matrix $\mathbf{K}_3\in\mathbb{R}^{3\times3}$ is positive definite, $\mathbf{K}_3=diag(k_{3,1},k_{3,2},k_{3,3})$ where $k_{3,1}$, $k_{3,2}$ and $k_{3,3}$ are positive constants to be defined later. Thus $\boldsymbol{\alpha}_\omega$ is defined as follows:

$$\boldsymbol{\alpha}_\omega=\mathbf{R}_2^{-1}(\boldsymbol{q})\left(\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d+\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\right)+(\mathbf{A}_{1,d}\boldsymbol{\Omega}+\boldsymbol{e}_3)\dot{\alpha}_\psi\right)\right.$$
$$\left.-\mathbf{K}_3\boldsymbol{q}_e-\frac{2\gamma_1\|\boldsymbol{\Omega}\|\mathbf{L}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{v}_{1e}}{\gamma_2m\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\right). \tag{4-98}$$

Substituting (4-98) into (4-96):

$$
\begin{aligned}
\dot{V}_3 \leq & -\gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e}) \mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} - \frac{2\gamma_1 \boldsymbol{v}_{1e}^{\mathrm{T}} \mathbf{K}_2 \sigma(\boldsymbol{v}_{1e})}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}} - 2\gamma_1 c \boldsymbol{v}_{1e}^{\mathrm{T}} \left(\frac{2\boldsymbol{\alpha}_1 + \boldsymbol{v}_{1e}}{2\Delta_1(\hat{\boldsymbol{v}}_1)} \right) \\
& + \frac{2\gamma_1 \|\boldsymbol{v}_{1e}\|^2}{(1+\|\boldsymbol{v}_{1e}\|^2)} \left(\frac{\|\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})\|}{\Delta_1(\hat{\boldsymbol{v}}_1)} \sqrt{1+\|\boldsymbol{v}_{1e}\|^2} \right) - \frac{2\gamma_1 \lambda_m(\mathbf{D}_1) \|\boldsymbol{v}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}} \\
& * \left(1 - \frac{\varepsilon_2}{8} - \frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{8\varepsilon_2} \right) - \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}} \mathbf{K}_{01} \tilde{\boldsymbol{\eta}}_1 + \tilde{\boldsymbol{v}}_1^{\mathrm{T}} (\mathbf{I}_{3\times 3} - \mathbf{K}_{02}) \tilde{\boldsymbol{\eta}}_1 - \tilde{\boldsymbol{v}}_1^{\mathrm{T}} \boldsymbol{h}_5 \\
& - \lambda_m(\mathbf{D}_1) \|\tilde{\boldsymbol{v}}_1\|^2 \left(1 - \frac{\varepsilon_2}{2} \right) - \gamma_2 \boldsymbol{q}_e^{\mathrm{T}} \mathbf{K}_3 \boldsymbol{q}_e + \gamma_2 \boldsymbol{q}_e^{T} \mathbf{R}_2(\boldsymbol{q}) \boldsymbol{\omega}_e \\
& + \left(\frac{2\gamma_1 \boldsymbol{v}_{1e}^{\mathrm{T}} \mathbf{G}_1}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}} - \gamma_2 \boldsymbol{q}_e^{T} \mathbf{R}_2(\boldsymbol{\alpha}_q) \mathbf{T}(\boldsymbol{\alpha}_2)^{-1} \mathbf{A}_{2,d} \frac{\partial \boldsymbol{\Omega}}{\partial \hat{\boldsymbol{v}}_1} \right) \boldsymbol{h}_5 \\
& + \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}} \left(-\boldsymbol{h}_1 + \frac{2\gamma_1 \mathbf{K}_{02}^{\mathrm{T}} \mathbf{G}_1^{\mathrm{T}} \boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}} + \mathbf{K}_{02}^{\mathrm{T}} \left(\gamma_2 \boldsymbol{q}_e^{\mathrm{T}} \mathbf{R}_2(\boldsymbol{\alpha}_q) \mathbf{T}(\boldsymbol{\alpha}_2)^{-1} \mathbf{A}_{2,d} \frac{\partial \boldsymbol{\Omega}}{\partial \hat{\boldsymbol{v}}_1} \right)^{\mathrm{T}} \right) \\
& + \frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{2\varepsilon_1 \lambda_m(\mathbf{D}_1) m^2}.
\end{aligned}
\tag{4-99}
$$

To prepare for the next step we calculate $\dot{\boldsymbol{\alpha}}_{\boldsymbol{\omega}}$ by applying the stochastic differentiation rule differentiating both sides of (4-98) we obtain.

$$
\begin{aligned}
\dot{\boldsymbol{\alpha}}_\omega = & \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \mathbf{q}} \dot{\mathbf{q}} + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \hat{\boldsymbol{v}}_1} \dot{\hat{\boldsymbol{v}}}_1 + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \alpha_\psi} \dot{\alpha}_\psi + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \dot{\alpha}_\psi} \ddot{\alpha}_\psi + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\eta}_{1e}} \dot{\boldsymbol{\eta}}_{1e} + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\eta}_{1d}} \dot{\boldsymbol{\eta}}_{1d} + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \dot{\boldsymbol{\eta}}_{1d}} \ddot{\boldsymbol{\eta}}_{1d} \\
& + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \ddot{\boldsymbol{\eta}}_{1d}} \dddot{\boldsymbol{\eta}}_{1d} + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \dddot{\boldsymbol{\eta}}_{1d}} \ddddot{\boldsymbol{\eta}}_{1d},
\end{aligned}
\tag{4-100}
$$

where the partial derivative of $\boldsymbol{\alpha}_\omega$ are presented in Appendix E. We will define the following:

$$
\dot{\boldsymbol{\alpha}}_\omega = \boldsymbol{\alpha}_{\omega,d} + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\eta}_{1e}} \frac{\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} \tilde{\boldsymbol{v}}_1 + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \hat{\boldsymbol{v}}_1} \mathbf{K}_{01} \tilde{\boldsymbol{\eta}}_1,
\tag{4-101}
$$

where we define:

$$
\begin{aligned}
\boldsymbol{\alpha}_{\omega,d} = & \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \mathbf{q}} \dot{\mathbf{q}} - \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \hat{\boldsymbol{v}}_1} \left(\frac{f}{m} \mathbf{R}(\boldsymbol{\eta}_2) \boldsymbol{e}_3 - \mathbf{D}_1 \hat{\boldsymbol{v}}_1 - g \boldsymbol{e}_3 + \boldsymbol{h}_2 + \boldsymbol{h}_3 + \boldsymbol{h}_4 + \boldsymbol{h}_5 \right) + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \alpha_\psi} \dot{\alpha}_\psi \\
& + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \dot{\alpha}_\psi} \ddot{\alpha}_\psi + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\eta}_{1e}} (\hat{\boldsymbol{v}}_1 - \dot{\boldsymbol{\eta}}_{1d}.) + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\eta}_{1d}} \dot{\boldsymbol{\eta}}_{1d} + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \dot{\boldsymbol{\eta}}_{1d}} \ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \ddot{\boldsymbol{\eta}}_{1d}} \dddot{\boldsymbol{\eta}}_{1d} + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \dddot{\boldsymbol{\eta}}_{1d}} \ddddot{\boldsymbol{\eta}}_{1d}.
\end{aligned}
\tag{4-102}
$$

Step 2: Now we will design a control law for the torque vector $\boldsymbol{\tau}$ to stabilize $\boldsymbol{\omega}_e$ at the origin. If we denote the error between the actual and the observed aerodynamic torque disturbance as

$$\boldsymbol{\tau}_{de} = \bar{\boldsymbol{\tau}}_{Aero} - \hat{\boldsymbol{\tau}}_{Aero}. \tag{4-103}$$

Replacing $\boldsymbol{\tau}_{Aero}$ with the observed disturbance $\hat{\boldsymbol{\tau}}_{Aero}$ and error $\boldsymbol{\tau}_{de}$ in the second equation of (4-14), $\dot{\boldsymbol{\omega}}$ becomes:

$$\dot{\boldsymbol{\omega}} = -\mathbf{I}_H^{-1}\boldsymbol{\omega}\times(\mathbf{I}_H\boldsymbol{\omega}) + \mathbf{I}_H^{-1}\boldsymbol{\tau} + \mathbf{I}_H^{-1}(\hat{\boldsymbol{\tau}}_{Aero} + \boldsymbol{\tau}_{de}). \tag{4-104}$$

Differentiating both sides of (4-80) along the solutions of (4-104) and the solution of (4-102) yields $\dot{\boldsymbol{\omega}}_e$ as follows:

$$\dot{\boldsymbol{\omega}}_e = -\mathbf{I}_H^{-1}\boldsymbol{\tau} - \mathbf{I}_H^{-1}\boldsymbol{\omega}\times(\mathbf{I}_H\boldsymbol{\omega}) + \mathbf{I}_H^{-1}(\hat{\boldsymbol{\tau}}_{Aero} + \boldsymbol{\tau}_{de}) - -\boldsymbol{\alpha}_{\omega,d} - \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1e}}\tilde{\boldsymbol{\nu}}_1 - \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\hat{\boldsymbol{\nu}}_1}\mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1. \tag{4-105}$$

To analyse the stability of $\boldsymbol{\omega}_e$ at the origin of we consider the Lyapunov function:

$$V_4 = V_3 + \frac{\gamma_3}{2}\sqrt{1+\|\boldsymbol{\omega}_e\|^4} + +\tfrac{1}{2}\|\boldsymbol{\tau}_{de}\|^2 - \frac{\gamma_3}{2}. \tag{4-106}$$

The derivative of (4-106) along the solutions of (4-99) and (4-105) is:

$$\begin{aligned}
\dot{V}_4 \le & -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} - \frac{2\gamma_1\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{K}_2\sigma(\boldsymbol{\nu}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} - 2\gamma_1 c\boldsymbol{\nu}_{1e}^{\mathrm{T}}\left(\frac{2\boldsymbol{\alpha}_1+\boldsymbol{\nu}_{1e}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right) \\
& + \frac{2\gamma_1\|\boldsymbol{\nu}_{1e}\|^2}{(1+\|\boldsymbol{\nu}_{1e}\|^2)}\left(\frac{\|\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\|}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}\right) - \frac{2\gamma_1\lambda_m(\mathbf{D}_1)\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \\
& * \left(1 - \frac{\varepsilon_2}{8} - \frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{8\varepsilon_2}\right) - \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1 + \tilde{\boldsymbol{\nu}}_1^{\mathrm{T}}(\mathbf{I}_{3\times3} - \mathbf{K}_{02})\tilde{\boldsymbol{\eta}}_1 - \tilde{\boldsymbol{\nu}}_1^{\mathrm{T}}\boldsymbol{h}_5 \\
& -\lambda_m(\mathbf{D}_1)\|\tilde{\boldsymbol{\nu}}_1\|^2\left(1 - \frac{\varepsilon_2}{2}\right) - \gamma_2\boldsymbol{q}_e^{\mathrm{T}}\mathbf{K}_3\boldsymbol{q}_e + \gamma_2\boldsymbol{q}_e^{T}\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e \\
& + \left(\frac{2\gamma_1\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{G}_1}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} - \gamma_2\boldsymbol{q}_e^{T}\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{\nu}}_1}\right)\boldsymbol{h}_5 \\
& + \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\left(-\boldsymbol{h}_1 + \frac{2\gamma_1\mathbf{K}_{02}^{\mathrm{T}}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} + \mathbf{K}_{02}^{\mathrm{T}}\left(\gamma_2\boldsymbol{q}_e^{\mathrm{T}}\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{\nu}}_1}\right)^{\mathrm{T}}\right) \\
& + \gamma_3\frac{\|\boldsymbol{\omega}_e\|^2\boldsymbol{\omega}_e^{\mathrm{T}}}{\sqrt{1+\|\boldsymbol{\omega}_e\|^4}}\Bigg((\mathbf{I}_A + \mathbf{I}_H)^{-1}(-\mathbf{D}_2\boldsymbol{\omega} - \mathbf{C}_{B22}(\boldsymbol{\omega})\boldsymbol{\omega} - \mathbf{C}_{A22}(\boldsymbol{\omega})\boldsymbol{\omega} + \boldsymbol{\tau} + \bar{\boldsymbol{\tau}}_{Aero}) - \boldsymbol{\alpha}_{\omega,d} \\
& \qquad - \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1e}}\tilde{\boldsymbol{\nu}}_1 - \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\hat{\boldsymbol{\nu}}_1}\mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1\Bigg) + \boldsymbol{\tau}_{de}^{\mathrm{T}}\dot{\boldsymbol{\tau}}_{de} + + \frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2}.
\end{aligned} \tag{4-107}$$

Where we define:

$$-\gamma_3\frac{\|\boldsymbol{\omega}_e\|^2}{\sqrt{1+\|\boldsymbol{\omega}_e\|^4}}\left(\boldsymbol{\omega}_{qe}^{T}\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} = \boldsymbol{h}_5. \tag{4-108}$$

Applying Young's inequality to (4-107) yields:

$$\boldsymbol{q}_{qe}^{T}\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e \leq \varepsilon_3\|\boldsymbol{q}_e\|^2 + \varepsilon_4\left(\frac{1}{4\varepsilon_3}\right)^2\frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^3} + \frac{\boldsymbol{q}_{qe}^{T}\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{3/4}} + \frac{1}{4\varepsilon_4}, \tag{4-109}$$

$$\frac{\gamma_3\|\boldsymbol{\omega}_e\|^2\boldsymbol{\omega}_e^{\mathrm{T}}(\mathbf{I}_A+\mathbf{I}_H)^{-1}\boldsymbol{\tau}_{de}}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}} \leq \frac{\varepsilon_6}{4}\gamma_3^2\frac{\gamma_3\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^1}\left\|\boldsymbol{\omega}_e^{\mathrm{T}}(\mathbf{I}_A+\mathbf{I}_H)^{-1}\right\|^2 + \frac{\|\boldsymbol{\tau}_{de}\|^2}{\varepsilon_6}, \tag{4-110}$$

Where ε_i are positive constants using (4-108)-(4-110), we rewrite (4-107) as:

$$\begin{aligned}
\dot{V}_4 \leq & -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} - \frac{2\gamma_1\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{K}_2\sigma(\boldsymbol{\nu}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} - 2\gamma_1 c\boldsymbol{\nu}_{1e}^{\mathrm{T}}\left(\frac{2\boldsymbol{\alpha}_1+\boldsymbol{\nu}_{1e}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)\\
& + \frac{2\gamma_1\|\boldsymbol{\nu}_{1e}\|^2}{(1+\|\boldsymbol{\nu}_{1e}\|^2)}\left(\frac{\|\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\|}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}\right) - \frac{2\gamma_1\lambda_m(\mathbf{D}_1)\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\\
& * \left(1-\frac{\varepsilon_2}{8}-\frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{8\varepsilon_2}\right) - \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1 + \tilde{\boldsymbol{\nu}}_1^{\mathrm{T}}(\mathbf{I}_{3\times3}-\mathbf{K}_{02})\tilde{\boldsymbol{\eta}}_1 - \lambda_m(\mathbf{D}_1)\|\tilde{\boldsymbol{\nu}}_1\|^2\left(\frac{2-\varepsilon_2}{2}\right)\\
& + \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\Bigg(-\boldsymbol{h}_1 + \frac{2\gamma_1\mathbf{K}_{02}^{\mathrm{T}}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} - \mathbf{K}_{02}^{\mathrm{T}}\left(\gamma_2\boldsymbol{q}_e^{\mathrm{T}}\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{\nu}}_1}\right)^{\mathrm{T}}\\
& \qquad - \left(\frac{\gamma_3\|\boldsymbol{\omega}_e\|^2\boldsymbol{\omega}_e^{T}}{(1+\|\boldsymbol{\omega}_e\|^4)^{3/4}}\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\hat{\boldsymbol{\nu}}_1}\mathbf{K}_{01}\right)^{\mathrm{T}}\Bigg) - \gamma_2\boldsymbol{q}_e^{\mathrm{T}}(\mathbf{K}_3\boldsymbol{q}_e - \varepsilon_3\boldsymbol{q}_e)\\
& + \frac{\gamma_3\|\boldsymbol{\omega}_e\|^2\boldsymbol{\omega}_e^{T}}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}\Bigg((\mathbf{I}_A+\mathbf{I}_H)^{-1}(-\mathbf{D}_2\boldsymbol{\omega} - \mathbf{C}_{B22}(\boldsymbol{\omega})\boldsymbol{\omega} - \mathbf{C}_{A22}(\boldsymbol{\omega})\boldsymbol{\omega} + \boldsymbol{\tau} + \hat{\boldsymbol{\tau}}_{Aero})\\
& \qquad - \boldsymbol{\alpha}_{\omega,d} - \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1e}}\left(\frac{2\gamma_1\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{G}_1}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} - \gamma_2\boldsymbol{q}_e^{T}\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{\nu}}_1}\right)^{\mathrm{T}}\\
& \qquad + \left(\frac{\gamma_2}{\gamma_3}\left(\left(\frac{1}{4\varepsilon_3}\right)^2\frac{\varepsilon_4\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{2.25}} + \frac{\boldsymbol{q}_{qe}^{T}\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e}{(1+\|\boldsymbol{\omega}_e\|^4)^{1/4}}\right)\right.\\
& \qquad \left. + \gamma_3\frac{\varepsilon_6}{4}\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\right\|^2\frac{\|\boldsymbol{\omega}_e\|^2}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}\right)\boldsymbol{\omega}_e\Bigg)\\
& -\boldsymbol{\tau}_{de}^{\mathrm{T}}\dot{\hat{\boldsymbol{\tau}}}_{Aero} + \left(\frac{1}{\varepsilon_6}+\frac{\varepsilon_7}{4}\right)\|\boldsymbol{\tau}_{de}\|^2 + \frac{\left\|\bar{\boldsymbol{f}}_{Aero}\right\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2} + \frac{\left\|\dot{\boldsymbol{\tau}}_{Aero}\right\|^2}{\varepsilon_7} + \frac{1}{4\varepsilon_4}.
\end{aligned} \tag{4-111}$$

We must now define a disturbance observer to estimate the disturbance torque acting on the system, by implementing the disturbance observer presented in (2-124) we obtain:

$$\begin{aligned}
\hat{\boldsymbol{\tau}}_{Aero} &= \boldsymbol{\xi}_1 + \mathbf{K}_{2d}(\mathbf{I}_A+\mathbf{I}_H)\boldsymbol{\omega}_e\\
\dot{\boldsymbol{\xi}}_1 &= \mathbf{K}_{2d}(\mathbf{I}_A+\mathbf{I}_H)\Bigg(\Big((\mathbf{I}_A+\mathbf{I}_H)^{-1}(-\mathbf{D}_2\boldsymbol{\omega} - \mathbf{C}_{B22}(\boldsymbol{\omega})\boldsymbol{\omega} - \mathbf{C}_{A22}(\boldsymbol{\omega})\boldsymbol{\omega} + \boldsymbol{\tau}) - \boldsymbol{\alpha}_{\omega,d}\\
&\qquad - \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\hat{\boldsymbol{\nu}}_1}\mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1 - \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1e}}\tilde{\boldsymbol{\nu}}_1\Big)\Bigg),\\
\boldsymbol{\xi}_1(t_0) &= -\mathbf{K}_{2d}\boldsymbol{\omega}_{\boldsymbol{e}}(t_0),
\end{aligned} \tag{4-112}$$

where the matrix $\mathbf{K}_{2\mathrm{d}} \in \mathbb{R}^{3\times3}$ is positive definite, $\mathbf{K}_{2\mathrm{d}} = diag(k_{2d,1}, k_{2d,2}, k_{2d,3})$ where $k_{2d,1}$, $k_{2d,2}$ and $k_{2d,3}$ are positive constants to be defined later. We will define the integral of $\boldsymbol{\xi}_1$ as follows:

$$\boldsymbol{\xi}_1 = \mathbf{K}_{2d}(\mathbf{I}_A + \mathbf{I}_H)\Bigg(\int_{t_0}^{t}\Bigg((\mathbf{I}_A + \mathbf{I}_H)^{-1}\Big(-\mathbf{D}_2\boldsymbol{\omega}(\sigma) - \mathbf{C}_{B22}\big(\boldsymbol{\omega}(\sigma)\big)\boldsymbol{\omega}(\sigma) - \mathbf{C}_{A22}\big(\boldsymbol{\omega}(\sigma)\big)\boldsymbol{\omega}(\sigma) + \boldsymbol{\tau}(\sigma)\Big) - \boldsymbol{\alpha}_{\omega,d}(\sigma) - \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\hat{\boldsymbol{\nu}}_1}\mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1(\sigma) + \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1e}}\hat{\boldsymbol{\nu}}_1(\sigma)\Bigg)d\sigma - \int_{\boldsymbol{\eta}_1(t_0)}^{\boldsymbol{\eta}_1(t)}\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1e}}\boldsymbol{\sigma}(s)d\boldsymbol{\sigma}(s)\Bigg), \tag{4-113}$$

Due to the selection of $\boldsymbol{\xi}_1$ in (4-112)and taking the total derivative of both side of (4-112) gives:

$$\dot{\boldsymbol{\tau}}_{de} = -\mathbf{K}_{2d}\boldsymbol{\tau}_{de} + \dot{\boldsymbol{\tau}}_{Aero}, \tag{4-114}$$

furthermore,

$$\|\boldsymbol{\tau}_{de}\| \leq \sqrt{\left(\|\boldsymbol{\tau}_{de}(t_0)\|^2 - \frac{\tau_{1dN}^2 + \varepsilon_6\varepsilon_7^2}{2\lambda_m^2(\mathbf{K}_{2d})}\right)e^{-\lambda_m(\mathbf{K}_{2d})(t-t_0)} + \frac{\tau_{1dN}^2 + \varepsilon_6\varepsilon_7^2}{2\lambda_m^2\mathbf{K}_{2d}}} \coloneqq \varrho_{3\tau}, \tag{4-115}$$
$$\|\hat{\boldsymbol{\tau}}_{Aero}\| \leq \frac{\lambda_M(\mathbf{K}_{2d})}{\lambda_m(\mathbf{K}_{2d})}\tau_{dM} + \frac{\varepsilon_6}{2}\|\boldsymbol{\tau}_{de}\|^2\varepsilon_7^2 + \frac{1}{2\varepsilon_6}\|\boldsymbol{f}_{de}\|^2 \coloneqq \varrho_{4\tau}.$$

Using Young's inequality, we can state:

$$\boldsymbol{\tau}_{de}^{\mathrm{T}}\dot{\boldsymbol{\tau}}_{Aero} \leq \frac{\varepsilon_9}{4}\|\boldsymbol{\tau}_{de}\|^2 + \frac{\left\|\dot{\boldsymbol{\tau}}_{Aero}^{\mathrm{M}}\right\|^2}{\varepsilon_9}. \tag{4-116}$$

The control torque is chosen as:

$$\boldsymbol{\tau} = (\mathbf{I}_A + \mathbf{I}_H)\Bigg[-(\mathbf{I}_A + \mathbf{I}_H)^{-1}(\mathbf{D}_2\boldsymbol{\alpha}_\omega - \mathbf{C}_{B22}(\boldsymbol{\omega})\boldsymbol{\omega} - \mathbf{C}_{A22}(\boldsymbol{\omega})\boldsymbol{\omega}) + \boldsymbol{\alpha}_{\omega,d} - \mathbf{K}_4\boldsymbol{\omega}_e - \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1e}}\left(\frac{2\gamma_1\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{G}_1}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} - \gamma_2\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{\nu}}_1}\right)^{\mathrm{T}} - \left(\frac{\gamma_2}{\gamma_3}\left(\left(\frac{1}{4\varepsilon_3}\right)^2\frac{\varepsilon_4\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{2.25}} + \frac{\boldsymbol{q}_{qe}^T\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e}{(1+\|\boldsymbol{\omega}_e\|^4)^{1/4}}\right) + \gamma_3\frac{\varepsilon_6}{4}\left\|(\mathbf{I}_A + \mathbf{I}_H)^{-1}\right\|^2\frac{\|\boldsymbol{\omega}_e\|^2}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}\right)\boldsymbol{\omega}_e\Bigg] - \hat{\boldsymbol{\tau}}_{Aero}. \tag{4-117}$$

We will now define the interlace term $\boldsymbol{h}_1$:

$$\boldsymbol{h}_1 = \mathbf{K}_{02}^{\mathrm{T}}\left(\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e} + \gamma_2\left(\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\partial\hat{\boldsymbol{\nu}}_1}\right)^{\mathrm{T}} - \gamma_3\left(\boldsymbol{\omega}_e^{\mathrm{T}}\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\hat{\boldsymbol{\nu}}_1}\right)^{\mathrm{T}}\right). \tag{4-118}$$

In addition, we now obtain:

$$\dot{V}_4 \leq -\gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} - \frac{2\gamma_1 \boldsymbol{v}_{1e}^{\mathrm{T}} \mathbf{K}_2 \sigma(\boldsymbol{v}_{1e})}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}} - 2\gamma_1 c \boldsymbol{v}_{1e}^{\mathrm{T}} \left(\frac{2\boldsymbol{\alpha}_1 + \boldsymbol{v}_{1e}}{2\Delta_1(\hat{\boldsymbol{v}}_1)}\right)$$
$$+ \frac{2\gamma_1\|\boldsymbol{v}_{1e}\|^2}{(1+\|\boldsymbol{v}_{1e}\|^2)} \left(\frac{\|\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\|}{\Delta_1(\hat{\boldsymbol{v}}_1)} \sqrt{1+\|\boldsymbol{v}_{1e}\|^2}\right) - \frac{2\gamma_1\lambda_m(\mathbf{D}_1)\|\boldsymbol{v}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}$$
$$* \left(1 - \frac{\varepsilon_2}{8} - \frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{8\varepsilon_2}\right) - \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}} \mathbf{K}_{01} \tilde{\boldsymbol{\eta}}_1 + \tilde{\boldsymbol{v}}_1^{\mathrm{T}} (\mathbf{I}_{3\times 3} - \mathbf{K}_{02}) \tilde{\boldsymbol{\eta}}_1 - \lambda_m(\mathbf{D}_1)\|\tilde{\boldsymbol{v}}_1\|^2 \left(1 - \frac{\varepsilon_2}{2}\right)$$
$$-\gamma_2 \boldsymbol{q}_e^{\mathrm{T}}(\mathbf{K}_3\boldsymbol{q}_e - \varepsilon_5 \boldsymbol{q}_e) - \gamma_3 \boldsymbol{\omega}_e^T(\mathbf{K}_4 + \mathbf{D}_2)\boldsymbol{\omega}_e - \left(\lambda_m(\mathbf{K}_{1d}) - \frac{1}{\varepsilon_6} - \frac{\varepsilon_7}{4}\right)\|\boldsymbol{\tau}_{de}\|^2$$
$$+ \frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2} + \frac{\|\dot{\bar{\boldsymbol{\tau}}}_{Aero}\|^2}{\varepsilon_7} + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{4} \frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\varepsilon_2\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}} + \frac{1}{4}\varepsilon_4, \tag{4-119}$$

We will now substitute (4-110) into (4-22) to give:

$$\dot{\tilde{\boldsymbol{\eta}}}_1 = \tilde{\boldsymbol{v}}_1 - \mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1 - \frac{2\gamma_1 \mathbf{K}_{02}^{\mathrm{T}} \mathbf{G}_1^T \boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}} + \gamma_2 \mathbf{K}_{02}^{\mathrm{T}} \left(\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d} \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \boldsymbol{q}_e$$
$$+\gamma_3 \mathbf{K}_{02}^{\mathrm{T}} \left(\frac{\partial \boldsymbol{\alpha}_\omega}{\partial \hat{\boldsymbol{v}}_1}\right)^{\mathrm{T}} \boldsymbol{\omega}_e. \tag{4-120}$$

We will now substitute (4-53), (4-54), (4-93) and (4-108) into (4-26) to give:

$$\dot{\tilde{\boldsymbol{v}}}_1 = \left(\mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1 - \mathbf{D}_1\tilde{\boldsymbol{v}}_1 + \tfrac{1}{m}\bar{\boldsymbol{f}}_{Aero} - \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\sigma(\boldsymbol{\eta}_{1e}) - \frac{2\gamma_1 \mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e}) \mathbf{G}_1^T \boldsymbol{v}_{1e}}{\Delta_1(\hat{\boldsymbol{v}}_1)\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\right.$$
$$\left. + \gamma_2 \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathbf{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_e + \gamma_3 \left(\frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \boldsymbol{\omega}_e\right). \tag{4-121}$$

Let us state:

$$\frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2} + \frac{\|\dot{\bar{\boldsymbol{\tau}}}_{Aero}\|^2}{\varepsilon_7} + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{4} \frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\varepsilon_2\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}} + \frac{1}{4}\varepsilon_4$$
$$\leq \left(\frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2} + \frac{\|\dot{\bar{\boldsymbol{\tau}}}_{Aero}\|^2}{\varepsilon_7} + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{4} \frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\varepsilon_2\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}} + \frac{1}{4}\varepsilon_4\right)\left(\frac{2+\|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{v}}_1)}\right.$$
$$\left. + \left(\boldsymbol{v}_{1e}^{\mathrm{T}} \frac{2\boldsymbol{\alpha}_1 + \boldsymbol{v}_{1e}}{2\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\right), \tag{4-122}$$

Since $\boldsymbol{\alpha}_1$ is bounded as per the use of the one-step ahead backstepping method, it can be stated:

$$\|\boldsymbol{\alpha}_1\| \leq \varrho_{\boldsymbol{\alpha}_1}, \tag{4-123}$$

This implies that:

$$\lambda_m(\bar{\mathbf{K}}_1) = \lambda_m(\mathbf{K}_1) - \frac{1}{2\varepsilon_2}\frac{2+\varrho_{\boldsymbol{\alpha}_1}^2}{2} > 0. \tag{4-124}$$

Furthermore, because:

$$\|\sigma(\boldsymbol{\eta}_{1e})\|^2 + \sum_{i=1}^{3} \frac{1}{1+\boldsymbol{\eta}_{1e,i}^2} = 3, \tag{4-125}$$

It can be stated that:

$$\begin{aligned}\left(\frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2} + \frac{\|\dot{\bar{\boldsymbol{\tau}}}_{Aero}\|^2}{\varepsilon_7} + \frac{1}{4}\varepsilon_4\right)\frac{2+\|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{v}}_1)} &= \frac{2+\|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{v}}_1)}\left(\frac{1}{3}\sum_{i=1}^{3}\frac{1}{1+\boldsymbol{\eta}_{1e,i}^2}\left(\frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2} + \frac{\|\dot{\bar{\boldsymbol{\tau}}}_{Aero}\|^2}{\varepsilon_7} + \frac{1}{4}\varepsilon_4\right)\right. \\ &\left. + \frac{1}{3}\sum_{i=1}^{3}\frac{\boldsymbol{\eta}_{1e,i}^2}{1+\boldsymbol{\eta}_{1e,i}^2}\left(\frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2} + \frac{\|\dot{\bar{\boldsymbol{\tau}}}_{Aero}\|^2}{\varepsilon_7} + \frac{1}{4}\varepsilon_4\right)\right),\end{aligned} \tag{4-126}$$

$$\lambda_m(\mathbf{K}_1)\frac{\lambda_m(\mathbf{D}_1)}{2\Delta_1(\hat{\boldsymbol{v}}_1)}c_2 = \frac{1}{3}\left(\frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2} + \frac{\|\dot{\bar{\boldsymbol{\tau}}}_{Aero}\|^2}{\varepsilon_7} + \frac{1}{4}\varepsilon_4\right)\frac{2+\varrho_{\alpha_1}^2}{2\Delta_1(\hat{\boldsymbol{v}}_1)}, \tag{4-127}$$

$$c_3 = \frac{1}{3}\left(\frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2} + \frac{\|\dot{\bar{\boldsymbol{\tau}}}_{Aero}\|^2}{\varepsilon_7} + \frac{1}{4}\varepsilon_4\right)\frac{2+\varrho_{\alpha_1}^2}{2}, \tag{4-128}$$

We thus obtain

$$\begin{aligned}\dot{V}_4 \le & -\frac{\lambda_m(\mathbf{D}_1)}{2}\lambda_m(\mathbf{K}_1)(\gamma_1 - c_2)\frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\Delta_1(\hat{\boldsymbol{v}}_1)} - \frac{2\gamma_1\boldsymbol{v}_{1e}^{\mathrm{T}}\mathbf{K}_2\sigma(\boldsymbol{v}_{1e})}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}} \\ & -\left(2\gamma_1 c - \frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2} - \frac{\|\dot{\bar{\boldsymbol{\tau}}}_{Aero}\|^2}{\varepsilon_7} - \frac{1}{4}\varepsilon_4\right)\boldsymbol{v}_{1e}^{\mathrm{T}}\left(\frac{2\boldsymbol{\alpha}_1 + \boldsymbol{v}_{1e}}{2\Delta_1(\hat{\boldsymbol{v}}_1)}\right) \\ & + \frac{2\gamma_1\|\boldsymbol{v}_{1e}\|^2}{(1+\|\boldsymbol{v}_{1e}\|^2)}\left(\frac{\|\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\|}{\Delta_1(\hat{\boldsymbol{v}}_1)}\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}\right) - \frac{2\gamma_1\lambda_m(\mathbf{D}_1)\|\boldsymbol{v}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}} \\ & * \left(1 - \frac{\varepsilon_2}{8} - \frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{8\varepsilon_2}\right) - \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1 + \tilde{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{I}_{3\times 3} - \mathbf{K}_{02})\tilde{\boldsymbol{\eta}}_1 \\ & -\lambda_m(\mathbf{D}_1)\|\tilde{\boldsymbol{v}}_1\|^2\left(1 - \frac{\varepsilon_2}{2}\right) - \gamma_2\boldsymbol{q}_e^{\mathrm{T}}(\mathbf{K}_3\boldsymbol{q}_e - \varepsilon_3\boldsymbol{q}_e) - \gamma_3\frac{\|\boldsymbol{\omega}_e\|^2\boldsymbol{\omega}_e^T(\mathbf{K}_4 + \mathbf{D}_2)\boldsymbol{\omega}_e}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}} \\ & -\left(\lambda_m(\mathbf{K}_{1d}) - \frac{1}{\varepsilon_6} - \frac{\varepsilon_7}{4}\right)\|\boldsymbol{\tau}_{de}\|^2 + \sum_{i=1}^{3}\frac{c_3}{\Delta_1(\hat{\boldsymbol{v}}_1(t))(1+\boldsymbol{\eta}_{1e,i}^2)}.\end{aligned} \tag{4-129}$$

In addition, we make the following conditions on the control gains:

$$b_1 < \frac{1+\|\boldsymbol{v}_{1e}\|^2}{\Delta_1(\hat{\boldsymbol{v}}_1)} < b_2, \qquad \forall \hat{\boldsymbol{v}}_1(t), \boldsymbol{\eta}_{1e}(t), \dot{\boldsymbol{\eta}}_{1d}(t) \in \mathbb{R}^{3\times 3}, t \in \mathbb{R}, \tag{4-130}$$

$$\bar{\gamma}_1 = \gamma_1 - c_4 > 0, \tag{4-131}$$

$$\|\sigma(\boldsymbol{\eta}_{1e})\|^2 < 3, \tag{4-132}$$

$$\lambda_m(\bar{\mathbf{K}}_2) = \lambda_m(\mathbf{K}_2) - \left(\|\mathbf{K}_1\|\sqrt{b_2}\right) > 0, \tag{4-133}$$

$$c = \frac{1}{2\gamma_1}\left(\frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2} + \frac{\|\dot{\bar{\boldsymbol{\tau}}}_{Aero}\|^2}{\varepsilon_7} + \frac{1}{4}\varepsilon_4\right), \tag{4-134}$$

$$\lambda_m(\bar{\mathbf{K}}_3) = \lambda_m(\mathbf{K}_3) - \varepsilon_3 > 0, \tag{4-135}$$

$$\lambda_m(\bar{\mathbf{D}}_1) = \lambda_m(\mathbf{D}_1)\left(1 - \frac{\varepsilon_2}{8} - \frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{8\varepsilon_2}\right) > 0, \quad (4\text{-}136)$$

$$\lambda_m(\bar{\bar{\mathbf{D}}}_1) = \lambda_m(\mathbf{D}_1)\left(1 - \frac{\varepsilon_2}{2}\right) > 0, \quad (4\text{-}137)$$

$$\lambda_m(\bar{\mathbf{K}}_{1d}) = \lambda_m(\mathbf{K}_{1d}) - \frac{1}{\varepsilon_6} - \frac{\varepsilon_7}{4} > 0, \quad (4\text{-}138)$$

$$c_3 = \frac{1}{3}\left(\frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2} + \frac{\|\dot{\bar{\boldsymbol{\tau}}}_{Aero}\|^2}{\varepsilon_7} + \frac{1}{4}\varepsilon_4\right)\frac{2 + \varrho_{\boldsymbol{\alpha}_1}^2}{2}. \quad (4\text{-}139)$$

we now state:

$$\begin{aligned}\dot{V}_4 \leq\ & -\bar{\gamma}_1\frac{\lambda_m(\mathbf{D}_1)}{2}\lambda_m(\bar{\mathbf{K}}_1)\frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\Delta_1(\hat{\boldsymbol{v}}_1)} - 2\gamma_1\frac{\lambda_m(\bar{\mathbf{K}}_2)\boldsymbol{\nu}_{1e}^{\mathrm{T}}\sigma(\boldsymbol{\nu}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} - 2\gamma_1\frac{\lambda_m(\bar{\mathbf{D}}_1)\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \\ & -\lambda_m(\mathbf{K}_{01})\|\tilde{\boldsymbol{\eta}}_1\|^2 - \lambda_m(\bar{\bar{\mathbf{D}}}_1)\|\tilde{\boldsymbol{v}}_1\|^2 - \gamma_2\lambda_m(\bar{\mathbf{K}}_3)\|\boldsymbol{q}_e\|^2 - \frac{\gamma_3\lambda_m(\mathbf{K}_4)\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}} \\ & -\lambda_m(\bar{\mathbf{K}}_{1d})\|\boldsymbol{\tau}_{de}\|^2 + \sum_{i=1}^{3}\frac{c_3}{\Delta_1(\hat{\boldsymbol{v}}_1(t))(1+\boldsymbol{\eta}_{1e,i}^2)}.\end{aligned} \quad (4\text{-}140)$$

The control design has been completed, it can be stated that the closed loop system consisting of $\tilde{\boldsymbol{\eta}}_1, \tilde{\boldsymbol{v}}_1, \boldsymbol{\eta}_{1e}, \boldsymbol{\nu}_{1e}, \boldsymbol{q}_e, \boldsymbol{\omega}_e$ and $\boldsymbol{\tau}_{de}$ and (2-122) is asymptotically stable. We will not specify the initial conditions or control gains in this section this will be done in the next section as the steps taken for the stochastic control design and the deterministic controller are near identical.

4.1.4 Stability Analysis and Initial Condition Specification

It should be noted that the selection of initial conditions and gains $\mathbf{K}_3, \mathbf{K}_4$, and μ must be chosen through trial and error. However, it is possible to prove exponential stability to a large ball centred at the solution and in this case provide explicit conditions on all control gains and initial conditions. To do this we must first note that the use of the Modified Rodriquez Parameter implies that a singularity occurs at an angular position of $\pm n2\pi,\ n = 1,2,3 \ldots$ To address this we must recall that:

$$\boldsymbol{q}(\boldsymbol{\eta}_2) = \begin{bmatrix} \dfrac{\sin\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) - \cos\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right)}{1 + \cos\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) + \sin\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right)} \\ \dfrac{\cos\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) + \sin\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right)}{1 + \cos\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) + \sin\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right)} \\ \dfrac{\cos\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right) - \sin\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right)}{1 + \cos\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) + \sin\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right)} \end{bmatrix} \quad (4\text{-}141)$$

The MRP experiences a singularity at the following conditions

Euler angle			MRP		
ϕ	θ	ψ	q_1	q_2	q_3
0	0	*2π*	Singularity	Singularity	Singularity
0	0	*-2π*	Singularity	Singularity	Singularity
0	*2π*	0	Singularity	Singularity	Singularity
0	*-2π*	0	Singularity	Singularity	Singularity
2π	0	0	Singularity	Singularity	Singularity
-2π	0	0	Singularity	Singularity	Singularity
2π	*2π*	*2π*	Singularity	Singularity	Singularity
2π	*2π*	*-2π*	Singularity	Singularity	Singularity
2π	*-2π*	*2π*	Singularity	Singularity	Singularity
-2π	*2π*	*2π*	Singularity	Singularity	Singularity
2π	*-2π*	*-2π*	Singularity	Singularity	Singularity
-2π	*2π*	*-2π*	Singularity	Singularity	Singularity
-2π	*-2π*	*2π*	Singularity	Singularity	Singularity
-2π	*-2π*	*-2π*	Singularity	Singularity	Singularity
π	*π*	*-π*	Singularity	Singularity	Singularity
π	*-π*	*π*	Singularity	Singularity	Singularity
-π	*π*	*π*	Singularity	Singularity	Singularity
-π	*-π*	*-π*	Singularity	Singularity	Singularity

Table 4-1: MRP Regions of Singularity.

Therefore, it can be stated:

$$|\boldsymbol{\eta}_2| \leq b_3 < 2\pi, \tag{4-142}$$
$$|\boldsymbol{q}(\boldsymbol{\eta}_2)| \leq b_4, \tag{4-143}$$
$$|\boldsymbol{q}| \leq b_4, \tag{4-144}$$
$$|\boldsymbol{q}_e| \leq b_4, \tag{4-145}$$

hence we wish to obtain

$$V_4 \leq b_5 \leq b_4, \tag{4-146}$$

where b_1, b_2 and b_3 are positive constants. To define the initial conditions, we will first state:

$$\begin{aligned} &\frac{1}{2}\left\|\boldsymbol{\Gamma}_{\boldsymbol{\tau}}^{-\mathbf{1}}\right\|\left\|\bar{\boldsymbol{\tau}}_{de,Aero}\right\|^2 + \frac{1}{2\mu_1}\tilde{\delta}_\tau^2 + \frac{1}{2\mu_2}\tilde{\delta}_2^2 \\ &\quad \leq \frac{1}{2}\lambda_m\left(\boldsymbol{\Gamma}_{\boldsymbol{\tau}}^{-\mathbf{1}}\right)(2\varrho_\tau + \xi_\tau)^2 + \frac{1}{2\mu_{\mathbf{1}}}\left(2\Delta_{1,M}^2 + \xi_{1,\delta}\right)^2 + \frac{1}{2\mu_2}\left(2\Delta_{2,M}^2 + \xi_{2,\delta}\right)^2 := b_6, \end{aligned} \tag{4-147}$$

where b_4 is a constant. Therefore, we can now state:

$$\begin{aligned} V_4(t_0) \leq{}& \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\int_0^{\boldsymbol{\eta}_{1e}} \sigma^{\mathrm{T}}(\mathbf{s})\,d\boldsymbol{s} + 2\gamma_1\sqrt{1 + \|\boldsymbol{\nu}_{1e}(t_0)\|^2} + \frac{1}{2}\left(\|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 + \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2\right) \\ &+ \tfrac{\gamma_2}{2}\|\boldsymbol{q}_e(t_0)\|^2 + \tfrac{\gamma_3}{2}\sqrt[2]{1 + \|\boldsymbol{\omega}_e(t_0)\|^4} + b_6 - 2\gamma_1 - \gamma_3 \ \leq b_5, \end{aligned} \tag{4-148}$$

therefore, it can be stated:

$$\|\boldsymbol{\eta}_{1e}(t_0)\| + \|\boldsymbol{\nu}_{1e}(t_0)\| + \|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 + \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2 + \|\boldsymbol{q}_e(t_0)\|^2 + \|\boldsymbol{\omega}_e\|^2 + \|\boldsymbol{\tau}_{de}(t_0)\|^2 + \frac{b_8}{b_7} \leq b_5. \tag{4-149}$$

Hence we are now able to choose bounds on the initial conditions of $\boldsymbol{\eta}_{1e}(t_0)$, $\boldsymbol{\nu}_{1e}(t_0)$, $\boldsymbol{q}_e(t_0)$, $\boldsymbol{\omega}_e(t_0)$, $\tilde{\boldsymbol{\eta}}_1(t_0)$, $\tilde{\boldsymbol{\nu}}_1(t_0)$. Now before we can provide conditions on the intimal conditions and all control gains we must first recognise that:

$$b_1 < \frac{1 + \|\boldsymbol{\nu}_{1e}\|^2}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} < b_2, \qquad \forall \hat{\boldsymbol{\nu}}_1(t) \in \mathbb{R}^3, t \in \mathbb{R} \tag{4-150}$$

where b_5 and b_6 are constant which are dependent on the selection of $\mathbf{K}_1$ and the maximum rate of change of the linear position reference signal, with this in mind it can be stated:

$$\begin{aligned}\dot{V}_4 \leq -b_1 \sum_{i=1}^{3} \frac{\bar{\gamma}_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \lambda_m(\bar{\mathbf{K}}_1)\boldsymbol{\eta}_{1e,i}^2 - c_3}{(1 + \|\boldsymbol{\nu}_{1e}\|^2)(1 + \boldsymbol{\eta}_{1e,i}^2)} - 2\gamma_1 \lambda_m(\bar{\mathbf{K}}_2) \frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}} \sigma(\boldsymbol{\eta}_{1e})}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}(t_0)\|^2}} \\ -2\gamma_1 \lambda_m(\bar{\mathbf{D}}_1) \frac{\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}(t_0)\|^2}} - \lambda_m(\mathbf{K}_{01})\|\tilde{\boldsymbol{\eta}}_1\|^2 - \lambda_m(\bar{\mathbf{D}}_1)\|\tilde{\boldsymbol{\nu}}_1\|^2 \\ -\gamma_2 \lambda_m(\bar{\mathbf{K}}_3)\|\boldsymbol{q}_e\|^2 - \gamma_3 \lambda_m(\mathbf{K}_4) \frac{\|\boldsymbol{\omega}_e\|^4}{(1 + \|\boldsymbol{\omega}_e\|^4)^{0.5}} - \lambda_m(\bar{\mathbf{K}}_{1d})\|\boldsymbol{\tau}_{de}\|^2.\end{aligned} \tag{4-151}$$

By adding and subtracting the right hand side and left hand side of the inequality of (4-375) respectively and dividing both by $(1 + \|\boldsymbol{\nu}_{1e}\|^2)(1 + \boldsymbol{\eta}_{1e,i}^2)/b_5$ to the right side of (4-379) we can now state:

$$\begin{aligned}\dot{V}_4 \leq -b_1 \sum_{i=1}^{3} \frac{\bar{\gamma}_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \lambda_m(\bar{\mathbf{K}}_1)\boldsymbol{\eta}_{1e,i}^2 - \frac{b_6}{3}}{(1 + \|\boldsymbol{\nu}_{1e}\|^2)(1 + \boldsymbol{\eta}_{1e,i}^2)} - 2\gamma_1 \lambda_m(\bar{\mathbf{K}}_2) \frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}} \sigma(\boldsymbol{\eta}_{1e})}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}(t_0)\|^2}} \\ -2\gamma_1 \lambda_m(\bar{\mathbf{D}}_1) \frac{\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}(t_0)\|^2}} - \lambda_m(\mathbf{K}_{01})\|\tilde{\boldsymbol{\eta}}_1\|^2 - \lambda_m(\bar{\bar{\mathbf{D}}}_1)\|\tilde{\boldsymbol{\nu}}_1\|^2 \\ -\gamma_2 \lambda_m(\bar{\mathbf{K}}_3)\|\boldsymbol{q}_e\|^2 - \gamma_3 \lambda_m(\mathbf{K}_4) \frac{\|\boldsymbol{\omega}_e\|^4}{(1 + \|\boldsymbol{\omega}_e\|^4)^{0.5}} - \lambda_m(\bar{\mathbf{K}}_{1d})\|\boldsymbol{\tau}_{de}\|^2.\end{aligned} \tag{4-152}$$

therefore, using (3-147), (4-376) and (4-380) it can be stated that:

$$\dot{V}_4 \leq -b_7 V_4 + b_8. \tag{4-153}$$

where:

$$b_7 = \min\left(\frac{2\gamma_1\,\lambda_m(\bar{\mathbf{D}}_1)}{2\gamma_1},\frac{\lambda_m(\mathbf{K}_{01})}{\frac{1}{2}},\frac{\lambda_m(\bar{\bar{\mathbf{D}}}_1)}{\frac{1}{2}},\frac{\gamma_2\,\lambda_m(\bar{\mathbf{K}}_3)}{\frac{\gamma_2}{2}},\frac{\gamma_3\,\lambda_m(\mathbf{K}_4)}{\frac{\gamma_3}{2}},\frac{\lambda_m(\bar{\mathbf{K}}_{1d})}{\frac{1}{2}}\right.$$
$$\frac{b_1\sum_{i=1}^{3}\frac{\bar{\gamma}_1\frac{\lambda_m(\mathbf{D}_1)}{2}\lambda_m(\bar{\mathbf{K}}_1)}{3(1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2)\left(1+\boldsymbol{\eta}_{1e,i}^2(t_0)\right)}}{\frac{\gamma_1\lambda_m(\mathbf{D}_1)}{2}},\frac{\sum_{i=1}^{3}\frac{b_1\frac{1}{6}\lambda_m(\boldsymbol{\Gamma}^{-1})}{(1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2)\left(1+\boldsymbol{\eta}_{1e,i}^2(t_0)\right)}}{\frac{1}{2}\|\boldsymbol{\Gamma}^{-1}\|},$$
$$\left.\frac{\sum_{i=1}^{3}\frac{b_1\frac{1}{6\mu_1}}{(1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2)\left(1+\boldsymbol{\eta}_{1e,i}^2(t_0)\right)}}{\frac{1}{2\mu_1}},\frac{\sum_{i=1}^{3}\frac{b_1\frac{1}{6\mu_2}}{(1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2)\left(1+\boldsymbol{\eta}_{1e,i}^2(t_0)\right)}}{\frac{1}{2\mu_2}}\right), \tag{4-154}$$

$$b_8 = \max\left(1,\min\left(\sum_{i=1}^{3}\frac{b_1\left(c_3+\frac{b_6}{3}\right)}{(1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2)\left(1+\boldsymbol{\eta}_{1e,i}^2(t_0)\right)}\right)\right). \tag{4-155}$$

Hence with the above equation we are able to choose all initial conditions as well as the controller gains $\mathbf{K}_3, \mathbf{K}_4, \boldsymbol{\Gamma}_\tau^{-1}$ and μ, we are able to state:

$$V_4(t) \le \left(V_4(t_0) - \frac{b_8}{b_7}\right)\mathrm{e}^{-b_7(t-t_0)} + \frac{b_8}{b_7}. \tag{4-156}$$

Which intern implies that:

$$\int_0^{\boldsymbol{\eta}_{1e(t)}} \sigma^{\mathrm{T}}(\mathbf{s})\,d\boldsymbol{s} + \sqrt{1+\|\boldsymbol{\nu}_{1e}(t)\|^2} + \|\tilde{\boldsymbol{\eta}}_1(t)\|^2 + \|\tilde{\boldsymbol{\nu}}_1(t)\|^2 + \|\boldsymbol{q}_e(t)\|^2 + \sqrt[2]{1+\|\boldsymbol{\omega}_e(t)\|^4}$$
$$+\|\boldsymbol{\tau}_{de}(t)\|^2 - 2$$
$$\le \int_0^{\boldsymbol{\eta}_{1e(t_0)}} \sigma^{\mathrm{T}}(\mathbf{s})\,d\boldsymbol{s} + \sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} + \|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 + \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2 + \|\boldsymbol{q}_e(t_0)\|^2$$
$$+\sqrt[2]{1+\|\boldsymbol{\omega}_e(t_0)\|^4} + \|\boldsymbol{\tau}_{de}(t_0)\|^2 + \frac{b_8}{b_7} - 2 \le b_5,, \tag{4-157}$$

We will first calculate the immediate upper bounds of

$$|\boldsymbol{\eta}_{1e}| \le \boldsymbol{\eta}_{1e}^M, \tag{4-158}$$
$$|\boldsymbol{\nu}_{1e}| \le \boldsymbol{\nu}_{1e}^M, \tag{4-159}$$
$$|\boldsymbol{q}_e| \le \boldsymbol{q}_e^M, \tag{4-160}$$
$$|\boldsymbol{\omega}_{1e}| \le \boldsymbol{\omega}_e^M, \tag{4-161}$$
$$|\tilde{\boldsymbol{\eta}}_1| \le \tilde{\boldsymbol{\eta}}_1^M, \tag{4-162}$$
$$|\tilde{\boldsymbol{\nu}}_1| \le \tilde{\boldsymbol{\nu}}_1^M, \tag{4-163}$$

Due to the design processes it can be stated the following conditions

$$\|\boldsymbol{\alpha}_1\| \le \lambda_M(\mathbf{K}_1)(2+\varrho_1), \tag{4-164}$$
$$\|\boldsymbol{\alpha}_2\| \le \boldsymbol{\alpha}_2^{\mathrm{M}}, \tag{4-165}$$
$$\|\boldsymbol{\alpha}_q\| \le \boldsymbol{\alpha}_q^{\mathrm{M}}. \tag{4-166}$$

Furthermore, under condition (4-164) - (4-166), (4-143) and (4-147) it can be now stated that:

$$\|\boldsymbol{\alpha}_q\| \le \boldsymbol{\alpha}_{\boldsymbol{\omega}}^{\mathrm{M}}. \tag{4-167}$$

Therefore, we can choose the values of $\mathbf{K}_3$ by stating:

$$V_4(t_0) \le \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\int_0^{\boldsymbol{\eta}_{1e(t_0)}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s} + 2\gamma_1\sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} + \frac{1}{2}\|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 + \frac{1}{2}\|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2 + \frac{\gamma_2}{2}\|\boldsymbol{q}_e(t_0)\|^2 + \frac{\gamma_3}{2}\sqrt[2]{1+\|\boldsymbol{\omega}_e(t_0)\|^4} + \tfrac{1}{2}\|\boldsymbol{\tau}_{de}(t_0)\|^2 - 2\gamma_1 - \tfrac{\gamma_3}{2} \le b_5, \tag{4-168}$$

$$V_4(t_0) \le \int_0^{\boldsymbol{\eta}_{1e(t_0)}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s} + \sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} + (\|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 + \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2) + \|\boldsymbol{q}_e(t_0)\|^2 + \sqrt[2]{1+\|\boldsymbol{\omega}_e(t_0)\|^4} + \|\boldsymbol{\tau}_{de}(t_0)\|^2 - 2 + \frac{b_8}{b_7} \le b_5, \tag{4-169}$$

$$\sqrt[2]{1+\|\boldsymbol{\omega}_e(t_0)\|^4} \le b_5 - \left(\int_0^{\boldsymbol{\eta}_{1e(t_0)}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s} + \sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} + \|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 + \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2 + \|\boldsymbol{q}_e(t_0)\|^2 + \|\boldsymbol{\tau}_{de}(t_0)\|^2 - 2 + \frac{b_8}{b_7}\right), \tag{4-170}$$

$$\|\boldsymbol{\omega}_e(t_0)\| \le \left(\left(b_5 - \left(\int_0^{\boldsymbol{\eta}_{1e(t_0)}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s} + \sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} + \|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 + \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2 + \|\boldsymbol{q}_e(t_0)\|^2 + \|\boldsymbol{\tau}_{de}(t_0)\|^2 - 2 + \frac{b_8}{b_7}\right)\right)^2 - 1\right)^{0.25}, \tag{4-171}$$

We need to choose the initial conditions for $\boldsymbol{q}_e(t_0)$ such that:

$$b_5 - \left(\int_0^{\boldsymbol{\eta}_{1e(t_0)}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s} + \|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 + \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2 + \sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} + \|\boldsymbol{q}_e(t_0)\|^2 + \|\boldsymbol{\tau}_{de}(t_0)\|^2 - 2 + \frac{b_8}{b_7}\right) > 1 \tag{4-172}$$

$$\|\boldsymbol{q}_e(t_0)\|^2 < b_5 - 1 - \int_0^{\boldsymbol{\eta}_{1e(t_0)}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s} - \|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 - \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2 - \sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} - \|\boldsymbol{\tau}_{de}(t_0)\|^2 + 2 - \frac{b_8}{b_7}, \tag{4-173}$$

$$\|\boldsymbol{q}_e(t_0)\| < \left(b_5 - 1 - \int_0^{\boldsymbol{\eta}_{1e(t_0)}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s} - \|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 - \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2 - \sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} - \left(\|\boldsymbol{\tau}_{de}(t_0)\|^2 - 2 + \frac{b_8}{b_7}\right)\right)^{0.5} \tag{4-174}$$

holds true, now we can choose the values of the control gains matrix $\mathbf{K}_3$, by noting that:

$$\sup_{t\in R^n}\|\mathbf{L}(t)\| = \varrho_9, \tag{4-175}$$

$$\sup_{t\in R^n}\|\boldsymbol{\Omega}(t)\| = \varrho_{10}, \tag{4-176}$$

$$\sup_{t\in R^n}\left\|\frac{2\gamma_1\|\boldsymbol{\Omega}(t)\|\mathbf{L}(t)\mathbf{G}_1^{\mathrm{T}}(t)\boldsymbol{\nu}_{1e}(t)}{\gamma_2\sqrt{1+\|\boldsymbol{\nu}_{1e}(t)\|^2}}\right\| \leq \frac{2\gamma_1\varrho_9\varrho_{10}}{\gamma_2 m}\left(1+0.46\big(\mathbf{K}_{1,11}+\mathbf{K}_{1,22}+\mathbf{K}_{1,33}\big)\right), \tag{4-177}$$

$$\sup_{t\in R^n}\left\|\mathbf{R}_2\left(\boldsymbol{\alpha}_q(t)\right)\right\| \leq 4, \tag{4-178}$$

$$\sup_{t\in R^n}\left\|\mathbf{T}\big(\boldsymbol{\alpha}_2(t)\big)^{-1}\right\| \leq 2, \tag{4-179}$$

$$\sup_{t\in R^n}\|\mathbf{A}_{2,d}(t)\| \leq \sup_{t\in R^n}\left\{\frac{1}{\cos^2\left(\alpha_\phi(t)\right)\|\boldsymbol{\Omega}(t)\|^2}+\cos^2\big(\alpha_\theta(t)\big)\frac{1}{\Omega_3^2(t)}\right\} \coloneqq \|\mathbf{A}_{2,d}\|^{\mathrm{M}}, \tag{4-180}$$

$$\sup_{t\in R^n}\left\|\boldsymbol{\Omega}_d(t)+\frac{\partial\boldsymbol{\Omega}(t)}{\partial\boldsymbol{\eta}_{1e}(t)}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}(t)\boldsymbol{\nu}_{1e}(t)}{\sqrt{1+\|\boldsymbol{\nu}_{1e}(t)\|^2}}\right\| \leq \left\|\boldsymbol{\Omega}_d+\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right\|^{\mathrm{M}}, \tag{4-181}$$

$$\sup_{t\in R^n}\|\mathbf{A}_{1,d}(t)\boldsymbol{\Omega}(t)\| \coloneqq \|\mathbf{A}_{2,d}\boldsymbol{\Omega}\|^{\mathrm{M}}, \tag{4-182}$$

$$\sup_{t\in R^n}\|(\mathbf{A}_{1,d}(t)\boldsymbol{\Omega}(t)+\boldsymbol{e}_3)\dot{\alpha}_\psi(t)\| \leq \left(\|\mathbf{A}_{2,d}\boldsymbol{\Omega}\|^{\mathrm{M}}+1\right)\varrho_6, \tag{4-183}$$

for all $t \geq t_0 \geq 0$.

We can sate that:

$$\alpha_p = \begin{bmatrix}1\\0\\0\end{bmatrix}^{\mathrm{T}}\mathbf{R}_2^{-1}(\boldsymbol{q})\left(\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d+\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\right.\right.$$
$$\left.\left.+(\mathbf{A}_{1,d}\boldsymbol{\Omega}+\boldsymbol{e}_3)\dot{\alpha}_\psi\right)-\mathbf{K}_{3,11}(q_1-\alpha_{q_1})-\begin{bmatrix}1\\0\\0\end{bmatrix}^{\mathrm{T}}\frac{2\gamma_1\|\boldsymbol{\Omega}\|\mathbf{L}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\gamma_2 m\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right), \tag{4-184}$$

$$\alpha_q = \begin{bmatrix}0\\1\\0\end{bmatrix}^{\mathrm{T}}\mathbf{R}_2^{-1}(\boldsymbol{q})\left(\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d+\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\right.\right.$$
$$\left.\left.+(\mathbf{A}_{1,d}\boldsymbol{\Omega}+\boldsymbol{e}_3)\dot{\alpha}_\psi\right)-\mathbf{K}_{3,22}(q_{22}-\alpha_{q_{22}})-\begin{bmatrix}0\\1\\0\end{bmatrix}^{\mathrm{T}}\frac{2\gamma_1\|\boldsymbol{\Omega}\|\mathbf{L}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\gamma_2 m\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right), \tag{4-185}$$

$$\alpha_r = \begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}}\mathbf{R}_2^{-1}(\boldsymbol{q})\left(\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d+\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\right.\right.$$
$$\left.\left.+(\mathbf{A}_{1,d}\boldsymbol{\Omega}+\boldsymbol{e}_3)\dot{\alpha}_\psi\right)-\mathbf{K}_{3,33}(q_3-\alpha_{q_3})-\begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}}\frac{2\gamma_1\|\boldsymbol{\Omega}\|\mathbf{L}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\gamma_2 m\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right), \tag{4-186}$$

and therefore state that:

$$\|\alpha_p\| \leq \alpha_p^{\mathrm{M}}, \qquad \|\alpha_q\| \leq \alpha_q^{\mathrm{M}}, \qquad \|\alpha_r\| \leq \alpha_r^{\mathrm{M}}, \qquad \|\boldsymbol{\alpha}_{\boldsymbol{\omega}}\| \leq \alpha_{\boldsymbol{\omega}}^{\mathrm{M}}. \tag{4-187}$$

And therefore it can be stated that:

$$\|\boldsymbol{\omega}(t_0)-\boldsymbol{\alpha}_{\boldsymbol{\omega}}(t_0)\| \leq \left(\left(b_5-\left(\int_0^{\boldsymbol{\eta}_{1e(t_0)}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s}+\|\widetilde{\boldsymbol{\eta}}_1(t_0)\|^2+\|\widetilde{\boldsymbol{v}}_1(t_0)\|^2+\sqrt{1+\|\boldsymbol{v}_{1e}(t_0)\|^2}+\|\boldsymbol{q}_e(t_0)\|^2++\|\boldsymbol{\tau}_{de}(t_0)\|^2-2+\frac{b_8}{b_7}\right)\right)^2-1\right)^{0.25}, \tag{4-188}$$

$$\|\boldsymbol{\alpha}_{\boldsymbol{\omega}}(t_0)\| \leq \left(\left(b_5-\left(\int_0^{\boldsymbol{\eta}_{1e(t_0)}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s}+\|\widetilde{\boldsymbol{\eta}}_1(t_0)\|^2+\|\widetilde{\boldsymbol{v}}_1(t_0)\|^2+\sqrt{1+\|\boldsymbol{v}_{1e}(t_0)\|^2}+\|\boldsymbol{q}_e(t_0)\|^2+\|\boldsymbol{\tau}_{de}(t_0)\|^2-2+\frac{b_8}{b_7}\right)\right)^2-1\right)^{0.25}-\|\boldsymbol{\omega}(t_0)\|>0, \tag{4-189}$$

$$\begin{aligned} 0<&\left\|\mathbf{R}_2^{-1}(\boldsymbol{q}(t_0))\left(\mathbf{R}_2\left(\boldsymbol{\alpha}_q(t_0)\right)\mathbf{T}(\boldsymbol{\alpha}_2(t_0))^{-1}\left(\mathbf{A}_{2,d}(t_0)\left(\boldsymbol{\Omega}_d(t_0)\right.\right.\right.\right.\\ &\left.\left.+\frac{\partial\boldsymbol{\Omega}(t_0)}{\partial\boldsymbol{\eta}_{1e(t_0)}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}(t_0)\boldsymbol{v}_{1e}(t_0)}{\sqrt{1+\|\boldsymbol{v}_{1e}(t_0)\|^2}}\right)+\left(\mathbf{A}_{1,d}(t_0)\boldsymbol{\Omega}(t_0)+\boldsymbol{e}_3\right)\dot{\alpha}_\psi(t_0)\right)\\ &\left.\left.-\mathbf{K}_3\left(\boldsymbol{q}(t_0)-\boldsymbol{\alpha}_q(t_0)\right)-\frac{2\gamma_1\|\boldsymbol{\Omega}(t_0)\|\mathbf{L}(t_0)\mathbf{G}_1^{\mathrm{T}}(t_0)\boldsymbol{v}_{1e}(t_0)}{\gamma_2 m\sqrt{1+\|\boldsymbol{v}_{1e}(t_0)\|^2}}\right)\right\|\\ \leq&\left(\left(b_5-\left(\int_0^{\boldsymbol{\eta}_{1e(t_0)}}\sigma^{\mathrm{T}}(\mathbf{s})\,d\boldsymbol{s}+\|\widetilde{\boldsymbol{\eta}}_1(t_0)\|^2+\|\widetilde{\boldsymbol{v}}_1(t_0)\|^2+\sqrt{1+\|\boldsymbol{v}_{1e}(t_0)\|^2}+\|\boldsymbol{q}_e(t_0)\|^2\right.\right.\right.\\ &\left.\left.\left.+\|\boldsymbol{\tau}_{de}(t_0)\|^2-2+\frac{b_8}{b_7}\right)\right)^2-1\right)^{0.25}-\|\boldsymbol{\omega}(t_0)\|, \end{aligned} \tag{4-190}$$

$$\begin{aligned} 0<&\left\|\mathbf{K}_3\left(\boldsymbol{q}(t_0)-\boldsymbol{\alpha}_q(t_0)\right)\right\|\\ &\leq\left(\left(b_5-\int_0^{\boldsymbol{\eta}_{1e(t_0)}}\sigma^{\mathrm{T}}(\mathbf{s})\,d\boldsymbol{s}-\|\widetilde{\boldsymbol{\eta}}_1(t_0)\|^2-\|\widetilde{\boldsymbol{v}}_1(t_0)\|^2-\sqrt{1+\|\boldsymbol{v}_{1e}(t_0)\|^2}\right.\right.\\ &\left.\left.-\|\boldsymbol{q}_e(t_0)\|^2-\|\boldsymbol{\tau}_{de}(t_0)\|^2+2-\frac{b_8}{b_7}\right)^2-1\right)^{0.25}\\ &-\left\|\mathbf{R}_2^{-1}(\boldsymbol{q}(t_0))\left(\mathbf{R}_2\left(\boldsymbol{\alpha}_q(t_0)\right)\mathbf{T}(\boldsymbol{\alpha}_2(t_0))^{-1}\left(\mathbf{A}_{2,d}(t_0)\left(\boldsymbol{\Omega}_d(t_0)\right.\right.\right.\right.\\ &\left.\left.+\frac{\partial\boldsymbol{\Omega}(t_0)}{\partial\boldsymbol{\eta}_{1e(t_0)}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}(t_0)\boldsymbol{v}_{1e}(t_0)}{\sqrt{1+\|\boldsymbol{v}_{1e}(t_0)\|^2}}\right)+\left(\mathbf{A}_{1,d}(t_0)\boldsymbol{\Omega}(t_0)+\boldsymbol{e}_3\right)\dot{\alpha}_\psi(t_0)\right)\\ &\left.\left.-\frac{2\gamma_1\|\boldsymbol{\Omega}(t_0)\|\mathbf{L}(t_0)\mathbf{G}_1^{\mathrm{T}}(t_0)\boldsymbol{v}_{1e}(t_0)}{\gamma_2 m\sqrt{1+\|\boldsymbol{v}_{1e}(t_0)\|^2}}\right)\right\|-\|\boldsymbol{\omega}(t_0)\|, \end{aligned} \tag{4-191}$$

It is now possible to consider the case for actuator saturation and therefore provide bounds on the gains $\mathbf{K}_4, \boldsymbol{\Gamma}_\tau^{-1}$ and μ, we are able to state first we will state that:

$$\sup_{t\in R^n} \|\boldsymbol{\tau}(t)\| = \boldsymbol{\tau}^M, \qquad \forall t \geq t_0 \geq 0. \tag{4-192}$$

Now we can make the conditions on the gains $\mathbf{K}_4, \boldsymbol{\Gamma}, \mu_1$ and μ_2, we are able to state:

$$\begin{aligned}\|\boldsymbol{\tau}\| \leq \Bigg\| (\mathbf{I}_A + \mathbf{I}_H)\Bigg[& -(\mathbf{I}_A + \mathbf{I}_H)^{-1}(\mathbf{D}_2\boldsymbol{\alpha}_\omega - \mathbf{C}_B(\boldsymbol{\omega})\boldsymbol{\omega} - \mathbf{C}_{A2}(\boldsymbol{\omega})\boldsymbol{\omega}) + \boldsymbol{\alpha}_{\omega,d} - \mathbf{K}_4\boldsymbol{\omega}_e \\ & - \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{2\gamma_1 \boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{G}_1}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} + \gamma_2 \boldsymbol{q}_e^T \mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial \boldsymbol{\Omega}}{\partial \hat{\boldsymbol{v}}_1}\right)^{\mathrm{T}} \\ & - \left(\frac{\gamma_2}{\gamma_3}\left(\left(\frac{1}{4\varepsilon_5}\right)^2 \frac{\varepsilon_6\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{2.25}} + \frac{\boldsymbol{q}_{qe}^T\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e}{(1+\|\boldsymbol{\omega}_e\|^4)^{1/4}}\right)\right. \\ & \left. + \gamma_3 \frac{\varepsilon_6}{4}\left\|(\mathbf{I}_A + \mathbf{I}_H)^{-1}\right\|^2 \frac{\|\boldsymbol{\omega}_e\|^2}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}\right)\boldsymbol{\omega}_e\Bigg]\hat{\boldsymbol{\tau}}_{Aero}\Bigg\| \leq \boldsymbol{\tau}^M := \left\|\begin{bmatrix} U_2^M \\ U_3^M \\ U_4^M\end{bmatrix}\right\|\end{aligned} \tag{4-193}$$

Recall from (4-309) that:

$$\begin{aligned}\boldsymbol{\alpha}_{\omega,d} = & \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \mathbf{q}}\dot{\mathbf{q}} - \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \hat{\boldsymbol{v}}_1}\left(\tfrac{f}{m}\mathbf{R}(\boldsymbol{\eta}_2)\boldsymbol{e}_3 - \mathbf{D}_1\hat{\boldsymbol{v}}_1 - g\boldsymbol{e}_3 + \boldsymbol{h}_2 + \boldsymbol{h}_3 + \boldsymbol{h}_4 + \boldsymbol{h}_5\right) + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \alpha_\psi}\dot{\alpha}_\psi \\ & + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \dot{\alpha}_\psi}\ddot{\alpha}_\psi + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\eta}_{1e}}(\hat{\boldsymbol{v}}_1 - \dot{\boldsymbol{\eta}}_{1d}.) + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\eta}_{1d}}\dot{\boldsymbol{\eta}}_{1d} + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d} + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \dddot{\boldsymbol{\eta}}_{1d}}\ddddot{\boldsymbol{\eta}}_{1d}.\end{aligned} \tag{4-194}$$

from Appendix E in addition with the conditions outlined in Assumption 4.5 to Assumption 4.9 and condition (4-372) it can be seen that:

$$\begin{aligned}&\sup_{t\in R^n}\left\|\frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \boldsymbol{q}(t)}\right\| \leq \varrho_{11}, \quad \sup_{t\in R^n}\left\|\frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \hat{\boldsymbol{v}}_1(t)}\dot{\hat{\boldsymbol{v}}}_1(t)\right\| \leq \varrho_{12}, \quad \sup_{t\in R^n}\left\|\frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \alpha_\psi(t)}\dot{\alpha}_\psi(t)\right\| \leq \varrho_{13}, \\ &\sup_{t\in R^n}\left\|\frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \dot{\alpha}_\psi(t)}\ddot{\alpha}_\psi(t)\right\| \leq \varrho_{14}, \quad \sup_{t\in R^n}\left\|\frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \boldsymbol{\eta}_{1e}(t)}(\hat{\boldsymbol{v}}_1(t) - \dot{\boldsymbol{\eta}}_{1d}(t))\right\| \leq \varrho_{15}, \\ &\sup_{t\in R^n}\left\|\frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \boldsymbol{\eta}_{1d}(t)}\dot{\boldsymbol{\eta}}_{1d}(t)\right\| \leq \varrho_{16}, \quad \sup_{t\in R^n}\left\|\frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \dot{\boldsymbol{\eta}}_{1d}(t)}\ddot{\boldsymbol{\eta}}_{1d}(t)\right\| \leq \varrho_{17}, \\ &\sup_{t\in R^n}\left\|\frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \ddot{\boldsymbol{\eta}}_{1d}(t)}\dddot{\boldsymbol{\eta}}_{1d}(t)\right\| \leq \varrho_{18}, \quad \sup_{t\in R^n}\left\|\frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \dddot{\boldsymbol{\eta}}_{1d}(t)}\ddddot{\boldsymbol{\eta}}_{1d}(t)\right\| \leq \varrho_{19},.\end{aligned} \tag{4-195}$$

We can therefore state that:

$$\sup_{t\in R^n}\|\boldsymbol{\alpha}_{\omega,d}(t)\| \leq \varrho_{20}, \tag{4-196}$$

$$\sup_{t\in R^n}\|\boldsymbol{\alpha}_{\omega,d,p}(t)\| \leq \varrho_{21}, \quad \sup_{t\in R^n}\|\boldsymbol{\alpha}_{\omega,d,q}(t)\| \leq \varrho_{22}, \quad \sup_{t\in R^n}\|\boldsymbol{\alpha}_{\omega,d,r}(t)\| \leq \varrho_{23}, \tag{4-197}$$

where:

$$\boldsymbol{\alpha}_{\omega,d} = [\boldsymbol{\alpha}_{\omega,d,p} \quad \boldsymbol{\alpha}_{\omega,d,q} \quad \boldsymbol{\alpha}_{\omega,d,r}]. \tag{4-198}$$

This gives:

$$\left\|\frac{I_z+I_{z,A}-I_y-I_{y,A}}{I_x+I_{x,A}}\right\| q^{M} r^{M}+\varrho_{21}+\frac{\gamma_2}{\gamma_3}\|\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{q})\|p_e^M$$
$$+\left\|\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1e}}\right\|\left(2\gamma_1\left(1+0.46(\mathbf{K}_{1,11}+\mathbf{K}_{1,22}+\mathbf{K}_{1,33})\right)\right.$$
$$\left.+\gamma_2\left\|\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\right\|\|\mathbf{A}_{2,d}^{\mathrm{T}}\|\|\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\|\mathbf{R}_2^T(\boldsymbol{\alpha}_q)\boldsymbol{q}_e\right)$$
$$+\left(\left(\mathbf{K}_{4,11}-\frac{\gamma_2}{\gamma_3}\left(\frac{1}{4\varepsilon_5}\right)^2\frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1+\|p_e^M\|^4)^{2.25}}-\gamma_3\frac{\varepsilon_6}{4}\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{1}}\right\|^2\right)p_e^M\le\frac{U_2^M}{I_x+I_{x,A}},\right. \tag{4-199}$$

$$\left\|\frac{I_x+I_{x,A}-I_z-I_{z,A}}{I_y+I_{y,A}}\right\| r^{M} p^{M}+\varrho_{22}+\frac{1}{\gamma_2}\|\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{q})\|q_e^M$$
$$+\left\|\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1e}}\right\|\left(2\gamma_1\left(1+0.46(\mathbf{K}_{1,11}+\mathbf{K}_{1,22}+\mathbf{K}_{1,33})\right)\right.$$
$$\left.+\gamma_2\left\|\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\right\|\|\mathbf{A}_{2,d}^{\mathrm{T}}\|\|\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\|\mathbf{R}_2^T(\boldsymbol{\alpha}_q)\boldsymbol{q}_e\right)$$
$$+\left(\left(\mathbf{K}_{4,11}-\frac{\gamma_2}{\gamma_3}\left(\frac{1}{4\varepsilon_5}\right)^2\frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1+\|p_e^M\|^4)^{2.25}}-\gamma_3\frac{\varepsilon_6}{4}\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{1}}\right\|^2\right)q_e^M\le\frac{U_3^M}{I_y+I_{y,A}},\right. \tag{4-200}$$

$$\left\|\frac{I_y+I_{y,A}-I_x-I_{x,A}}{I_z+I_{z,A}}\right\| p^{M} q^{M}+\varrho_{23}+\frac{1}{\gamma_2}\|\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{q})\|r_e^M$$
$$+\left\|\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1e}}\right\|\left(2\gamma_1\left(1+0.46(\mathbf{K}_{1,11}+\mathbf{K}_{1,22}+\mathbf{K}_{1,33})\right)\right.$$
$$\left.+\gamma_2\left\|\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\right\|\|\mathbf{A}_{2,d}^{\mathrm{T}}\|\|\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\|\mathbf{R}_2^T(\boldsymbol{\alpha}_q)\boldsymbol{q}_e\right)$$
$$+\left(\left(\mathbf{K}_{4,11}-\frac{\gamma_2}{\gamma_3}\left(\frac{1}{4\varepsilon_5}\right)^2\frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1+\|p_e^M\|^4)^{2.25}}-\gamma_3\frac{\varepsilon_6}{4}\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{1}}\right\|^2\right)r_e^M\le\frac{U_4^M}{I_z+I_{z,A}}.\right. \tag{4-201}$$

Rearranging (4-427), (4-428) and (4-429) gives:

$$0<\mathbf{K}_{2,11}\le\frac{\frac{U_2^M}{I_x+I_{x,A}}-\left(\left\|\frac{I_z+I_{z,A}-I_y-I_{y,A}}{I_x+I_{x,A}}\right\|q^M r^M+\frac{1}{I_x+I_{x,A}}\hat{\overline{\boldsymbol{\tau}}}_p^M\right)}{p_e^M}$$
$$-\frac{1}{p_e^M}\left(\dot{\alpha}_p^M+\frac{1}{\gamma_2}\|\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{q})\|p_e^M-\frac{\varepsilon_2}{\gamma_2}\left(\frac{1}{4\varepsilon_5}\right)^2\frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1+\|p_e^M\|^4)^{2.25}}p_e^M\right.$$
$$\left.-\gamma_3\frac{\varepsilon_6}{4}\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{1}}\right\|^2 p_e^M\right), \tag{4-202}$$

$$0<\mathbf{K}_{2,22}\le\frac{\frac{U_3^M}{I_y+I_{y,A}}-\left(\left\|\frac{I_x+I_{x,A}-I_z-I_{z,A}}{I_y+I_{y,A}}\right\|r^M p^M+\frac{1}{I_y+I_{y,A}}\hat{\overline{\boldsymbol{\tau}}}_q^M\right)}{q_e^M}$$
$$-\frac{1}{q_e^M}\left(\dot{\alpha}_q^M+\frac{1}{\gamma_2}\|\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{q})\|q_e^M-\frac{\varepsilon_2}{\gamma_2}\left(\frac{1}{4\varepsilon_5}\right)^2\frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1+\|q_e^M\|^4)^{2.25}}q_e^M\right.$$
$$\left.-\gamma_3\frac{\varepsilon_6}{4}\left\|(\mathbf{I}_A+\mathbf{I}_H)^{-\mathbf{1}}\right\|^2 q_e^M\right), \tag{4-203}$$

$$0 < \mathbf{K}_{2,33} \leq \frac{\frac{U_4^M}{I_z + I_{z,A}} - \left(\left\| \frac{I_y + I_{y,A} - I_x - I_{x,A}}{I_z + I_{z,A}} \right\| p^M q^M + \frac{1}{I_z + I_{z,A}} \hat{\hat{\boldsymbol{\tau}}}_r^M \right)}{r_e^M} - \frac{1}{r_e^M} \left(\dot{\alpha}_r^{\mathrm{M}} + \frac{1}{\gamma_2} \| \boldsymbol{q}_e^T \mathbf{R}_2(\boldsymbol{q}) \| q_e^M - \frac{\varepsilon_2}{\gamma_2} \left(\frac{1}{4\varepsilon_5} \right)^2 \frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1 + \|r_e^M\|^4)^{2.25}} r_e^M - \gamma_3 \frac{\varepsilon_6}{4} \left\| (\mathbf{I}_A + \mathbf{I}_H)^{-\mathbf{1}} \right\|^2 r_e^M \right), \tag{4-204}$$

Therefore, the control design is complete and control objective 4.2 is met. Furthermore, it has been proven that the control law is exponentially stable in probability.

4.2 New Stochastic Output Feedback Control Design

In the previous section the control of the quadrotor has been deterministic, the environmental disturbance inducing forces and moments acting on the aircraft during flight were assumed to be deterministic. This fundamental assumption does not hold in practice as wind always contain both laminar and turbulent flow components which we will consider the latter to be stochastic. The control performance then deteriorates and the closed loop system can become unstable.

4.2.1 Model Assumptions

Listed below are the assumptions used when modelling the quadrotor under stochastic aerodynamic loads.

Assumption 4.5

Assume that the unknown parametric aeronautic force disturbance vector $\boldsymbol{f}_{Aero}$ *and its derivative are bounded, i.e., there exist nonnegative constants such that:*

$$\sup_{t\in R^n}\left\|\bar{\boldsymbol{f}}_{Aero}(t)\right\| \leq \varrho_f, \qquad \sup_{t\in R^n}\left\|\dot{\bar{\boldsymbol{f}}}_{Aero\boldsymbol{\eta}}(t)\right\| = 0, \forall t \geq t_0 \geq 0. \tag{4-205}$$

Furthermore, there exist constants ϱ_{f1}, ϱ_{f2} *and* ϱ_{f3} *such that*:

$$\sup_{t\in R^n}\left|\bar{\boldsymbol{f}}_{Aero,x}(t)\right| \leq \varrho_{f1}, \sup_{t\in R^n}\left|\bar{\boldsymbol{f}}_{Aero,y}(t)\right| \leq \varrho_{f2} \sup_{t\in R^n}\left|\bar{\boldsymbol{f}}_{Aero,z}(t)\right| \leq \varrho_{f3}. \tag{4-206}$$

The unknown aerodynamic covariance matrix $\Delta_{\boldsymbol{1}}(t)$ is bounded, i.e., there exist $\Delta_{1,M}$ such that:

$$\sup_{t\in R^n}\left\|\Delta_{\boldsymbol{1}}(t)\Delta_{\boldsymbol{1}}^{\mathrm{T}}(t)\right\|_{\infty} \leq \Delta_{\boldsymbol{1},M}. \tag{4-207}$$

Assumption 4.6

Assume that unknown parameter *aerodynamic torque disturbance vector* $\boldsymbol{\tau}_{Aero}$ *and its derivative are bounded, i.e., there exist nonnegative constants such that*

$$\sup_{t\in R^n}\left\|\boldsymbol{\tau}_{Aero}(t)\right\| \leq \varrho_{\tau}, \qquad \sup_{t\in R^n}\left\|\dot{\boldsymbol{\tau}}_{Aero,\boldsymbol{\eta}}(t)\right\| = 0, \forall t \geq t_0 \geq 0. \tag{4-208}$$

Furthermore, three exists constants $\varrho_{\tau 1}$, $\varrho_{\tau 2}$ *and* $\varrho_{\tau 3}$ *such that:*

$$\sup_{t\in R^n}\left|\bar{\boldsymbol{\tau}}_{Aero,\boldsymbol{p}}(t)\right| \leq \varrho_{\tau 1}, \sup_{t\in R^n}\left|\bar{\boldsymbol{\tau}}_{Aero,\boldsymbol{q}}(t)\right| \leq \varrho_{\tau 2} \sup_{t\in R^n}\left|\bar{\boldsymbol{\tau}}_{Aero,\boldsymbol{r}}(t)\right| \leq \varrho_{\tau 3}. \tag{4-209}$$

The unknown covariance matrix $\Delta_2(t)$ and $\Delta_3(t)$ are bounded, i.e., there exist $\Delta_{2,M}$ and $\Delta_{3,M}$ such that:

$$\sup_{t\in R^n}\left\|\Delta_{\boldsymbol{2}}(\boldsymbol{t})\Delta_{\boldsymbol{2}}^{\mathrm{T}}(\boldsymbol{t})\right\|_{\infty} \leq \Delta_{\boldsymbol{2},M}, \tag{4-210}$$

$$\delta_1 = \left\|\Delta_{\boldsymbol{1}}(t)\Delta_{\boldsymbol{1}}^{\mathrm{T}}(t)\right\|_{\infty}, \tag{4-211}$$

$$\sup_{t \in R^n} \left\| \Delta_3(\boldsymbol{t}) \Delta_3^{\mathrm{T}}(\boldsymbol{t}) \right\|_\infty \leq \Delta_{3,M}, \tag{4-212}$$

$$\delta_2 = \left\| \Delta_3(t) \Delta_3^{\mathrm{T}}(t) \right\|_\infty, \tag{4-213}$$

Assumption 4.7

The added mass to the aircraft induced by the relative airflow is a point mass that is:

$$\mathbf{M}_A = \mathrm{m}_A \mathbf{I}_{3\times 3}. \tag{4-214}$$

Assumption 4.8

Assume that the reference position trajectory $\boldsymbol{\eta}_{1d}(t) = [x_d(t) \quad y_d(t) \quad z_d(t)]^{\mathrm{T}}$ is sufficiently smooth, i.e., its first four derivatives exist and are bounded, that is there exist non-negative constants ϱ_1, ϱ_2, ϱ_3 and ϱ_4 such that:

$$\sup_{t \in R^n} \|\dot{\boldsymbol{\eta}}_{1d}(t)\| \leq \varrho_1, \sup_{t \in R^n} \|\ddot{\boldsymbol{\eta}}_{1d}(t)\| \leq \varrho_2, \sup_{t \in R^n} \|\dddot{\boldsymbol{\eta}}_{1d}(t)\| \leq \varrho_3, \sup_{t \in R^n} \left\|\ddddot{\boldsymbol{\eta}}_{1d}(t)\right\| \leq \varrho_4 \tag{4-215}$$

Furthermore, the second derivative of $\boldsymbol{z}_d(t)$ is assumed strictly less than g and the aerodynamic disturbance force acting along the vertical axis of the E-frame:

$$\sup_{t \in R^n} |\ddot{z}_d(t)| \leq g - \varrho_5 - \varrho_{f3}, \tag{4-216}$$

where ϱ_5 is a strictly positive constant as well ϱ_{f3} which is defined in (4-206), *the reference yaw angle $\psi_d(t)$ is also assumed as being sufficiently smooth. That is the first two derivatives exist and are bounded, i.e. there are non-negative constants ϱ_6 and ϱ_7 exist such that:*

$$\sup_{t \in R^n} \left|\dot{\psi}_d(t)\right| \leq \varrho_6, \sup_{t \in R^n} \left|\ddot{\psi}_d(t)\right| \leq \varrho_7, \tag{4-217}$$

Assumption 4.9

Assume that reference yawl trajectory is bounded by a positive constant ϱ_8, between $\pm 2\pi$ such that

$$\sup_{t \in R^n} |\psi_d(t)| \leq \varrho_8, \qquad \varrho_8 < 2\pi. \tag{4-218}$$

Under the above assumptions, it is possible to design the control inputs U_1, U_2, U_3 and U_4 in such a manner that the linear position $\boldsymbol{\eta}_1(t)$ and yaw angle $\psi(t)$ of the aircraft stochastically practically asymptotically track their reference trajectories $\boldsymbol{\eta}_{1d}(t)$ and $\psi_d(t)$, i.e.,

$$\lim_{t\to\infty} \left(\boldsymbol{\eta}_1(t) - \boldsymbol{\eta}_{1d}(t)\right) = 0, \qquad \lim_{t\to\infty} \left(\psi_1(t) - \psi_{1d}(t)\right) = 0. \tag{4-219}$$

While keeping all other states of the aircraft dynamics bounded for all initial conditions $\boldsymbol{\eta}_1(t_0) \in R^n$, $\boldsymbol{\nu}_1(t_0) \in R^n$, $\boldsymbol{\eta}_2(t_0) \in R^n$ and $\boldsymbol{\omega}(t_0) \in R^n$. The condition (4-216) implies that the aircraft is not allowed to descend faster than it freely falls under the gravitational force. In a practical sense, this implies that the aircrafts propellers cannot spin in the opposing direction to that which provides lift nor can the quadrotor fly in an inverted manner. This condition is essential when designing the control scheme for the under actuated quadrotor. The system states are defined as shown below

$$\boldsymbol{\eta}_1 = \begin{bmatrix} x \\ y \\ z \end{bmatrix},\ \boldsymbol{\nu}_1 = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix},\ \boldsymbol{\eta}_2 = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix},\ \boldsymbol{q} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix},\ \boldsymbol{\omega} = \begin{bmatrix} p \\ q \\ r \end{bmatrix},\ \boldsymbol{f} = [U_1],\ \boldsymbol{\tau} = \begin{bmatrix} U_2 \\ U_3 \\ U_4 \end{bmatrix}, \quad \text{(4-220)}$$

where $\boldsymbol{\eta}_1$ denotes the linear position of the system, while $\boldsymbol{\nu}_1$ denotes the linear velocity of the system $\boldsymbol{\eta}_2$ denotes the attitude of the quadrotor through the use of Euler angles, while $\boldsymbol{q}$ denotes the attitude of the aircraft through the use of MRP and $\boldsymbol{\omega}$ denotes the angular velocity of the quadrotor with respect to the predefined body frame of reference. We will design a control law for the system described in equation (2-119) and (2-120) under the conditions prescribed in Assumption 4.7. For convenience we restate the equations here:

$$\begin{aligned} &\dot{\boldsymbol{\eta}}_1 = \boldsymbol{\nu}_1, && \text{(4-221)} \\ &d\boldsymbol{\nu}_1 = \left(-\mathbf{D}_1\boldsymbol{\nu}_1 - g\boldsymbol{e}_3 + \tfrac{1}{m}f\mathbf{R}_1(\boldsymbol{q}) + \tfrac{1}{m}\bar{\boldsymbol{f}}_{Aero}\right)dt + \tfrac{1}{m}\Delta_{\boldsymbol{\nu}}(t)d\boldsymbol{w}_1,, \\ &\dot{\boldsymbol{q}} = \mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega} && \text{(4-222)} \\ &d\boldsymbol{\omega} = (\mathbf{I}_A + \mathbf{I}_H)^{-1}(-\mathbf{D}_2\boldsymbol{\omega} - \mathbf{C}_B(\boldsymbol{\omega})\boldsymbol{\omega} - \mathbf{C}_{A2}(\boldsymbol{\omega})\boldsymbol{\omega} + \boldsymbol{\tau} + \boldsymbol{\tau}_{Aero})dt \\ &\qquad + (\mathbf{I}_A + \mathbf{I}_H)^{-1}\boldsymbol{\theta}\Delta_2(t)d\boldsymbol{w}_2 + (\mathbf{I}_A + \mathbf{I}_H)^{-1}\Delta_3(t)d\boldsymbol{w}_3. \end{aligned}$$

where $\boldsymbol{\Delta}_1(t), \boldsymbol{\Delta}_2(t)$ and $\boldsymbol{\Delta}_3(t)$ are (time-varying) covariance matrices, $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ are three-dimensional vectors of standard Wiener process and let:

$$\boldsymbol{\theta} = (\mathbf{C}_{A2}(\boldsymbol{\omega}) + \boldsymbol{S}(\boldsymbol{\omega})\mathbf{I}_A + \mathbf{D}_2). \quad \text{(4-223)}$$

<u>Control objective 4.2</u>

1. *Stochastic robust objective: Under Assumption 4.5 to Assumption 4.9 design the control inputs U_1, U_2, U_3 and U_4 and update laws $\bar{\boldsymbol{f}}_{Aero}$, δ_f, $\bar{\boldsymbol{\tau}}_{Aero}$ and $\delta_{\boldsymbol{\tau}}$ to force the position vector $\boldsymbol{\eta}_1$ and yaw angle ψ of the aircraft to ultimately track $\boldsymbol{\eta}_{1d}$ and ψ_d in probability while keeping all other states of the aircraft bounded.*
2. Keep the aircraft's attitude between the range of $\pm 2\pi$.

4.2.2 Linear Position Tracking.

In this stage, we are concerned with the two equations of (4-221) because we are considering an output feedback system, were the liner position is known and the liner velocity is unknown, we will define the error of the state and state estimate as:

$$\tilde{\boldsymbol{\eta}}_1 = \boldsymbol{\eta}_1 - \hat{\boldsymbol{\eta}}_1, \tag{4-224}$$
$$\tilde{\boldsymbol{\nu}}_1 = \boldsymbol{\nu}_1 - \hat{\boldsymbol{\nu}}_1. \tag{4-225}$$

We will define the derivative of the linear position estimate as follows:

$$\dot{\hat{\boldsymbol{\eta}}}_1 = \hat{\boldsymbol{\nu}}_1 + \mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1 + \boldsymbol{h}_1. \tag{4-226}$$

We will define the derivative of the linear velocity estimate latter. In the meantime, we will define the tracking error as:

$$\begin{aligned} \boldsymbol{\eta}_{1e} &= \boldsymbol{\eta}_1 - \boldsymbol{\eta}_{1d}, \\ \boldsymbol{\nu}_{1e} &= \hat{\boldsymbol{\nu}}_1 - \boldsymbol{\alpha}_1. \end{aligned} \tag{4-227}$$

Where $\boldsymbol{\eta}_{1e}$ denotes the tracking error between the desired position $\boldsymbol{\eta}_{1d}$ and current position $\boldsymbol{\eta}_1$ in the second equation $\boldsymbol{\nu}_{1e}$ denotes the tracking error between the estimate of the systems linear velocity and virtual control signal $\boldsymbol{\alpha}_1$ of the velocity. The first step of the design process is to design $\boldsymbol{\alpha}_1$ to stabilise the tracking error of $\boldsymbol{\eta}_{1e}(t)$ and $\boldsymbol{\nu}_{1e}(t)$ at the origin. By substituting (4-227) into the first equation of (4-221) we obtain:

$$\dot{\boldsymbol{\eta}}_{1e} = \tilde{\boldsymbol{\nu}}_1 + \boldsymbol{\nu}_{1e} + \boldsymbol{\alpha}_1 - \dot{\boldsymbol{\eta}}_{1d}. \tag{4-228}$$

Therefore, we will choose the derivative of the estimate of the liner velocity:

$$\dot{\hat{\boldsymbol{\nu}}}_1 = -\mathbf{D}_1\hat{\boldsymbol{\nu}}_1 - g\boldsymbol{e}_3 + \frac{f}{m}\mathbf{R}(\boldsymbol{\eta}_2)\boldsymbol{e}_3 + \mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1 + \boldsymbol{h}_2 + \boldsymbol{h}_3 + \boldsymbol{h}_4 + \boldsymbol{h}_5, \tag{4-229}$$

and hence define the derivative of the error of the estimate of the linear velocity as follows:

$$d\tilde{\boldsymbol{\nu}}_1 = \left(-\mathbf{D}_1\tilde{\boldsymbol{\nu}}_1 + \mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1 + \frac{1}{m}\bar{\boldsymbol{f}}_{Aero} + \boldsymbol{h}_2 + \boldsymbol{h}_3 + \boldsymbol{h}_4 + \boldsymbol{h}_5\right)dt + \frac{1}{m}\Delta_1(t)\, d\boldsymbol{w}_1. \tag{4-230}$$

Considering the positive definite Lyapunov function candidate:

$$V_1 = \int_0^{\boldsymbol{\eta}_{1e}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s}. \tag{4-231}$$

The infinite generator of both sides of (4-231) along the solution of (4-228) results in the following infinite generator:

$$\mathcal{L}V_1 = \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})(\boldsymbol{\alpha}_1 - \dot{\boldsymbol{\eta}}_{1d}) + \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\boldsymbol{\nu}_{1e} + \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\tilde{\boldsymbol{\nu}}_1. \tag{4-232}$$

To stabilise $\boldsymbol{\eta}_{1e}$ we will implement the one-step ahead backstepping method to design the virtual control signal $\boldsymbol{\alpha}_1$ as follows:

$$\boldsymbol{\alpha}_1 = -\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \dot{\boldsymbol{\eta}}_{1d}. \tag{4-233}$$

Defining the matrix $\mathbf{K}_1 \in \mathbb{R}^{3\times3}$, $\mathbf{K}_1 = diag(k_{1,1}, k_{1,2}, k_{1,3})$ where $k_{1,1}$, $k_{1,2}$ and $k_{1,3}$ are positive constants and $\sigma(\boldsymbol{s})$ and $\Delta_1(\boldsymbol{s})$ are defined as follows:

$$\sigma(s) = \frac{s}{\sqrt{1+\|s\|^2}}, \quad \Delta_1(\boldsymbol{s}) = 1 + \frac{1}{2}\|\boldsymbol{s}\|^2. \tag{4-234}$$

Substituting (4-233) into (4-232) yields:

$$\mathcal{L}V_1 \leq -\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} + \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\boldsymbol{v}_{1e} + \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\tilde{\boldsymbol{v}}_1. \tag{4-235}$$

Substituting (4-233) into (4-228) yields:

$$\dot{\boldsymbol{\eta}}_{1e} = -\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} + \tilde{\boldsymbol{v}}_1 + \boldsymbol{v}_{1e}. \tag{4-236}$$

The derivative of (4-233) along the solutions of (4-236) and (4-229) yields:

$$\dot{\boldsymbol{\alpha}}_1 = \mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\dot{\hat{\boldsymbol{v}}}_1 - \mathbf{K}_1 \frac{\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\left(-\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} + \tilde{\boldsymbol{v}}_1 + \boldsymbol{v}_{1e}\right) + \ddot{\boldsymbol{\eta}}_{1d}. \tag{4-237}$$

To prepare for the next step, we calculate $\dot{\boldsymbol{v}}_{1e}$, by taking the derivative of the second equation of (4-227) along the solution of (4-229) and (4-237) as follows:

$$\dot{\boldsymbol{v}}_{1e} = \mathbf{G}_1\left(-\mathbf{D}_1\hat{\boldsymbol{v}}_1 - g\boldsymbol{e}_3 + \frac{f}{m}\mathbf{R}_1(\boldsymbol{q})\boldsymbol{e}_3 + \mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1 + \sum_{i=2}^{5}\boldsymbol{h}_i\right) + \boldsymbol{F}_1 + \boldsymbol{E}_2 - \ddot{\boldsymbol{\eta}}_{1d}, \tag{4-238}$$

Where the terms $\mathbf{G}_1, \boldsymbol{F}_1$ and $\boldsymbol{E}_2$ are defined as follows:

$$\mathbf{G}_1 = \left(\mathbf{I}_{3\times3} - \mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\right). \tag{4-239}$$

$$\boldsymbol{F}_1 = \left(\frac{\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\sigma'(\boldsymbol{\eta}_{1e})\left(-\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} + \boldsymbol{v}_{1e}\right), \quad \sigma'(\boldsymbol{\eta}_{1e}) = \frac{\partial\sigma(\boldsymbol{\eta}_{1e})}{\partial\boldsymbol{\eta}_{1e}}. \tag{4-240}$$

$$\boldsymbol{E}_2 = \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\tilde{\boldsymbol{v}}_1. \tag{4-241}$$

We need to consider the conditions for which $\mathbf{G}_1$ is invertible, clearly $\det(\mathbf{G}_1) \geq 1 - k_{1,1} - k_{1,2} - k_{1,3}$, It should be noted that every element in $\mathbf{G}_1$ is bounded between -1 and 1. This implies that the determinant of $\mathbf{G}_1$ is bounded between -2 and 4. Furthermore the constants $k_{1,1}$, $k_{1,2}$ and $k_{1,3}$ should be chosen such that:

$$1 - 0.28 * (k_{1,1} - k_{1,2} - k_{1,3}) > 0. \tag{4-242}$$

The above is based on the determinant equation for a 3x3 matrix. The matrix $\mathbf{G}_1$ obeys:

$$\|\mathbf{G}_1\| \leq 1 + \|\mathbf{K}_1\|. \tag{4-243}$$

The vector $\boldsymbol{F}_1$ is bounded by:

$$\|\boldsymbol{F}_1\| \leq \lambda_M(\mathbf{K}_1)(2+\varrho_1). \tag{4-244}$$

Where $\lambda_M(\mathbf{K}_1)$ denotes the maximum Eigen value of $\mathbf{K}_1$.

Step 2: We define the coordinate transformation representing the error between the current system angular position and the virtual control signal $\boldsymbol{\alpha}_q$ as:

$$\boldsymbol{q}_e = \boldsymbol{q} - \boldsymbol{\alpha}_q. \tag{4-245}$$

Where $\boldsymbol{q}_e$ is the tracking error of the attitude and $\boldsymbol{\alpha}_q = [\alpha_{q,1} \quad \alpha_{q,2} \quad \alpha_{q,3}]^{\mathrm{T}}$, the MRP virtual control signal for the attitude. Furthermore, substituting (4-245) into (2-90) yields:

$$\mathbf{R_1}(\boldsymbol{q}) = \mathbf{H}(\boldsymbol{q}_e, \boldsymbol{\alpha}_q) + \mathbf{R_1}(\boldsymbol{\alpha}_q), \tag{4-246}$$

where

$$\mathbf{H}(\boldsymbol{q}_e, \boldsymbol{\alpha}_q) = \mathbf{R}_2(\boldsymbol{q}_e + \boldsymbol{\alpha}_q) - \mathbf{R_1}(\boldsymbol{\alpha}_q), \tag{4-247}$$

$$\begin{aligned}\mathbf{H}(\boldsymbol{q}_e, \boldsymbol{\alpha}_q) = {} & 8\frac{\boldsymbol{S}(\boldsymbol{q}_e)(\mathbf{S}(\boldsymbol{q}) - \mathbf{I}_{3\times3}) + \mathbf{S}(\boldsymbol{\alpha}_q)\boldsymbol{S}(\boldsymbol{q}_e)}{(1+\|\boldsymbol{q}\|^2)^2} + \mathbf{S}(\boldsymbol{\alpha}_q)(\mathbf{S}(\boldsymbol{\alpha}_q) - \mathbf{I}_{3\times3}) \\ & * \left(\frac{8}{\left(1+\|\boldsymbol{q_e}+\boldsymbol{\alpha}_q\|^2\right)^2} - \frac{8}{\left(1+\|\boldsymbol{\alpha}_q\|^2\right)^2}\right) \\ & - 4\mathbf{S}(\boldsymbol{\alpha}_q)\left(\frac{1}{1+\|\boldsymbol{q_e}+\boldsymbol{\alpha}_q\|^2} - \frac{1}{1+\|\boldsymbol{\alpha}_q\|^2}\right).\end{aligned} \tag{4-248}$$

It is difficult to design the virtual control signal $\boldsymbol{\alpha}_q$ to stabilise the error vector $\boldsymbol{\nu}_{1e}$ at the origin. To get around this and stabilise $\boldsymbol{\eta}_{1e}$ and $\boldsymbol{\nu}_{1e}$ we will design the virtual control signal $\boldsymbol{\alpha}_2$ and then calculate $\boldsymbol{\alpha}_q$ after. The Euler virtual control for the attitude:

$$\boldsymbol{\alpha}_2 = [\alpha_\phi \quad \alpha_\theta \quad \alpha_\psi]^{\mathrm{T}}, \tag{4-249}$$

where, α_ϕ denotes the virtual control for the pitch, α_θ the virtual control for the roll and α_ψ virtual control for the yaw angle:

$$\alpha_\psi = \psi_d. \tag{4-250}$$

We will now define each component of $\boldsymbol{\alpha}_q$ as follows:

$$\alpha_{q_1} = \frac{\sin(\frac{\alpha_\phi}{2})\cos(\frac{\alpha_\theta}{2})\cos(\frac{\alpha_\psi}{2}) - \cos(\frac{\alpha_\phi}{2})\sin(\frac{\alpha_\theta}{2})\sin(\frac{\alpha_\psi}{2})}{1+\cos(\frac{\alpha_\phi}{2})\cos(\frac{\alpha_\theta}{2})\cos(\frac{\alpha_\psi}{2}) + \sin(\frac{\alpha_\phi}{2})\sin(\frac{\alpha_\theta}{2})\sin(\frac{\alpha_\psi}{2})}, \tag{4-251}$$

$$\alpha_{q_2} = \frac{\cos(\frac{\alpha_\phi}{2})\sin(\frac{\alpha_\theta}{2})\cos(\frac{\alpha_\psi}{2}) + \sin(\frac{\alpha_\phi}{2})\cos(\frac{\alpha_\theta}{2})\sin(\frac{\alpha_\psi}{2})}{1+\cos(\frac{\alpha_\phi}{2})\cos(\frac{\alpha_\theta}{2})\cos(\frac{\alpha_\psi}{2}) + \sin(\frac{\phi_d}{2})\sin(\frac{\alpha_\theta}{2})\sin(\frac{\alpha_\psi}{2})}, \tag{4-252}$$

$$\alpha_{q_3} = \frac{\cos(\frac{\alpha_\phi}{2})\cos(\frac{\alpha_\theta}{2})\sin(\frac{\alpha_\psi}{2}) - \sin(\frac{\alpha_\phi}{2})\sin(\frac{\alpha_\theta}{2})\cos(\frac{\alpha_\psi}{2})}{1+\cos(\frac{\alpha_\phi}{2})\cos(\frac{\alpha_\theta}{2})\cos(\frac{\alpha_\psi}{2}) + \sin(\frac{\alpha_\phi}{2})\sin(\frac{\alpha_\theta}{2})\sin(\frac{\alpha_\psi}{2})}. \tag{4-253}$$

we can write:

$$\mathbf{R}_1(\boldsymbol{\alpha}_q) = \mathbf{R}_1(\boldsymbol{\alpha}_2). \tag{4-254}$$

To stabilise $\boldsymbol{\nu}_{1e}$ and design the linear velocity $\hat{\boldsymbol{\nu}}_1$ estimate we will consider the Lyapunov function candidate:

$$V_2 = \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} V_1 + \tfrac{1}{2}\left(4\gamma_1\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2} + \|\tilde{\boldsymbol{\eta}}_1\|^2 + \|\tilde{\boldsymbol{\nu}}_1\|^2\right) - 2\gamma_1. \tag{4-255}$$

The inclusion of the term $-2\gamma_1$ is to counteract the offset in the term $2\gamma_1\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}$ that is:

$$0 < 2\gamma_1\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2} - 2\gamma_1, \qquad \forall \|\boldsymbol{\nu}_{1e}\|! = 0. \tag{4-256}$$

Which implies that the function V_2 is strictly positive definite , only being equal to zero when:

$$\|\boldsymbol{\eta}_{1e}\|, \|\boldsymbol{\nu}_{1e}\|, \|\tilde{\boldsymbol{\eta}}_1\|, \|\tilde{\boldsymbol{\nu}}_1\| = 0 \tag{4-257}$$

To stabilise the error vector $\boldsymbol{\nu}_{1e}$ at the origin we will design the virtual control $\boldsymbol{\alpha}_2$. Only components α_ϕ and α_θ are considered as α_ψ has already been defined in (4-250). To design the control f and the virtual control $\boldsymbol{\alpha}_2$ we calculate the infinite generator of (4-255) along the solution of (4-235), (4-236)and the derivative of (4-225) gives:

$$\begin{aligned}\mathcal{L}V_2 \leq & -\gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \gamma_1 \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\boldsymbol{\nu}_{1e} \\ & + \frac{2\gamma_1 \boldsymbol{\nu}_{1e}^{\mathrm{T}}}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}} \Bigg(\mathbf{G}_1\Bigg(-g\boldsymbol{e}_3 + \mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1 + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\sigma(\boldsymbol{\eta}_{1e}) + \frac{f}{m}\mathbf{R}_1(\boldsymbol{\alpha}_q)\boldsymbol{e}_3 \\ & + \boldsymbol{h}_3 + \boldsymbol{h}_4 + \boldsymbol{h}_5\Bigg) + \boldsymbol{F}_1 + \boldsymbol{E}_2 - \ddot{\boldsymbol{\eta}}_{1d} + \mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^T}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\mathbf{D}_1\hat{\boldsymbol{\nu}}_1 - \mathbf{D}_1\boldsymbol{\nu}_{1e} - \mathbf{D}_1\boldsymbol{\alpha}_1 \\ & + \mathbf{G}_1\tfrac{f}{m}\mathbf{H}(\boldsymbol{q}_e, \boldsymbol{\alpha}_q)\boldsymbol{e}_3\Bigg) - \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1 + \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}(-\boldsymbol{h}_1) - \tilde{\boldsymbol{\nu}}_1^{\mathrm{T}}\mathbf{D}_1\tilde{\boldsymbol{\nu}}_1 \\ & + \tilde{\boldsymbol{\nu}}_1^{\mathrm{T}}(-\boldsymbol{h}_2 - \boldsymbol{h}_3 - \boldsymbol{h}_4 - \boldsymbol{h}_5) + \tilde{\boldsymbol{\nu}}_1^{\mathrm{T}}(\mathbf{I}_{3\times3} - \mathbf{K}_{02})\tilde{\boldsymbol{\eta}}_1 + \tfrac{1}{m}\tilde{\boldsymbol{\nu}}_1^{\mathrm{T}}\bar{\boldsymbol{f}}_{Aero} \\ & + \frac{1}{2}\mathrm{Tr}\left\{\Delta_1^{\mathrm{T}}(t)\tfrac{1}{m}\frac{\partial^2 V_3}{\partial \tilde{\boldsymbol{\nu}}_{11e}^{\ 2}}\tfrac{1}{m}\Delta_1(t)\right\}.\end{aligned} \tag{4-258}$$

We will choose the interlace term $\boldsymbol{h}_2$ and $\boldsymbol{h}_3$ as:

$$\boldsymbol{h}_2 = \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\sigma(\boldsymbol{\eta}_{1e}), \tag{4-259}$$

$$\boldsymbol{h}_3 = \frac{2\gamma_1}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}} \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\boldsymbol{\nu}_{1e}. \tag{4-260}$$

We now use Young's inequality:

$$\tfrac{1}{m}\tilde{\boldsymbol{\nu}}_1^{\mathrm{T}}\bar{\boldsymbol{f}}_{Aero} \leq \lambda_m(\mathbf{D}_1)\frac{\varepsilon_1}{2}\|\tilde{\boldsymbol{\nu}}_1\|^2 + \frac{\left\|\bar{\boldsymbol{f}}_{Aero}\right\|^2}{\varepsilon_1 2m^2}, \tag{4-261}$$

$$\frac{1}{2}\mathrm{Tr}\left\{\Delta_1^{\mathrm{T}}(t)\tfrac{1}{m}\frac{\partial^2 V_3}{\partial \tilde{\boldsymbol{\nu}}_{11e}^{\ 2}}\tfrac{1}{m}\Delta_1(t)\right\} \leq \frac{1}{2m^2}\left\|\Delta_1(t)\Delta_1^{\mathrm{T}}(t)\right\|_\infty^1, \tag{4-262}$$

$$\gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \boldsymbol{\nu}_{1e}^{\mathrm{T}} \sigma(\boldsymbol{\eta}_{1e}) \leq \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{4} \left(\varepsilon_2 \frac{\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} + \frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\varepsilon_2} \frac{1+\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \right). \tag{4-263}$$

Moreover,

$$-1 \leq \frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}} \sigma(\boldsymbol{\eta}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \leq 1, \tag{4-264}$$

$$\frac{2\gamma_1^2 \boldsymbol{\nu}_{1e}^{\mathrm{T}} \sigma(\boldsymbol{\eta}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \frac{\lambda_m(\mathbf{D}_1)}{2} \leq 2\gamma_1^2 \frac{\lambda_m(\mathbf{D}_1)}{2} \left(\frac{\|\boldsymbol{\nu}_{1e}\|^2 + 2\boldsymbol{\nu}_{1e}^{\mathrm{T}} \boldsymbol{\alpha}_1}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}} \sigma(\boldsymbol{\eta}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \frac{2+\|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} \right), \tag{4-265}$$

this is true because:

$$\left\| \frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}} \sigma(\boldsymbol{\eta}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \right\| \leq 1, \tag{4-266}$$

$$\frac{\Delta_1(\hat{\boldsymbol{\nu}}_1)}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} = \frac{2 + \|\boldsymbol{\alpha}_1\|^2 + \|\boldsymbol{\nu}_{1e}\|^2 + 2\boldsymbol{\nu}_{1e}^{\mathrm{T}} \boldsymbol{\alpha}_1}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} = 1. \tag{4-267}$$

By substituting (4-259) to (4-263) and (4-265) back into (4-258) gives:

$$\begin{aligned}
\mathcal{L}V_2 \leq& -\gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e}) \frac{\mathbf{K}_1 \sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \frac{2\gamma_1 \boldsymbol{\nu}_{1e}^{\mathrm{T}}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \\
&* \Bigg(\mathbf{G}_1 \left(-g\boldsymbol{e}_3 + \mathbf{K}_{02} \tilde{\boldsymbol{\eta}}_1 + \frac{f}{m} \mathbf{R}_1(\boldsymbol{\alpha}_q) \boldsymbol{e}_3 + \frac{\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \frac{2\gamma_1 \boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \right) \\
&- \frac{\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \frac{\mathbf{K}_1 \sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} - \mathbf{D}_1 \boldsymbol{\alpha}_1 - \mathbf{D}_1 \boldsymbol{\nu}_{1e} - \ddot{\boldsymbol{\eta}}_{1d} + \mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e}) \hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2} \mathbf{D}_1 \hat{\boldsymbol{\nu}}_1 \\
&- \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e}) \hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2} \sigma(\boldsymbol{\eta}_{1e}) + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \frac{\boldsymbol{\nu}_{1e} + 2\boldsymbol{\alpha}_1}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} \sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2} \\
&+ \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \sigma(\boldsymbol{\eta}_{1e}) \frac{2+\|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} \Bigg) \\
&+ \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{4} \left(\varepsilon_2 \frac{\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} + \frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\varepsilon_2} \frac{1+\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \right) \\
&+ \frac{2\gamma_1 \|\boldsymbol{\nu}_{1e}\|^2}{(1+\|\boldsymbol{\nu}_{1e}\|^2)} \left(\frac{\|\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})\|}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2} \right) + \boldsymbol{\nu}_{1e}^{\mathrm{T}} \mathbf{G}_1 (\boldsymbol{h}_4 + \boldsymbol{h}_5) \\
&+ \boldsymbol{\nu}_{1e}^{\mathrm{T}} \mathbf{G}_1 \frac{f}{m} \mathbf{H}(\boldsymbol{q}_e, \boldsymbol{\alpha}_q) \boldsymbol{e}_3 - \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}} \mathbf{K}_{01} \tilde{\boldsymbol{\eta}}_1 + \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}} (-\boldsymbol{h}_1 + \mathbf{K}_{02}^{\mathrm{T}} \mathbf{G}_1^{\mathrm{T}} \boldsymbol{\nu}_{1e}) - \tilde{\boldsymbol{\nu}}_1^{\mathrm{T}} \mathbf{D}_1 \tilde{\boldsymbol{\nu}}_1 \\
&+ \lambda_m(\mathbf{D}_1) \frac{\varepsilon_1}{2} \|\tilde{\boldsymbol{\nu}}_1\|^2 + \tilde{\boldsymbol{\nu}}_1^{\mathrm{T}} (\mathbf{I}_{3\times 3} - \mathbf{K}_{02}) \tilde{\boldsymbol{\eta}}_1 + \tilde{\boldsymbol{\nu}}_1^{\mathrm{T}} (-\boldsymbol{h}_4 - \boldsymbol{h}_5) \\
&+ \frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{\varepsilon_1 2m^2 \lambda_m(\mathbf{D}_1)} + \frac{\|\Delta_1(t) \Delta_1^{\mathrm{T}}(t)\|_\infty^1}{2m^2}.
\end{aligned} \tag{4-268}$$

This suggests the choosing of:

$$\mathbf{G}_1\left(-g\boldsymbol{e}_3+\frac{f}{m}\mathbf{R}_1(\boldsymbol{\alpha}_q)\boldsymbol{e}_3+\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\frac{2\gamma_1\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\right)-\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\sigma(\boldsymbol{\eta}_{1e})$$
$$-\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}-\mathbf{D}_1\boldsymbol{\alpha}_1+\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\mathbf{D}_1\hat{\boldsymbol{v}}_1+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{\boldsymbol{v}_{1e}+2\boldsymbol{\alpha}_1}{2\Delta_1(\hat{\boldsymbol{v}}_1)}\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}$$
$$+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma(\boldsymbol{\eta}_{1e})\frac{2+\|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{v}}_1)}-\ddot{\boldsymbol{\eta}}_{1d}\qquad=-\mathbf{K}_2\sigma(\boldsymbol{v}_{1e})-c\frac{2\boldsymbol{\alpha}_1+\boldsymbol{v}_{1e}}{2\Delta_1(\hat{\boldsymbol{v}}_1)}\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}, \tag{4-269}$$

where the matrix $\mathbf{K}_2\in\mathbb{R}^{3\times3}$, $\mathbf{K}_2=diag(k_{2,1},k_{2,2},k_{2,3})$ where $k_{2,1}$, $k_{2,2}$ and $k_{2,3}$ are positive constants to be defined later.

Substituting (4-269), back into (4-268) gives:

$$\mathcal{L}V_2\le-\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}-\frac{2\gamma_1\boldsymbol{v}_{1e}^{\mathrm{T}}\mathbf{K}_2\sigma(\boldsymbol{v}_{1e})}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}-\frac{2\gamma_1\boldsymbol{v}_{1e}^{\mathrm{T}}\mathbf{D}_1\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}$$
$$-2\gamma_1c\boldsymbol{v}_{1e}^{\mathrm{T}}\left(\frac{2\boldsymbol{\alpha}_1+\boldsymbol{v}_{1e}}{2\Delta_1(\hat{\boldsymbol{v}}_1)}\right)+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{4}\left(\varepsilon_2\frac{\|\boldsymbol{v}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}+\frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\varepsilon_2}\frac{1+\|\boldsymbol{v}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\right)$$
$$+\frac{2\gamma_1\|\boldsymbol{v}_{1e}\|^2}{(1+\|\boldsymbol{v}_{1e}\|^2)}\left(\frac{\|\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\|}{\Delta_1(\hat{\boldsymbol{v}}_1)}\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}\right)+\frac{2\gamma_1\boldsymbol{v}_{1e}^{\mathrm{T}}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\mathbf{G}_1(\boldsymbol{h}_4+\boldsymbol{h}_5)$$
$$-\tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1+\tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\left(-\boldsymbol{h}_1+\frac{2\gamma_1\mathbf{K}_{02}^{\mathrm{T}}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\right)-\lambda_m(\mathbf{D}_1)\left(1-\frac{\varepsilon_1}{2}\right)\|\tilde{\boldsymbol{v}}_1\|^2$$
$$+\tilde{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{I}_{3\times3}-\mathbf{K}_{02})\tilde{\boldsymbol{\eta}}_1-\tilde{\boldsymbol{v}}_1^{\mathrm{T}}(\boldsymbol{h}_4+\boldsymbol{h}_5)+\frac{\varepsilon_2\lambda_m(\mathbf{D}_1)}{2}\|\tilde{\boldsymbol{v}}_1\|^2+\frac{\left\|\bar{\boldsymbol{f}}_{Aero}\right\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2}$$
$$+\frac{\left\|\Delta_1(t)\Delta_1^{\mathrm{T}}(t)\right\|_\infty^1}{2m^2}+\frac{2\gamma_1\boldsymbol{v}_{1e}^{\mathrm{T}}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\mathbf{G}_1\frac{f}{m}\mathbf{H}(\boldsymbol{q}_e,\boldsymbol{\alpha}_q)\boldsymbol{e}_3. \tag{4-270}$$

The last stage of (4-270) will be considered in the second stage of the control design for the time being we do not consider it for the design of $f\mathbf{R}(\boldsymbol{\alpha}_q)\boldsymbol{e}_3$. To achieve stochastic practical asymptotic stability, we choose:

$$f\mathbf{R}_1(\boldsymbol{\alpha}_q)\boldsymbol{e}_3=m\mathbf{G}_1^{-1}\left(\mathbf{D}_1\boldsymbol{\alpha}_1-\mathbf{K}_2\sigma(\boldsymbol{v}_{1e})-\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma(\boldsymbol{\eta}_{1e})\frac{2+\|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{v}}_1)}\right.$$
$$+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\sigma(\boldsymbol{\eta}_{1e})-\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\mathbf{D}_1\hat{\boldsymbol{v}}_1$$
$$+\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}-\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{2\boldsymbol{\alpha}_1+\boldsymbol{v}_{1e}}{8\Delta_1(\hat{\boldsymbol{v}}_1)}\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}$$
$$\left.+\ddot{\boldsymbol{\eta}}_{1d}\right)+m\left(g\boldsymbol{e}_3-\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\frac{2\gamma_1\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\right)\quad\stackrel{\text{def}}{=}\boldsymbol{\Omega}. \tag{4-271}$$

Because:

$$\mathbf{R}(\boldsymbol{\alpha}_2)=\mathbf{R}(\boldsymbol{\alpha}_q), \tag{4-272}$$

equation (4-271) can now be written as:

$$f = \mathbf{R}_1^{-1}(\boldsymbol{\alpha}_2)\boldsymbol{\Omega}. \tag{4-273}$$

Let the components of $\boldsymbol{\Omega}$ be Ω_1, Ω_2 and Ω_3 i.e. $\boldsymbol{\Omega} = [\Omega_1 \quad \Omega_2 \quad \Omega_3]^{\mathrm{T}}$. Taking the norms of both sides of (4-273) gives:

$$\sqrt{ff\boldsymbol{e}_3^{\mathrm{T}}\boldsymbol{e}_3} = \sqrt{\mathbf{R}_1^{-\mathrm{T}}(\boldsymbol{\alpha}_2)\mathbf{R}_1^{-1}(\boldsymbol{\alpha}_2)\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}, \tag{4-274}$$

$$\sqrt{ff} = \sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}, \tag{4-275}$$

$$f = \sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}. \tag{4-276}$$

Where $\mathbf{R}_1^{-\mathrm{T}}(\boldsymbol{\alpha}_2)\mathbf{R}_1^{-1}(\boldsymbol{\alpha}_2) = \mathbf{I}_{3\times 3}$. By expanding out (4-273) and using the fact that $\mathbf{R}_1^{-1}(\boldsymbol{\alpha}_2) = \mathbf{R}_1^{\mathrm{T}}(\boldsymbol{\alpha}_2)$ we obtain:

$$\begin{aligned}
\left(\mathrm{c}_{\alpha_\psi}\mathrm{c}_{\alpha_\theta}\right)\Omega_1 + \left(\mathrm{s}_{\alpha_\psi}\mathrm{c}_{\alpha_\theta}\right)\Omega_2 - \left(\mathrm{s}_{\alpha_\theta}\right)\Omega_3 &= 0,\\
\left(-\mathrm{s}_{\alpha_\psi}\mathrm{c}_{\alpha_\phi} + \mathrm{c}_{\alpha_\psi}\mathrm{s}_{\alpha_\theta}\mathrm{s}_{\alpha_\phi}\right)\Omega_1 + \left(\mathrm{c}_{\alpha_\psi}\mathrm{c}_{\alpha_\phi} + \mathrm{s}_{\alpha_\psi}\mathrm{s}_{\alpha_\theta}\mathrm{s}_{\alpha_\phi}\right)\Omega_2 + \left(\mathrm{c}_{\alpha_\theta}\mathrm{s}_{\alpha_\phi}\right)\Omega_3 &= 0,\\
\left(\mathrm{s}_{\alpha_\psi}\mathrm{s}_{\alpha_\phi} + \mathrm{c}_{\alpha_\psi}\mathrm{s}_{\alpha_\theta}\mathrm{c}_{\alpha_\phi}\right)\Omega_1 + \left(-\mathrm{c}_{\alpha_\psi}\mathrm{s}_{\alpha_\phi} + \mathrm{s}_{\alpha_\psi}\mathrm{s}_{\alpha_\theta}\mathrm{c}_{\alpha_\phi}\right)\Omega_2 + \left(\mathrm{c}_{\alpha_\theta}\mathrm{c}_{\alpha_\phi}\right)\Omega_3 &= f.
\end{aligned} \tag{4-277}$$

Multiplying the second equation of (4-277) through by $-\mathrm{c}_{\alpha_\phi}$ and multiplying the third equation of (4-277) by s_{α_ϕ} and adding the resultant together yields:

$$\alpha_\phi = \sin^{-1}\left(\frac{\mathrm{s}_{\alpha_\psi}\Omega_1 - \mathrm{c}_{\alpha_\psi}\Omega_2}{\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}}\right). \tag{4-278}$$

Which is well defined since $\left|\mathrm{s}_{\alpha_\psi}\Omega_1 - \mathrm{c}_{\alpha_\psi}\Omega_2\right| \le \sqrt{\Omega_1^2 + \Omega_2^2} \le \sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}} > 0$. Furthermore, from the first equation of (4-277) we can obtain:

$$\alpha_\theta = \tan^{-1}\left(\frac{\mathrm{c}_{\alpha_\psi}\Omega_1 + \mathrm{s}_{\alpha_\psi}\Omega_2}{\Omega_3}\right). \tag{4-279}$$

With:

$$\begin{aligned}
\boldsymbol{\Omega} = m\mathbf{G}_1^{-1}\Bigg(&\mathbf{D}_1\boldsymbol{\alpha}_1 - \mathbf{K}_2\sigma(\boldsymbol{\nu}_{1e}) - \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma(\boldsymbol{\eta}_{1e})\frac{2 + \|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\\
&+ \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\sigma(\boldsymbol{\eta}_{1e}) - \frac{\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}\mathbf{D}_1\hat{\boldsymbol{\nu}}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})\\
&+ \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e}) - \left(c + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{2\boldsymbol{\alpha}_1 + \boldsymbol{\nu}_{1e}}{8\Delta_1(\hat{\boldsymbol{\nu}}_1)}\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}\\
&+ \ddot{\boldsymbol{\eta}}_{1d}\Bigg) + m\left(g\boldsymbol{e}_3 - \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\frac{2\gamma_1\boldsymbol{\nu}_{1e}}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}\right).
\end{aligned} \tag{4-280}$$

To ensure that (4-277) is well defined, we will show that $\boldsymbol{\Omega}$ is bounded and there are initial values of the states and control gains such that $\Omega_3(t) > 0$. We state:

$$\|\boldsymbol{\Omega}\| \leq m\|\varrho_0\|\Bigg(\|\mathbf{D}_1\|(\|\mathbf{K}_1\|+\varrho_1)+\|\mathbf{K}_2\|+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+(\|\mathbf{K}_1\|+\varrho_1)}{2} + 0.8\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\|\mathbf{K}_1\|+0.5\|\mathbf{K}_1\|\|\mathbf{D}_1\|+\|\mathbf{K}_1\|^2 + \left(\frac{c}{8}+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{16}\right)\Big(\sqrt{0.5+(0.46\|\mathbf{K}_1\|+0.7\varrho_1)^2+1.3\|\mathbf{K}_1\|+1.6\varrho_1+4} + (\|\mathbf{K}_1\|+\varrho_1)\sqrt{1+(\|\mathbf{K}_1\|+\varrho_1)^2+1.3\|\mathbf{K}_1\|+1.6\varrho_1+0.5}\Big)+\varrho_2\Bigg) + m(g+2\gamma_1\|\mathbf{K}_1\|). \tag{4-281}$$

In addition, ϱ_0 is defined as shown below:

$$\varrho_0 = 1-0.27\big(\mathbf{K}_{1,1}+\mathbf{K}_{1,2}+\mathbf{K}_{1,3}\big)>0. \tag{4-282}$$

We can list the upper bounds for Ω_1, Ω_2 and Ω_3 as follows:

$$\lfloor\Omega_1\rceil \leq \Omega_{1\mathrm{M}} = \frac{m}{\varrho_0}\Bigg(\big(1-k_{1,2}-k_{1,3}\big)\Bigg(\mathbf{D}_{1,1}\big(k_{1,1}+\varrho_{1,1}\big)+k_{2,1} + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2}+0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,1}+0.5\lambda_M(\mathbf{D}_1)k_{1,1} + k_{1,1}^2+\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right) * \left(\big(k_{1,1}+\varrho_{1,1}\big)\frac{1+k_{1,1}+\varrho_{1,1}}{2}+\sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right)++\varrho_{2,1}\Bigg) + \Big(k_{1,1}\big(2k_{1,3}+1\big)\Big)\Bigg(\mathbf{D}_{2,2}\big(k_{1,2}+\varrho_{1,2}\big)+k_{2,2} + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2}++0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,2} + 0.5\lambda_M(\mathbf{D}_1)k_{1,2}+k_{1,2}^2+\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right) * \left(\big(k_{1,2}+\varrho_{1,2}\big)\frac{1+k_{1,2}+\varrho_{1,2}}{2}+\sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right)+\varrho_{2,2}\Bigg) + \Big(k_{1,1}\big(2k_{1,2}+1\big)\Big)\Bigg(\mathbf{D}_{2,2}\big(k_{1,3}+\varrho_{1,3}\big)+k_{2,3} + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2}++0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,3} + 0.5\lambda_M(\mathbf{D}_1)k_{1,3}+k_{1,3}^2+\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right) * \left(\big(k_{1,3}+\varrho_{1,3}\big)\frac{1+k_{1,3}+\varrho_{1,3}}{2}+\sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right)+\varrho_{2,3}\Bigg)\Bigg) + m\big(-k_{1,1}+2\gamma_1\big), \tag{4-283}$$

$$\lfloor \Omega_2 \rfloor \leq \Omega_{2\mathrm{M}} = \frac{m}{\varrho_0}\Bigg(\Big(k_{1,2}(2k_{1,3}+1)\Big)\Bigg(\mathbf{D}_{1,1}(k_{1,1}+\varrho_{1,1})+k_{2,1}$$
$$+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2}+0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,1}+0.5\lambda_M(\mathbf{D}_1)k_{1,1}$$
$$+k_{1,1}^2+\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)$$
$$*\left((k_{1,1}+\varrho_{1,1})\frac{1+k_{1,1}+\varrho_{1,1}}{2}+\sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right)++\varrho_{2,1}\Bigg)$$
$$+(1-k_{1,1}-k_{1,3})\Bigg(\mathbf{D}_{2,2}(k_{1,2}+\varrho_{1,2})+k_{2,2}$$
$$+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2}++0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,2}$$
$$+0.5\lambda_M(\mathbf{D}_1)k_{1,2}+k_{1,2}^2+\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)$$
$$*\left((k_{1,2}+\varrho_{1,2})\frac{1+k_{1,2}+\varrho_{1,2}}{2}+\sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right)+\varrho_{2,2}\Bigg)$$
$$+\Big(k_{1,2}(2kk_{1,1}+1)\Big)\Bigg(\mathbf{D}_{2,2}(k_{1,3}+\varrho_{1,3})+k_{2,3}$$
$$+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2}++0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,3}$$
$$+0.5\lambda_M(\mathbf{D}_1)k_{1,3}+k_{1,3}^2+\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)$$
$$*\left((k_{1,3}+\varrho_{1,3})\frac{1+k_{1,3}+\varrho_{1,3}}{2}+\sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right)+\varrho_{2,3}\Bigg)\Bigg)$$
$$+mk_{1,2}(k_{1,2}+\varrho_{1,2})+2\varepsilon_2\gamma_1^2\frac{1+k_{1,1}+k_{1,2}+k_{1,3}}{\lambda_m(\mathbf{D}_1)}, \tag{4-284}$$

$$\lfloor \Omega_3 \rfloor \leq \Omega_{3\mathrm{M}} = \frac{m}{\varrho_0}\Bigg(\Big(k_{1,3}\big(2k_{1,2}+1\big)\Big)\Bigg(\mathbf{D}_{1,1}\big(k_{1,1}+\varrho_{1,1}\big)+k_{2,1} + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2} + 0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,1} + 0.5\lambda_M(\mathbf{D}_1)k_{1,1} + k_{1,1}^2 + \left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right) * \left(\big(k_{1,1}+\varrho_{1,1}\big)\frac{1+k_{1,1}+\varrho_{1,1}}{2} + \sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right) + +\varrho_{2,1}\Bigg) + \Big(k_{1,3}\big(2\mathrm{i}k_{1,1}+1\big)\Big)\Bigg(\mathbf{D}_{2,2}\big(k_{1,2}+\varrho_{1,2}\big)+k_{2,2} + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2} + +0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,2} + 0.5\lambda_M(\mathbf{D}_1)k_{1,2} + k_{1,2}^2 + \left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right) * \left(\big(k_{1,2}+\varrho_{1,2}\big)\frac{1+k_{1,2}+\varrho_{1,2}}{2} + \sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right) + \varrho_{2,2}\Bigg) + \big(1-k_{1,1}-k_{1,2}\big)\Bigg(\mathbf{D}_{2,2}\big(k_{1,3}+\varrho_{1,3}\big)+k_{2,3} + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2} + +0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,3} + 0.5\lambda_M(\mathbf{D}_1)k_{1,3} + k_{1,3}^2 + \left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right) * \left(\big(k_{1,3}+\varrho_{1,3}\big)\frac{1+k_{1,3}+\varrho_{1,3}}{2} + \sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right) + \varrho_{2,3}\Bigg)\Bigg) + mg + mk_{1,3}\big(k_{1,3}+\varrho_{1,3}\big) + 2\varepsilon_2\gamma_1^2\frac{1+k_{1,1}+k_{1,2}+k_{1,3}}{\lambda_m(\mathbf{D}_1)}. \tag{4-285}$$

Where $\Omega_{1\mathrm{M}}$, $\Omega_{2\mathrm{M}}$ and $\Omega_{3\mathrm{M}}$ are the upper bounds of the control signals Ω_1, Ω_2 and Ω_3 of Referring back to (4-216) where we specified that the aircraft cannot descend faster than it freely falls due to ϱ_5 which is a strictly positive constant. The control gains γ_1, $\mathbf{K}_1$, $\mathbf{K}_2$ and $\mathbf{K}_{1\mathrm{d},3}$ need to be selected so that the following holds true:

$$\begin{aligned}\Omega_3 \geq -\frac{m}{\varrho_0}\Bigg(& k_{1,3}(2k_{1,2}+1)\Bigg(\mathbf{D}_{1,1}(k_{1,1}+\varrho_{1,1})+k_{2,1}+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2} \\ & +0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,1}+0.5\lambda_M(\mathbf{D}_1)k_{1,1}+k_{1,1}^2+\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right) \\ & *\left((k_{1,1}+\varrho_{1,1})\frac{1+k_{1,1}+\varrho_{1,1}}{2}+\sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right)++\varrho_{2,1}\Bigg) \\ & +k_{1,3}(2k_{1,1}+1) \\ & *\Bigg(\mathbf{D}_{2,2}(k_{1,2}+\varrho_{1,2})+k_{2,2}+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2} \\ & ++0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,2}+0.5\lambda_M(\mathbf{D}_1)k_{1,2}+k_{1,2}^2+\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right) \\ & *\left((k_{1,2}+\varrho_{1,2})\frac{1+k_{1,2}+\varrho_{1,2}}{2}+\sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right)+\varrho_{2,2}\Bigg) \\ & +(1-k_{1,1}-k_{1,2}) \\ & *\Bigg(\mathbf{D}_{2,2}(k_{1,3}+\varrho_{1,3})+k_{2,3}+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{2+\|(\|\mathbf{K}_1\|+\varrho_1)\|^2}{2} \\ & ++0.46\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}k_{1,3}+0.5\lambda_M(\mathbf{D}_1)k_{1,3}+k_{1,3}^2+\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right) \\ & *\left((k_{1,3}+\varrho_{1,3})\frac{1+k_{1,3}+\varrho_{1,3}}{2}+\sqrt{(1+\lambda_M(\mathbf{K}_1)+\varrho_1)/2}\right)+\varrho_{2,3}\Bigg)\Bigg)+mg \\ & -mk_{1,3}(k_{1,3}+\varrho_{1,3})-2\varepsilon_2\gamma_1^2\frac{1+k_{1,1}+k_{1,2}+k_{1,3}}{\lambda_m(\mathbf{D}_1)}\geq\Omega_3^*>0.\end{aligned} \quad (4\text{-}286)$$

Where Ω_3^* is the lower bound of Ω_3 which strictly greater than zero and is a product of the effects ϱ_5 and ϱ_{f3} have on (4-216), this implies that Ω_3 has an upper and lower bound. It can now be said that because $\Omega_3 > 0$, α_θ defined in (4-279) is well defined. Once the virtual control $\boldsymbol{\alpha}_2 = col(\alpha_\phi, \alpha_\theta, \alpha_\psi)$ is available from (4-278), (4-279) and (4-250), the virtual control $\boldsymbol{\alpha}_q$ is determined from (4-251) to (4-253).

4.2.3 Stabilization of Attitude System

In this stage, we consider the two equations of (4-222) to be able to use the backstepping method for this stage we define the coordinate transformation:

$$\boldsymbol{\omega}_e = \boldsymbol{\omega} - \boldsymbol{\alpha}_{\omega}. \tag{4-287}$$

Where $\boldsymbol{\omega}_e$ is the tracking error of the angular velocity and $\boldsymbol{\alpha}_{\boldsymbol{\omega}}$ the virtual control for the angular velocity. In the first step, we design $\boldsymbol{\alpha}_{\omega}$ to stabilize in probability the tracking error $\boldsymbol{q}_e$ at the origin. In the second step, we design the torque vector $\boldsymbol{\tau}$ to stochastically practically asymptotically stabilise the tracking error $\boldsymbol{\omega}_e$ at the origin.

Step 1: Before calculating $\dot{\boldsymbol{q}}_e$ let us first calculate $\dot{\boldsymbol{\alpha}}_2$ as:

$$\dot{\boldsymbol{\alpha}}_2 = [\dot{\alpha}_{\phi} \quad \dot{\alpha}_{\theta} \quad \dot{\alpha}_{\psi}]^{\mathrm{T}}. \tag{4-288}$$

Furthermore, from (4-278) and (4-279) we obtain:

$$\alpha_{\phi} = \sin^{-1}\left(\frac{s_{\alpha_\psi}\Omega_1 - c_{\alpha_\psi}\Omega_2}{\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}}\right), \qquad \alpha_{\theta} = \tan^{-1}\left(\frac{c_{\alpha_\psi}\Omega_1 + s_{\alpha_\psi}\Omega_2}{\Omega_3}\right). \tag{4-289}$$

We can state that:

$$\dot{\boldsymbol{\Omega}} = \frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\left(-\mathbf{D}_1\hat{\boldsymbol{v}}_1 - g\boldsymbol{e}_3 + \frac{\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}}{m}\mathbf{R}_1(\boldsymbol{q})\boldsymbol{e}_3 + \mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1 + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma(\boldsymbol{\eta}_{1e}) + \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\boldsymbol{v}_{1e} + \boldsymbol{h}_4 + \boldsymbol{h}_5\right) + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}(\tilde{\boldsymbol{v}}_1 + \hat{\boldsymbol{v}}_1 - \dot{\boldsymbol{\eta}}_{1d}) + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1d}}\dot{\boldsymbol{\eta}}_{1d} + \frac{\partial\boldsymbol{\Omega}}{\partial\dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial\boldsymbol{\Omega}}{\partial\ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d}, \tag{4-290}$$

$$\boldsymbol{\Omega}_d = \frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\left(-\mathbf{D}_1\hat{\boldsymbol{v}}_1 - g\boldsymbol{e}_3 + \frac{\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}}{m}\mathbf{R}_1(\boldsymbol{q})\boldsymbol{e}_3 + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma(\boldsymbol{\eta}_{1e}) + \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\boldsymbol{v}_{1e} + \boldsymbol{h}_4\right) + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\left(-\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} + \boldsymbol{v}_{1e}\right) + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1d}}\dot{\boldsymbol{\eta}}_{1d} + \frac{\partial\boldsymbol{\Omega}}{\partial\dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial\boldsymbol{\Omega}}{\partial\ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d}, \tag{4-291}$$

where the partial derivatives of $\boldsymbol{\Omega}$ are defined in Appendix D. We will state:

$$\dot{\boldsymbol{\Omega}} = \boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\tilde{\boldsymbol{v}}_1 + \frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\boldsymbol{h}_5 + \frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1. \tag{4-292}$$

Therefore, we state:

$$\dot{\boldsymbol{\alpha}}_2 = \mathbf{A}_{2,d}\boldsymbol{\Omega}_d + \mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\tilde{\boldsymbol{v}}_1 + \mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\boldsymbol{h}_5 + \mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1 + \begin{bmatrix}0\\0\\1\end{bmatrix}\dot{\alpha}_{\psi} + \mathbf{A}_{1,d}\boldsymbol{\Omega}\dot{\alpha}_{\psi}, \tag{4-293}$$

$$\mathbf{A}_{1,d} = \begin{bmatrix} c_{\alpha_\psi}\dfrac{1}{\cos(\alpha_\phi)\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}} & s_{\alpha_\psi}\dfrac{1}{\cos(\alpha_\phi)\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}} & 0 \\ -s_{\alpha_\psi}\dfrac{\cos^2(\alpha_\theta)}{\Omega_3} & c_{\alpha_\psi}\dfrac{\cos^2(\alpha_\theta)}{\Omega_3} & 0 \\ 0 & 0 & 0 \end{bmatrix}, \tag{4-294}$$

$$\mathbf{A}_{2,d} = \begin{bmatrix} \frac{s_{\alpha_\psi}}{\cos(\alpha_\phi)\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}} & \frac{-c_{\alpha_\psi}}{\cos(\alpha_\phi)\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}} & 0 \\ \cos^2(\alpha_\theta)\frac{c_{\alpha_\psi}}{\Omega_3} & \cos^2(\alpha_\theta)\frac{s_{\alpha_\psi}}{\Omega_3} & -\cos^2(\alpha_\theta)\frac{\tan(\alpha_\theta)}{\Omega_3} \\ 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} \frac{\tan(\alpha_\phi)}{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}} \\ 0 \\ 0 \end{bmatrix} \boldsymbol{\Omega}^{\mathrm{T}}. \tag{4-295}$$

As mentioned earlier we will use the modified Rodrigues parameter as such we will state:

$$\dot{\boldsymbol{\alpha}}_q = \mathbf{R}_2(\boldsymbol{\alpha}_2)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\dot{\boldsymbol{\alpha}}_2. \tag{4-296}$$

Differentiating both sides of (4-245) along the solutions of (4-296) and (4-287) yields:

$$\dot{\boldsymbol{q}}_e = \mathbf{R}_2(\boldsymbol{q})(\boldsymbol{\alpha}_{\boldsymbol{\omega}} + \boldsymbol{\omega}_e) - \mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\dot{\boldsymbol{\alpha}}_2. \tag{4-297}$$

To analyse the stability of the origin of $\boldsymbol{q}_e$ we consider the Lyapunov function candidate as follows:

$$V_3 = V_2 + \tfrac{\gamma_2}{2}\|\boldsymbol{q}_e\|^2. \tag{4-298}$$

Taking the infinitesimal generator of both sides of (4-298) along the solution of (4-270) and (4-297) gives:

$$\begin{aligned} \mathcal{L}V_3 \le & -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} - \frac{2\gamma_1\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{K}_2\sigma(\boldsymbol{\nu}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} - 2\gamma_1 c\boldsymbol{\nu}_{1e}^{\mathrm{T}}\left(\frac{2\boldsymbol{\alpha}_1+\boldsymbol{\nu}_{1e}}{2\Delta_1(\hat{\boldsymbol{v}}_1)}\right) \\ & + \frac{2\gamma_1\|\boldsymbol{\nu}_{1e}\|^2}{(1+\|\boldsymbol{\nu}_{1e}\|^2)}\left(\frac{\|\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\|}{\Delta_1(\hat{\boldsymbol{v}}_1)}\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}\right) - \frac{2\gamma_1\lambda_m(\mathbf{D}_1)\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \\ & * \left(1 - \frac{1}{8}\left(\varepsilon_2 + \frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\varepsilon_2}\right)\right) + \frac{2\gamma_1\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{G}_1}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}(\boldsymbol{h}_4+\boldsymbol{h}_5) - \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1 \\ & + \tilde{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{I}_{3\times 3} - \mathbf{K}_{02})\tilde{\boldsymbol{\eta}}_1 - \tilde{\boldsymbol{v}}_1^{\mathrm{T}}(\boldsymbol{h}_4+\boldsymbol{h}_5) - \lambda_m(\mathbf{D}_1)\|\tilde{\boldsymbol{v}}_1\|^2\left(1-\frac{\varepsilon_2}{2}\right) \\ & + \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\left(-\boldsymbol{h}_1 + \frac{2\gamma_1\mathbf{K}_{02}^{\mathrm{T}}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) + \frac{2\gamma_1\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{G}_1\frac{\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}}{m}\mathbf{H}(\boldsymbol{q}_e,\boldsymbol{\alpha}_q)\boldsymbol{e}_3}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \\ & + \gamma_2\boldsymbol{q}_e^T\left(\mathbf{R}_2(\boldsymbol{q})(\boldsymbol{\alpha}_{\boldsymbol{\omega}}+\boldsymbol{\omega}_e) - \dot{\boldsymbol{\alpha}}_q\right) + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{4}\left(\frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\varepsilon_2\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) + \frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2} \\ & + \frac{\|\Delta_1(t)\Delta_1^{\mathrm{T}}(t)\|_\infty^1}{2m^2}. \end{aligned} \tag{4-299}$$

We will define the following interlace term:

$$-\gamma_2\boldsymbol{q}_e^{\mathrm{T}}\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\tilde{\boldsymbol{v}}_1 = \tilde{\boldsymbol{v}}_1^{\mathrm{T}}\boldsymbol{h}_4. \tag{4-300}$$

We will now define

$$\boldsymbol{q}_e^{\mathrm{T}}\mathbf{L} = \left(\mathbf{H}(\boldsymbol{q}_e,\boldsymbol{\alpha}_q)\boldsymbol{e}_3\right)^{\mathrm{T}}. \tag{4-301}$$

This implies that:

$$\mathbf{L}=\frac{8}{(1+\|\boldsymbol{q}\|^2)^2}\begin{bmatrix}\boldsymbol{q}_3 & 1 & -\boldsymbol{q}_1-\boldsymbol{\alpha}_{q,1}\\ -1 & \boldsymbol{q}_3 & -\boldsymbol{q}_2-\boldsymbol{\alpha}_{q,2}\\ \boldsymbol{\alpha}_{q,1} & \boldsymbol{\alpha}_{q,2} & 0\end{bmatrix}+\left(\frac{4\mathbf{I}_{3\times3}}{(1+\|\boldsymbol{q}\|^2)}-\frac{4(\boldsymbol{q}+\boldsymbol{\alpha}_q)\boldsymbol{\alpha}_q^{\mathrm{T}}}{(1+\|\boldsymbol{q}\|^2)\left(1+\|\boldsymbol{\alpha}_q\|^2\right)}\right)$$
$$\times\begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}-8\left(\frac{\boldsymbol{q}_e\left(\|\boldsymbol{q}_e\|^2+2+2\|\boldsymbol{\alpha}_q\|^2\right)+4\boldsymbol{\alpha}_q\left(1+\|\boldsymbol{q}\|^2-\boldsymbol{q}_e^{\mathrm{T}}\boldsymbol{\alpha}_q\right)}{(1+\|\boldsymbol{q}\|^2)^2\left(1+\|\boldsymbol{\alpha}_q\|^2\right)^2}\right)$$
$$\times\left(\begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}}\left(\boldsymbol{\alpha}_q\boldsymbol{\alpha}_q^{\mathrm{T}}-\|\boldsymbol{\alpha}_q\|^2\mathbf{I}_{3\times3}\right)+\boldsymbol{\alpha}_q^{\mathrm{T}}\begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\right). \tag{4-302}$$

Substituting (4-300) - (4-302) into (4-299) gives

$$\mathcal{L}V_3\leq-\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}-\frac{2\gamma_1\boldsymbol{v}_{1e}^{\mathrm{T}}\mathbf{K}_2\sigma(\boldsymbol{v}_{1e})}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}-2\gamma_1 c\boldsymbol{v}_{1e}^{\mathrm{T}}\left(\frac{2\boldsymbol{\alpha}_1+\boldsymbol{v}_{1e}}{2\Delta_1(\hat{\boldsymbol{v}}_1)}\right)$$
$$+\frac{2\gamma_1\|\boldsymbol{v}_{1e}\|^2}{(1+\|\boldsymbol{v}_{1e}\|^2)}\left(\frac{\|\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\|}{\Delta_1(\hat{\boldsymbol{v}}_1)}\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}\right)-\frac{2\gamma_1\lambda_m(\mathbf{D}_1)\|\boldsymbol{v}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}$$
$$*\left(1-\frac{\varepsilon_2}{8}-\frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{8\varepsilon_2}\right)+\frac{2\gamma_1\boldsymbol{v}_{1e}^{\mathrm{T}}\mathbf{G}_1}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}(\boldsymbol{h}_4+\boldsymbol{h}_5)-\tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1$$
$$+\tilde{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{I}_{3\times3}-\mathbf{K}_{02})\tilde{\boldsymbol{\eta}}_1-\tilde{\boldsymbol{v}}_1^{\mathrm{T}}\boldsymbol{h}_5-\lambda_m(\mathbf{D}_1)\|\tilde{\boldsymbol{v}}_1\|^2\left(1-\frac{\varepsilon_2}{2}\right)-\frac{2\gamma_1\boldsymbol{v}_{1e}^{\mathrm{T}}(\boldsymbol{h}_5)}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}$$
$$-\tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\left(-\boldsymbol{h}_1+\frac{2\gamma_1\mathbf{K}_{02}^{\mathrm{T}}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}+\mathbf{K}_{02}^{\mathrm{T}}\left(\boldsymbol{q}_e^{T}\mathbf{R}_2(\boldsymbol{\alpha}_2)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\right)^{\mathrm{T}}\right)$$
$$+\gamma_2\boldsymbol{q}_e^{T}\left(-\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}+\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\alpha}_{\boldsymbol{\omega}}-\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\right.$$
$$\left.*\mathbf{A}_{2,d}\boldsymbol{\Omega}_d-\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}(\mathbf{A}_{1,d}\boldsymbol{\Omega}+\boldsymbol{e}_3)\dot{\alpha}_\psi+\frac{2\gamma_1\|\boldsymbol{\Omega}\|\mathbf{L}}{\gamma_2 m}\frac{\mathbf{G}_1^{\mathrm{T}}\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\right)$$
$$+\gamma_2\boldsymbol{q}_e^{T}\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e-\gamma_2\boldsymbol{q}_e^{T}\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\boldsymbol{h}_5+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{4}\left(\frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\varepsilon_2\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\right)$$
$$+\frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2}+\frac{\|\Delta_1(t)\|_\infty^2}{2m^2}. \tag{4-303}$$

To stabilise $\boldsymbol{q}_e$ and ensures that $\mathcal{L}V_3$ is stochastically practically asymptotically stable we will implement the standard backstopping method to design the virtual control signal $\boldsymbol{\alpha}_{\boldsymbol{\omega}}$ as follows:

$$-\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d+\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\right)+(\mathbf{A}_{1,d}\boldsymbol{\Omega}+\boldsymbol{e}_3)\dot{\alpha}_\psi\right)$$
$$+2\gamma_1\frac{\|\boldsymbol{\Omega}\|\mathbf{L}}{\gamma_2 m}\frac{\mathbf{G}_1^{\mathrm{T}}\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}+\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\alpha}_{\boldsymbol{\omega}}=-\mathbf{K}_3\boldsymbol{q}_e. \tag{4-304}$$

Where the matrix $\mathbf{K}_3\in\mathbb{R}^{3\times3}$ is positive definite, $\mathbf{K}_3=diag(k_{3,1},k_{3,2},k_{3,3})$ where $k_{3,1}$, $k_{3,2}$ and $k_{3,3}$ are positive constants to be defined later. Thus $\boldsymbol{\alpha}_\omega$ is defined as follows:

$$\boldsymbol{\alpha}_\omega = \mathbf{R}_2^{-1}(\boldsymbol{q})\left(\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) + \left(\mathbf{A}_{1,d}\boldsymbol{\Omega} + \boldsymbol{e}_3\right)\dot{\alpha}_\psi\right) - \mathbf{K}_3\boldsymbol{q}_e - \frac{2\gamma_1\|\boldsymbol{\Omega}\|\mathbf{L}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\gamma_2 m\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right). \tag{4-305}$$

Substituting (4-305) into (4-303):

$$\begin{aligned}\mathcal{L}V_3 \le & -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} - \frac{2\gamma_1\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{K}_2\sigma(\boldsymbol{\nu}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} - 2\gamma_1 c\boldsymbol{\nu}_{1e}^{\mathrm{T}}\left(\frac{2\boldsymbol{\alpha}_1 + \boldsymbol{\nu}_{1e}}{2\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\\ & + \frac{2\gamma_1\|\boldsymbol{\nu}_{1e}\|^2}{(1+\|\boldsymbol{\nu}_{1e}\|^2)}\left(\frac{\|\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\|}{\Delta_1(\hat{\boldsymbol{v}}_1)}\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}\right) - \frac{2\gamma_1\lambda_m(\mathbf{D}_1)\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\\ & * \left(1 - \frac{\varepsilon_2}{8} - \frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{8\varepsilon_2}\right) - \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1 + \tilde{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{I}_{3\times3} - \mathbf{K}_{02})\tilde{\boldsymbol{\eta}}_1 - \tilde{\boldsymbol{v}}_1^{\mathrm{T}}\boldsymbol{h}_5\\ & - \lambda_m(\mathbf{D}_1)\|\tilde{\boldsymbol{v}}_1\|^2\left(1 - \frac{\varepsilon_2}{2}\right) - \gamma_2\boldsymbol{q}_e^{\mathrm{T}}\mathbf{K}_3\boldsymbol{q}_e + \gamma_2\boldsymbol{q}_e^{T}\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e\\ & + \left(\frac{2\gamma_1\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{G}_1}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} - \gamma_2\boldsymbol{q}_e^{T}\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\right)\boldsymbol{h}_5\\ & + \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\left(-\boldsymbol{h}_1 + \frac{2\gamma_1\mathbf{K}_{02}^{\mathrm{T}}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} + \mathbf{K}_{02}^{\mathrm{T}}\left(\gamma_2\boldsymbol{q}_e^{\mathrm{T}}\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\right)^{\mathrm{T}}\right)\\ & + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{4}\left(\frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\varepsilon_2\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) + \frac{\left\|\bar{\boldsymbol{f}}_{Aero}\right\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2} + + \frac{\|\Delta_1(t)\|_\infty^2}{2m^2}.\end{aligned} \tag{4-306}$$

To prepare for the next step we calculate $\dot{\boldsymbol{\alpha}}_{\boldsymbol{\omega}}$ by applying the stochastic differentiation rule differentiating both sides of (4-305) we obtain.

$$\begin{aligned}\dot{\boldsymbol{\alpha}}_\omega = & \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\mathbf{q}}\dot{\mathbf{q}} + \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\hat{\boldsymbol{v}}_1}\dot{\hat{\boldsymbol{v}}}_1 + \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\alpha_\psi}\dot{\alpha}_\psi + \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\dot{\alpha}_\psi}\ddot{\alpha}_\psi + \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e} + \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1d}}\dot{\boldsymbol{\eta}}_{1d} + \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d}\\ & + \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d} + \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\dddot{\boldsymbol{\eta}}_{1d}}\ddddot{\boldsymbol{\eta}}_{1d},\end{aligned} \tag{4-307}$$

where the partial derivative of $\boldsymbol{\alpha}_\omega$ are presented in Appendix E. We will define the following:

$$\dot{\boldsymbol{\alpha}}_\omega = \boldsymbol{\alpha}_{\omega,d} + \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1e}}\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\tilde{\boldsymbol{v}}_1 + \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\hat{\boldsymbol{v}}_1}\mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1, \tag{4-308}$$

where we define:

$$\begin{aligned}\boldsymbol{\alpha}_{\omega,d} = & \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\mathbf{q}}\dot{\mathbf{q}} - \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\hat{\boldsymbol{v}}_1}\left(\frac{f}{m}\mathbf{R}(\boldsymbol{\eta}_2)\boldsymbol{e}_3 - \mathbf{D}_1\hat{\boldsymbol{v}}_1 - g\boldsymbol{e}_3 + \boldsymbol{h}_2 + \boldsymbol{h}_3 + \boldsymbol{h}_4 + \boldsymbol{h}_5\right) + \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\alpha_\psi}\dot{\alpha}_\psi\\ & + \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\dot{\alpha}_\psi}\ddot{\alpha}_\psi + \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1e}}(\hat{\boldsymbol{v}}_1 - \dot{\boldsymbol{\eta}}_{1d\cdot}) + \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1d}}\dot{\boldsymbol{\eta}}_{1d} + \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d} + \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\dddot{\boldsymbol{\eta}}_{1d}}\ddddot{\boldsymbol{\eta}}_{1d}.\end{aligned} \tag{4-309}$$

Step 2: Now we will design a control law for the torque vector $\boldsymbol{\tau}$ to stabilize $\boldsymbol{\omega}_e$ at the origin. Differentiating both sides of (4-287) along the solutions of (4-307) and (4-222) yields $d\boldsymbol{\omega}_e$ as follows:

$$d\boldsymbol{\omega}_e = \left((\mathbf{I}_A + \mathbf{I}_H)^{-1}(-\mathbf{D}_2\boldsymbol{\omega} - \mathbf{C}_{B22}(\boldsymbol{\omega})\boldsymbol{\omega} - \mathbf{C}_{A22}(\boldsymbol{\omega})\boldsymbol{\omega} + \boldsymbol{\tau}) - \dot{\boldsymbol{\alpha}}_\omega\right)dt + (\mathbf{I}_A + \mathbf{I}_H)^{-1}\boldsymbol{\theta}\Delta_2(t)d\boldsymbol{w}_2 + (\mathbf{I}_A + \mathbf{I}_H)^{-1}\Delta_3(t)d\boldsymbol{w}_3, \tag{4-310}$$

$$d\boldsymbol{\omega}_e = \Bigg((\mathbf{I}_A + \mathbf{I}_H)^{-1}(-\mathbf{D}_2\boldsymbol{\omega} - \mathbf{C}_{B22}(\boldsymbol{\omega})\boldsymbol{\omega} - \mathbf{C}_{A22}(\boldsymbol{\omega})\boldsymbol{\omega} + \boldsymbol{\tau}) - \boldsymbol{\alpha}_{\omega,d} - \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1e}}\tilde{\boldsymbol{\nu}}_1 - \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\hat{\boldsymbol{\nu}}_1}\mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1\Bigg)dt + (\mathbf{I}_A + \mathbf{I}_H)^{-1}\boldsymbol{\theta}\Delta_2(t)d\boldsymbol{w}_2 + (\mathbf{I}_A + \mathbf{I}_H)^{-1}\Delta_3(t)d\boldsymbol{w}_3. \tag{4-311}$$

To analyse the stability of $\boldsymbol{\omega}_e$ at the origin of we consider the Lyapunov function:

$$V_4 = V_3 + \frac{\gamma_3}{2}\sqrt{1 + \|\boldsymbol{\omega}_e\|^4} + \frac{1}{2}\bar{\boldsymbol{\tau}}_{de,Aero}^{\mathrm{T}}\boldsymbol{\Gamma}^{-1}\bar{\boldsymbol{\tau}}_{de,Aero} + \frac{1}{2\mu_1}\tilde{\delta}_1^2 + \frac{1}{2\mu_2}\tilde{\delta}_2^2 - \frac{\gamma_3}{2}, \tag{4-312}$$

where the inclusion of the term $-\frac{\gamma_3}{2}$ is to counteract the offset in the term $\frac{\gamma_3}{2}\sqrt{1 + \|\boldsymbol{\omega}_e\|^4}$ that is:

$$0 < \frac{\gamma_3}{2}\sqrt{1 + \|\boldsymbol{\omega}_e\|^4} - \frac{\gamma_3}{2}, \qquad \forall\|\boldsymbol{\omega}_e\|! = 0. \tag{4-313}$$

Which implies that the function V_4 is strictly positive definite, only being equal to zero when:

$$\|\boldsymbol{\eta}_{1e}\|, \|\boldsymbol{\nu}_{1e}\|, \|\tilde{\boldsymbol{\eta}}_1\|, \|\tilde{\boldsymbol{\nu}}_1\|, \|\boldsymbol{q}_e\|, \|\boldsymbol{\omega}_e\|, \left\|\bar{\boldsymbol{\tau}}_{de,Aero}\right\|, \left\|\tilde{\delta}_1\right\|, \left\|\tilde{\delta}_2\right\| = 0. \tag{4-314}$$

Moreover, the matrix $\boldsymbol{\Gamma}$ is a positive definite matrix, μ_1 and μ_2 are positive constants, and we have defined the following uncertainty errors:

$$\tilde{\delta}_1 = \left\|\Delta_2(t)\Delta_2^{\mathrm{T}}(t)\right\|_\infty^2 - \hat{\delta}_1, \tag{4-315}$$

$$\tilde{\delta}_2 = \left\|\Delta_3(t)\Delta_3^{\mathrm{T}}(t)\right\|_\infty^2 - \hat{\delta}_2, \tag{4-316}$$

$$\bar{\boldsymbol{\tau}}_{de,Aero} = \bar{\boldsymbol{\tau}}_{Aero} - \hat{\boldsymbol{\tau}}_{Aero}. \tag{4-317}$$

The infinite generator of (4-312) along the solutions of (4-306) and (4-311) is:

$$\begin{aligned}\mathcal{L}V_4 \leq & -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} - \frac{2\gamma_1\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{K}_2\sigma(\boldsymbol{\nu}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} - 2\gamma_1 c\boldsymbol{\nu}_{1e}^{\mathrm{T}}\left(\frac{2\boldsymbol{\alpha}_1 + \boldsymbol{\nu}_{1e}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right) \\ & + \frac{2\gamma_1\|\boldsymbol{\nu}_{1e}\|^2}{(1+\|\boldsymbol{\nu}_{1e}\|^2)}\left(\frac{\|\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\|}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}\right) - \frac{2\gamma_1\lambda_m(\mathbf{D}_1)\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \\ & * \left(1 - \frac{\varepsilon_2}{8} - \frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{8\varepsilon_2}\right) - \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1 + \tilde{\boldsymbol{\nu}}_1^{\mathrm{T}}(\mathbf{I}_{3\times3} - \mathbf{K}_{02})\tilde{\boldsymbol{\eta}}_1 - \tilde{\boldsymbol{\nu}}_1^{\mathrm{T}}\boldsymbol{h}_5 \\ & -\lambda_m(\mathbf{D}_1)\|\tilde{\boldsymbol{\nu}}_1\|^2\left(1 - \frac{\varepsilon_2}{2}\right) - \gamma_2\boldsymbol{q}_e^{\mathrm{T}}\mathbf{K}_3\boldsymbol{q}_e + \gamma_2\boldsymbol{q}_e^{T}\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e \\ & + \left(\frac{2\gamma_1\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{G}_1}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} - \gamma_2\boldsymbol{q}_e^{T}\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{\nu}}_1}\right)\boldsymbol{h}_5 \\ & + \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\left(-\boldsymbol{h}_1 + \frac{2\gamma_1\mathbf{K}_{02}^{\mathrm{T}}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} + \mathbf{K}_{02}^{\mathrm{T}}\left(\gamma_2\boldsymbol{q}_e^{\mathrm{T}}\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{\nu}}_1}\right)^{\mathrm{T}}\right)\end{aligned} \tag{4-318}$$

$$+\gamma_3\frac{\|\boldsymbol{\omega}_e\|^2\boldsymbol{\omega}_e^{\mathrm{T}}}{(1+\|\boldsymbol{\omega}_e\|^4)^{3/4}}\Big((\mathbf{I}_A+\mathbf{I}_H)^{-1}(-\mathbf{D}_2\boldsymbol{\omega}-\mathbf{C}_{B22}(\boldsymbol{\omega})\boldsymbol{\omega}-\mathbf{C}_{A22}(\boldsymbol{\omega})\boldsymbol{\omega}+\boldsymbol{\tau}+\bar{\boldsymbol{\tau}}_{Aero})$$
$$-\boldsymbol{\alpha}_{\omega,d}-\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1e}}\tilde{\boldsymbol{v}}_1-\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\hat{\boldsymbol{v}}_1}\mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1\Big)-\bar{\boldsymbol{\tau}}_{de,Aero}^{\mathrm{T}}\boldsymbol{\Gamma}^{-1}\dot{\hat{\boldsymbol{\tau}}}_{Aero}$$
$$+\frac{\gamma_3}{2}\mathrm{Tr}\left\{\Delta_2^{\mathrm{T}}(t)(\mathbf{I}_A+\mathbf{I}_H)^{-\mathrm{T}}\boldsymbol{\theta}^{\mathrm{T}}\frac{\partial^2V_4}{\partial\boldsymbol{\omega}_e^2}(\mathbf{I}_A+\mathbf{I}_H)^{-1}\boldsymbol{\theta}\Delta_2(t)\right\}-\frac{1}{\mu_1}\tilde{\delta}_1\dot{\hat{\delta}}_1-\frac{1}{\mu_2}\tilde{\delta}_2\dot{\hat{\delta}}_2$$
$$+\frac{\gamma_3}{2}\mathrm{Tr}\left\{\Delta_3^{\mathrm{T}}(t)(\mathbf{I}_A+\mathbf{I}_H)^{-\mathrm{T}}\frac{\partial^2V_4}{\partial\boldsymbol{\omega}_e^2}(\mathbf{I}_A+\mathbf{I}_H)^{-1}\Delta_3(t)\right\}+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{4}\left(\frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\varepsilon_2\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\right)$$
$$+\frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2}++\frac{\|\Delta_1(t)\|_\infty^2}{2m^2}.$$

Where we define:

$$\frac{\partial^2V_5}{\partial\boldsymbol{\omega}_e^2}=\frac{\mathbf{I}_{3\times3}\|\boldsymbol{\omega}_e\|^2+2\boldsymbol{\omega}_e\boldsymbol{\omega}_e^T}{(1+\|\boldsymbol{\omega}_e\|^4)^{3/4}}-3\frac{\|\boldsymbol{\omega}_e\|^4\boldsymbol{\omega}_e\boldsymbol{\omega}_e^T}{(1+\|\boldsymbol{\omega}_e\|^4)^{7/4}},\tag{4-319}$$

$$-\gamma_3\frac{\|\boldsymbol{\omega}_e\|^2}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}\left(\boldsymbol{\omega}_{qe}^T\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}}=\boldsymbol{h}_5.\tag{4-320}$$

Applying Young's inequality to (4-318)yields:

$$\boldsymbol{q}_{qe}^T\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e\le\varepsilon_3\|\boldsymbol{q}_e\|^2+\varepsilon_4\left(\frac{1}{4\varepsilon_3}\right)^2\frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^3}+\frac{\boldsymbol{q}_{qe}^T\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{3/4}}+\frac{1}{4\varepsilon_4},\tag{4-321}$$

$$\frac{\gamma_3}{2}\mathrm{Tr}\left\{\Delta_2^{\mathrm{T}}(t)(\mathbf{I}_A+\mathbf{I}_H)^{-\mathrm{T}}\boldsymbol{\theta}^{\mathrm{T}}\frac{\partial^2V_4}{\partial\boldsymbol{\omega}_e^2}(\mathbf{I}_A+\mathbf{I}_H)^{-1}\boldsymbol{\theta}\Delta_2(t)\right\}$$
$$\le\frac{9\gamma_3^2}{4}\frac{\varepsilon_5\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\boldsymbol{\theta}\|^4\|\boldsymbol{\omega}_e\|^4\hat{\delta}_1}{(1+\|\boldsymbol{\omega}_e\|^4)^{1.5}}+\frac{1}{4\varepsilon_5}.\tag{4-322}$$

$$\frac{\gamma_3}{2}\mathrm{Tr}\left\{\Delta_3^{\mathrm{T}}(t)(\mathbf{I}_A+\mathbf{I}_H)^{-\mathrm{T}}\frac{\partial^2V_4}{\partial\boldsymbol{\omega}_e^2}(\mathbf{I}_A+\mathbf{I}_H)^{-1}\Delta_3(t)\right\}\le\frac{9\gamma_3^2}{4}\frac{\varepsilon_6\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\|^4\|\boldsymbol{\omega}_e\|^4\hat{\delta}_2}{(1+\|\boldsymbol{\omega}_e\|^4)^{1.5}}+\frac{1}{4\varepsilon_6}.\tag{4-323}$$

Where ε_i are positive constants using (4-320), (4-321), (4-322) and (4-323), we rewrite (4-318) as:

$$\mathcal{L}V_4\le-\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}-\frac{2\gamma_1\boldsymbol{v}_{1e}^{\mathrm{T}}\mathbf{K}_2\sigma(\boldsymbol{v}_{1e})}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}-2\gamma_1c\boldsymbol{v}_{1e}^{\mathrm{T}}\left(\frac{2\boldsymbol{\alpha}_1+\boldsymbol{v}_{1e}}{2\Delta_1(\hat{\boldsymbol{v}}_1)}\right)$$
$$+\frac{2\gamma_1\|\boldsymbol{v}_{1e}\|^2}{(1+\|\boldsymbol{v}_{1e}\|^2)}\left(\frac{\|\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\|}{\Delta_1(\hat{\boldsymbol{v}}_1)}\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}\right)-\frac{2\gamma_1\lambda_m(\mathbf{D}_1)\|\boldsymbol{v}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}$$
$$*\left(1-\frac{\varepsilon_2}{8}-\frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{8\varepsilon_2}\right)-\tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1+\tilde{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{I}_{3\times3}-\mathbf{K}_{02})\tilde{\boldsymbol{\eta}}_1-\lambda_m(\mathbf{D}_1)\|\tilde{\boldsymbol{v}}_1\|^2\left(\frac{2-\varepsilon_2}{2}\right)$$
$$+\tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\Bigg(-\boldsymbol{h}_1+\frac{2\gamma_1\mathbf{K}_{02}^{\mathrm{T}}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}-\mathbf{K}_{02}^{\mathrm{T}}\left(\gamma_2\boldsymbol{q}_e^{\mathrm{T}}\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\right)^{\mathrm{T}}$$
$$-\left(\frac{\gamma_3\|\boldsymbol{\omega}_e\|^2\boldsymbol{\omega}_e^T}{(1+\|\boldsymbol{\omega}_e\|^4)^{3/4}}\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\hat{\boldsymbol{v}}_1}\mathbf{K}_{01}\right)^{\mathrm{T}}\Bigg)-\gamma_2\boldsymbol{q}_e^{\mathrm{T}}(\mathbf{K}_3\boldsymbol{q}_e-\varepsilon_3\boldsymbol{q}_e)\tag{4-324}$$

$$
\begin{aligned}
&+\frac{\gamma_3\|\boldsymbol{\omega}_e\|^2\boldsymbol{\omega}_e^T}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}\Bigg((\mathbf{I}_A+\mathbf{I}_H)^{-1}(-\mathbf{D}_2\boldsymbol{\omega}-\mathbf{C}_{B22}(\boldsymbol{\omega})\boldsymbol{\omega}-\mathbf{C}_{A22}(\boldsymbol{\omega})\boldsymbol{\omega}+\boldsymbol{\tau}+\hat{\boldsymbol{\tau}}_{Aero})\\
&\qquad-\boldsymbol{\alpha}_{\omega,d}-\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1e}}\left(\frac{2\gamma_1\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{G}_1}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}-\gamma_2\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{\nu}}_1}\right)^{\mathrm{T}}\\
&\qquad+\frac{\gamma_2}{\gamma_3}\left(\left(\frac{1}{4\varepsilon_3}\right)^2\frac{\varepsilon_4\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{2.25}}+\frac{\boldsymbol{q}_{qe}^T\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e}{(1+\|\boldsymbol{\omega}_e\|^4)^{1/4}}\right)\boldsymbol{\omega}_e\\
&\qquad+\frac{9}{4}\gamma_3\varepsilon_5\frac{\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\boldsymbol{\theta}\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.75}}\hat{\delta}_1\boldsymbol{\omega}_e+\frac{9}{4}\gamma_3\varepsilon_6\frac{\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\|^4\hat{\delta}_2}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.75}}\boldsymbol{\omega}_e\Bigg)\\
&+\bar{\boldsymbol{\tau}}_{de,Aero}^{\mathrm{T}}\left(\gamma_3\frac{\|\boldsymbol{\omega}_e\|^2(\mathbf{I}_A+\mathbf{I}_H)^{-\mathrm{T}}\boldsymbol{\omega}_e}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.75}}-\boldsymbol{\Gamma}^{-1}\dot{\hat{\boldsymbol{\tau}}}_{Aero}\right)+\frac{\|\Delta_1(t)\|_\infty^2}{2m^2}+\frac{\|\bar{\boldsymbol{f}}_{Aero}\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2}\\
&+\tilde{\delta}_1\left(\gamma_3^2\frac{9}{4}\frac{\varepsilon_5\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\boldsymbol{\theta}\|^4\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{1.5}}-\frac{1}{\mu_1}\dot{\hat{\delta}}_1\right)+\frac{1}{4\varepsilon_5(1+\|\boldsymbol{\omega}_e\|^4)^{3/4}}\\
&+\tilde{\delta}_2\left(\gamma_3^2\frac{9}{4}\frac{\varepsilon_6\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\|^4\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{1.5}}-\frac{1}{\mu_2}\dot{\hat{\delta}}_2\right)+\frac{1}{4\varepsilon_6(1+\|\boldsymbol{\omega}_e\|^4)^{3/4}}+\frac{1}{4\varepsilon_4}\\
&+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{4}\left(\frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\varepsilon_2\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right).
\end{aligned}
$$

The dependence of the time varying disturbances $\Delta_2(t)$ and $\Delta_3(t)$ on t results in the generation of the terms $\frac{1}{4\varepsilon_5}$ and $\frac{1}{4\varepsilon_6}$, to overcome the effects of the disturbance on the system the control law for $\boldsymbol{\tau}$ and update law for $\hat{\delta}_1$ and $\hat{\delta}_2$ is designed using the projection algorithm presented in section 2.7.3 as follows:

$$
\begin{aligned}
\boldsymbol{\tau}=(\mathbf{I}_A+\mathbf{I}_H)\Bigg[&-(\mathbf{I}_A+\mathbf{I}_H)^{-1}(\mathbf{D}_2\boldsymbol{\alpha}_\omega-\mathbf{C}_{B22}(\boldsymbol{\omega})\boldsymbol{\omega}-\mathbf{C}_{A22}(\boldsymbol{\omega})\boldsymbol{\omega})+\boldsymbol{\alpha}_{\omega,d}-\mathbf{K}_4\boldsymbol{\omega}_e\\
&+\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1e}}\left(\frac{2\gamma_1\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{G}_1}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}-\gamma_2\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{\nu}}_1}\right)^{\mathrm{T}}\\
&-\Bigg(\frac{\gamma_2}{\gamma_3}\left(\left(\frac{1}{4\varepsilon_3}\right)^2\frac{\varepsilon_4\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{2.25}}+\frac{\boldsymbol{q}_{qe}^T\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega}_e}{(1+\|\boldsymbol{\omega}_e\|^4)^{1/4}}\right)\\
&+\gamma_3\frac{9}{4}\frac{\varepsilon_5\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\boldsymbol{\theta}\|^4}{(1+\|\omega_e\|^4)^{0.5}}\hat{\delta}_1+\gamma_3\frac{9}{4}\varepsilon_6\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\|^4\hat{\delta}_2\Bigg)\boldsymbol{\omega}_e\Bigg]-\hat{\boldsymbol{\tau}}_{Aero}.
\end{aligned}
\tag{4-325}
$$

To overcome the effects of the disturbance on the system the update laws for $\hat{\boldsymbol{\tau}}_{Aero}$, $\hat{\delta}_1$ and $\hat{\delta}_2$ designed using the projection algorithm presented in section 2.7.3 are as follows:

$$
\dot{\hat{\boldsymbol{\tau}}}_{Aero}=\boldsymbol{\Gamma}\mathrm{proj}\left(\gamma_3\frac{\|\boldsymbol{\omega}_e\|^2\mathbf{I}_H^{-T}\boldsymbol{\omega}_e}{(1+\|\omega_e\|^4)^{0.5}},\hat{\bar{\boldsymbol{\tau}}}_{Aero}\right),\tag{4-326}
$$

$$
\dot{\hat{\delta}}_1=\mu_1\mathrm{proj}\left(\frac{9}{4}\gamma_3^2\varepsilon_5\frac{\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\boldsymbol{\theta}\|^4\|\boldsymbol{\omega}_e\|^4}{(1+\|\omega_e\|^4)^1},\hat{\delta}_1\right),\tag{4-327}
$$

$$
\dot{\hat{\delta}}_2=\mu_2\mathrm{proj}\left(\frac{9}{4}\gamma_3^2\varepsilon_6\frac{\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\|^4\|\boldsymbol{\omega}_e\|^4}{(1+\|\omega_e\|^4)^{0.5}},\hat{\delta}_2\right).\tag{4-328}
$$

In addition, the update initial values are selected in accordance with:

$$\left(\hat{\boldsymbol{\tau}}_{Aero}(t_0), \hat{\delta}_1(t_0), \hat{\delta}_2(t_0)\right) \in Q_0, \tag{4-329}$$

$$Q_0 := \left(\hat{\boldsymbol{\tau}}_{Aero}^{min}, \hat{\boldsymbol{\tau}}_{Aero}^{Max}\right) \times \left(\hat{\delta}_1^{min}, \hat{\delta}_1^{Max}\right) \times \left(\hat{\delta}_2^{min}, \hat{\delta}_2^{Max}\right). \tag{4-330}$$

We will now define the interlace term $\boldsymbol{h}_1$:

$$\boldsymbol{h}_1 = \mathbf{K}_{02}^{\mathrm{T}}\left(\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e} + \gamma_2\left(\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\partial\hat{\boldsymbol{\nu}}_1}\right)^{\mathrm{T}} - \frac{\gamma_3\|\boldsymbol{\omega}_e\|^2}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}\left(\boldsymbol{\omega}_e^{\mathrm{T}}\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\hat{\boldsymbol{\nu}}_1}\right)^{\mathrm{T}}\right). \tag{4-331}$$

In addition, we now obtain:

$$\begin{aligned}
\mathcal{L}V_4 \leq& -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} - \frac{2\gamma_1\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{K}_2\sigma(\boldsymbol{\nu}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} - 2\gamma_1 c\boldsymbol{\nu}_{1e}^{\mathrm{T}}\left(\frac{2\boldsymbol{\alpha}_1+\boldsymbol{\nu}_{1e}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)\\
&+\frac{2\gamma_1\|\boldsymbol{\nu}_{1e}\|^2}{(1+\|\boldsymbol{\nu}_{1e}\|^2)}\left(\frac{\|\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\|}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}\right) - \frac{2\gamma_1\lambda_m(\mathbf{D}_1)\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\\
&*\left(1-\frac{\varepsilon_2}{8}-\frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{8\varepsilon_2}\right) - \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}}\mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1 + \tilde{\boldsymbol{\nu}}_1^{\mathrm{T}}(\mathbf{I}_{3\times3}-\mathbf{K}_{02})\tilde{\boldsymbol{\eta}}_1 - \lambda_m(\mathbf{D}_1)\|\tilde{\boldsymbol{\nu}}_1\|^2\left(1-\frac{\varepsilon_2}{2}\right)\\
&-\gamma_2\boldsymbol{q}_e^{\mathrm{T}}(\mathbf{K}_3\boldsymbol{q}_e - \varepsilon_5\boldsymbol{q}_e) - \gamma_3\frac{\|\boldsymbol{\omega}_e\|^2\boldsymbol{\omega}_e^T(\mathbf{K}_4+\mathbf{D}_2)\boldsymbol{\omega}_e}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}} + \frac{\left\|\bar{\boldsymbol{f}}_{Aero}\right\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2}\\
&+\frac{\left\|\Delta_1(t)\Delta_1^T(t)\right\|_\infty^1}{2m^2} + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{4}\frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\varepsilon_2\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} + \frac{1}{4}\sum_{i=4}^{6}\varepsilon_i,
\end{aligned} \tag{4-332}$$

we will now substitute (4-331) into (4-228) to give:

$$\begin{aligned}
\dot{\tilde{\boldsymbol{\eta}}}_1 =& \tilde{\boldsymbol{\nu}}_1 - \mathbf{K}_{01}\tilde{\boldsymbol{\eta}}_1 - \frac{2\gamma_1\mathbf{K}_{02}^{\mathrm{T}}\mathbf{G}_1^T\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} + \gamma_2\mathbf{K}_{02}^{\mathrm{T}}\left(\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}}\boldsymbol{q}_e\\
&+\gamma_3\mathbf{K}_{02}^{\mathrm{T}}\left(\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\hat{\boldsymbol{\nu}}_1}\right)^{\mathrm{T}}\frac{\|\boldsymbol{\omega}_e\|^2\boldsymbol{\omega}_e}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}.
\end{aligned} \tag{4-333}$$

We will now substitute (4-259), (4-260), (4-300)and (4-320) into (4-230) to give:

$$\begin{aligned}
d\tilde{\boldsymbol{\nu}}_1 =& \left(\mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1 - \mathbf{D}_1\tilde{\boldsymbol{\nu}}_1 + \tfrac{1}{m}\bar{\boldsymbol{f}}_{Aero} - \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma(\boldsymbol{\eta}_{1e}) - \frac{2\gamma_1\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\mathbf{G}_1^T\boldsymbol{\nu}_{1e}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right.\\
&\left.+\gamma_2\left(\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}}\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_e + \gamma_3\left(\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}}\frac{\|\boldsymbol{\omega}_e\|^2\boldsymbol{\omega}_e}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}\right)dt\\
&+\tfrac{1}{m}\Delta_1(t)\,d\boldsymbol{w}_1.
\end{aligned} \tag{4-334}$$

Let us state:

$$\frac{\left\|\bar{\boldsymbol{f}}_{Aero}\right\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2}+\frac{\left\|\Delta_1(t)\Delta_1^T(t)\right\|_\infty^1}{2m^2}\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{4}\frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\varepsilon_2\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}+\frac{1}{4\varepsilon_4}+\frac{1}{4\varepsilon_5}+\frac{1}{4\varepsilon_6}$$
$$\leq\left(\frac{\left\|\bar{\boldsymbol{f}}_{Aero,MAX}\right\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2}+\frac{\|\Delta_1(t)\|_\infty^2}{2m^2}+\frac{1}{4}\sum_{i=4}^{6}\varepsilon_i\right)\left(\frac{2+\|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}+\left(\boldsymbol{\nu}_{1e}^{\mathrm{T}}\frac{2\boldsymbol{\alpha}_1+\boldsymbol{\nu}_{1e}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)\right)$$
$$+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{4}\frac{1}{\varepsilon_2\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\left(\|\sigma(\boldsymbol{\eta}_{1e})\|^2\frac{2+\|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}+3\left(\boldsymbol{\nu}_{1e}^{\mathrm{T}}\frac{2\boldsymbol{\alpha}_1+\boldsymbol{\nu}_{1e}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)\right). \tag{4-335}$$

Since $\boldsymbol{\alpha}_1$ is bounded as per the use of the one-step ahead backstepping method, it can be stated:

$$\|\boldsymbol{\alpha}_1\|\leq\varrho_{\boldsymbol{\alpha}_1}, \tag{4-336}$$

This implies that:

$$\lambda_m(\bar{\mathbf{K}}_1)=\lambda_m(\mathbf{K}_1)-\frac{1}{2\varepsilon_2}\frac{2+\varrho_{\boldsymbol{\alpha}_1}^2}{2}>0. \tag{4-337}$$

Furthermore, because:

$$\|\sigma(\boldsymbol{\eta}_{1e})\|^2+\sum\nolimits_{i=1}^{3}\frac{1}{1+\boldsymbol{\eta}_{1e,i}^2}=3, \tag{4-338}$$

It can be stated that:

$$\left(\frac{\left\|\bar{\boldsymbol{f}}_{Aero,MAX}\right\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2}+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{4}\frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\varepsilon_2\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}+\frac{\|\Delta_1(t)\|_\infty^2}{2m^2}+\frac{1}{4}\sum_{i=4}^{6}\varepsilon_i\right)\frac{2+\|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}$$
$$=\frac{2+\|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\left(\frac{1}{3}\sum\nolimits_{i=1}^{3}\frac{1}{1+\boldsymbol{\eta}_{1e,i}^2}\left(\frac{\left\|\bar{\boldsymbol{f}}_{Aero,MAX}\right\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2}+\frac{\|\Delta_1(t)\|_\infty^2}{2m^2}+\frac{1}{4}\sum_{i=4}^{6}\varepsilon_i\right)\right.$$
$$\left.+\frac{1}{3}\sum\nolimits_{i=1}^{3}\frac{\boldsymbol{\eta}_{1e,i}^2}{1+\boldsymbol{\eta}_{1e,i}^2}\left(\frac{\left\|\bar{\boldsymbol{f}}_{Aero,MAX}\right\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2}+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{4}+\frac{\|\Delta_1(t)\|_\infty^2}{2m^2}+\frac{1}{4}\sum_{i=4}^{6}\varepsilon_i\right)\right), \tag{4-339}$$

$$\lambda_m(\mathbf{K}_1)\frac{\lambda_m(\mathbf{D}_1)}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}c_2=\frac{1}{3}\left(\frac{\left\|\bar{\boldsymbol{f}}_{Aero,MAX}\right\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2}+\frac{\|\Delta_1(t)\|_\infty^2}{2m^2}+\frac{1}{4}\sum_{i=4}^{6}\varepsilon_i\right)\frac{2+\varrho_{\boldsymbol{\alpha}_1}^2}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}, \tag{4-340}$$

$$c_3=\frac{1}{3}\left(\frac{\left\|\bar{\boldsymbol{f}}_{Aero,MAX}\right\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2}+\frac{\|\Delta_1(t)\|_\infty^2}{2m^2}+\frac{1}{4}\sum_{i=4}^{6}\varepsilon_i\right)\frac{2+\varrho_{\boldsymbol{\alpha}_1}^2}{2}, \tag{4-341}$$

We thus obtain

$$\mathcal{L}V_4\leq-\frac{\lambda_m(\mathbf{D}_1)}{2}\lambda_m(\mathbf{K}_1)(\gamma_1-c_2)\frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}-\frac{2\gamma_1\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{K}_2\sigma(\boldsymbol{\nu}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}$$
$$-\left(2\gamma_1c-\frac{\left\|\bar{\boldsymbol{f}}_{Aero,MAX}\right\|^2}{2\varepsilon_1\lambda_m(\mathbf{D}_1)m^2}-\frac{\|\Delta_1(t)\|_\infty^2}{2m^2}-3\frac{\lambda_m(\mathbf{D}_1)}{2\varepsilon_2}-\frac{1}{4}\sum_{i=4}^{6}\varepsilon_i\right)\boldsymbol{\nu}_{1e}^{\mathrm{T}}\left(\frac{2\boldsymbol{\alpha}_1+\boldsymbol{\nu}_{1e}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)$$
$$+\frac{2\gamma_1\|\boldsymbol{\nu}_{1e}\|^2}{(1+\|\boldsymbol{\nu}_{1e}\|^2)}\left(\frac{\|\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\|}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)-\frac{2\gamma_1\lambda_m(\mathbf{D}_1)\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \tag{4-342}$$

$$* \left(1 - \frac{\varepsilon_2}{8} - \frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{8\varepsilon_2}\right) - \tilde{\boldsymbol{\eta}}_1^{\mathrm{T}} \mathbf{K}_{01} \tilde{\boldsymbol{\eta}}_1 + \tilde{\boldsymbol{\nu}}_1^{\mathrm{T}} (\mathbf{I}_{3\times 3} - \mathbf{K}_{02}) \tilde{\boldsymbol{\eta}}_1$$
$$-\lambda_m(\mathbf{D}_1)\|\tilde{\boldsymbol{\nu}}_1\|^2 \left(1 - \frac{\varepsilon_2}{2}\right) - \gamma_2 \boldsymbol{q}_e^{\mathrm{T}} (\mathbf{K}_3 \boldsymbol{q}_e - \varepsilon_3 \boldsymbol{q}_e) - \gamma_3 \frac{\|\boldsymbol{\omega}_e\|^2 \boldsymbol{\omega}_e^T (\mathbf{K}_4 + \mathbf{D}_2) \boldsymbol{\omega}_e}{(1 + \|\boldsymbol{\omega}_e\|^4)^{0.5}}$$
$$+ \sum_{i=1}^{3} \frac{c_3}{\Delta_1(\hat{\boldsymbol{\nu}}_1(t))(1 + \boldsymbol{\eta}_{1e,i}^2)}.$$

In addition, we make the following conditions on the control gains:

$$b_1 < \frac{1 + \|\boldsymbol{\nu}_{1e}\|^2}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} < b_2, \qquad \forall \hat{\boldsymbol{\nu}}_1(t), \boldsymbol{\eta}_{1e}(t), \dot{\boldsymbol{\eta}}_{1d}(t) \in \mathbb{R}^{3\times 3}, t \in \mathbb{R}, \tag{4-343}$$
$$\bar{\gamma}_1 = \gamma_1 - c_4 > 0, \tag{4-344}$$
$$\|\sigma(\boldsymbol{\eta}_{1e})\|^2 < 3, \tag{4-345}$$
$$\lambda_m(\bar{\mathbf{K}}_2) = \lambda_m(\mathbf{K}_2) - \left(\|\mathbf{K}_1\|\sqrt{b_2}\right) > 0, \tag{4-346}$$
$$c = \frac{1}{2\gamma_1}\left(\frac{\left\|\bar{\boldsymbol{f}}_{Aero,MAX}\right\|^2}{2\varepsilon_1 \lambda_m(\mathbf{D}_1) m^2} + \frac{\|\Delta_1(t)\|_\infty^2}{2m^2} + \frac{1}{4}\sum_{i=4}^{6} \varepsilon_i\right), \tag{4-347}$$
$$\lambda_m(\bar{\mathbf{K}}_3) = \lambda_m(\mathbf{K}_3) - \varepsilon_3 > 0, \tag{4-348}$$
$$\lambda_m(\bar{\mathbf{D}}_1) = \lambda_m(\mathbf{D}_1)\left(1 - \frac{\varepsilon_2}{8} - \frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{8\varepsilon_2}\right) > 0, \tag{4-349}$$
$$\lambda_m(\bar{\bar{\mathbf{D}}}_1) = \lambda_m(\mathbf{D}_1)\left(1 - \frac{\varepsilon_2}{2}\right) > 0, \tag{4-350}$$
$$c_3 = \frac{1}{3}\left(\frac{\left\|\bar{\boldsymbol{f}}_{Aero,MAX}\right\|^2}{2\varepsilon_1 \lambda_m(\mathbf{D}_1) m^2} + \frac{\|\Delta_1(t)\|_\infty^2}{2m^2} + \frac{1}{4}\sum_{i=4}^{6} \varepsilon_i\right)\frac{2 + \varrho_{\alpha_1}^2}{2},. \tag{4-351}$$

we now state:

$$\mathcal{L}V_4 \leq -\bar{\gamma}_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \lambda_m(\bar{\mathbf{K}}_1) \frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} - 2\gamma_1 \frac{\lambda_m(\bar{\mathbf{K}}_2) \boldsymbol{\nu}_{1e}^{\mathrm{T}} \sigma(\boldsymbol{\nu}_{1e})}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}} - 2\gamma_1 \frac{\lambda_m(\bar{\mathbf{D}}_1)\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}} \tag{4-352}$$
$$-\lambda_m(\mathbf{K}_{01})\|\tilde{\boldsymbol{\eta}}_1\|^2 - \lambda_m(\bar{\bar{\mathbf{D}}}_1)\|\tilde{\boldsymbol{\nu}}_1\|^2 - \gamma_2 \lambda_m(\bar{\mathbf{K}}_3)\|\boldsymbol{q}_e\|^2 - \frac{\gamma_3 \lambda_m(\mathbf{K}_4)\|\boldsymbol{\omega}_e\|^4}{(1 + \|\boldsymbol{\omega}_e\|^4)^{0.5}}$$
$$+ \sum_{i=1}^{3} \frac{c_3}{\Delta_1(\hat{\boldsymbol{\nu}}_1(t))(1 + \boldsymbol{\eta}_{1e,i}^2)}.$$

Due to the presence of the term c_3 in the equation above, the closed loop system is stochastically practically asymptotically stable and not stochastically asymptotically stable. The control design has been completed and we will summarise the main results in the next subsection.

4.2.4 Stability Analysis

The control design to meet the control objective to achieve stochastic robust stability was completed in the previous sub-section in this sub-section we will perform the stability analysis of the controller designed. Firstly, we will state the following theorem and then prove this to be so:

Theorem 4.1 Under Assumption 4.5 to Assumption 4.9 the control and update laws of (4-276), (4-325), (4-326) (4-327) and (4-328) solve control objective 4.2 as long as the update initial values are chosen as specified in (4-329) and (4-330) and the control gains $\mathbf{K}_1, \mathbf{K}_2, \mathbf{K}_3, \mathbf{K}_4, \mathbf{K}_{01}$, $\mathbf{K}_{02}, \mathbf{\Gamma}, \mu_1, \mu_2$, $\gamma, \varepsilon_1, \varepsilon_2, \varepsilon_3, \varepsilon_4$ ε_5 and ε_6 are chosen such that the conditions (4-282) to (4-286) and (4-343) to (4-351) hold. In practice the below results will hold:

1. *The control inputs $U_i, i = 1,2,3,4$ to the i[th] rotor of the aircraft can be found by solving* (2-121) *with f and $\boldsymbol{\tau}$ defined in (4-276) and (4-325) respectively*
2. *The closed loop system consisting of* (4-228)*,* (4-331)*,* (4-333)*,* (4-334)*,* (4-297)*,* (4-311)*,* (4-325)*, (4-326),* (4-327) *and* (4-331) *has a unique strong solution and is stochastically practically asymptotically stable.*
3. *All parameter estimates are contained within their limits as specified in* (4-329) and(4-330).
4. *The tracking errors converge to a ball centred at the origin in probability i.e.*

$$P\left\{\lim_{t\to\infty}\sum_{i=1}^{3}\frac{W\big(\mathbf{X}_e(t)\big)-c_3}{\Delta_1\big(\hat{\boldsymbol{v}}_1(t)\big)\big(1+\boldsymbol{\eta}_{1e,i}^2\big)}=0\right\}=1 \tag{4-353}$$

Where

$$\mathbf{X}_e = \mathrm{col}(\boldsymbol{\eta}_{1e}, \boldsymbol{\nu}_{1e}, \tilde{\boldsymbol{\eta}}_1, \tilde{\boldsymbol{v}}_1, \boldsymbol{q}_e, \boldsymbol{\omega}_e) \tag{4-354}$$

$$W(\mathbf{X}_e) = \bar{\gamma}_1\lambda_m(\bar{\mathbf{K}}_1)\boldsymbol{\eta}_{1e,i}^2 + 2\gamma_1\frac{\Delta_1(\hat{\boldsymbol{v}}_1)\big(1+\boldsymbol{\eta}_{1e,i}^2\big)}{3}\left(\frac{\lambda_m(\bar{\mathbf{K}}_2)\boldsymbol{\nu}_{1e}^{\mathrm{T}}\sigma(\boldsymbol{\nu}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}+\frac{\lambda_m(\bar{\mathbf{D}}_1)\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} + \frac{\lambda_m(\mathbf{K}_{01})}{2\gamma_1}\|\tilde{\boldsymbol{\eta}}_1\|^2 + \frac{\lambda_m(\bar{\bar{\mathbf{D}}}_1)}{2\gamma_1}\|\tilde{\boldsymbol{v}}_1\|^2 + \frac{\gamma_2\lambda_m(\bar{\mathbf{K}}_3)}{2\gamma_1}\|\boldsymbol{q}_e\|^2 + \frac{\gamma_3}{2\gamma_1}\frac{\lambda_m(\mathbf{K}_4)\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}\right). \tag{4-355}$$

To prove the above theorem, recall from (4-312)that::

$$V_4 = \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\int_0^{\boldsymbol{\eta}_{1e}}\sigma^{\mathrm{T}}(\mathbf{s})\,d\boldsymbol{s} + 2\gamma_1\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2} + \tfrac{1}{2}(\|\tilde{\boldsymbol{\eta}}_1\|^2+\|\tilde{\boldsymbol{v}}_1\|^2) + \frac{\gamma_2}{2}\|\boldsymbol{q}_e\|^2 + \frac{\gamma_3}{2}\sqrt[2]{1+\|\boldsymbol{\omega}_e\|^4} + \frac{1}{2}\bar{\boldsymbol{\tau}}_{de,Aero}^{\mathrm{T}}\mathbf{\Gamma}^{-1}\bar{\boldsymbol{\tau}}_{de,Aero} + \frac{1}{2\mu_1}\tilde{\delta}_1^2 + \frac{1}{2\mu_2}\tilde{\delta}_2^2 - 2\gamma_1 - \frac{\gamma_3}{2}, \tag{4-356}$$

Recall from (4-255) and (4-313) that the including of $\left(-2\gamma_1-\frac{\gamma_3}{2}\right)$ is to counterect the offset caused by the terms $2\gamma_1\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}$ and $\frac{\gamma_3}{2}\sqrt[2]{1+\|\boldsymbol{\omega}_e\|^4}$ and ensure that V_4 is positive deffinte and only equal to zero when:

$$\|\boldsymbol{\eta}_{1e}\|, \|\boldsymbol{\nu}_{1e}\|, \|\tilde{\boldsymbol{\eta}}_1\|, \|\tilde{\boldsymbol{v}}_1\|, \|\boldsymbol{q}_e\|, \|\boldsymbol{\omega}_e\|, \|\bar{\boldsymbol{\tau}}_{de,Aero}\|, \|\tilde{\delta}_1\|, \|\tilde{\delta}_2\| = 0. \tag{4-357}$$

Furthermore recall the infinite generator of V_4 is:

$$\mathcal{L}V_4 \leq -\bar{\gamma}_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\lambda_m(\bar{\mathbf{K}}_1)\frac{\|\sigma(\boldsymbol{\eta}_{1e})\|^2}{\Delta_1(\hat{\boldsymbol{v}}_1)} - 2\gamma_1\lambda_m(\bar{\mathbf{K}}_2)\frac{\boldsymbol{v}_{1e}^{\mathrm{T}}\sigma(\boldsymbol{\eta}_{1e})}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}$$
$$-2\gamma_1\frac{\lambda_m(\bar{\mathbf{D}}_1)\|\boldsymbol{v}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}} - \lambda_m(\mathbf{K}_{01})\|\tilde{\boldsymbol{\eta}}_1\|^2 - \lambda_m(\bar{\bar{\mathbf{D}}}_1)\|\tilde{\boldsymbol{v}}_1\|^2 - \gamma_2\lambda_m(\bar{\mathbf{K}}_3)\|\boldsymbol{q}_e\|^2$$
$$-\gamma_3\frac{\lambda_m(\mathbf{K}_4)\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}} + \sum_{i=1}^{3}\frac{c_3}{\Delta_1(\hat{\boldsymbol{v}}_1)\left(1+\boldsymbol{\eta}_{1e,i}^2\right)}. \tag{4-358}$$

From (4-231), (4-255), (4-298) and (4-312) we can state that:

$$V_4 = W_4(\mathbf{X}_e) + \frac{1}{2}\bar{\boldsymbol{\tau}}_{de,Aero}^{\mathrm{T}}\boldsymbol{\Gamma}^{-1}\bar{\boldsymbol{\tau}}_{de,Aero} + \frac{1}{2\mu_1}\tilde{\delta}_1^2 + \frac{1}{2\mu_2}\tilde{\delta}_2^2, \tag{4-359}$$

where:

$$W_4(\mathbf{X}_e) = \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\int_0^{\boldsymbol{\eta}_{1e}}\sigma^{\mathrm{T}}(\mathbf{s})\,d\boldsymbol{s} + 2\gamma_1\sqrt{1+\|\boldsymbol{v}_{1e}\|^2} + \tfrac{1}{2}(\|\tilde{\boldsymbol{\eta}}_1\|^2 + \|\tilde{\boldsymbol{v}}_1\|^2) + \frac{\gamma_2}{2}\|\boldsymbol{q}_e\|^2$$
$$+\frac{\gamma_3}{2}\sqrt{1+\|\boldsymbol{\omega}_e\|^4} - 2\gamma_1 - \frac{\gamma_3}{2}. \tag{4-360}$$

Is a class $\mathcal{K}_\infty$ function of $\|\mathbf{X}_e\|$.From (4-355) it is clear $W(\mathbf{X}_e)$ is a class $\mathcal{K}_\infty$ function of $\|\mathbf{X}_e\|$, Hence, we can state that there is a class a class $\mathcal{K}_\infty$ function of $\|\mathbf{X}_e\|$. Therefore, there exist a class $\mathcal{K}_\infty$ functions $\bar{W}(\bar{\mathbf{X}}_e)$ and of $\bar{W}_3(\bar{\mathbf{X}}_e)$ of a vector $\bar{\mathbf{X}}_e$ such that:

$$\bar{W}(\bar{\mathbf{X}}_e) = W(\mathbf{X}_e) - \lambda_m(\mathbf{K}_1)c_2, \tag{4-361}$$
$$\mathcal{L}\bar{V}_4 = \mathcal{L}V_4, \tag{4-362}$$

Where:

$$\bar{V}_4 = \bar{W}_4(\bar{\mathbf{X}}_e) + \frac{1}{2}\bar{\boldsymbol{\tau}}_{de,Aero}^{\mathrm{T}}\boldsymbol{\Gamma}^{-1}\bar{\boldsymbol{\tau}}_{de,Aero} + \frac{1}{2\mu_1}\tilde{\delta}_1^2 + \frac{1}{2\mu_2}\tilde{\delta}_2^2. \tag{4-363}$$

Taking:

$$\mathbf{X}_e = \mathrm{col}(\boldsymbol{\eta}_{1e}, \bar{\boldsymbol{v}}_{1e}, \bar{\boldsymbol{q}}_e, \bar{\boldsymbol{\omega}}_e), \tag{4-364}$$

Substituting (4-361) and (4-362) into (4-358) yields:

$$\mathcal{L}\bar{V}_4 \leq -\sum_{i=1}^{3}\frac{\bar{W}(\bar{\mathbf{X}}_e)}{\Delta_1(\hat{\boldsymbol{v}}_1(t))\left(1+\boldsymbol{\eta}_{1e,i}^2\right)}. \tag{4-365}$$

it holds from Theorem 2.6 that:

$$P\left\{\lim_{t\to\infty}\sum_{i=1}^{3}\frac{\bar{W}(\bar{\mathbf{X}}_e(t))}{\Delta_1(\hat{\boldsymbol{v}}_1(t))\left(1+\boldsymbol{\eta}_{1e,i}^2\right)} = 0\right\} = 1. \tag{4-366}$$

This in turn gives:

$$P\left\{\lim_{t\to\infty}\sum_{i=1}^{3}\bar{W}\big(\bar{\mathbf{X}}_e(t)\big)=0\right\}=1. \tag{4-367}$$

This proves the last result of Theorem 4.1 since:

$$\bar{W}(\bar{\mathbf{X}}_e)=W(\mathbf{X}_e)-\lambda_m(\mathbf{K}_1)c_2, \tag{4-368}$$

and the definition of $\tilde{\boldsymbol{\eta}}_1, \tilde{\boldsymbol{v}}_1, \boldsymbol{\eta}_{1e}, \boldsymbol{v}_{1e}, \boldsymbol{q}_e$ and $\boldsymbol{\omega}_e$ in (4-224), (4-225), (4-227), (4-245) and (4-287). It should be noted that the selection of initial conditions and gains $\mathbf{K}_3, \mathbf{K}_4$, and μ must be chosen through trial and error. However, it is possible to prove exponential stability in probability to a greater ball centred at the solution and in this case provide explicit conditions on all control gains and initial conditions. To do this we must first note that the use of the Modified Rodriquez Parameter implies that a singularity occurs at an angular position of $\pm n2\pi,\ n=1,2,3\ldots$ To address this we must recall that:

$$\boldsymbol{q}(\boldsymbol{\eta}_2)=\begin{bmatrix}\dfrac{\sin(\frac{\phi}{2})\cos(\frac{\theta}{2})\cos(\frac{\psi}{2})-\cos(\frac{\phi}{2})\sin(\frac{\theta}{2})\sin(\frac{\psi}{2})}{1+\cos(\frac{\phi}{2})\cos(\frac{\theta}{2})\cos(\frac{\psi}{2})+\sin(\frac{\phi}{2})\sin(\frac{\theta}{2})\sin(\frac{\psi}{2})}\\ \dfrac{\cos(\frac{\phi}{2})\sin(\frac{\theta}{2})\cos(\frac{\psi}{2})+\sin(\frac{\phi}{2})\cos(\frac{\theta}{2})\sin(\frac{\psi}{2})}{1+\cos(\frac{\phi}{2})\cos(\frac{\theta}{2})\cos(\frac{\psi}{2})+\sin(\frac{\phi}{2})\sin(\frac{\theta}{2})\sin(\frac{\psi}{2})}\\ \dfrac{\cos(\frac{\phi}{2})\cos(\frac{\theta}{2})\sin(\frac{\psi}{2})-\sin(\frac{\phi}{2})\sin(\frac{\theta}{2})\cos(\frac{\psi}{2})}{1+\cos(\frac{\phi}{2})\cos(\frac{\theta}{2})\cos(\frac{\psi}{2})+\sin(\frac{\phi}{2})\sin(\frac{\theta}{2})\sin(\frac{\psi}{2})}\end{bmatrix} \tag{4-369}$$

The MRP experiences a singularity at the following conditions

Euler angle			**MRP**		
ϕ	θ	ψ	q_1	q_2	q_3
0	0	$\pm 2\pi$	Singularity	Singularity	Singularity
0	$\pm 2\pi$	0	Singularity	Singularity	Singularity
$\pm 2\pi$	0	0	Singularity	Singularity	Singularity
2π	2π	2π	Singularity	Singularity	Singularity
2π	2π	-2π	Singularity	Singularity	Singularity
2π	-2π	2π	Singularity	Singularity	Singularity
-2π	2π	2π	Singularity	Singularity	Singularity
2π	-2π	-2π	Singularity	Singularity	Singularity
-2π	2π	-2π	Singularity	Singularity	Singularity
-2π	-2π	2π	Singularity	Singularity	Singularity
-2π	-2π	-2π	Singularity	Singularity	Singularity
π	π	$-\pi$	Singularity	Singularity	Singularity
π	$-\pi$	π	Singularity	Singularity	Singularity
$-\pi$	π	π	Singularity	Singularity	Singularity
$-\pi$	$-\pi$	$-\pi$	Singularity	Singularity	Singularity

Table 4-2: MRP Regions of Singularity.

Therefore, it can be stated:

$$|\boldsymbol{\eta}_2| \le b_3 < 2\pi, \tag{4-370}$$
$$|\boldsymbol{q}(\boldsymbol{\eta}_2)| \le b_4, \tag{4-371}$$
$$|\boldsymbol{q}| \le b_4, \tag{4-372}$$
$$|\boldsymbol{q}_e| \le b_4, \tag{4-373}$$

hence we wish to obtain

$$V_4 \le b_5 \le b_4, \tag{4-374}$$

where b_1, b_2 and b_3 are positive constants. To define the initial conditions, we will first state:

$$\begin{aligned}&\frac{1}{2}\left\|\boldsymbol{\Gamma}_{\boldsymbol{\tau}}^{-\mathbf{1}}\right\|\left\|\overline{\boldsymbol{\tau}}_{de,Aero}\right\|^2 + \frac{1}{2\mu_1}\tilde{\delta}_\tau^2 + \frac{1}{2\mu_2}\tilde{\delta}_2^2 \\ &\quad\le \frac{1}{2}\lambda_m\left(\boldsymbol{\Gamma}_{\boldsymbol{\tau}}^{-\mathbf{1}}\right)(2\varrho_\tau + \xi_\tau)^2 + \frac{1}{2\mu_1}\left(2\Delta_{1,M}^2 + \xi_{1,\delta}\right)^2 + \frac{1}{2\mu_2}\left(2\Delta_{2,M}^2 + \xi_{2,\delta}\right)^2 \coloneqq b_6,\end{aligned} \tag{4-375}$$

where b_4 is a constant. Therefore, we can now state:

$$\begin{aligned}V_4(t_0) \le{}& \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\int_0^{\eta_{1e}}\sigma^{\mathrm{T}}(\mathbf{s})\,d\boldsymbol{s} + 2\gamma_1\sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} + \frac{1}{2}(\|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 + \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2) \\ &+\tfrac{\gamma_2}{2}\|\boldsymbol{q}_e(t_0)\|^2 + \tfrac{\gamma_3}{2}\sqrt[2]{1+\|\boldsymbol{\omega}_e(t_0)\|^4} + b_6 - 2\gamma_1 - \gamma_3 \ \le b_5,\end{aligned} \tag{4-376}$$

therefore, it can be stated:

$$\begin{aligned}&\|\boldsymbol{\eta}_{1e}(t_0)\| + \|\boldsymbol{\nu}_{1e}(t_0)\| + \|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 + \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2 + \|\boldsymbol{q}_e(t_0)\|^2 + \|\boldsymbol{\omega}_e\|^2 \\ &\qquad + \left\|\overline{\boldsymbol{\tau}}_{de,Aero}(t_0)\right\|^2 + \tilde{\delta}_1^2(t_0) + \tilde{\delta}_2^2(t_0) + \frac{b_8}{b_7} \le b_5.\end{aligned} \tag{4-377}$$

Hence we are now able to choose bounds on the initial conditions of $\boldsymbol{\eta}_{1e}(t_0)$, $\boldsymbol{\nu}_{1e}(t_0)$, $\boldsymbol{q}_e(t_0)$, $\boldsymbol{\omega}_e(t_0)$, $\tilde{\boldsymbol{\eta}}_1(t_0)$, $\tilde{\boldsymbol{\nu}}_1(t_0)$. Now before we can provide conditions on the intimal conditions and all control gains we must first recognise that:

$$b_1 < \frac{1+\|\boldsymbol{\nu}_{1e}\|^2}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} < b_2, \qquad \forall\hat{\boldsymbol{\nu}}_1(t) \in \mathbb{R}^3, t \in \mathbb{R} \tag{4-378}$$

where b_5 and b_6 are constant which are dependent on the selection of $\mathbf{K}_1$ and the maximum rate of change of the linear position reference signal, with this in mind it can be stated:

$$\begin{aligned}\mathcal{L}V_4 \le{}& -b_1\sum_{i=1}^{3}\frac{\bar{\gamma}_1\frac{\lambda_m(\mathbf{D}_1)}{2}\lambda_m(\overline{\mathbf{K}}_1)\boldsymbol{\eta}_{1e,i}^2 - c_3}{(1+\|\boldsymbol{\nu}_{1e}\|^2)(1+\boldsymbol{\eta}_{1e,i}^2)} - 2\gamma_1\lambda_m(\overline{\mathbf{K}}_2)\frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}}\sigma(\boldsymbol{\eta}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2}} \\ &-2\gamma_1\lambda_m(\overline{\mathbf{D}}_1)\frac{\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2}} - \lambda_m(\mathbf{K}_{01})\|\tilde{\boldsymbol{\eta}}_1\|^2 - \lambda_m(\overline{\mathbf{D}}_1)\|\tilde{\boldsymbol{\nu}}_1\|^2 \\ &-\gamma_2\lambda_m(\overline{\mathbf{K}}_3)\|\boldsymbol{q}_e\|^2 - \gamma_3\lambda_m(\mathbf{K}_4)\frac{\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}.\end{aligned} \tag{4-379}$$

By adding and subtracting the right hand side and left hand side of the inequality of (4-375) respectively and dividing both by $(1+\|\boldsymbol{\nu}_{1e}\|^2)(1+\boldsymbol{\eta}_{1e,i}^2)/b_5$ to the right side of (4-379) we can now state:

$$\mathcal{L}V_4 \leq -b_1\sum_{i=1}^{3}\frac{\bar{\gamma}_1\frac{\lambda_m(\mathbf{D}_1)}{2}\lambda_m(\overline{\mathbf{K}}_1)\boldsymbol{\eta}_{1e,i}^2+\frac{1}{3}\|\boldsymbol{\Gamma}_{\boldsymbol{\tau}}^{-1}\|\left\|\bar{\boldsymbol{\tau}}_{de,Aero}\right\|^2+\frac{1}{6}\sum_{i=1}^{2}\frac{\tilde{\delta}_i^2}{\mu_i}-c_3-\frac{b_6}{3}}{(1+\|\boldsymbol{\nu}_{1e}\|^2)(1+\boldsymbol{\eta}_{1e,i}^2)} \\ -2\gamma_1\lambda_m(\overline{\mathbf{K}}_2)\frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}}\sigma(\boldsymbol{\eta}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2}}-2\gamma_1\lambda_m(\overline{\mathbf{D}}_1)\frac{\|\boldsymbol{\nu}_{1e}\|^2}{\sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2}}-\lambda_m(\mathbf{K}_{01})\|\widetilde{\boldsymbol{\eta}}_1\|^2 \\ -\lambda_m(\overline{\overline{\mathbf{D}}}_1)\|\tilde{\boldsymbol{\nu}}_1\|^2-\gamma_2\lambda_m(\overline{\mathbf{K}}_3)\|\boldsymbol{q}_e\|^2-\gamma_3\lambda_m(\mathbf{K}_4)\frac{\|\boldsymbol{\omega}_e\|^4}{(1+\|\boldsymbol{\omega}_e\|^4)^{0.5}}. \tag{4-380}$$

therefore, using (3-147), (4-376) and (4-380) it can be stated that:

$$\mathcal{L}V_4 \leq -b_7V_4+b_8. \tag{4-381}$$

where:

$$b_7=\min\left(\frac{2\gamma_1\,\lambda_m(\overline{\mathbf{D}}_1)}{2\gamma_1},\frac{\lambda_m(\mathbf{K}_{01})}{\frac{1}{2}},\frac{\lambda_m(\overline{\overline{\mathbf{D}}}_1)}{\frac{1}{2}},\frac{\gamma_2\,\lambda_m(\overline{\mathbf{K}}_3)}{\frac{\gamma_2}{2}},\frac{\gamma_3\,\lambda_m(\mathbf{K}_4)}{\frac{\gamma_3}{2}},\right. \\ \frac{b_1\sum_{i=1}^{3}\frac{\bar{\gamma}_1\frac{\lambda_m(\mathbf{D}_1)}{2}\lambda_m(\overline{\mathbf{K}}_1)}{3(1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2)\left(1+\boldsymbol{\eta}_{1e,i}^2(t_0)\right)}}{\frac{\gamma_1\lambda_m(\mathbf{D}_1)}{2}},\frac{\sum_{i=1}^{3}\frac{b_1\frac{1}{6}\lambda_m(\boldsymbol{\Gamma}^{-1})}{(1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2)\left(1+\boldsymbol{\eta}_{1e,i}^2(t_0)\right)}}{\frac{1}{2}\|\boldsymbol{\Gamma}^{-1}\|}, \\ \left.\frac{\sum_{i=1}^{3}\frac{b_1\frac{1}{6\mu_1}}{(1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2)\left(1+\boldsymbol{\eta}_{1e,i}^2(t_0)\right)}}{\frac{1}{2\mu_1}},\frac{\sum_{i=1}^{3}\frac{b_1\frac{1}{6\mu_2}}{(1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2)\left(1+\boldsymbol{\eta}_{1e,i}^2(t_0)\right)}}{\frac{1}{2\mu_2}}\right), \tag{4-382}$$

$$b_8=\max\left(1,\min\left(\sum_{i=1}^{3}\frac{b_1\left(c_3+\frac{b_6}{3}\right)}{(1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2)\left(1+\boldsymbol{\eta}_{1e,i}^2(t_0)\right)}\right)\right). \tag{4-383}$$

Hence with the above equation we are able to choose all initial conditions as well as the controller gains $\mathbf{K}_3,\mathbf{K}_4,\boldsymbol{\Gamma}_{\tau}^{-1}$ and μ, we are able to state:

$$V_4(t)\leq\left(V_4(t_0)-\frac{b_8}{b_7}\right)\mathrm{e}^{-b_7(t-t_0)}+\frac{b_8}{b_7}. \tag{4-384}$$

Which in turn implies that:

$$\int_0^{\boldsymbol{\eta}_{1e(t)}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s} + \sqrt{1+\|\boldsymbol{\nu}_{1e}(t)\|^2} + \|\tilde{\boldsymbol{\eta}}_1(t)\|^2 + \|\tilde{\boldsymbol{\nu}}_1(t)\|^2 + \|\boldsymbol{q}_e(t)\|^2 + \sqrt[2]{1+\|\boldsymbol{\omega}_e(t)\|^4}$$
$$+\left(2\Delta_{2,M}^2+\xi_{1,\delta}\right)^2 + \left(2\Delta_{3,M}^2+\xi_{3,\delta}\right)^2 - 2$$
$$\leq \int_0^{\boldsymbol{\eta}_{1e(t_0)}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s} + \sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} + \|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 + \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2 + \|\boldsymbol{q}_e(t_0)\|^2$$
$$+\sqrt[2]{1+\|\boldsymbol{\omega}_e(t_0)\|^4} + \frac{1}{2}\lambda_m(\boldsymbol{\Gamma}^{-1})\left(2\boldsymbol{\tau}_{Aero_M}+\xi_{\tau_{Aero}}\right)^2 + \left(2\Delta_{2,M}^2+\xi_{1,\delta}\right)^2$$
$$+\left(2\Delta_{3,M}^2+\xi_{2,\delta}\right)^2 + \frac{b_8}{b_7} - 2 \leq b_5,, \tag{4-385}$$

We will first calculate the immediate upper bounds of

$$|\boldsymbol{\eta}_{1e}| \leq \boldsymbol{\eta}_{1e}^M, \tag{4-386}$$
$$|\boldsymbol{\nu}_{1e}| \leq \boldsymbol{\nu}_{1e}^M, \tag{4-387}$$
$$|\boldsymbol{q}_e| \leq \boldsymbol{q}_e^M, \tag{4-388}$$
$$|\boldsymbol{\omega}_{1e}| \leq \boldsymbol{\omega}_e^M, \tag{4-389}$$
$$|\tilde{\boldsymbol{\eta}}_1| \leq \tilde{\boldsymbol{\eta}}_1^M, \tag{4-390}$$
$$|\tilde{\boldsymbol{\nu}}_1| \leq \tilde{\boldsymbol{\nu}}_1^M, \tag{4-391}$$

Due to the design processes it can be stated the following conditions

$$\|\boldsymbol{\alpha}_1\| \leq \lambda_M(\mathbf{K}_1)(2+\varrho_1), \tag{4-392}$$
$$\|\boldsymbol{\alpha}_2\| \leq \boldsymbol{\alpha}_2^{\mathrm{M}}, \tag{4-393}$$
$$\left\|\boldsymbol{\alpha}_q\right\| \leq \boldsymbol{\alpha}_q^{\mathrm{M}}. \tag{4-394}$$

Furthermore, under condition (4-386)-(4-391), (4-371) and (4-374) it can be now stated that:

$$\left\|\boldsymbol{\alpha}_q\right\| \leq \boldsymbol{\alpha}_{\boldsymbol{\omega}}^{\mathrm{M}}. \tag{4-395}$$

Therefore, we can choose the values of $\mathbf{K}_3$ by stating:

$$V_4(t_0) \leq \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \int_0^{\boldsymbol{\eta}_{1e(t_0)}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s} + 2\gamma_1\sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} + \frac{1}{2}\|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2$$
$$+\frac{1}{2}\|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2 + \frac{\gamma_2}{2}\|\boldsymbol{q}_e(t_0)\|^2 + \frac{\gamma_3}{2}\sqrt[2]{1+\|\boldsymbol{\omega}_e(t_0)\|^4}$$
$$+\frac{1}{2\mu_1}\left(2\Delta_{2,M}^2+\xi_{1,\delta}\right)^2 + \frac{1}{2\mu_2}\left(2\Delta_{3,M}^2+\xi_{3,\delta}\right)^2 - 2\gamma_1 - \frac{\gamma_3}{2} \leq b_5, \tag{4-396}$$

$$V_4(t_0) \leq \int_0^{\boldsymbol{\eta}_{1e(t_0)}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s} + \sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} + (\|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 + \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2)$$
$$+\|\boldsymbol{q}_e(t_0)\|^2 + \sqrt[2]{1+\|\boldsymbol{\omega}_e(t_0)\|^4} + \left(2\Delta_{2,M}^2+\xi_{1,\delta}\right)^2 + \left(2\Delta_{3,M}^2+\xi_{2,\delta}\right)^2 - 2 + \frac{b_8}{b_7}$$
$$\leq b_5, \tag{4-397}$$

$$\sqrt[2]{1+\|\boldsymbol{\omega}_e(t_0)\|^4} \leq b_5 - \left(\int_0^{\boldsymbol{\eta}_{1e(t_0)}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s} + \sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} + \|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 + \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2\right.$$
$$+\|\boldsymbol{q}_e(t_0)\|^2 + \frac{1}{2}\lambda_m(\boldsymbol{\Gamma}_{\boldsymbol{\tau}}^{-1})(2\varrho_\tau+\xi_\tau)^2 + \left(2\Delta_{2,M}^2+\xi_{1,\delta}\right)^2$$
$$\left.++\left(2\Delta_{3,M}^2+\xi_{2,\delta}\right)^2 - 2 + \frac{b_8}{b_7}\right), \tag{4-398}$$

$$\|\boldsymbol{\omega}_e(t_0)\| \leq \left(\left(\left(b_5 - \left(\int_0^{\boldsymbol{\eta}_{1e(t_0)}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s} + \sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} + \|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 + \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2 + \|\boldsymbol{q}_e(t_0)\|^2 + \frac{1}{2}\lambda_m(\boldsymbol{\Gamma}_{\boldsymbol{\tau}}^{-1})(2\varrho_\tau + \xi_\tau)^2 + \left(2\Delta_{2,M}^2 + \xi_{1,\delta}\right)^2 + \left(2\Delta_{3,M}^2 + \xi_{2,\delta}\right)^2 - 2 + \frac{b_8}{b_7}\right)\right)^2 - 1\right)^{0.25}, \tag{4-399}$$

We need to choose the initial conditions for $\boldsymbol{q}_e(t_0)$ such that:

$$b_5 - \left(\int_0^{\boldsymbol{\eta}_{1e(t_0)}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s} + \|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 + \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2 + \sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} + \|\boldsymbol{q}_e(t_0)\|^2 + \frac{1}{2}\lambda_m(\boldsymbol{\Gamma}_{\boldsymbol{\tau}}^{-1})(2\varrho_\tau + \xi_\tau)^2 + \left(2\Delta_{2,M}^2 + \xi_{1,\delta}\right)^2 + \left(2\Delta_{3,M}^2 + \xi_{2,\delta}\right)^2 - 2 + \frac{b_8}{b_7}\right) > 1 \tag{4-400}$$

$$\|\boldsymbol{q}_e(t_0)\|^2 < b_5 - 1 - \int_0^{\boldsymbol{\eta}_{1e(t_0)}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s} - \|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 - \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2 - \sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} - \frac{1}{2}\lambda_m(\boldsymbol{\Gamma}_{\boldsymbol{\tau}}^{-1})(2\varrho_\tau + \xi_\tau)^2 - \left(2\Delta_{2,M}^2 + \xi_{1,\delta}\right)^2 - \left(2\Delta_{3,M}^2 + \xi_{2,\delta}\right)^2 + 2 - \frac{b_8}{b_7}, \tag{4-401}$$

$$\|\boldsymbol{q}_e(t_0)\| < \left(b_5 - 1 - \int_0^{\boldsymbol{\eta}_{1e(t_0)}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s} - \|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 - \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2 - \sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} - \left(\frac{1}{2}\lambda_m(\boldsymbol{\Gamma}_{\boldsymbol{\tau}}^{-1})(2\varrho_\tau + \xi_\tau)^2 + \left(2\Delta_{2,M}^2 + \xi_{1,\delta}\right)^2 + \left(2\Delta_{3,M}^2 + \xi_{2,\delta}\right)^2 - 2 + \frac{b_8}{b_7}\right)\right)^{0.5} \tag{4-402}$$

holds true, now we can choose the values of the control gains matrix $\mathbf{K}_3$, by noting that:

$$\sup_{t\in R^n}\|\mathbf{L}(t)\| = \varrho_9, \tag{4-403}$$

$$\sup_{t\in R^n}\|\boldsymbol{\Omega}(t)\| = \varrho_{10}, \tag{4-404}$$

$$\sup_{t\in R^n}\left\|\frac{2\gamma_1\|\boldsymbol{\Omega}(t)\|\mathbf{L}(t)\mathbf{G}_1^{\mathrm{T}}(t)\boldsymbol{\nu}_{1e}(t)}{\gamma_2\sqrt{1+\|\boldsymbol{\nu}_{1e}(t)\|^2}}\right\| \leq \frac{2\gamma_1\varrho_9\varrho_{10}}{\gamma_2 m}\left(1 + 0.46\left(\mathbf{K}_{1,11} + \mathbf{K}_{1,22} + \mathbf{K}_{1,33}\right)\right), \tag{4-405}$$

$$\sup_{t\in R^n}\left\|\mathbf{R}_2\left(\boldsymbol{\alpha}_q(t)\right)\right\| \leq 4, \tag{4-406}$$

$$\sup_{t\in R^n}\left\|\mathbf{T}\left(\boldsymbol{\alpha}_2(t)\right)^{-1}\right\| \leq 2, \tag{4-407}$$

$$\sup_{t\in R^n}\|\mathbf{A}_{2,d}(t)\| \leq \sup_{t\in R^n}\left\{\frac{1}{\cos^2\left(\alpha_\phi(t)\right)\|\boldsymbol{\Omega}(t)\|^2} + \cos^2\left(\alpha_\theta(t)\right)\frac{1}{\Omega_3^2(t)}\right\} := \|\mathbf{A}_{2,d}\|^{\mathrm{M}}, \tag{4-408}$$

$$\sup_{t\in R^n}\left\|\boldsymbol{\Omega}_d(t) + \frac{\partial\boldsymbol{\Omega}(t)}{\partial\boldsymbol{\eta}_{1e}(t)}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}(t)\boldsymbol{\nu}_{1e}(t)}{\sqrt{1+\|\boldsymbol{\nu}_{1e}(t)\|^2}}\right\| \leq \left\|\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right\|^{\mathrm{M}}, \tag{4-409}$$

$$\sup_{t\in R^n}\|\mathbf{A}_{1,d}(t)\boldsymbol{\Omega}(t)\| := \|\mathbf{A}_{2,d}\boldsymbol{\Omega}\|^{\mathrm{M}}, \tag{4-410}$$

$$\sup_{t\in R^n}\left\|\left(\mathbf{A}_{1,d}(t)\boldsymbol{\Omega}(t) + \boldsymbol{e}_3\right)\dot{\alpha}_\psi(t)\right\| \leq \left(\|\mathbf{A}_{2,d}\boldsymbol{\Omega}\|^{\mathrm{M}} + 1\right)\varrho_6, \tag{4-411}$$

for all $t \geq t_0 \geq 0$.

We can sate that:

$$\alpha_p = \begin{bmatrix}1\\0\\0\end{bmatrix}^{\mathrm{T}} \mathbf{R}_2^{-1}(\boldsymbol{q})\left(\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) + \left(\mathbf{A}_{1,d}\boldsymbol{\Omega} + \boldsymbol{e}_3\right)\dot{\alpha}_\psi\right) - \mathbf{K}_{3,11}\left(q_1 - \alpha_{q_1}\right) - \begin{bmatrix}1\\0\\0\end{bmatrix}^{\mathrm{T}}\frac{2\gamma_1\|\boldsymbol{\Omega}\|\mathbf{L}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\gamma_2 m\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right), \tag{4-412}$$

$$\alpha_q = \begin{bmatrix}0\\1\\0\end{bmatrix}^{\mathrm{T}} \mathbf{R}_2^{-1}(\boldsymbol{q})\left(\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) + \left(\mathbf{A}_{1,d}\boldsymbol{\Omega} + \boldsymbol{e}_3\right)\dot{\alpha}_\psi\right) - \mathbf{K}_{3,22}\left(q_{22} - \alpha_{q_{22}}\right) - \begin{bmatrix}0\\1\\0\end{bmatrix}^{\mathrm{T}}\frac{2\gamma_1\|\boldsymbol{\Omega}\|\mathbf{L}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\gamma_2 m\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right), \tag{4-413}$$

$$\alpha_r = \begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}} \mathbf{R}_2^{-1}(\boldsymbol{q})\left(\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) + \left(\mathbf{A}_{1,d}\boldsymbol{\Omega} + \boldsymbol{e}_3\right)\dot{\alpha}_\psi\right) - \mathbf{K}_{3,33}\left(q_3 - \alpha_{q_3}\right) - \begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}}\frac{2\gamma_1\|\boldsymbol{\Omega}\|\mathbf{L}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\gamma_2 m\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right), \tag{4-414}$$

and therefore state that:

$$\|\alpha_p\| \le \alpha_p^{\mathrm{M}}, \qquad \|\alpha_q\| \le \alpha_q^{\mathrm{M}}, \qquad \|\alpha_r\| \le \alpha_r^{\mathrm{M}}, \qquad \|\boldsymbol{\alpha}_{\boldsymbol{\omega}}\| \le \alpha_{\boldsymbol{\omega}}^{\mathrm{M}}. \tag{4-415}$$

And therefore it can be stated that:

$$\|\boldsymbol{\omega}(t_0) - \boldsymbol{\alpha}_{\boldsymbol{\omega}}(t_0)\| \le \left(\left(b_5 - \left(\int_0^{\boldsymbol{\eta}_{1e}(t_0)} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s} + \|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 + \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2 + \sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} + \|\boldsymbol{q}_e(t_0)\|^2 + + \frac{1}{2}\lambda_m\left(\boldsymbol{\Gamma}_{\boldsymbol{\tau}}^{-\mathbf{1}}\right)(2\varrho_\tau + \xi_\tau)^2 + \left(2\Delta_{2,M}^2 + \xi_{1,\delta}\right)^2 + \left(2\Delta_{3,M}^2 + \xi_{2,\delta}\right)^2 - 2 + \frac{b_8}{b_7}\right)\right)^2 - 1\right)^{0.25}, \tag{4-416}$$

$$\|\boldsymbol{\alpha}_{\boldsymbol{\omega}}(t_0)\| \le \left(\left(b_5 - \left(\int_0^{\boldsymbol{\eta}_{1e}(t_0)} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s} + \|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 + \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2 + \sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} + \|\boldsymbol{q}_e(t_0)\|^2 + \frac{1}{2}\lambda_m\left(\boldsymbol{\Gamma}_{\boldsymbol{\tau}}^{-\mathbf{1}}\right)(2\varrho_\tau + \xi_\tau)^2 + \left(2\Delta_{2,M}^2 + \xi_{1,\delta}\right)^2 + \left(2\Delta_{3,M}^2 + \xi_{2,\delta}\right)^2 - 2 + \frac{b_8}{b_7}\right)\right)^2 - 1\right)^{0.25} - \|\boldsymbol{\omega}(t_0)\| > 0, \tag{4-417}$$

$$0 < \left\| \mathbf{R}_2^{-1}(\boldsymbol{q}(t_0)) \left(\mathbf{R}_2\left(\boldsymbol{\alpha}_q(t_0)\right) \mathbf{T}(\boldsymbol{\alpha}_2(t_0))^{-1} \left(\mathbf{A}_{2,d}(t_0) \left(\boldsymbol{\Omega}_d(t_0) + \frac{\partial \boldsymbol{\Omega}(t_0)}{\partial \boldsymbol{\eta}_{1e(t_0)}} \frac{2\gamma_1 \mathbf{G}_1^{\mathrm{T}}(t_0)\boldsymbol{\nu}_{1e}(t_0)}{\sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2}} \right) + \left(\mathbf{A}_{1,d}(t_0)\boldsymbol{\Omega}(t_0) + \boldsymbol{e}_3\right)\dot{\alpha}_\psi(t_0) \right) - \mathbf{K}_3\left(\boldsymbol{q}(t_0) - \boldsymbol{\alpha}_q(t_0)\right) - \frac{2\gamma_1 \|\boldsymbol{\Omega}(t_0)\| \mathbf{L}(t_0)\mathbf{G}_1^{\mathrm{T}}(t_0)\boldsymbol{\nu}_{1e}(t_0)}{\gamma_2 m\sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2}} \right) \right\| \leq \left(\left(b_5 - \left(\int_0^{\boldsymbol{\eta}_{1e(t_0)}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s} + \|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 + \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2 + \sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} + \|\boldsymbol{q}_e(t_0)\|^2 + \frac{1}{2}\lambda_m(\boldsymbol{\Gamma}_{\boldsymbol{\tau}}^{-1})(2\varrho_\tau + \xi_\tau)^2 + \left(2\Delta_{2,M}^2 + \xi_{1,\delta}\right)^2 + \left(2\Delta_{3,M}^2 + \xi_{2,\delta}\right)^2 - 2 + \frac{b_8}{b_7} \right) \right)^2 - 1 \right)^{0.25} - \|\boldsymbol{\omega}(t_0)\|, \tag{4-418}$$

$$0 < \left\| \mathbf{K}_3\left(\boldsymbol{q}(t_0) - \boldsymbol{\alpha}_q(t_0)\right) \right\| \leq \left(\left(b_5 - \int_0^{\boldsymbol{\eta}_{1e(t_0)}} \sigma^{\mathrm{T}}(\mathbf{s})\, d\boldsymbol{s} - \|\tilde{\boldsymbol{\eta}}_1(t_0)\|^2 - \|\tilde{\boldsymbol{\nu}}_1(t_0)\|^2 - \sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2} - \|\boldsymbol{q}_e(t_0)\|^2 - \left(2\Delta_{\boldsymbol{\tau},M}^2 + \xi_\delta\right)^2 + 2 - \frac{b_8}{b_7} \right)^2 - 1 \right)^{0.25} - \left\| \mathbf{R}_2^{-1}(\boldsymbol{q}(t_0)) \left(\mathbf{R}_2\left(\boldsymbol{\alpha}_q(t_0)\right) \mathbf{T}(\boldsymbol{\alpha}_2(t_0))^{-1} \left(\mathbf{A}_{2,d}(t_0) \left(\boldsymbol{\Omega}_d(t_0) + \frac{\partial \boldsymbol{\Omega}(t_0)}{\partial \boldsymbol{\eta}_{1e(t_0)}} \frac{2\gamma_1 \mathbf{G}_1^{\mathrm{T}}(t_0)\boldsymbol{\nu}_{1e}(t_0)}{\sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2}} \right) + \left(\mathbf{A}_{1,d}(t_0)\boldsymbol{\Omega}(t_0) + \boldsymbol{e}_3\right)\dot{\alpha}_\psi(t_0) \right) - \frac{2\gamma_1 \|\boldsymbol{\Omega}(t_0)\| \mathbf{L}(t_0)\mathbf{G}_1^{\mathrm{T}}(t_0)\boldsymbol{\nu}_{1e}(t_0)}{\gamma_2 m\sqrt{1+\|\boldsymbol{\nu}_{1e}(t_0)\|^2}} \right) \right\| - \|\boldsymbol{\omega}(t_0)\|, \tag{4-419}$$

It is now possible to consider the case for actuator saturation and therefore provide bounds on the gains $\mathbf{K}_4, \boldsymbol{\Gamma}_\tau^{-1}$ and μ, we are able to state first we will state that:

$$\sup_{t \in R^n} \|\boldsymbol{\tau}(t)\| = \boldsymbol{\tau}^M, \qquad \forall t \geq t_0 \geq 0. \tag{4-420}$$

Now we can make the conditions on the gains $\mathbf{K}_4, \mathbf{\Gamma}, \mu_1$ and μ_2, we are able to state:

$$\|\boldsymbol{\tau}\| \leq \left\| (\mathbf{I}_A + \mathbf{I}_H) \left[-(\mathbf{I}_A + \mathbf{I}_H)^{-1} (\mathbf{D}_2 \boldsymbol{\alpha}_\omega - \mathbf{C}_B(\boldsymbol{\omega})\boldsymbol{\omega} - \mathbf{C}_{A2}(\boldsymbol{\omega})\boldsymbol{\omega}) + \boldsymbol{\alpha}_{\omega,d} - \mathbf{K}_4 \boldsymbol{\omega}_e \right.\right.$$
$$- \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\eta}_{1e}} \left(\frac{2\gamma_1 \boldsymbol{\nu}_{1e}^{\mathrm{T}} \mathbf{G}_1}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}} + \gamma_2 \boldsymbol{q}_e^T \mathbf{R}_2(\boldsymbol{\alpha}_q) \mathbf{T}(\boldsymbol{\alpha}_2)^{-1} \mathbf{A}_{2,d} \frac{\partial \boldsymbol{\Omega}}{\partial \hat{\boldsymbol{v}}_1} \right)^{\mathrm{T}}$$
$$- \left(\frac{\gamma_2}{\gamma_3} \left(\left(\frac{1}{4\varepsilon_5} \right)^2 \frac{\varepsilon_6 \|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1 + \|\boldsymbol{\omega}_e\|^4)^{2.25}} + \frac{\boldsymbol{q}_{qe}^T \mathbf{R}_2(\boldsymbol{q}) \boldsymbol{\omega}_e}{(1 + \|\boldsymbol{\omega}_e\|^4)^{1/4}} \right) \right.$$
$$\left.\left.\left. + \gamma_3 \frac{9}{4} \left(\varepsilon_5 \frac{\|(\mathbf{I}_A + \mathbf{I}_H)^{-1} \boldsymbol{\theta}\|^4}{(1 + \|\omega_e\|^4)^{0.5}} \hat{\delta}_1 + \varepsilon_6 \|(\mathbf{I}_A + \mathbf{I}_H)^{-1}\|^4 \hat{\delta}_2 \right) \right) \boldsymbol{\omega}_e \right] \hat{\boldsymbol{\tau}}_{Aero} \right\| \leq \boldsymbol{\tau}^M$$
$$:= \left\| \begin{bmatrix} U_2^M \\ U_3^M \\ U_4^M \end{bmatrix} \right\| \tag{4-421}$$

Recall from (4-309) that:

$$\boldsymbol{\alpha}_{\omega,d} = \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \mathbf{q}} \dot{\mathbf{q}} - \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \hat{\boldsymbol{v}}_1} \left(\frac{f}{m} \mathbf{R}(\boldsymbol{\eta}_2) \boldsymbol{e}_3 - \mathbf{D}_1 \hat{\boldsymbol{v}}_1 - g \boldsymbol{e}_3 + \boldsymbol{h}_2 + \boldsymbol{h}_3 + \boldsymbol{h}_4 + \boldsymbol{h}_5 \right) + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \alpha_\psi} \dot{\alpha}_\psi$$
$$+ \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \dot{\alpha}_\psi} \ddot{\alpha}_\psi + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\eta}_{1e}} (\hat{\boldsymbol{v}}_1 - \dot{\boldsymbol{\eta}}_{1d}.) + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\eta}_{1d}} \dot{\boldsymbol{\eta}}_{1d} + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \dot{\boldsymbol{\eta}}_{1d}} \ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \ddot{\boldsymbol{\eta}}_{1d}} \dddot{\boldsymbol{\eta}}_{1d} + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \dddot{\boldsymbol{\eta}}_{1d}} \ddddot{\boldsymbol{\eta}}_{1d}. \tag{4-422}$$

from Appendix E in addition with the conditions outlined in Assumption 4.5 to Assumption 4.9 and condition (4-372) it can be seen that:

$$\sup_{t \in R^n} \left\| \frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \boldsymbol{q}(t)} \right\| \leq \varrho_{11}, \quad \sup_{t \in R^n} \left\| \frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \hat{\boldsymbol{v}}_1(t)} \dot{\hat{\boldsymbol{v}}}_1(t) \right\| \leq \varrho_{12}, \quad \sup_{t \in R^n} \left\| \frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \alpha_\psi(t)} \dot{\alpha}_\psi(t) \right\| \leq \varrho_{13},$$
$$\sup_{t \in R^n} \left\| \frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \dot{\alpha}_\psi(t)} \ddot{\alpha}_\psi(t) \right\| \leq \varrho_{14}, \quad \sup_{t \in R^n} \left\| \frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \boldsymbol{\eta}_{1e}(t)} (\hat{\boldsymbol{v}}_1(t) - \dot{\boldsymbol{\eta}}_{1d}(t)) \right\| \leq \varrho_{15},$$
$$\sup_{t \in R^n} \left\| \frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \boldsymbol{\eta}_{1d}(t)} \dot{\boldsymbol{\eta}}_{1d}(t) \right\| \leq \varrho_{16}, \quad \sup_{t \in R^n} \left\| \frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \dot{\boldsymbol{\eta}}_{1d}(t)} \ddot{\boldsymbol{\eta}}_{1d}(t) \right\| \leq \varrho_{17},$$
$$\sup_{t \in R^n} \left\| \frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \ddot{\boldsymbol{\eta}}_{1d}(t)} \dddot{\boldsymbol{\eta}}_{1d}(t) \right\| \leq \varrho_{18}, \quad \sup_{t \in R^n} \left\| \frac{\partial \boldsymbol{\alpha}_\omega(t)}{\partial \dddot{\boldsymbol{\eta}}_{1d}(t)} \ddddot{\boldsymbol{\eta}}_{1d}(t) \right\| \leq \varrho_{19},. \tag{4-423}$$

We can therefore state that:

$$\sup_{t \in R^n} \|\boldsymbol{\alpha}_{\omega,d}(t)\| \leq \varrho_{20}, \tag{4-424}$$

$$\sup_{t \in R^n} \|\boldsymbol{\alpha}_{\omega,d,p}(t)\| \leq \varrho_{21}, \quad \sup_{t \in R^n} \|\boldsymbol{\alpha}_{\omega,d,q}(t)\| \leq \varrho_{22}, \quad \sup_{t \in R^n} \|\boldsymbol{\alpha}_{\omega,d,r}(t)\| \leq \varrho_{23}, \tag{4-425}$$

where:

$$\boldsymbol{\alpha}_{\omega,d} = [\boldsymbol{\alpha}_{\omega,d,p} \quad \boldsymbol{\alpha}_{\omega,d,q} \quad \boldsymbol{\alpha}_{\omega,d,r}]. \tag{4-426}$$

This gives:

$$\left\|\frac{I_z + I_{z,A} - I_y - I_{y,A}}{I_x + I_{x,A}}\right\| q^{\boldsymbol{M}} r^M + \varrho_{21} + \frac{\gamma_2}{\gamma_3}\|\boldsymbol{q}_e^T \mathbf{R}_2(\boldsymbol{q})\| p_e^M$$
$$+ \left\|\frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\eta}_{1e}}\right\| \Bigg(2\gamma_1 \left(1 + 0.46(\mathbf{K}_{1,11} + \mathbf{K}_{1,22} + \mathbf{K}_{1,33})\right)$$
$$+ \gamma_2 \left\|\frac{\partial \boldsymbol{\Omega}}{\partial \hat{\boldsymbol{v}}_1}\right\| \|\mathbf{A}_{2,d}^{\mathrm{T}}\| \|\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\| \mathbf{R}_2^T(\boldsymbol{\alpha}_q)\boldsymbol{q}_e\Bigg)$$
$$+ \Bigg(\Bigg(\mathbf{K}_{4,11} - \frac{\gamma_2}{\gamma_3}\left(\frac{1}{4\varepsilon_5}\right)^2 \frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1 + \|p_e^M\|^4)^{2.25}}$$
$$- \gamma_3 \frac{9}{4}\left(\varepsilon_5 \frac{\|(\mathbf{I}_A + \mathbf{I}_H)^{-1}\boldsymbol{\theta}\|^4}{(1 + \|\omega_e\|^4)^{0.5}} \hat{\delta}_1^{\mathrm{M}} + \varepsilon_6 \|(\mathbf{I}_A + \mathbf{I}_H)^{-1}\|^4 \hat{\delta}_2^{\mathrm{M}}\right)\Bigg) p_e^M + \frac{\hat{\boldsymbol{\tau}}_p^M}{I_x + I_{x,A}}$$
$$\leq \frac{U_2^M}{I_x + I_{x,A}}, \tag{4-427}$$

$$\left\|\frac{I_x + I_{x,A} - I_z - I_{z,A}}{I_y + I_{y,A}}\right\| r^M p^M + \varrho_{22} + \frac{1}{\gamma_2}\|\boldsymbol{q}_e^T \mathbf{R}_2(\boldsymbol{q})\| q_e^M$$
$$+ \left\|\frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\eta}_{1e}}\right\| \Bigg(2\gamma_1 \left(1 + 0.46(\mathbf{K}_{1,11} + \mathbf{K}_{1,22} + \mathbf{K}_{1,33})\right)$$
$$+ \gamma_2 \left\|\frac{\partial \boldsymbol{\Omega}}{\partial \hat{\boldsymbol{v}}_1}\right\| \|\mathbf{A}_{2,d}^{\mathrm{T}}\| \|\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\| \mathbf{R}_2^T(\boldsymbol{\alpha}_q)\boldsymbol{q}_e\Bigg)$$
$$+ \Bigg(\Bigg(\mathbf{K}_{4,11} - \frac{\gamma_2}{\gamma_3}\left(\frac{1}{4\varepsilon_5}\right)^2 \frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1 + \|p_e^M\|^4)^{2.25}}$$
$$- \gamma_3 \frac{9}{4}\left(\varepsilon_5 \frac{\|(\mathbf{I}_A + \mathbf{I}_H)^{-1}\boldsymbol{\theta}\|^4}{(1 + \|\omega_e\|^4)^{0.5}} \hat{\delta}_1^{\mathrm{M}} + \varepsilon_6 \|(\mathbf{I}_A + \mathbf{I}_H)^{-1}\|^4 \hat{\delta}_2^{\mathrm{M}}\right)\Bigg) q_e^M + \frac{\hat{\boldsymbol{\tau}}_q^M}{I_y + I_{y,A}}$$
$$\leq \frac{U_3^M}{I_y + I_{y,A}}, \tag{4-428}$$

$$\left\|\frac{I_y + I_{y,A} - I_x - I_{x,A}}{I_z + I_{z,A}}\right\| p^M q^M + \varrho_{23} + \frac{1}{\gamma_2}\|\boldsymbol{q}_e^T \mathbf{R}_2(\boldsymbol{q})\| r_e^M$$
$$+ \left\|\frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\eta}_{1e}}\right\| \Bigg(2\gamma_1 \left(1 + 0.46(\mathbf{K}_{1,11} + \mathbf{K}_{1,22} + \mathbf{K}_{1,33})\right)$$
$$+ \gamma_2 \left\|\frac{\partial \boldsymbol{\Omega}}{\partial \hat{\boldsymbol{v}}_1}\right\| \|\mathbf{A}_{2,d}^{\mathrm{T}}\| \|\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\| \mathbf{R}_2^T(\boldsymbol{\alpha}_q)\boldsymbol{q}_e\Bigg)$$
$$+ \Bigg(\Bigg(\mathbf{K}_{4,11} - \frac{\gamma_2}{\gamma_3}\left(\frac{1}{4\varepsilon_5}\right)^2 \frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1 + \|p_e^M\|^4)^{2.25}}$$
$$- \gamma_3 \frac{9}{4}\left(\varepsilon_5 \frac{\|(\mathbf{I}_A + \mathbf{I}_H)^{-1}\boldsymbol{\theta}\|^4}{(1 + \|\omega_e\|^4)^{0.5}} \hat{\delta}_1^{\mathrm{M}} + \varepsilon_6 \|(\mathbf{I}_A + \mathbf{I}_H)^{-1}\|^4 \hat{\delta}_2^{\mathrm{M}}\right)\Bigg) r_e^M + \frac{\hat{\boldsymbol{\tau}}_r^M}{I_z + I_{z,A}}$$
$$\leq \frac{U_4^M}{I_z + I_{z,A}}. \tag{4-429}$$

Rearranging (4-427), (4-428) and (4-429) gives:

$$0<\mathbf{K}_{2,11}\leq\frac{\frac{U_2^M}{I_x+I_{x,A}}-\left(\left\|\frac{I_z+I_{z,A}-I_y-I_{y,A}}{I_x+I_{x,A}}\right\|q^Mr^M+\frac{1}{I_x+I_{x,A}}\hat{\bar{\boldsymbol{\tau}}}_p^M\right)}{p_e^M}$$
$$-\frac{1}{p_e^M}\left(\dot{\alpha}_p^{\mathrm{M}}+\frac{1}{\gamma_2}\|\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{q})\|p_e^M-\frac{\varepsilon_2}{\gamma_2}\left(\frac{1}{4\varepsilon_5}\right)^2\frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1+\|p_e^M\|^4)^{2.25}}p_e^M\right.$$
$$\left.-\gamma_3\frac{9}{4}\left(\varepsilon_5\frac{\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\boldsymbol{\theta}\|^4}{(1+\|\omega_e\|^4)^{0.5}}\hat{\delta}_1^{\mathrm{M}}+\varepsilon_6\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\|^4\hat{\delta}_2^{\mathrm{M}}\right)p_e^M\right), \tag{4-430}$$

$$0<\mathbf{K}_{2,22}\leq\frac{\frac{U_3^M}{I_y+I_{y,A}}-\left(\left\|\frac{I_x+I_{x,A}-I_z-I_{z,A}}{I_y+I_{y,A}}\right\|r^Mp^M+\frac{1}{I_y+I_{y,A}}\hat{\bar{\boldsymbol{\tau}}}_q^M\right)}{q_e^M}$$
$$-\frac{1}{q_e^M}\left(\dot{\alpha}_q^{\mathrm{M}}+\frac{1}{\gamma_2}\|\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{q})\|q_e^M-\frac{\varepsilon_2}{\gamma_2}\left(\frac{1}{4\varepsilon_5}\right)^2\frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1+\|q_e^M\|^4)^{2.25}}q_e^M\right.$$
$$\left.-\gamma_3\frac{9}{4}\left(\varepsilon_5\frac{\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\boldsymbol{\theta}\|^4}{(1+\|\omega_e\|^4)^{0.5}}\hat{\delta}_1^{\mathrm{M}}+\varepsilon_6\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\|^4\hat{\delta}_2^{\mathrm{M}}\right)q_e^M\right), \tag{4-431}$$

$$0<\mathbf{K}_{2,33}\leq\frac{\frac{U_4^M}{I_z+I_{z,A}}-\left(\left\|\frac{I_y+I_{y,A}-I_x-I_{x,A}}{I_z+I_{z,A}}\right\|p^Mq^M+\frac{1}{I_z+I_{z,A}}\hat{\bar{\boldsymbol{\tau}}}_r^M\right)}{r_e^M}$$
$$-\frac{1}{r_e^M}\left(\dot{\alpha}_r^{\mathrm{M}}+\frac{1}{\gamma_2}\|\boldsymbol{q}_e^T\mathbf{R}_2(\boldsymbol{q})\|q_e^M-\frac{\varepsilon_2}{\gamma_2}\left(\frac{1}{4\varepsilon_5}\right)^2\frac{\|\mathbf{R}_2(\boldsymbol{q})\|^4}{(1+\|r_e^M\|^4)^{2.25}}r_e^M\right.$$
$$\left.-\gamma_3\frac{9}{4}\left(\varepsilon_5\frac{\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\boldsymbol{\theta}\|^4}{(1+\|\omega_e\|^4)^{0.5}}\hat{\delta}_1^{\mathrm{M}}+\varepsilon_6\|(\mathbf{I}_A+\mathbf{I}_H)^{-1}\|^4\hat{\delta}_2^{\mathrm{M}}\right)r_e^M\right), \tag{4-432}$$

Therefore, the control design is complete and control objective 4.2 is met. Furthermore, it has been proven that the control law is exponentially stable in probability. In addition we can state that we have meet the second and sixth point of the project aims depicted in section 1.3.

5 Numerical Simulation

In this chapter the simulation results are presented for control schemes detailed in chapters 3 and 4. This chapter is broken up into two section.

In the first section, the problem of attitude position stabilisation is considered. Firstly, with deterministic control and the corresponding model as per section 3.1. Secondly stochastic control and the corresponding model as per section 3.3. In both cases the system response and behaviour under external loads is studied.

In the second section the problem of linear position tracking is considered. Firstly, with the deterministic control and model as per section 4.1 and then with the stochastic control scheme and model as per section4.2. In both cases the system response and behaviour under external loads is studied.

5.1 Angular Position Stabilisation

5.1.1 Deterministic Model and Controller

In this sub section we look at the performance of the deterministic control strategy for controlling and stabilizing the attitude of the quadrotor

Listed below are the system dynamics and properties:

Simulation Parameters						
Control gain		Disturbance Observer		System Properties		
$\mathbf{K}_{1,11}$	5	Property	Value	Property	Value	Units
$\mathbf{K}_{1,22}$	5	$\mathbf{K}_{d,11}$	10	I_x	0.016507	Kg/m^2
$\mathbf{K}_{1,33}$	1	$\mathbf{K}_{d,22}$	10	I_y	0.016507	Kg/m^2
$\mathbf{K}_{2,11}$	10	$\mathbf{K}_{d,33}$	10	I_z	0.016284	Kg/m^2
$\mathbf{K}_{2,22}$	10			$\bar{\boldsymbol{\tau}}_p(t)$	0.001	Nm
$\mathbf{K}_{2,33}$	2			$\bar{\boldsymbol{\tau}}_q(t)$	0.001	Nm
ε_1	5			$\bar{\boldsymbol{\tau}}_e(t)$	0.001	Nm
				Delta Time	10	µs
				Start time	0	s
				End Time	10	s

Table 5-1: System Dynamics and Properties.

The control and observer gains listed in the table above are chosen in accordance with conditions specified in (3-73), (3-75), (3-76), (3-83), (3-84) and (3-85).

Listed in the table below, are the initial states.

Initial State Values		
$\phi(0) = \frac{\pi}{8}$	$\theta(0) = -\frac{\pi}{8}$	$\psi(0) = 0$
$p(0) = 0$	$q(0) = -0$	$r(0) = 0$

Table 5-2: Initial Conditions.

Which are chosen in accordance with(3-64), (3-65), (3-66), (3-70), (3-71) and (3-72).

Listed in the table below, the attitude reference signals are specified.

Attitude Reference Signals		
$\phi_d(t) = \frac{\pi}{18}\tanh(t)$	$\theta_d(t) = -\frac{\pi}{18}\tanh(t)$	$\psi_d(t) = 0$

Table 5-3: Reference Attitude Signal.

Listed in the table below we specify the disturbance acting on the system.

Disturbance Torque Profile		
$\phi_d(t) = \frac{\pi}{18}\sin(t)$	$\theta_d(t) = -\frac{\pi}{18}\sin(t)$	$\psi_d(t) = 0$

Table 5-4: Disturbance Model.

Using the above information presented in Table 5-1- to Table 5-4 we present the simulation results, the MATLAB code for this simulation can be seen in Appendix H. The Euler attitude reference trajectory $\boldsymbol{\eta}_{2d}$ and the real Euler attitude trajectory $\boldsymbol{\eta}_2$ are plotted in Figure 5-1 the selection of these reference signals was done to provide a reasonable estimation of a changing reference signal during flight. The attitude error $\boldsymbol{\eta}_{2e}$ is plotted in Figure 5-2. The estimate of the disturbance torque is plotted in Figure 5-3, the disturbance observer error is plotted in Figure 5-4. The control torques are plotted in Figure 5-5, Lyapunov function V_{sum} and its bound $V_{sum\ bound}$ are plotted in Figure 5-6. It is seen from these figures, that all tracking errors asymptotically and the Lyapunov function exponentially converge to zero.

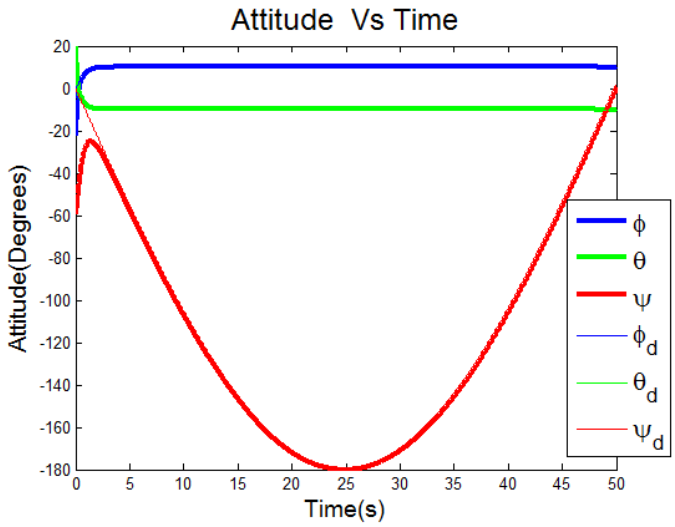

Figure 5-1: Attitude Response and Reference Signal Vs Time.

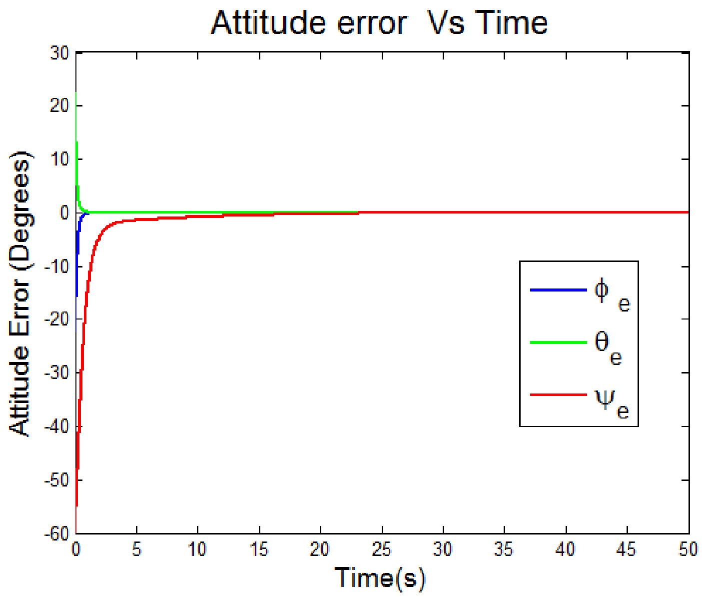

Figure 5-2: Attitude Error Vs Time.

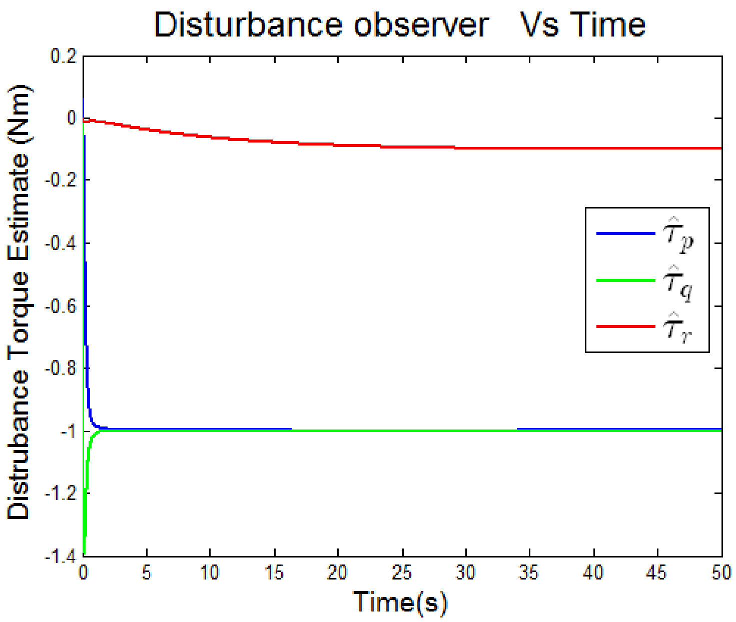

Figure 5-3: Disturbance Estimate Vs Time.

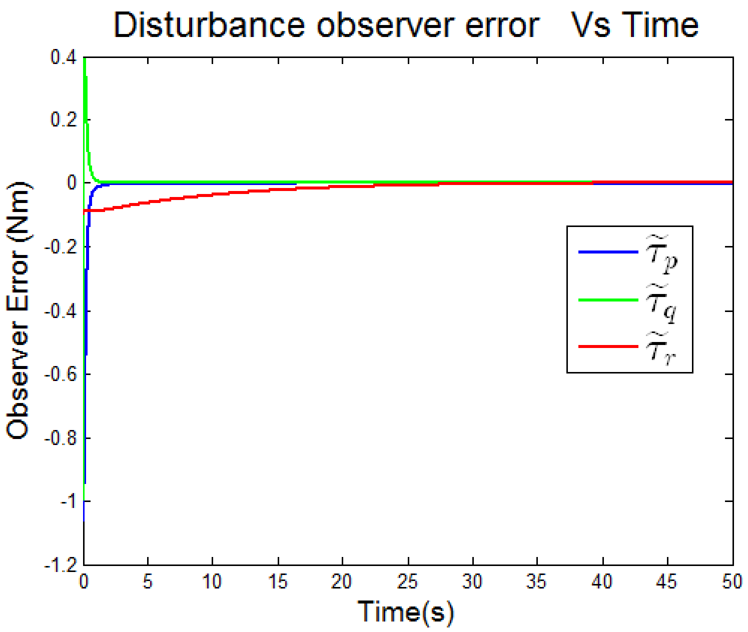

Figure 5-4: Disturbance Observer Error Vs Time.

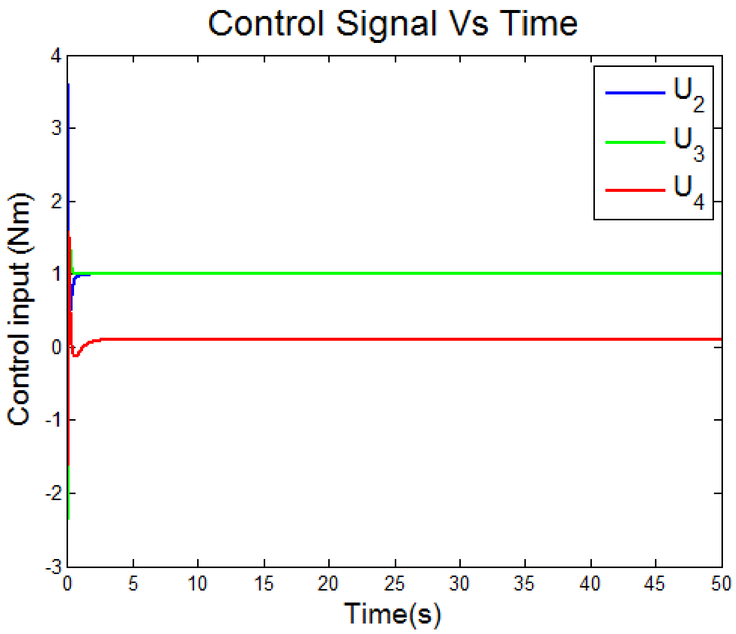

Figure 5-5: Control Torques Vs Time.

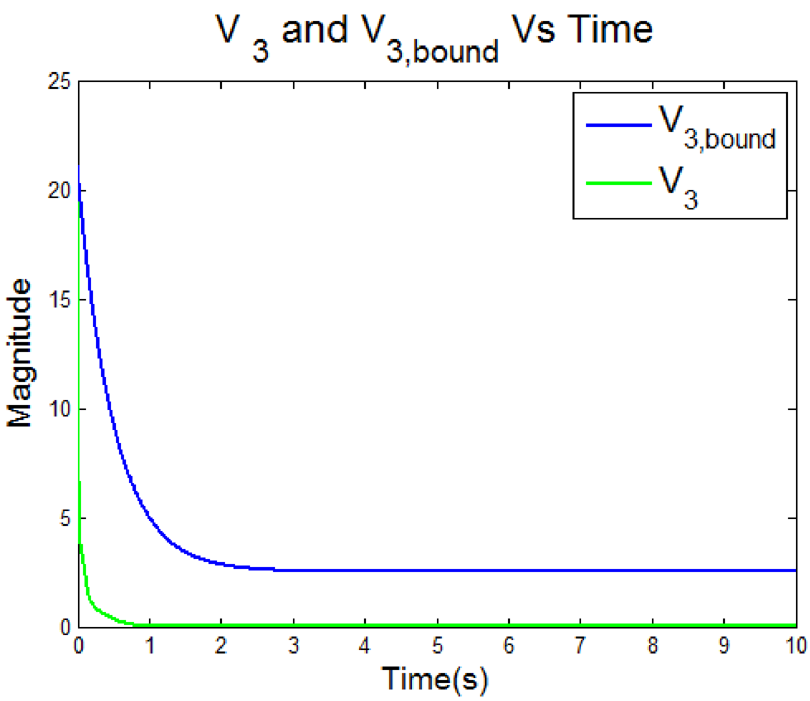

Figure 5-6: V_3 and $V_{3,\,bound}$ Vs Time.

In light of the Figure 5-1 to Figure 5-5 it can be stated that the backstepping controller presented in section 3.1 is more than acceptable for stabilizing the aircraft the conditions prescribed in Table 5-1- to Table 5-4.

5.1.2 Stochastic Model and Controller

Listed below are the system dynamics and properties

Simulation Parameters						
Control gain		Projection Algorithm		System Properties		
Property	Value	Property	Value	Property	Value	Units
$\mathbf{K}_{1,11}$	5.1	$\mathbf{\Gamma}_{11}$	5	I_x	0.016507	Kg/m^2
$\mathbf{K}_{1,22}$	5.1	$\mathbf{\Gamma}_{22}$	5	I_y	0.016507	Kg/m^2
$\mathbf{K}_{1,33}$	5.1	$\mathbf{\Gamma}_{33}$	3	I_z	0.016284	Kg/m^2
$\mathbf{K}_{2,11}$	5	$\omega_{\mathrm{M},\bar{\tau}_{Aero}}$	0.02	$\mathbf{I}_H$	$diag(I_x, I_y, I_z)$	Kg/m^2
$\mathbf{K}_{2,22}$	5	$\xi_{\bar{\tau}_{Aero}}$	0.1	$\mathbf{I}_A$	$0.5\mathbf{I}_H$	Kg/m^2
$\mathbf{K}_{2,33}$	5	μ_1	$4*10^9$	$\mathbf{D}_1$	$\mathbf{0}_{3\times3}$	Kg/(m^{2}s)
ε_1	5	$\omega_{1,\mathrm{M},\delta}$	1	$\mathbf{D}_2, 11$	$2.5 * 10^{-3}$	Kg/(m^{2}s)
ε_2	10	$\xi_{1,\delta}$	0.1	$\mathbf{D}_{2,11}$	$2.5 * 10^{-3}$	Kg/(m^{2}s)
ε_3	1	μ_2	10	$\mathbf{D}_2, 11$	0.0310^{-3}	Kg/(m^{2}s)
ε_4	1	$\omega_{2,\mathrm{M},\delta}$	0.0001	Delta Time	10	µs
γ_1	100	$\xi_{2,\delta}$	0.00001	Start time	0	s
γ_2	0.1			End Time	50	s

Table 5-5: System Dynamics and Properties.

Where $\bar{\boldsymbol{\tau}}_e(t)$, $\bar{\boldsymbol{\tau}}_e(t)$, and $\bar{\boldsymbol{\tau}}_e(t)$ are the deterministic components of the external disturbance acting on the aircraft.

Listed in the table below, are the initial states.

Initial state values		
$\phi(0) = -\frac{\pi}{8}$	$\theta(0) = \frac{\pi}{8}$	$\psi(0) = \frac{\pi}{3}$
$p(0) = 0$	$q(0) = -0$	$r(0) = 0$

Table 5-6: Initial Conditions.

Listed in the table below, is the reference signal.

Attitude Reference Signals		
$\phi_d(t) = \frac{\pi}{8}\tanh(t)$	$\theta_d(t) = -\frac{\pi}{18}\tanh(t)$	$\psi_d(t) = -\pi\, cos(t)$

Table 5-7: Reference Attitude Signal.

Disturbance Parameters					
Disturbance torque		Added mass due to airflow		Disturbance noise covariance	
Property	Value(Nm)	Property	Value(Kg/(m^{2}s))	Property	Value
$\bar{\boldsymbol{\tau}}_p(t)$	-1	$I_{x,A}$	$0.5I_x$	Δ_1	$0.1 * \mathbf{I}_{3\times3}$
$\bar{\boldsymbol{\tau}}_q(t)$	-1	$I_{y,A}$	$0.5I_y$	Δ_2	$1 * \mathbf{I}_{3\times3}$
$\bar{\boldsymbol{\tau}}_e(t)$	0.1	$I_{z,A}$	$0.5I_z$		

Table 5-8: Disturbance Properties

Listed in the table below, is the Wiener process profile used in this simulation.

Wiener process profile
$d\boldsymbol{w} = \sqrt{dt}\boldsymbol{\mathcal{N}}(0, \Delta_1)$

Table 5-9: Wiener Process Model.

The control gains, projection algorithm parameters and initial conditions have been chosen to adhere to the conditions specified by (3-126), (3-170), (3-171), (3-176), (3-177) and (3-178). The bounds specified by these equations have been calculated numerical using MATLAB and will not be written out here. Using the above information, we are able to present the simulation results.

5.1.2.1 System Response

Using the above information presented in Table 5-5 to Table 5-9 we present the simulation results, the MATLAB code for this simulation can be found in Appendix I. The Euler attitude reference trajectory $\boldsymbol{\eta}_{2d}$ and the real Euler attitude trajectory $\boldsymbol{\eta}_1$ are plotted in Figure 5-7. The attitude error represented in Euler angles $\boldsymbol{\eta}_{2e}$ is plotted in Figure 5-8, the attitude error represented in MRP is plotted in Figure 5-9. The estimate of the deterministic disturbance torque is plotted in Figure 5-10, estimate of the $\hat{\delta}_1$ and $\hat{\delta}_2$ is plotted in Figure 5-11. The control torques are plotted in Figure 5-12, Lyapunov function V_2 and its bound $V_{2\,bound}$ are plotted in Figure 5-13. It is seen from these figures, that all tracking errors asymptotically and the Lyapunov function exponentially converge to the prescribed bound.

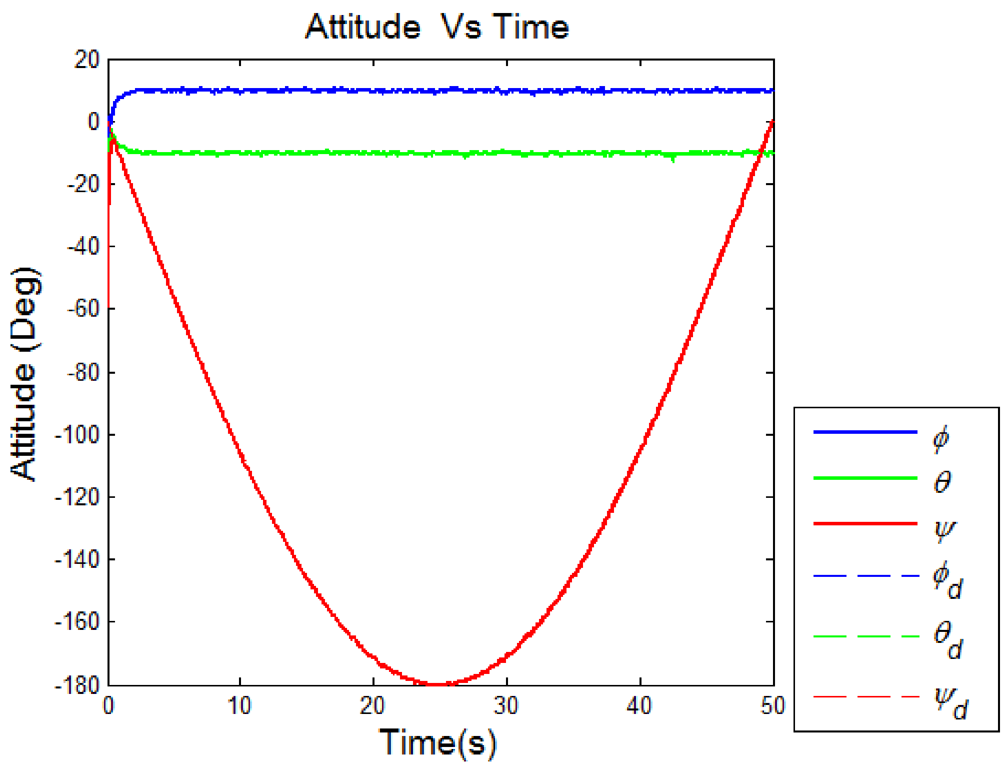

Figure 5-7: Attitude Response and Reference Signal Vs Time.

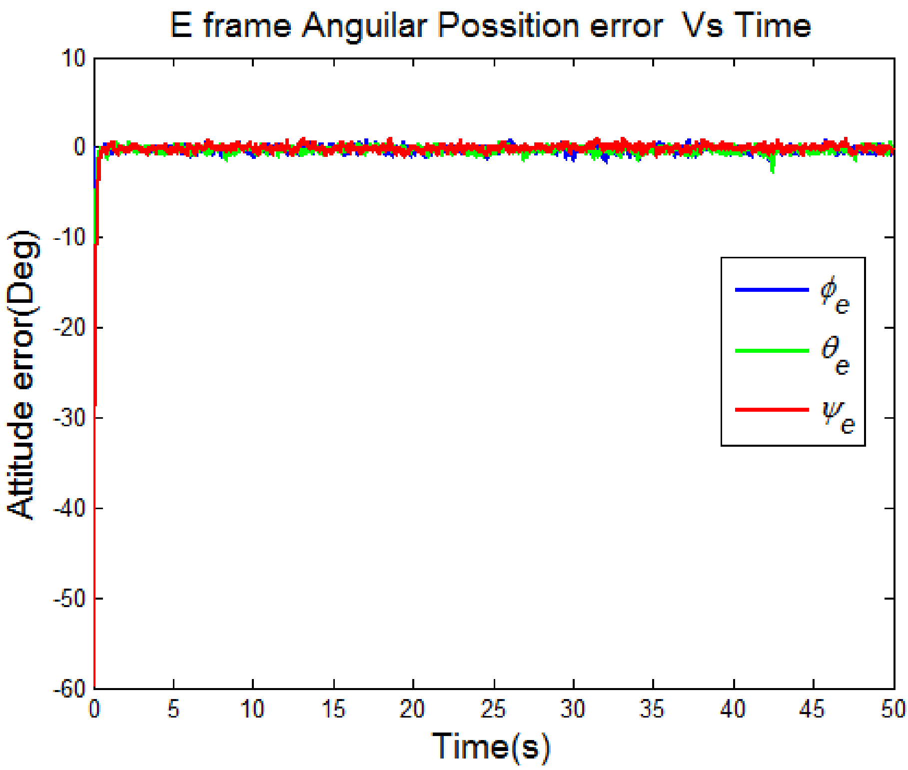

Figure 5-8: Attitude Euler Error Vs Time.

As can be seen in the figure above the error reaches a steady state error of 1^0, the rapid oscillation present is a direct result of the stochastic disturbance acting on the system

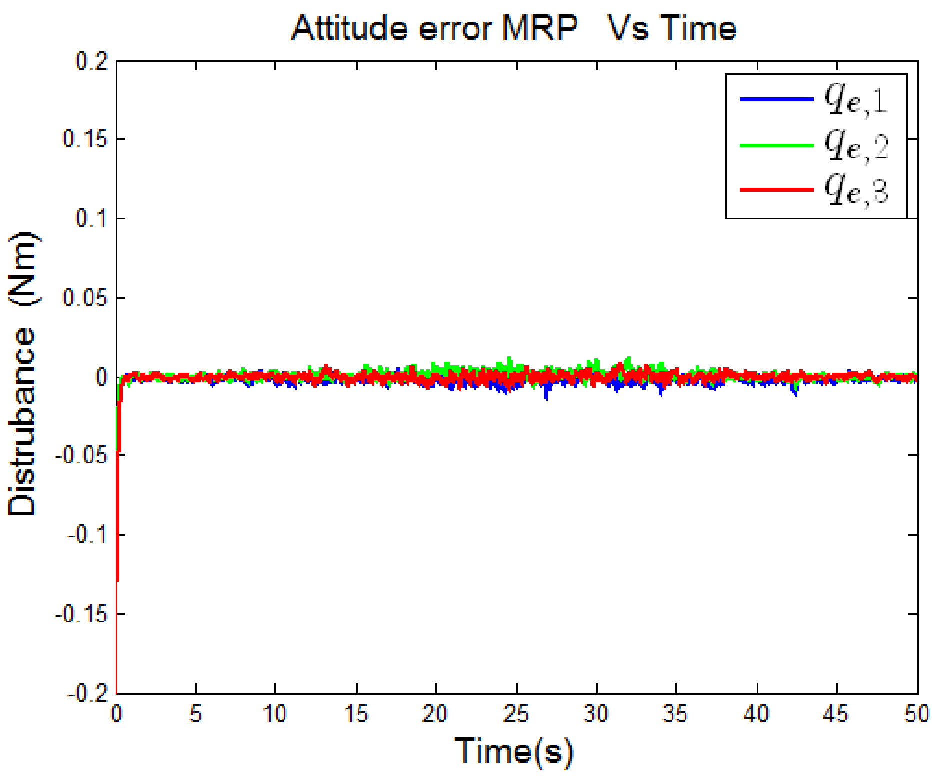

Figure 5-9: Attitude MRP Error Vs Time.

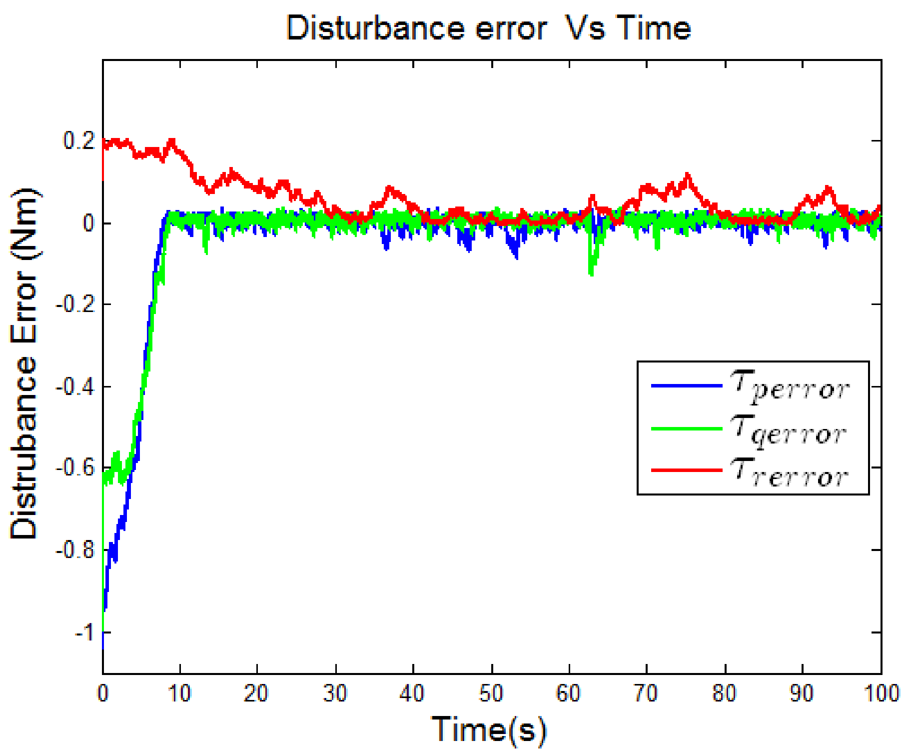

Figure 5-10: Deterministic Component of Disturbance Estimate Vs Time.

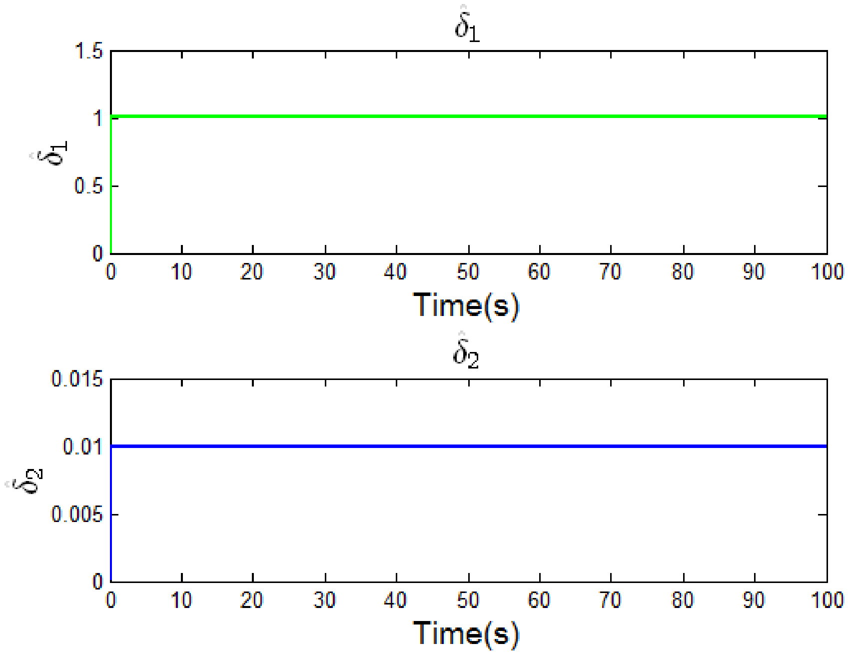

Figure 5-11: Estimate of $\hat{\delta}_1$ and $\hat{\delta}_2$ Vs Time.

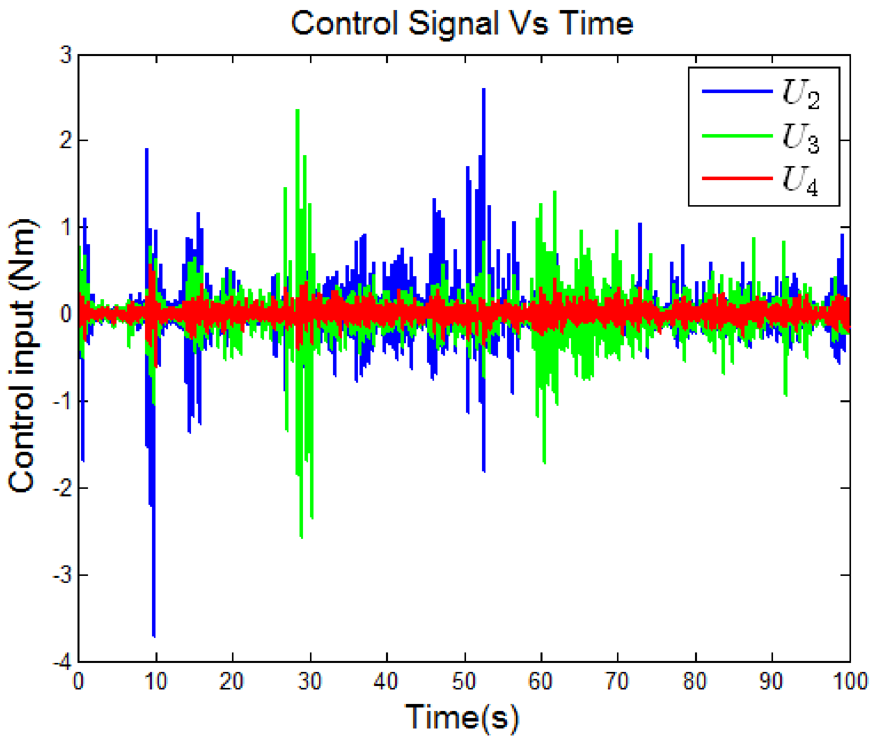

Figure 5-12: Control Torques Vs Time.

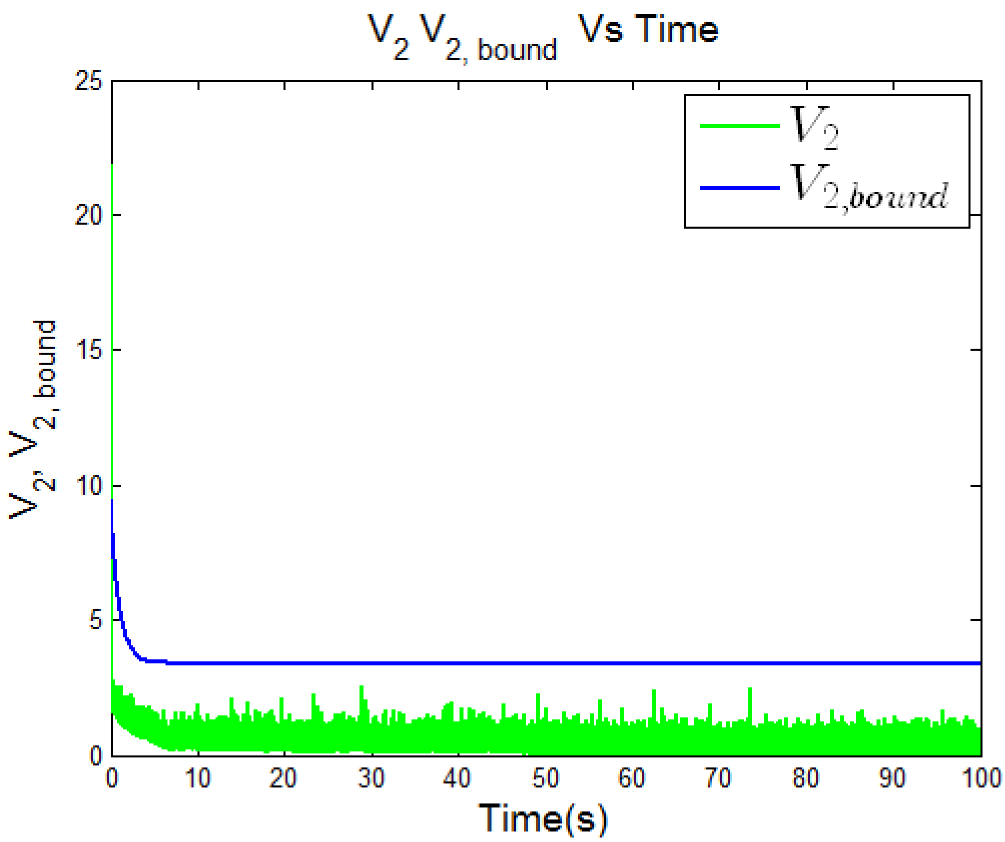

Figure 5-13: V_2 and $V_{2,\ bound}$ Vs Time.

In light of Figure 5-7. To Figure 5-13.it can be stated that the backstepping controller presented in section 3.2 is more than acceptable for stabilizing the aircraft the conditions prescribed in Table 5-5-Table 5-9. This is because the backstepping controller is able to obtain:

1. An attitude error of aircraft is within 0.5^0 as shown in Figure 5-9.
2. Control torques are bounded within 4 Nm see Figure 5-12,
3. The error of the estimate of the disturbance torque acting on the system settles to within 0.15Nm
4. Exponential convergence of system states to within a ball centred at the origin is shown in Figure 5-13.

These four points are reasonable conditions for ensuring stable flight

5.1.3 Stochastic Model and Deterministic Controller

In this sub section we look at the performance of the deterministic and stochastic control strategy for controlling and stabilizing the attitude of the quadrotor under the same stochastic conditions.

Simulation Parameters					
Deterministic Controller					
Control gain		Disturbance Observer		System Properties	
Property	Initial Value	Property	Value	Property	Value
$\mathbf{K}_{1,11}$	10	$\mathbf{K}_{d,11}$	10	I_x	0.016507
$\mathbf{K}_{1,22}$	10	$\mathbf{K}_{d,22}$	10	I_y	0.016507
$\mathbf{K}_{1,33}$	10	$\mathbf{K}_{d,33}$	10	I_z	0.016284
$\mathbf{K}_{2,11}$	10			$\mathbf{D}_{2,11}$	$2.5*10^{-3}$
$\mathbf{K}_{2,22}$	10			$\mathbf{D}_{2,22}$	$2.5*10^{-3}$
$\mathbf{K}_{2,33}$	10			$\mathbf{D}_{2,33}$	$0.03*10^{-3}$
ε_1	5			Delta Time	10µs
				Start time	0
				End Time	50s

Table 5-10: System Dynamics and Properties.

Where $\overline{\boldsymbol{\tau}}_e(t)$, $\overline{\boldsymbol{\tau}}_e(t)$, and $\overline{\boldsymbol{\tau}}_e(t)$ are the deterministic components of the external disturbance acting on the aircraft.

Disturbance Parameters					
Disturbance torque		Added mass due to airflow		**Disturbance noise covariance**	
Property	Value	Property	Value	Property	Value
$\overline{\boldsymbol{\tau}}_p(t)$	-1	$I_{x,A}$	$0.2I_x$	Δ_1	$0.1 * \mathbf{I}_{3\times3}$
$\overline{\boldsymbol{\tau}}_q(t)$	-1	$I_{y,A}$	$0.2I_y$	Δ_2	$1 * \mathbf{I}_{3\times3}$
$\overline{\boldsymbol{\tau}}_e(t)$	0.1	$I_{z,A}$	$0.2I_z$		

Table 5-11: Disturbance Parameters.

Listed in the table below, are the initial states.

Initial state values		
$\phi(0) = -\frac{\pi}{8}$	$\theta(0) = \frac{\pi}{8}$	$\psi(0) = \frac{\pi}{3}$
$p(0) = 0$	$q(0) = -0$	$r(0) = 0$

Table 5-12: Initial Conditions.

Listed in the table below, is the reference signal.

Attitude Reference Signals		
$\phi_d(t) = \frac{\pi}{8}\tanh(t)$	$\theta_d(t) = -\frac{\pi}{18}\tanh(t)$	$\psi_d(t) = -\pi\, cos(t)$

Table 5-13: Reference Attitude Signal.

Wiener process profile
$d\boldsymbol{w} = \sqrt{dt}\boldsymbol{\mathcal{N}}(0, \Delta_1)$

Table 5-14: Wiener Process Model.

Using the above information we are able to present the simulation results.

Using the above information presented in Table 5-10 to Table 5-14 we present the simulation results, the MATLAB code for this simulation can be found in Appendix I. The attitude error represented in Euler angles $\boldsymbol{\eta}_{2e}$ is plotted in Figure 5-14. The disturbance observer response is shown in Figure 5-15. The control torques are plotted in Figure 5-16, Lyapunov function V_3 and its bound $V_{2\ bound}$ are plotted in Figure 5-17. It is seen from these figures, that the introduction of stochastic noise prevents the determinist backstepping controller which in section 5.1.1 was capable of stabilizing the system is no longer able to do so.

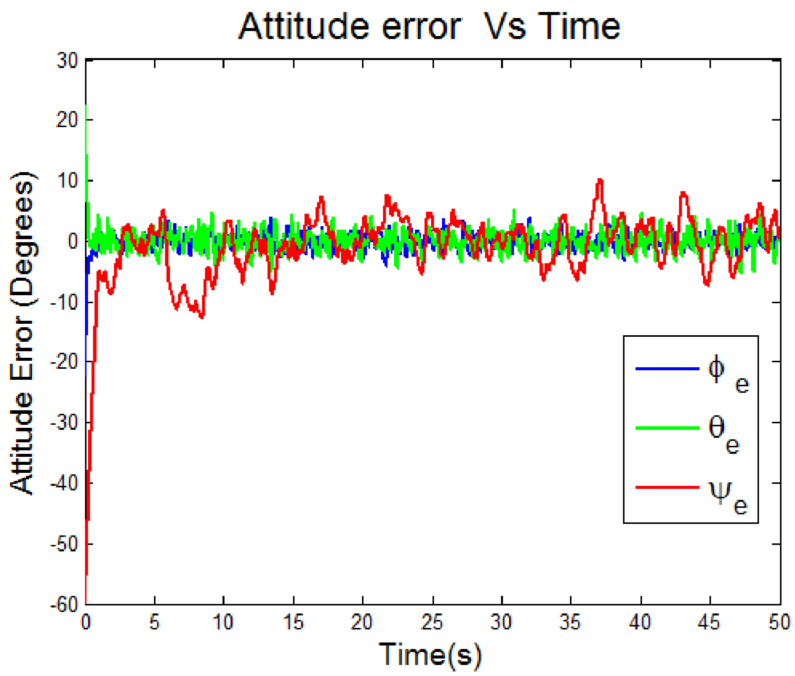

Figure 5-14: Attitude Euler Representation Error Vs Time.

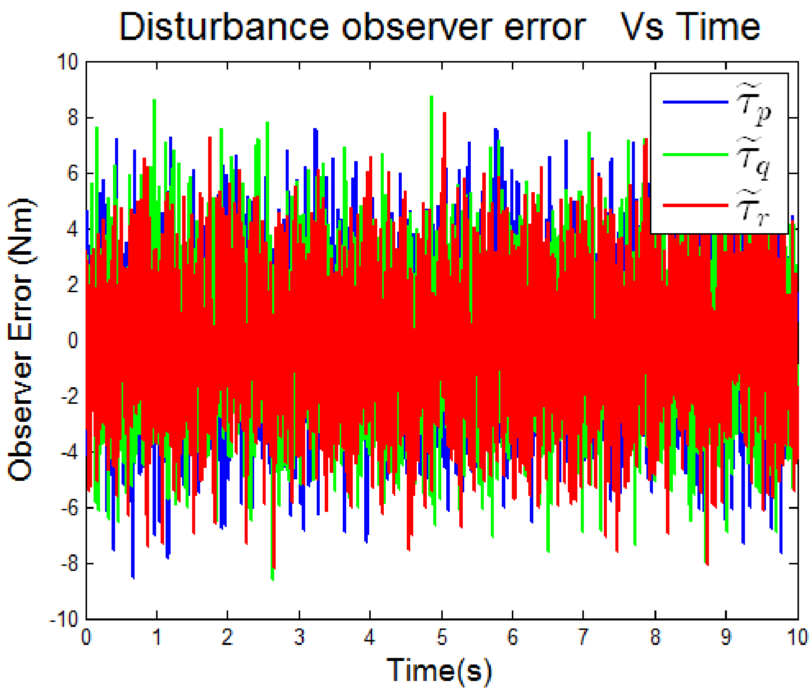

Figure 5-15: Deterministic Component of Disturbance Estimate Error Vs Time.

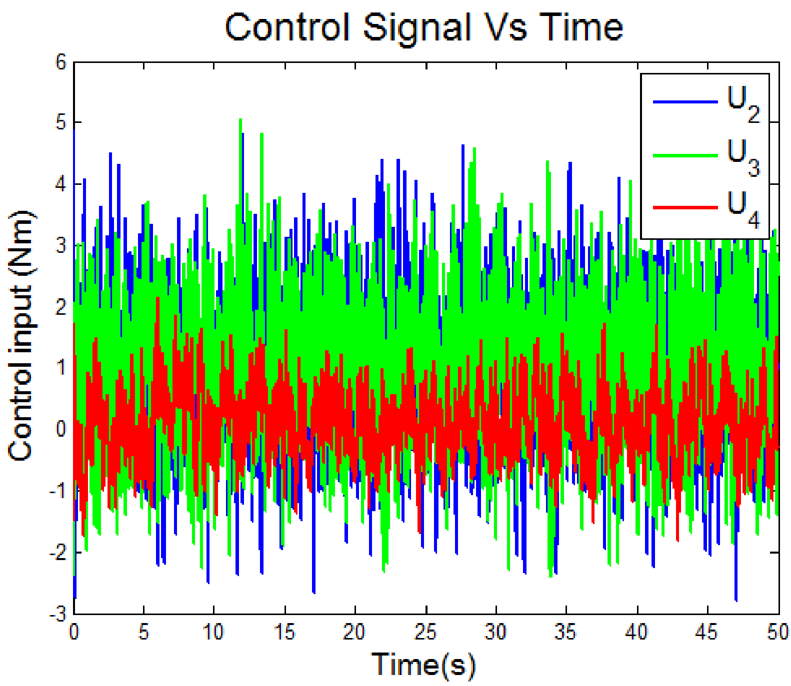

Figure 5-16: Control Signal Vs Time.

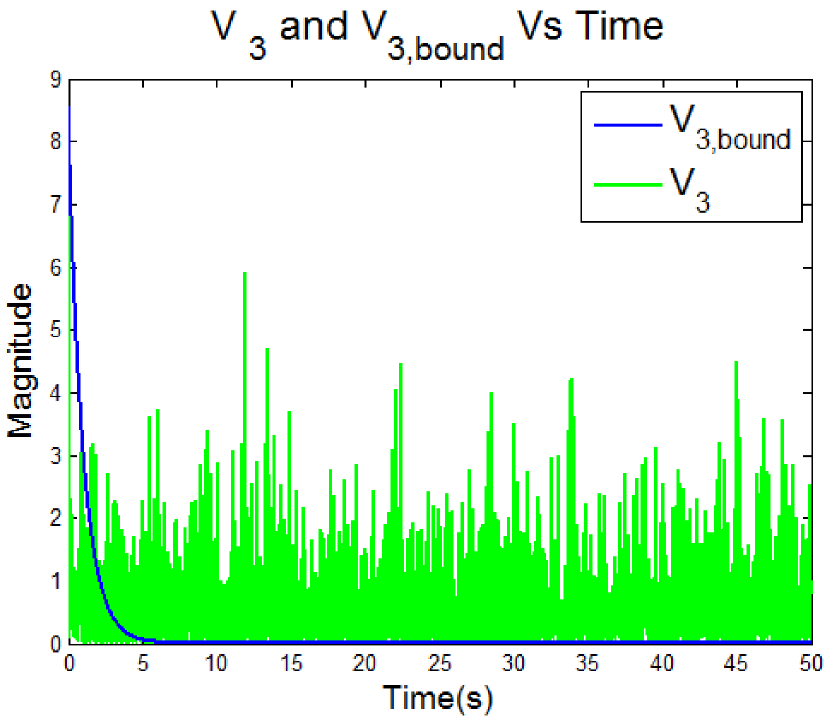

Figure 5-17: V_{sum} and $V_{sum, bound}$ Vs Time.

In light of Figure 5-14 to Figure 5-17 it can be stated that the backstepping controller presented in section 3.1 is not suitable for stabilizing the aircraft attitude under the conditions prescribed in Table 5-10 to Table 5-14. It is quite obvious comparing the results depicted in Figure 5-14 with Figure 5-8 the effect of not taking into account stochastic disturbances on the system when designing the control law. Even though tithe controller presented in section 3.1 is asymptotically stable for deterministic it is not suitable for stochastic conditions. Thus we have meet the fourth point of the project aims depicted in section 1.3.

5.2 Quadrotor Aircraft Linear and Angular Position Stabilisation

5.2.1 Deterministic Model and Controller

Listed below are the system dynamics and properties

Simulation Parameters						
		Disturbance Observer				
Property	Value	Property	Value	Property	Value	Units
γ_1	12	$\mathbf{K}_{D,11}$	100	I_x	0.016507	Kg/m^2
γ_2	100	$\mathbf{K}_{D,22}$	00	I_y	0.016507	Kg/m^2
γ_3	0.1	$\mathbf{K}_{D,33}$	100	I_z	0.016284	Kg/m^2
$\mathbf{K}_{1,11}$	0.8			$I_{x,A}$	$0.2I_x$	Kg/m^2
$\mathbf{K}_{1,22}$	0.8			$I_{y,A}$	$0.2I_y$	Kg/m^2
$\mathbf{K}_{1,33}$	0.8			$I_{z,A}$	$0.2I_z$	Kg/m^2
$\mathbf{K}_{2,11}$	4			$\mathbf{D}_{1,11}$	0.25	Kg/(m^{2}s)
$\mathbf{K}_{2,22}$	4			$\mathbf{D}_{1,22}$	0.25	Kg/(m^{2}s)
$\mathbf{K}_{2,33}$	4			$\mathbf{D}_{1,33}$	0.125	Kg/(m^{2}s)
$\mathbf{K}_{3,11}$	10			$\mathbf{D}_{2,11}$	$2.5*10^{-3}$	Kg/(m^{2}s)
$\mathbf{K}_{3,22}$	10			$\mathbf{D}_{2,22}$	$2.5*10^{-3}$	Kg/(m^{2}s)
$\mathbf{K}_{3,33}$	10			$\mathbf{D}_{2,33}$	$0.03*10^{-3}$	Kg/(m^{2}s)
$\mathbf{K}_{4,11}$	10			m	2.25	kg
$\mathbf{K}_{4,22}$	10			m_a	0	kg
$\mathbf{K}_{4,33}$	10			c_{T}	$2.5569*10^{-5}$	
ε_1	7.6			c_{R}	$5.7768*10^{-6}$	
ε_2	3.9			Delta Time	10	µs
ε_3	$\frac{5}{5.1}\lambda_m(\mathbf{K}_3)$			Start time	0	s
$\varepsilon_4, \varepsilon_5, \varepsilon_6$	10			End Time	100s	s

Table 5-15: System Dynamics and Properties.

Reference signal			
$\boldsymbol{\eta}_{1d,1} = 10\cos(0.1t)$	$\boldsymbol{\eta}_{1d,2} = 10\sin(0.1t)$	$\boldsymbol{\eta}_{1d,3} = 0.1t$	$\psi_d = 0.1t$

Table 5-16: Reference Signal.

Initial conditions							
State	Value	State	Value	State	Value	State	Value
$\boldsymbol{\eta}_{1,1}$	10(m)	$\boldsymbol{\nu}_{1,1}$	0	$\boldsymbol{\eta}_{2,1}$		$\boldsymbol{\omega}_{,1}$	0
$\boldsymbol{\eta}_{1,2}$	0(m)	$\boldsymbol{\nu}_{1,2}$	0	$\boldsymbol{\eta}_{2,2}$		$\boldsymbol{\omega}_{,2}$	0
$\boldsymbol{\eta}_{1,3}$	0(m)	$\boldsymbol{\nu}_{1,2}$	0	$\boldsymbol{\eta}_{2,3}$	0	$\boldsymbol{\omega}_{,3}$	0

Table 5-17: Initial Conditions.

The table below shows the deterministic component of the disturbances acting on the quadrotor

Deterministic Disturbance Torque Profile			
Linear Disturbance		Torque Disturbance	
Property	Value(N)	Property	Value(Nm)
$\bar{\boldsymbol{f}}_x(t)$	0.5	$\bar{\boldsymbol{\tau}}_p(t)$	-1
$\bar{\boldsymbol{f}}_y(t)$	0.5	$\bar{\boldsymbol{\tau}}_q(t)$	-1
$\bar{\boldsymbol{f}}_z(t)$	0.1	$\bar{\boldsymbol{\tau}}_e(t)$	0.1

Table 5-18: Disturbance Model.

The selection of the control gains, projection algorithm and system parameters and initial conditions specified in the above tables has been selected in accordance with the bounds prescribed by(4-174), (4-191), (4-202), (4-203) and (4-204). The bounds specified by these equations were calculated numerically inn MATLAB and as such will not be demonstrated here.

Using the above information presented in Table 5-15 - Table 5-18 we are able to present the simulation results, the MATLAB code for this simulation can be found in Appendix J. The position reference trajectory $\boldsymbol{\eta}_{1d}$ and the position real trajectory $\boldsymbol{\eta}_1$ are plotted in Figure 5-18 the position error $\boldsymbol{\eta}_{1e}$ are plotted in Figure 5-19, attitude Euler error $\boldsymbol{\eta}_{2e}$ are plotted in Figure 5-20, the error of the estimate of the aerodynamic disturbance torque is presented in Figure 5-21, the control thrust and control torques are plotted in Figure 5-22, Lyapunov function V_4, its bound $V_{4\ bound}$ are plotted in Figure 5-23. The error of the estimate of the linear velocity is plotted in Figure 5-24. It is seen from these figures, that all tracking errors asymptotically and the Lyapunov function exponentially converge to prescribed bound.

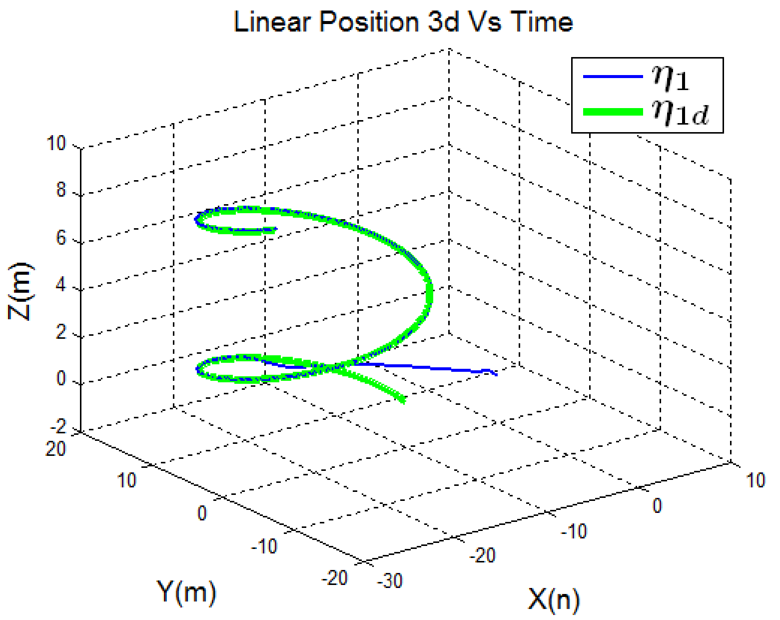

Figure 5-18: System 3-D Linear Position Vs Time.

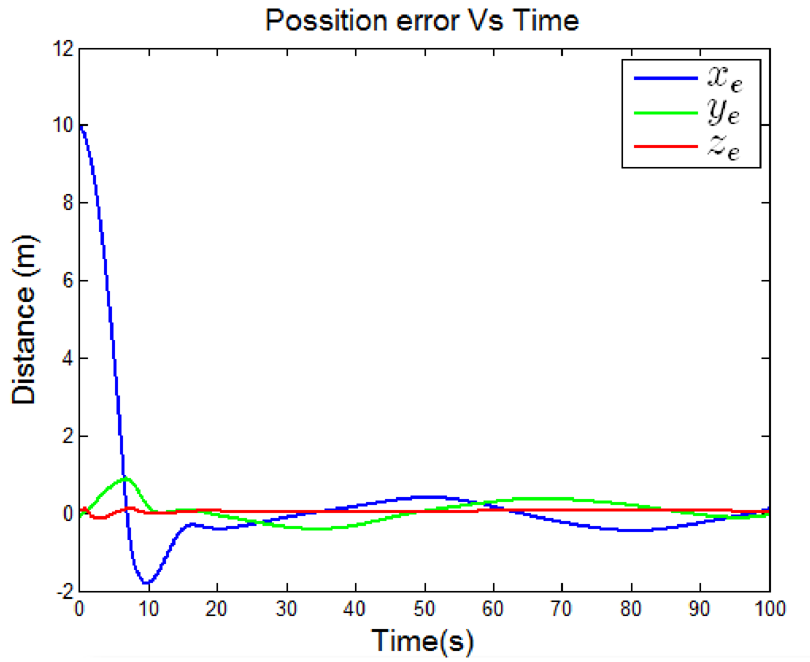

Figure 5-19: System Linear Position Error Vs Time.

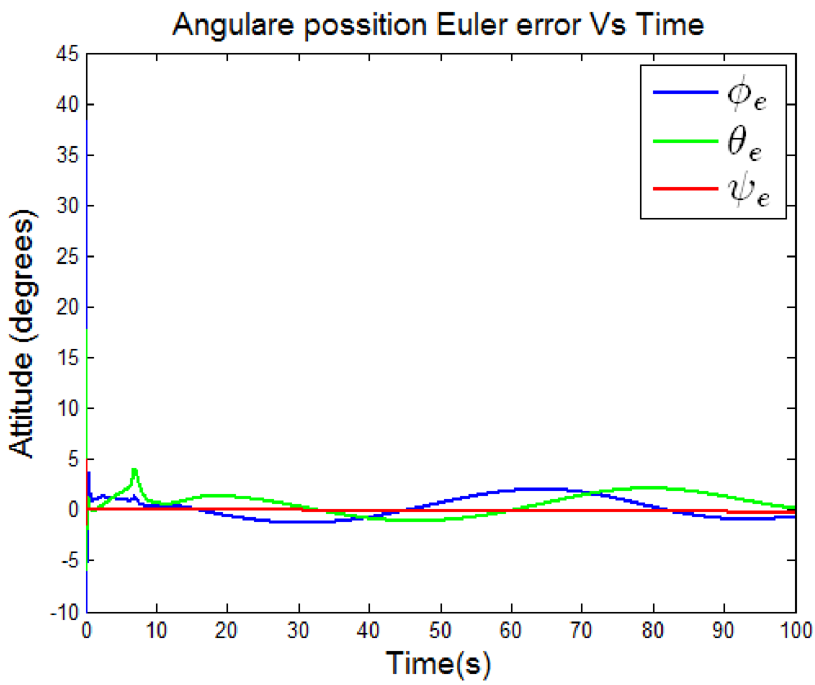

Figure 5-20: System Attitude Euler Representation Error Vs Time.

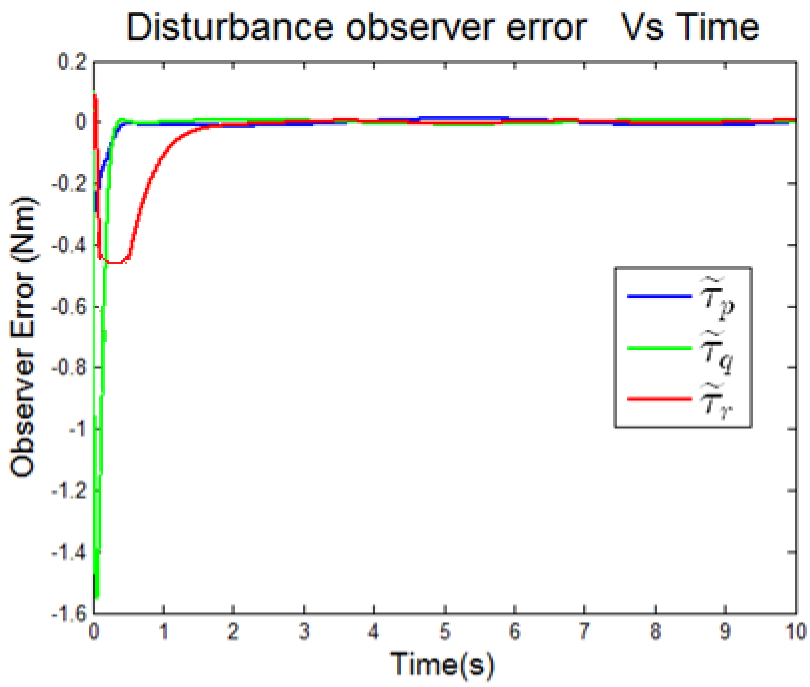

Figure 5-21: System Torque Disturbance Estimated error Vs Time.

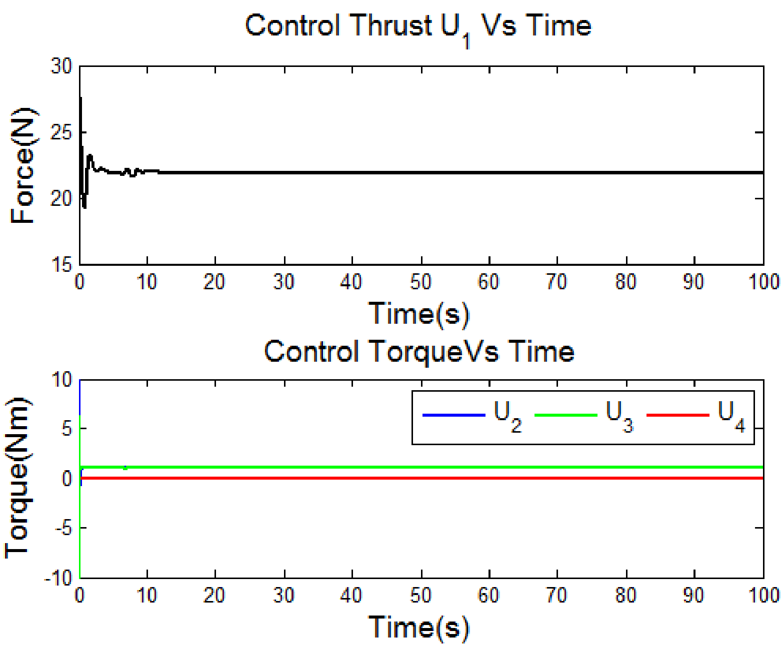

Figure 5-22: System Control Thrust and Torques Vs Time.

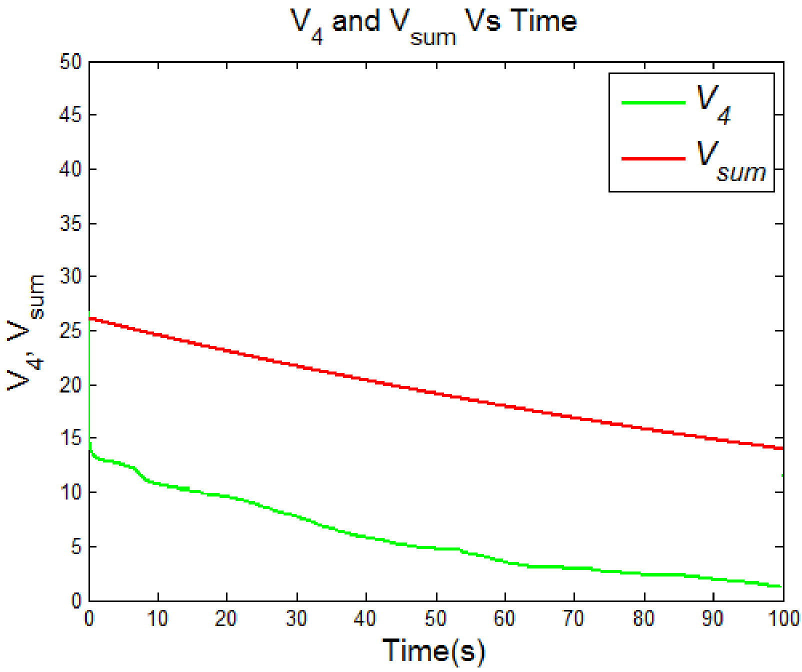

Figure 5-23: Lyapunov function V_4 and V_{bound} Vs Time.

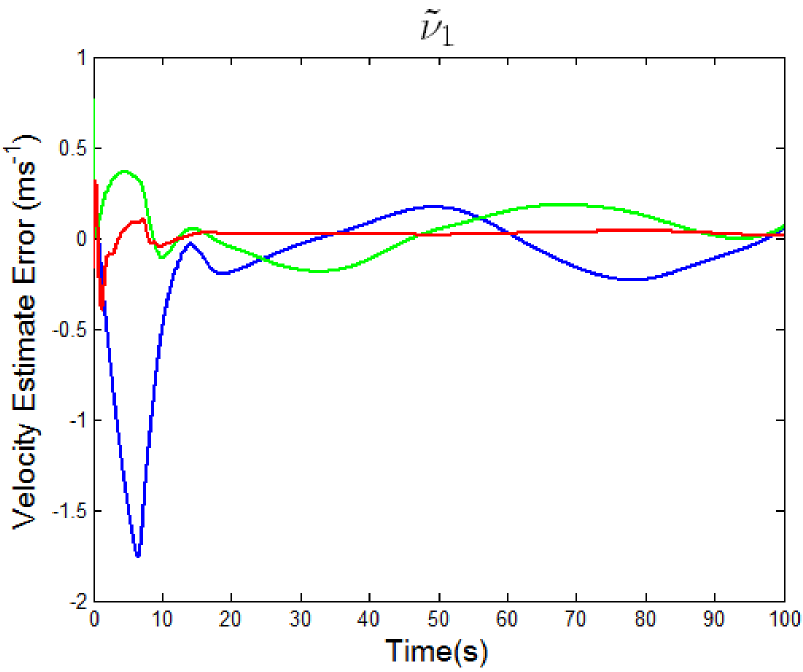

Figure 5-24: $\tilde{\nu}_1$ Vs Time.

5.2.2 Stochastic Model and Controller

Listed below are the system dynamics and properties the table below shows all controller gains and mechanical properties of the quadrotor helicopter used for simulation.

Simulation Parameters						
Control Gains		Projection Algorithm				
Property	Value	Property	Value	Property	Value	Units
γ_1	5.12	$\mathbf{\Gamma}_{11}$	5	I_x	0.016507	Kg/m^2
γ_2	200	$\mathbf{\Gamma}_{22}$	5	I_y	0.016507	Kg/m^2
γ_3	0.1	$\mathbf{\Gamma}_{33}$	3	I_z	0.016284	Kg/m^2
$\mathbf{K}_{1,11}$	0.8	$\omega_{\mathrm{M},\bar{\tau}_{Aero}}$	0.02	$I_{x,A}$	$0.2I_x$	Kg/m^2
$\mathbf{K}_{1,22}$	0.8	$\xi_{\bar{\tau}_{Aero}}$	0.1	$I_{y,A}$	$0.2I_y$	Kg/m^2
$\mathbf{K}_{1,33}$	0.8	μ_1	4*10^9	$I_{z,A}$	$0.2I_z$	Kg/m^2
$\mathbf{K}_{2,11}$	4	$\omega_{1,\mathrm{M},\delta}$	1	$\mathbf{D}_{1,11}$	0.25	Kg/(m^{2}s)
$\mathbf{K}_{2,22}$	4	$\xi_{1,\delta}$	0.1	$\mathbf{D}_{1,22}$	0.25	Kg/(m^{2}s)
$\mathbf{K}_{2,33}$	4	μ_2	10	$\mathbf{D}_{1,33}$	0.125	Kg/(m^{2}s)
$\mathbf{K}_{3,11}$	10	$\omega_{2,\mathrm{M},\delta}$	0.0001	$\mathbf{D}_{2,11}$	2.5*10^{-3}	Kg/(m^{2}s)
$\mathbf{K}_{3,22}$	10	$\xi_{2,\delta}$	0.00001	$\mathbf{D}_{2,22}$	2.5*10^{-3}	Kg/(m^{2}s)
$\mathbf{K}_{3,33}$	10			$\mathbf{D}_{2,33}$	0.03*10^{-3}	Kg/(m^{2}s)
$\mathbf{K}_{4,11}$	10			m	2.25	kg
$\mathbf{K}_{4,22}$	10			m_a	0	kg
$\mathbf{K}_{4,33}$	10			c_{T}	2.5569*10^{-5}	
ε_1	7.6			c_{R}	5.7768*10^{-6}	
ε_2	3.9			Delta Time	10	µs
ε_3	$\frac{5}{5.1}\lambda_m(\mathbf{K}_3)$			Start time	0	s
$\varepsilon_4, \varepsilon_5, \varepsilon_6$	10			End Time	100s	s

Table 5-19: System Dynamics add Properties.

The following table shows the behaviour of the reference signal that the quadrotor needs to track.

Reference signal			
$\boldsymbol{\eta}_{1d,1} = 10\cos(0.1t)$	$\boldsymbol{\eta}_{1d,2} = 10\sin(0.1t)$	$\boldsymbol{\eta}_{1d,3} = 0.1t$	$\psi_d = 0.1t$

Table 5-20: Reference Position Signal.

The following table shows the initial states of the aircraft.

Initial conditions							
State	Value	State	Value	State	Value	State	Value
$\boldsymbol{\eta}_{1,1}$	10(m)	$\boldsymbol{\nu}_{1,1}$	0	$\boldsymbol{\eta}_{2,1}$	0	$\boldsymbol{\omega}_{,1}$	0
$\boldsymbol{\eta}_{1,2}$	0(m)	$\boldsymbol{\nu}_{1,2}$	0	$\boldsymbol{\eta}_{2,2}$	0	$\boldsymbol{\omega}_{,2}$	0
$\boldsymbol{\eta}_{1,3}$	0(m)	$\boldsymbol{\nu}_{1,2}$	0	$\boldsymbol{\eta}_{2,3}$	0	$\boldsymbol{\omega}_{,3}$	0

Table 5-21: Initial Conditions.

The table below shows the deterministic component of the disturbances acting on the quadrotor.

Deterministic Disturbance Torque Profile			
Linear Disturbance		Torque Disturbance	
Property	Value(N)	Property	Value(Nm)
$\bar{\boldsymbol{f}}_x(t)$	0.5	$\bar{\boldsymbol{\tau}}_p(t)$	-1
$\bar{\boldsymbol{f}}_y(t)$	0.5	$\bar{\boldsymbol{\tau}}_q(t)$	-1
$\bar{\boldsymbol{f}}_z(t)$	0.1	$\bar{\boldsymbol{\tau}}_e(t)$	0.1

Table 5-22: Deterministic Disturbance Model.

In addition to the disturbances behaviour shown in the table above the stochastic component of the disturbances acting on the quadrotor is shown in the table below.

Stochastic Disturbance Torque Profile			
Linear Disturbance		Angular Disturbance	
Property	Value	Property	Value
$\Delta_{\boldsymbol{v}_1}$	$0.1\times\mathbf{I}_{3\times3}$	Δ_2	$1\times\mathbf{I}_{3\times3}$
		Δ_3	$0.1\times\mathbf{I}_{3\times3}$

Table 5-23: Stochastic Disturbance Model.

For clarification of the symbols please refer back to equation (2-119) and (2-120).

To be able to generate the stochastic disturbances used in this simulation the standard Wiener processes for both the linear position and angular position subsystems of the quadrotor are shown in the table below.

Wiener Process Profile		
Linear Disturbance	Angular disturbance	
$d\boldsymbol{w}_1 = \sqrt{dt}\boldsymbol{\mathcal{N}}(0,\Delta_{\boldsymbol{v}_1})$	$d\boldsymbol{w}_2 = \sqrt{dt}\boldsymbol{\mathcal{N}}(0,\Delta_2)$	$d\boldsymbol{w}_3 = \sqrt{dt}\boldsymbol{\mathcal{N}}(0,\Delta_3)$

Table 5-24: Wiener Process Profile.

The selection of the control gains, projection algorithm and system parameters and initial conditions specified in the above tables has been selected in accordance with the bounds prescribed by (4-402), (4-419), (4-430), (4-431) and (4-432). The bounds specified by these equations were calculated numerically in MATLAB and as such will not be demonstrated here.

Using the above information presented in Table 5-19 - Table 5-24 we present the simulation results, the MATLAB code for this simulation can be found in Appendix J. The position reference trajectory $\boldsymbol{\eta}_{1d}$ and the position real trajectory $\boldsymbol{\eta}_1$ are plotted in Figure 5-25 The position error $\boldsymbol{\eta}_{1e}$ are plotted in Figure 5-26, attitude Euler error $\boldsymbol{\eta}_{2e}$ are plotted in Figure 5-27, estimate of $\hat{\delta}_1$ and $\hat{\delta}_2$ i is plotted in Figure 5-28, the control thrust and control torques are plotted in Figure 5-29. The error between the estimate of the disturbance torque and actual disturbance torque is plotted is plotted in Figure 5-30. The velocity estimate error $\tilde{\boldsymbol{v}}_1$ is plotted in Figure 5-31. The Lyapunov function V_4 and its bound V4 bound are plotted in Figure 5-32. It is seen from these figures, that all tracking errors asymptotically and the Lyapunov function exponentially converge to bound.

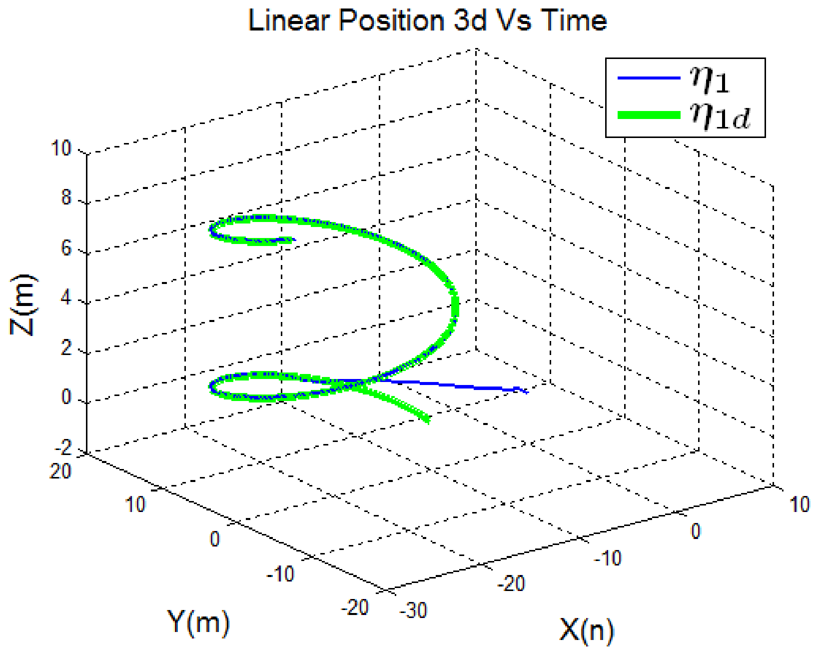

Figure 5-25: System 3-D linear position Vs Time.

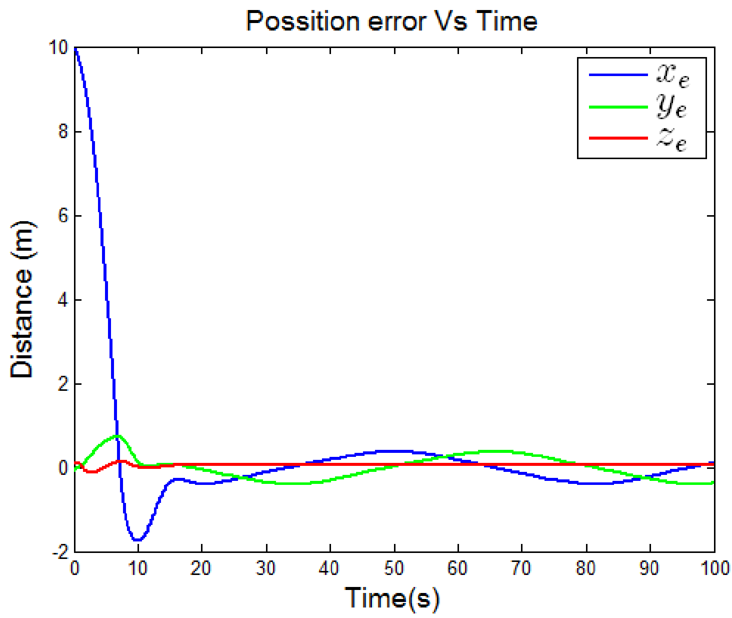

Figure 5-26: System Linear Position error Vs Time.

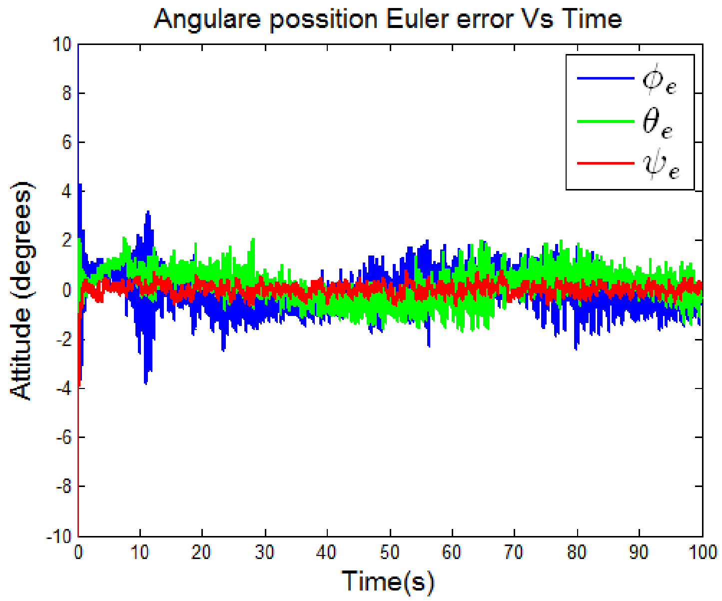

Figure 5-27: System Attitude Euler Error Vs Time.

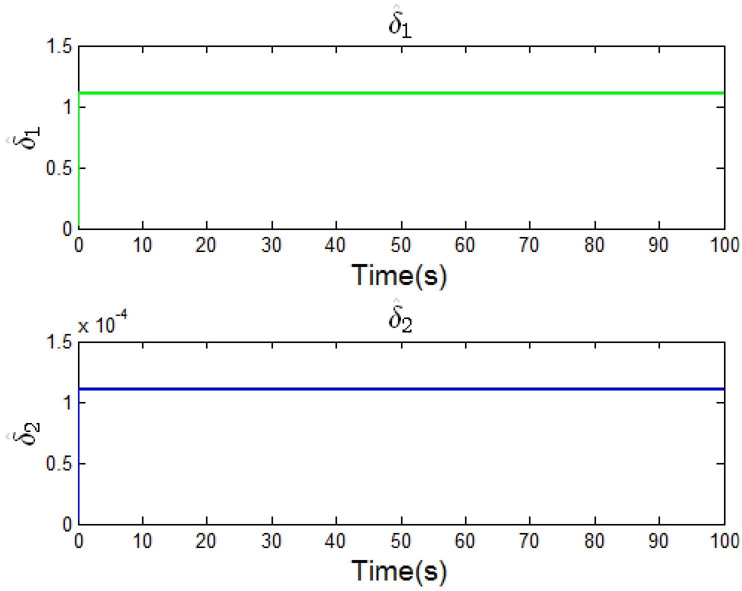

Figure 5-28: Estimate of $\hat{\delta}_1$ and $\hat{\delta}_2$ Vs Time.

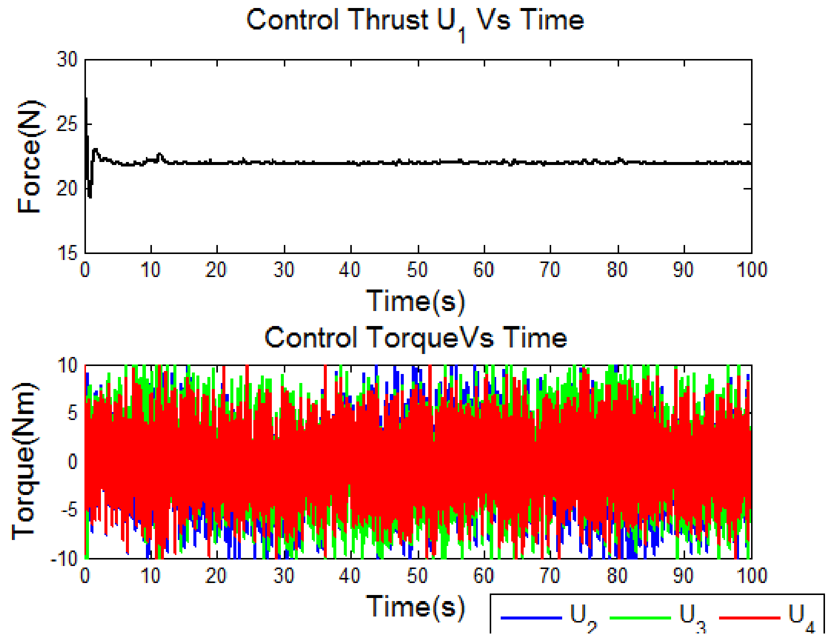

Figure 5-29: System Control Thrust and Torques Vs Time.

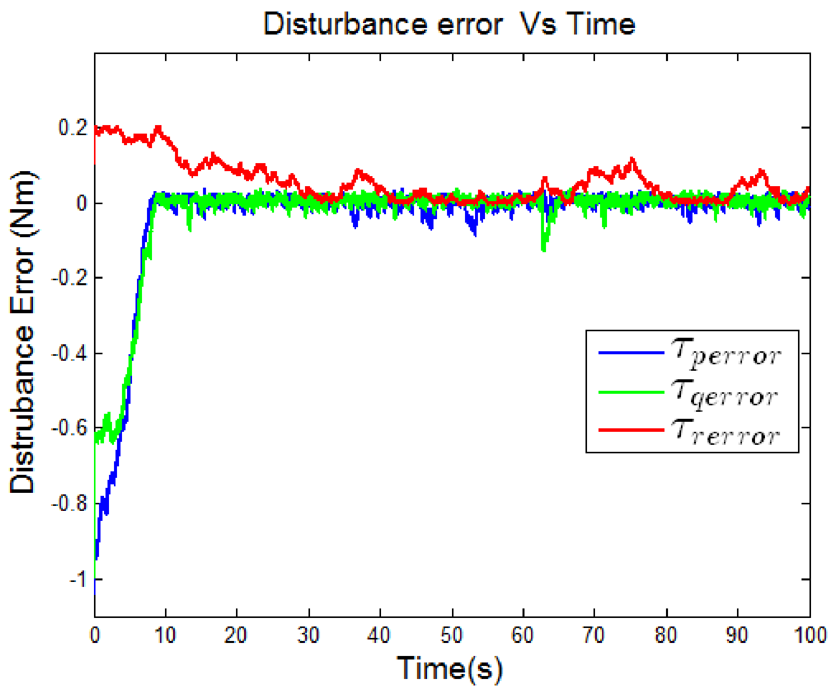

Figure 5-30: Deterministic Disturbance Estimate Error Vs Time.

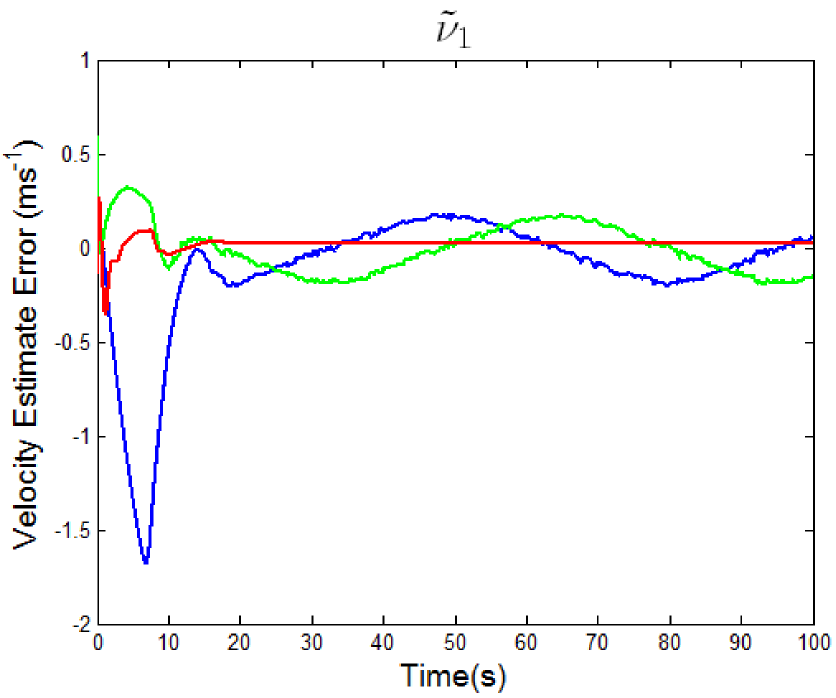

Figure 5-31: $\tilde{\nu}_1$ Vs Time.

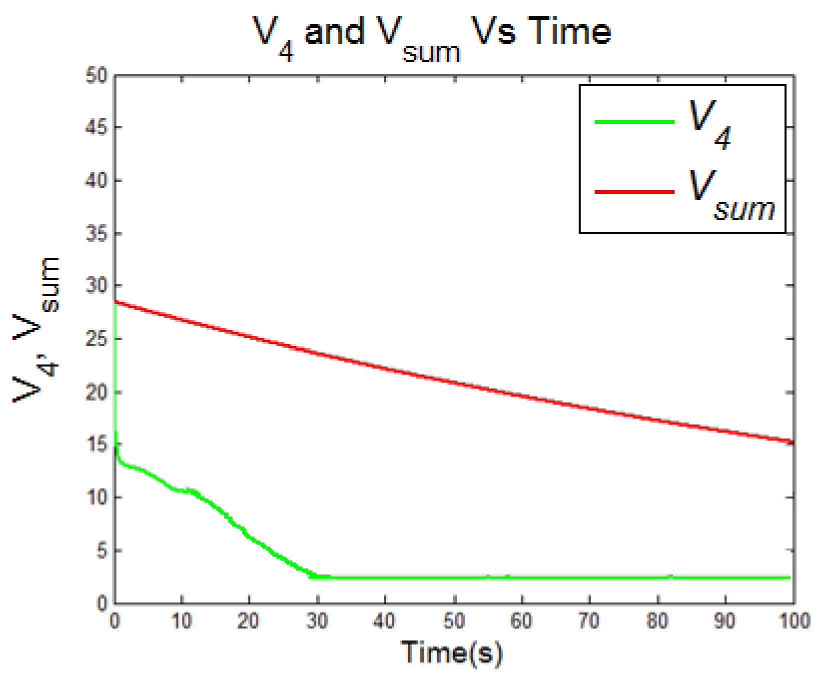

Figure 5-32: Lyapunov Function V_4 and V_{bound} Vs Time..

5.2.3 Stochastic Model and Deterministic Controller

Listed below are the system parameters. The table below shows all controller gains and mechanical properties of the quadrotor used for simulation.

Simulation Parameters						
Control Gains		Projection Algorithm				
Property	Value	Property	Value	Property	Value	Units
γ_1	5.12	$\mathbf{\Gamma}_{11}$	5	I_x	0.016507	Kg/m^2
γ_2	200	$\mathbf{\Gamma}_{22}$	5	I_y	0.016507	Kg/m^2
γ_3	0.1	$\mathbf{\Gamma}_{33}$	3	I_z	0.016284	Kg/m^2
$\mathbf{K}_{1,11}$	0.8			$I_{x,A}$	$0.2I_x$	Kg/m^2
$\mathbf{K}_{1,22}$	0.8			$I_{y,A}$	$0.2I_y$	Kg/m^2
$\mathbf{K}_{1,33}$	0.8			$I_{z,A}$	$0.2I_z$	Kg/m^2
$\mathbf{K}_{2,11}$	4			$\mathbf{D}_{1,11}$	0.25	$Kg/(m^2s)$
$\mathbf{K}_{2,22}$	4			$\mathbf{D}_{1,22}$	0.25	$Kg/(m^2s)$
$\mathbf{K}_{2,33}$	4			$\mathbf{D}_{1,33}$	0.125	$Kg/(m^2s)$
$\mathbf{K}_{3,11}$	10			$\mathbf{D}_{2,11}$	$2.5*10^{-3}$	$Kg/(m^2s)$
$\mathbf{K}_{3,22}$	10			$\mathbf{D}_{2,22}$	$2.5*10^{-3}$	$Kg/(m^2s)$
$\mathbf{K}_{3,33}$	10			$\mathbf{D}_{2,33}$	$0.03*10^{-3}$	$Kg/(m^2s)$
$\mathbf{K}_{4,11}$	10			m	2.25	kg
$\mathbf{K}_{4,22}$	10			m_a	0	kg
$\mathbf{K}_{4,33}$	10			c_{T}	$2.5569*10^{-5}$	
ε_1	7.6			c_{R}	$5.7768*10^{-6}$	
ε_2	3.9			Delta Time	10	µs
ε_3	$\frac{5}{5.1}\lambda_m(\mathbf{K}_3)$			Start time	0	s
$\varepsilon_4, \varepsilon_5, \varepsilon_6$	10			End Time	100s	s

Table 5-25: System Dynamics add Properties.

The following table shows the behaviour of the reference signal that the quadrotor needs to track.

Reference signal			
$\boldsymbol{\eta}_{1d,1} = 10\cos(0.1t)$	$\boldsymbol{\eta}_{1d,2} = 10\sin(0.1t)$	$\boldsymbol{\eta}_{1d,3} = 0.1t$	$\psi_d = 0.1t$

Table 5-26: Reference Position Signal.

The table below shows the initial conditions of quadrotors states.

Initial conditions							
State	Value	State	Value	State	Value	State	Value
$\boldsymbol{\eta}_{1,1}$	10(m)	$\boldsymbol{\nu}_{1,1}$	0	$\boldsymbol{\eta}_{2,1}$	0	$\boldsymbol{\omega}_{,1}$	0
$\boldsymbol{\eta}_{1,2}$	0(m)	$\boldsymbol{\nu}_{1,2}$	0	$\boldsymbol{\eta}_{2,2}$	0	$\boldsymbol{\omega}_{,2}$	0
$\boldsymbol{\eta}_{1,3}$	0(m)	$\boldsymbol{\nu}_{1,2}$	0	$\boldsymbol{\eta}_{2,3}$	0	$\boldsymbol{\omega}_{,3}$	0

Table 5-27: Initial Conditions.

The table below shows the deterministic components of the disturbances acting on the quadrotor.

Deterministic Disturbance Torque Profile			
Linear Disturbance		Torque Disturbance	
Property	Value(N)	Property	Value(Nm)
$\bar{\boldsymbol{f}}_x(t)$	0.5	$\bar{\boldsymbol{\tau}}_p(t)$	-1
$\bar{\boldsymbol{f}}_y(t)$	0.5	$\bar{\boldsymbol{\tau}}_q(t)$	-1
$\bar{\boldsymbol{f}}_z(t)$	0.1	$\bar{\boldsymbol{\tau}}_e(t)$	0.1

Table 5-28: Deterministic Disturbance Model.

In addition to the disturbances behaviour shown in the table above the stochastic component of the disturbances acting on the quadrotor is shown in the table below.

Stochastic Disturbance Torque Profile			
Linear Disturbance		Angular Disturbance	
Property	Value	Property	Value
$\Delta_{\boldsymbol{v}_1}$	$0.1\times\mathbf{I}_{3\times3}$	Δ_2	$1\times\mathbf{I}_{3\times3}$
		Δ_3	$0.1\times\mathbf{I}_{3\times3}$

Table 5-29: Stochastic Disturbance Model.

For clarification of the symbols please refer back to equation (2-119) and (2-120).

To be able to generate the stochastic disturbances used in this simulation the standard Wiener processes for both the linear position and angular position subsystems of the quadrotor are shown in the table below.

Wiener Process Profile		
Linear Disturbance	Angular disturbance	
$d\boldsymbol{w}_1 = \sqrt{dt}\boldsymbol{\mathcal{N}}(0,\Delta_{\boldsymbol{v}_1})$	$d\boldsymbol{w}_2 = \sqrt{dt}\boldsymbol{\mathcal{N}}(0,\Delta_2)$	$d\boldsymbol{w}_3 = \sqrt{dt}\boldsymbol{\mathcal{N}}(0,\Delta_3)$

Table 5-30: Wiener Process Profile.

The selection of the control gains, projection algorithm and system parameters and initial conditions specified in the above tables has been selected in accordance with the bounds prescribed by(4-174), (4-191), (4-202), (4-203) and (4-204). The bounds specified by these equations were calculated numerically in MATLAB and as such will not be demonstrated here.

Using the information presented in Table 5-26 to Table 5-30 we present the simulation results of the quadrotor in the following pages, the MATLAB code for this simulation can be found in Appendix J. The position reference trajectory $\boldsymbol{\eta}_{1d}$ and the position real trajectory $\boldsymbol{\eta}_1$ are plotted in Figure 5-33. The position error $\boldsymbol{\eta}_{1e}$ are plotted in Figure 5-34, attitude Euler error $\boldsymbol{\eta}_{2e}$ are plotted in Figure 5-35, the control thrust and control torques are plotted in Figure 5-36, the estimate of the disturbance torque error is plotted in Figure 5-37. The Lyapunov function V_4 and its bound $V_{4\,\text{bound}}$ are plotted in Figure 5-38. The error in the velocity estimate is plotted in Figure 5-31. It is seen from these figures, that all tracking errors do not converge to a bound.

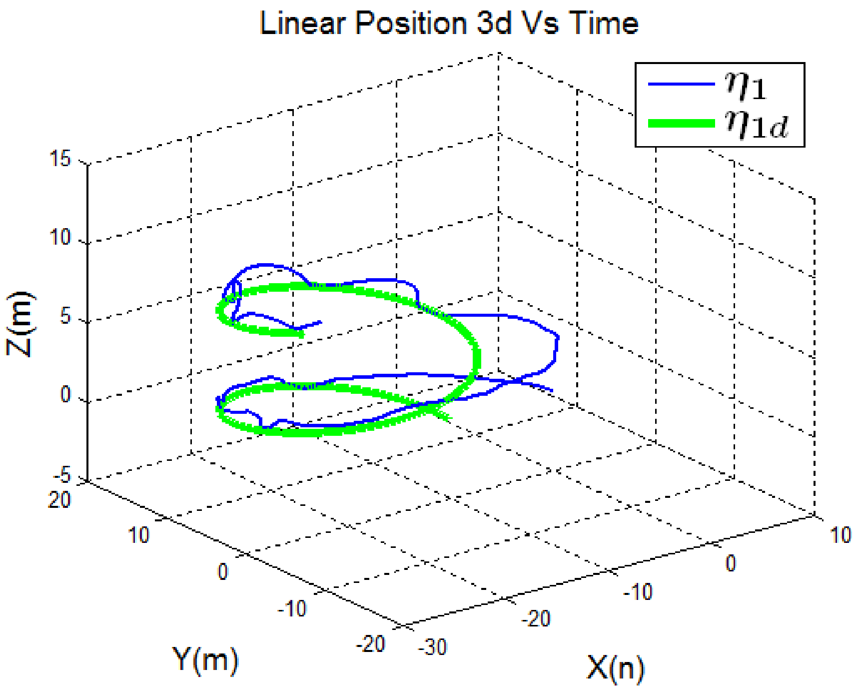

Figure 5-33: System 3-D Linear Position Vs Time.

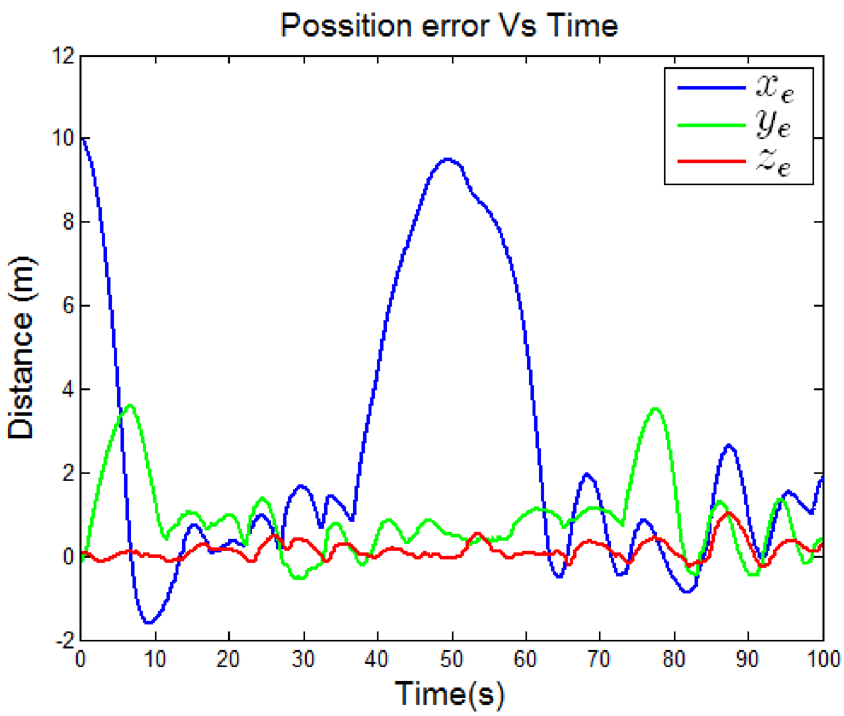

Figure 5-34: System Linear Position Error Vs Time.

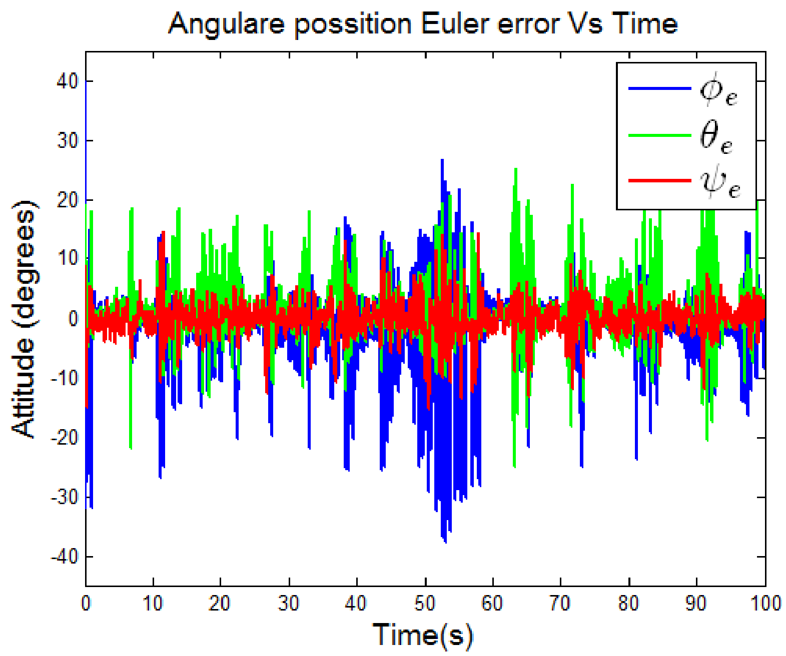

Figure 5-35: System Attitude Euler Error Vs Time.

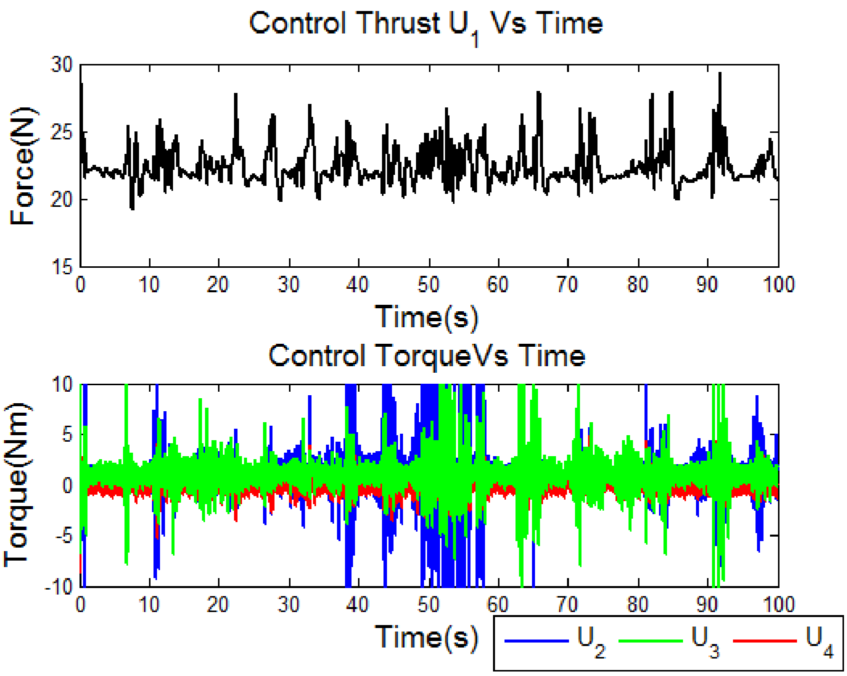

Figure 5-36: System Control Thrust and Torques Vs Time.

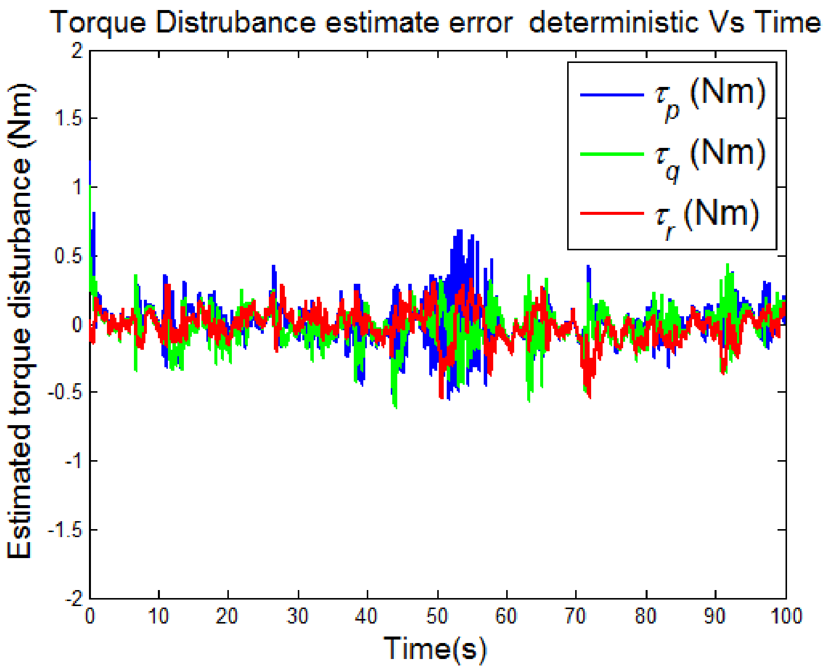

Figure 5-37: Deterministic Disturbance Estimate Error Vs Time.

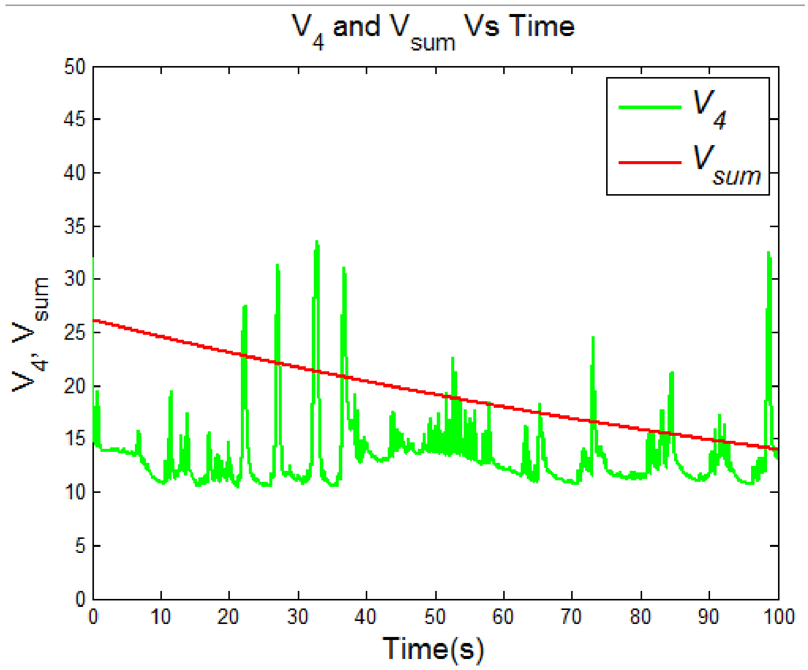

Figure 5-38: Lyapunov Function V_4 and V_{bound} Vs Time.

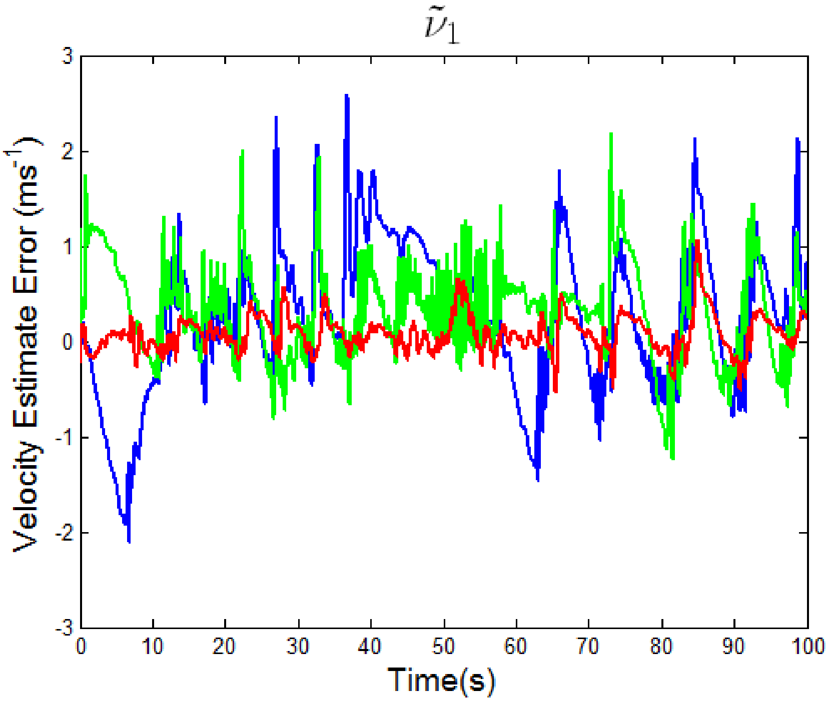

Figure 5-39: $\tilde{\nu}_1$ Vs Time.

In light of Figure 5-14 to Figure 5-17 it can be stated that the one -step ahead backstepping controller presented in section 4.1 is not suitable for stabilizing the aircraft under the conditions prescribed in Table 5-25 to Table 5-30. It is quite obvious comparing the results depicted in Figure 5-25 with Figure 5-33the effect of not taking into account stochastic disturbances on the system when designing the control law. Even though tithe controller presented in section 4.1 is asymptotically stable for deterministic conditions and shown to be good for these conditions in Figure 5-18 to Figure 5-24 it is not suitable for stochastic conditions. The new stochastic controller presented in section 4.2 has been shown in Figure 5-25 to Figure 5-32 to provide good tracking and stabilization in the presence of stochastic disturbances. respectively. Thus we have meet the last point of the project aims depicted in section 1.3.

5.3 Simulation Summery

In summery it can be stated that the new control scheme presented is effective and is capable of stabilizing the trajectory tracking of the quadrotor aircraft when operating in a disturbance filled environment. Furthermore, when compared to a deterministic controller of a similar type, the stochastic controller overwhelmingly outperformed the deterministic control scheme, highlighting the significance of taking into account the stochastic disturbances acting on the system. While the deterministic controller form section 4.1 does produces good results in the presence of deterministic conditions see for example the attitude error and the 3d plot of quadrotors trajectory in Figure 5-18 and Figure 5-20 respectively. Thus we have meet the last point of the project aims depicted in section 1.3.

6 Physical Implementation

In this chapter we implement the controllers simulated in chapter 5 on a physical system. This chapter is split up into two sections, equipment overview, attitude control and finally linear position control.

In the first section, we split up into three sub-sections firstly all hardware used and attached to the quadrotor is presented. In the second sub-section the method for obtaining control of the attitude system is presented. In the third sub-section, the test stand used to calibrate the attitude control system is presented and explained.

The second section is split up into four sub-sections. The first sub-section outlines the test setup and procedure used for calibrating the attitude control algorithm. In the second sub-section the behaviour of the quadrotor is presented when subjected to deterministic control. In the third section the system response is presented when subjected to stochastic control. Finally, comparisons between the simulation results and the actual experimental results are presented

6.1 Equipment

In this section all equipment used during all experiments are presented and the test stand constructed for obtaining attitude control of the quadrotor.

6.1.1 Quadrotor

Listed below are the parts comprising the quadrotor vehicle:

Part	Quantity	Manufacture	Part Name
Frame	1	Rc Timer	F-450 Flame wheel
Motors	4	RC timer	1100 kV
Propellers	4	APC	11x4.7
ESC	4	DJI	OPTO-30 A
Battery	1	REVLECTRIX	Lithium Polymer 4 Cell 5200 mAh at 45C
IMU	1	SparkFun	9 DOF Razor IMU
Arduino	1	Arduino	Mega 2560
GPS receiver module	1	U-Blox	Neo 7N
Antenna	1		
Operating software		MATLAB	R2015b
Wireless communication	6	XBee	XBee Pro 60mW Wire Antenna - Series 1 (802.15.4)
XBee cradle	3	SparkFun	SparkFun XBee Explorer Regulated
XBee USB connection	3	SparkFun	SparkFun XBee Explorer Dongle
Power	4	Energizer	AA battery
breadboard	1	SparkFun	Breadboard - Small Self-Adhesive
breadboard	2	SparkFun	Breadboard - Mini Modular
Altitude/Pressure Sensor Breakout	1	SparkFun	MPL3115A2

Table 6-1: All Equipment Used by Quadrotor.

All the parts presented above were either sourced from the online store SparkFun or local hobby shops around the Perth metropolitan area.

6.1.2 Control System

To obtain attitude control of the quadrotor and select the control gains needed to obtain attitude stabilisation all control processing was done through MATLAB running on a Windows pc, with the control signals sent to the Arduino via a serial port to generate the PWM signals for the ESC's. The IMU calculates the angular position estimate on board the quadrotor and sends the attitude and angular velocities about each axis back to MATLAB via a serial port. A flow chart of the control system is show below:

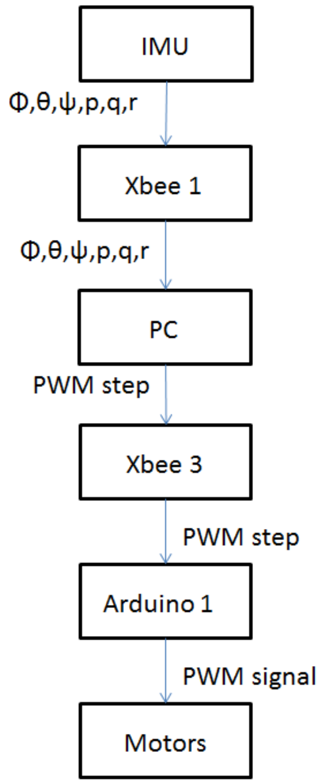

Figure 6-1: Quadrotor Attitude Control Flow Chart.

It should be noted that the code running on the IMU can be found in Appendix L and is a slightly modified version of the code recommended by the IMU manufacturer SparkFun which can be found on GitHub[42].

6.1.3 Test Stand

To allow for testing and control gain selection of the attitude stabilization control system a test stand was constricted and shown below, to hold the aircraft in place about a fixed pivot point

Figure 6-2: Quadrotor Test Stand.

Figure 6-3: Quadrotor Test Stand Base.

The test rig as shown above in Figure 6-2 is constructed of single timber leg with an angled piece of steel attached at the top with a ball and socket joint connected to the underside of a base plate to allow for rotation about all three axes. The base of the test stand is made of ply wood as shown in Figure 6-3 with house bricks used to keep the stand firmly anchored to the ground, all parts are listed below:

Part	Quantity	Manufacture	Part Name
Ball and socket joint	1	Chapman	UNF-Ball Joint Linkage 5/16
Spring	1	Great Vigor Models	MV1393314 1.4mm shock spring rear 1.4mm L =80.5, Blue
Right angle steel bracket	1	Bunnings	N/A
Pine wood leg	1	Bunnings	90 x 35 x 1200
Pine wood base	1	N/A	N/A
Pine wood top plate	1	N/A	120 x 180 x 12
House brick	6	N/A	220 x 70 x 100

Table 6-2: Test Stand Parts.

The ball and socket joint connecting the test plate to the platform leg is shown below:

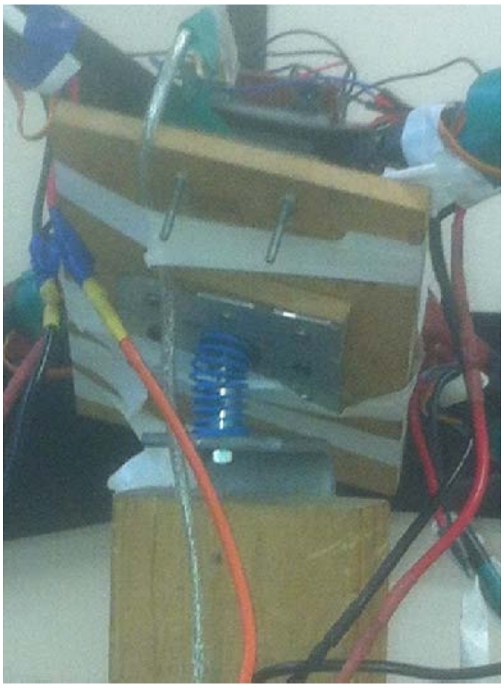

Figure 6-4: Quadrotor Test Stand, Ball and Socket Joint.

The inclusion of the spring around the ball and socket joint was introduced to cut down on vibration introduced by the high speed motor propeller system on the frame. Due to the high speeds at which the propellers rotate and the height of the stand, it was noticed that that vibration was travelling up and down the test stand leg and had a noticeable effect on the control performance. These vibrations where severally affecting the accelerometer readings on the IMU even when an exponential filter and a second order low pass filter were implemented on the IMU.

6.2 Angular Position Stabilisation

6.2.1 Set up

The test set up is shown below the Multi rotor is firmly attached to the stand via four M4 bolts passing through the base of the multi rotor and the pivoting platform on top of the test stand. As highlighted below

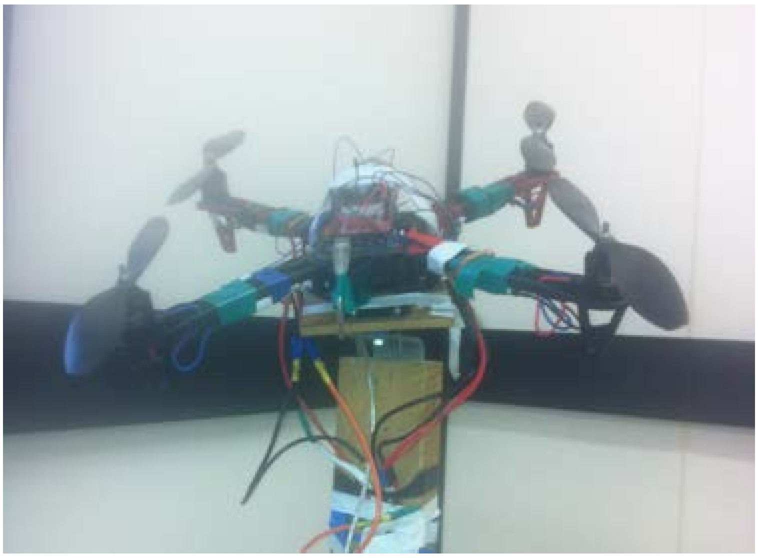

Figure 6-5: Quadrotor Attached to Test Stand.

The batteries where attached to the stand and not the quadrotor for initial testing as it made access to them when a change out was needed easier as highlighted in the figure below.

Important system parameters					
Frame Properties		Propeller Properties		Electronics Properties	
Property	Initial Value	Property	value	Property	Value
I_x	0.016507(kg/m^2)	$Prop\ Radis$	6.5"	Baud rate	115200
I_y	0.016507(kg/m^2)	Prop pitch	4.7	Controller update rate	55 (Hz)
I_z	0.016284(kg/m^2)	c_{T}	2.5569*10^{-5}	Battery Voltage	14.8(V)
$\mathbf{I}_A$	$0.5\mathbf{I}_H$	c_{R}	5.7768*10^{-6}		
$Copter\ Radis$	0.225(m)				

Table 6-3: Quadrotor System Physical and Electronic Properties.

6.2.2 System Response

In the preceding two sub-sections the quadrotor response when controlled by both deterministic and stochastic control is presented.

6.2.3 Deterministic Control

System Parameters					
Control gains				Disturbance Observer gains	
$\mathbf{K}_{1,11}$	10	$\mathbf{K}_{2,11}$	10	$\mathbf{K}_{d,11}$	10
$\mathbf{K}_{1,22}$	10	$\mathbf{K}_{2,22}$	10	$\mathbf{K}_{d,22}$	10
$\mathbf{K}_{1,33}$	2	$\mathbf{K}_{2,33}$	2	$\mathbf{K}_{d,33}$	10

Table 6-4: Deterministic Attitude Controller Properties.

The MATLAB code for this controller can be found in Appendix N. After many attempts stable flight was obtained with a steady state error of ±3.5^{O} for each axis as highlighted in the figure below

Using the above information presented in Table 6-4 we present the experimental results, the MATLAB code for this controller can be found in Appendix N. The attitude reference trajectory $\boldsymbol{\eta}_{2d}$ and the attitude real trajectory $\boldsymbol{\eta}_2$ are plotted in Figure 6-6. The deterministic torque estimate is presented in Figure 6-7. The control torques and motor PWM signals are presented in Figure 6-8 and Figure 6-9 respectively. It is seen from these figures, that the attitude converges to the reference trajectory.

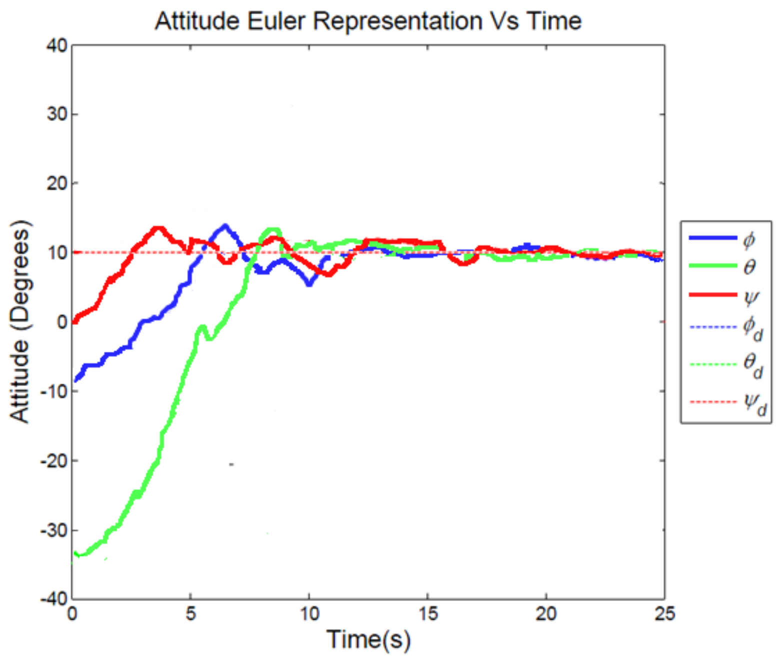

Figure 6-6: Deterministic Attitude Control Attitude Response Vs Time.

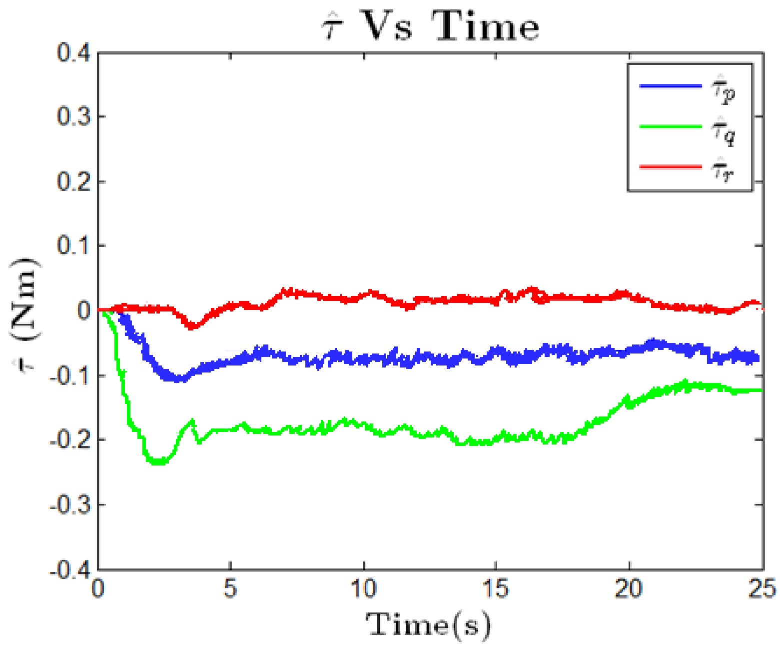

Figure 6-7: Deterministic Attitude Control Disturbance Estimate Vs Time.

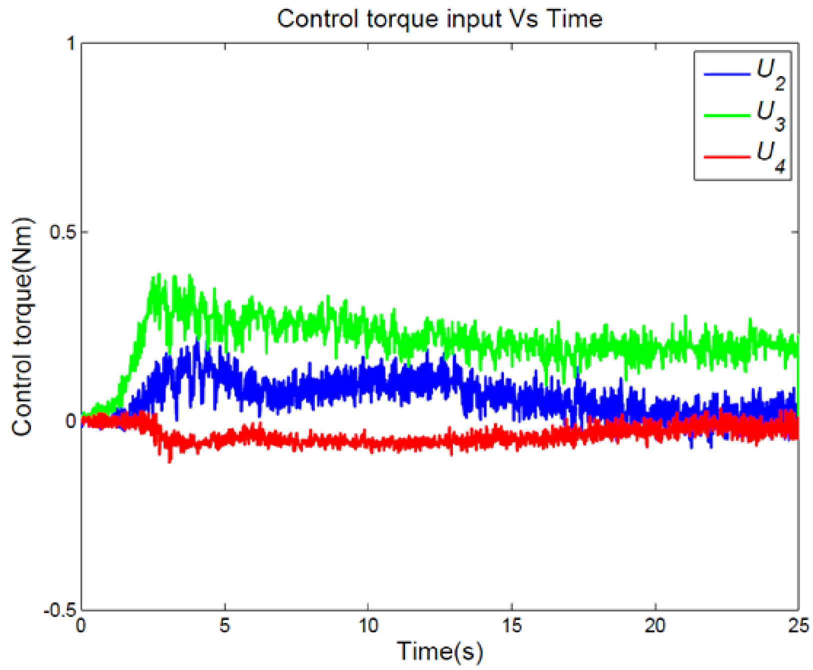

Figure 6-8: Deterministic Attitude Control, Control Torque Signals Vs Time.

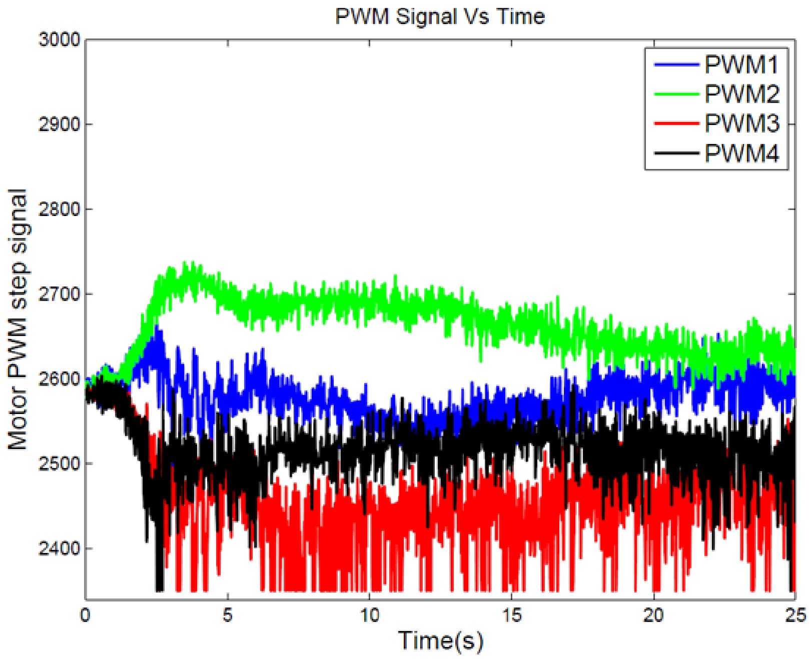

Figure 6-9: Deterministic Attitude Control Motor Signals Vs Time.

6.2.4 Stochastic Control

To obtain stable attitude control of the quadrotor the following control and projection algorithm parameters were used.

System Parameters			
Control Gains		Projection Algorithm	
$\mathbf{K}_{1,11}$	4.2660	$\mathbf{\Gamma}_{11}$	0.02
$\mathbf{K}_{1,22}$	4.2660	$\mathbf{\Gamma}_{22}$	0.02
$\mathbf{K}_{1,33}$	3.1284	$\mathbf{\Gamma}_{33}$	0.02
$\mathbf{K}_{2,11}$	3.4000	$\omega_{\mathrm{M},\bar{\boldsymbol{\tau}}_{Aero}}$	0.1
$\mathbf{K}_{2,22}$	3.4000	$\xi_{\bar{\boldsymbol{\tau}}_{Aero}}$	0.1
$\mathbf{K}_{2,33}$	2.0400	μ	10^{-8}
		$\omega_{\mathrm{M},\delta}$	10^{-4}
		ξ_{δ}	0.000051

Table 6-5: Stochastic attitude controller properties.

The MATLAB code for this controller can be found in Appendix O. Using the parameters listed in the table above table attitude control of the quadrotor was obtained, as evident in the figure below showing the difference between the attitude and reference signal

Using the above information presented in Table 6-5 we present the experimental results, the MATLAB code for this controller can be found in Appendix N. The attitude reference trajectory $\boldsymbol{\eta}_{2d}$ and the attitude real trajectory $\boldsymbol{\eta}_2$ are plotted in Figure 6-10. The deterministic torque estimate is presented in Figure 6-11 and the estimates of $\hat{\delta}_1$ and $\hat{\delta}_2$ in Figure 6-12 and Figure 6-13 respectively. The control torques and motor PWM signals are presented in Figure 6-14and Figure 6-15 respectively. It is seen from these figures, that the attitude converges to the reference trajectory.

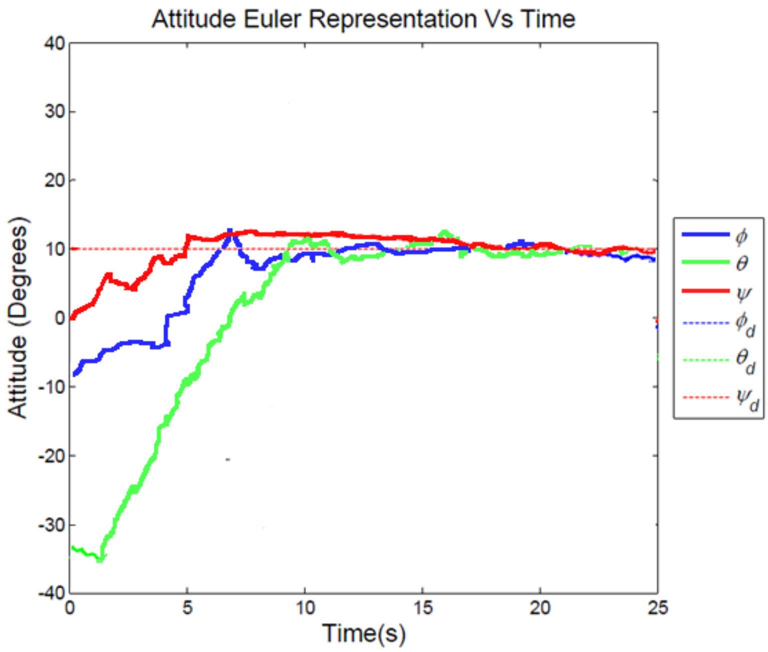

Figure 6-10: Stochastic attitude control Attitude Response Vs Time.

As can be seen in the figure above all 3 angular states reach a steady state error of within 1.5 degrees of the target attitude this is a good and reasonable result as it will allow for stable flight. Also it should be noted that the pivot point of the system is located 50mm below the centre of mass of the aircraft and no matter what precautions were made vibration in the test rig leg could not be removed, in fact the vibration induced by the motors could be felt travelling along the ground as far as 8 meters away. The figure below shows the estimate of the disturbance torque acting on the system.

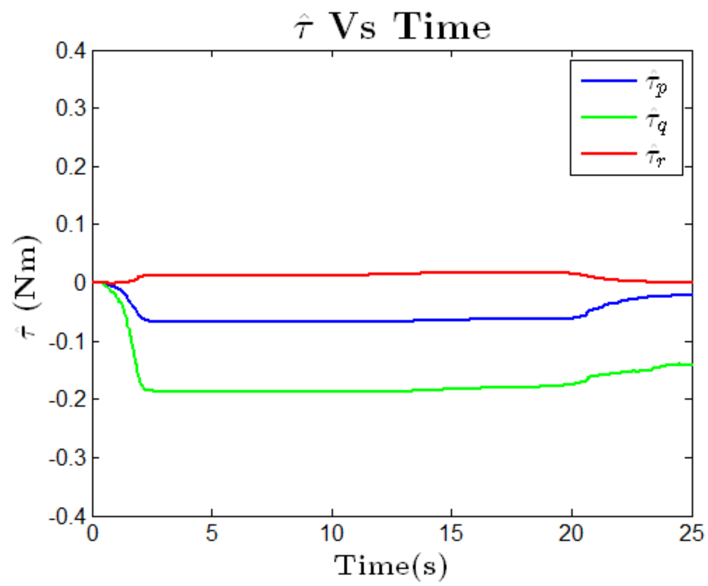

Figure 6-11: Stochastic Attitude Control Disturbance Estimate Vs Time.

Note the smooth estimate associated with the projection algorithm in contrast with the disturbance observer in Figure 6-7.

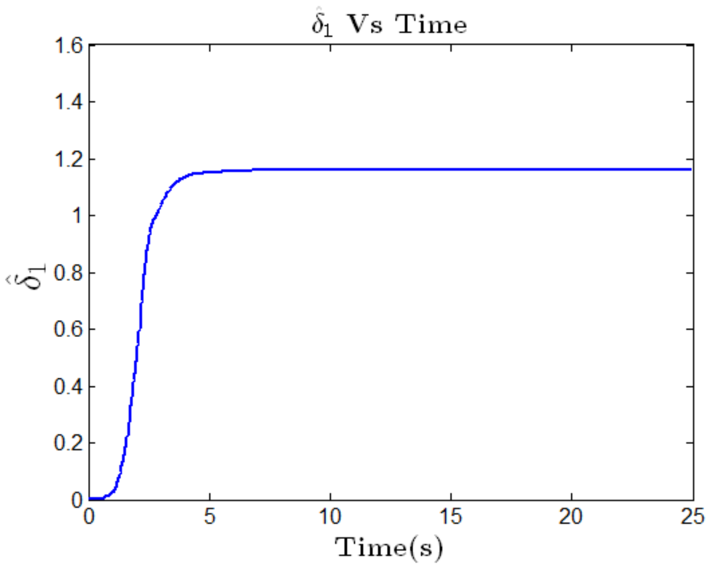

Figure 6-12: Stochastic Attitude Control, $\hat{\delta}_1$ Vs Time.

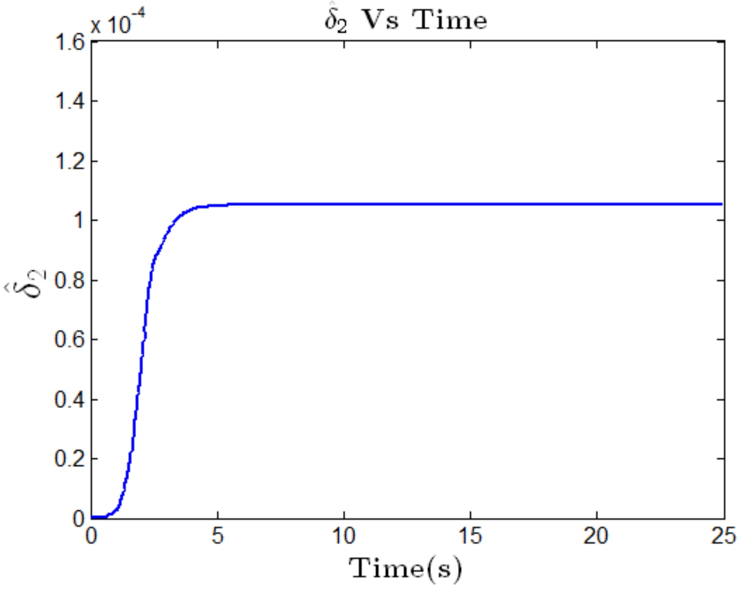

Figure 6-13: Stochastic Attitude Control, $\hat{\delta}_2$ Vs Time.

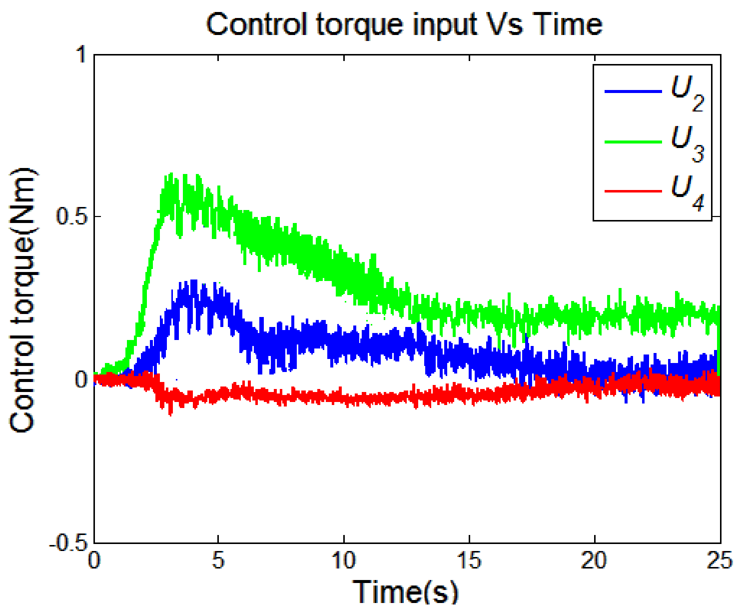

Figure 6-14: Stochastic Attitude Control Torques Vs Time.

The figure above show the control torques produced by the control scheme while there is noise present in the output this is to compensate for the stochastic noise present in the test rig but as we can see the control torque are bounded and well within reason.

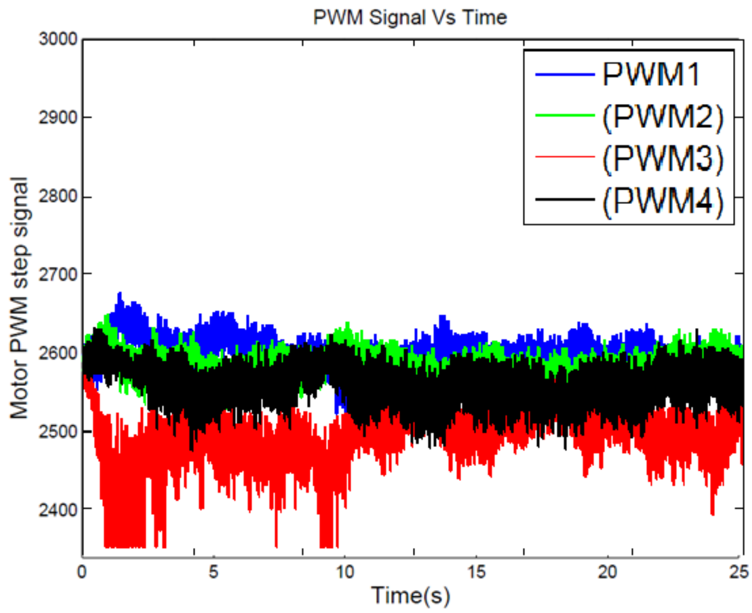

Figure 6-15: Stochastic Attitude Control Motor Signals Vs Time.

6.2.5 Comparison

Comparing the attitude response of the deterministic controller (see Figure 6-6) with the stochastic controller (see Figure 6-10), less noise and a smoother trajectory is observed for the stochastic controller. During testing there was no external wind source blowing on the aircraft. However disturbances were present in the form of vibrations and gamble lock in the test rig. These disturbances were able to highlight the stochastic backstepping controllers robustness to disturbances over the deterministic controller. This result warrants practical implementation of the new one step ahead backstepping stochastic controller presented in this thesis. Thus we have meet the fifth point of the project aims depicted in section 1.3.

7 Discussion & Conclusions

The quadrotor model used in this project represents all dynamics between the motor input and the motions of the quadrotor. Therefore, it is useful to derive and simulate a control law, that influences the aircraft's position by altering the motor input. Controller design incorporates stochastic and deterministic aerodynamic wind effects as well as state feedback to achieve velocity state estimation.

7.1 Discussion

The proposed control law design ensures that the closed loop quadrotor system is stochastically practically asymptotically stable. It allows for tracking errors, unmeasured state estimation errors and disturbance estimation errors. All state errors converge to a ball centred at the origin in sufficiently short time. Therefore, the control law is useful for tracking of a reference trajectory in a disturbance filled environment.

Furthermore, the initial conditions are reasonably relaxed, because the MRP's are used for attitude representation in the quadrotor model. This provides a wide flight envelope allowing the aircraft to fly at attitudes between $\pm 2\pi$. In order to avoid singularities, the absolute sum of each element of the quadrotors attitude must be bounded within a reasonably large domain of 2π. To ensure solution boundedness the initial state errors may only exist in a specifically large domain around the origin. Using the unit quaternion to represent the attitude of the aircraft would avoid restrictions of the pitch, roll and yaw angle and the initial states could be chosen more freely. However, conditions will still need to be placed on the size of the external disturbances acting on the system and their corresponding disturbance profile as to the maximum torque and thrust produced by each motor.

The numerical simulation verifies, that the stochastic control law is valid for the chosen model and performs better than a deterministic control law when faced with stochastic disturbances, (see Figure 5-26, Figure 5-27 and Figure 5-34, Figure 5-35).

Experimental flights are necessary to verify the simulations results. In order to transfer the simulation results to a real flight, the same restrictions as for the model and control design were applied. When testing the attitude stabilisation, it was found that the stochastic backstepping controller performed better than the deterministic backstepping controller for stabilizing the aircrafts attitude, (see Figure 6-6 and Figure 6-10)

While programming the simulation the problem of verifying the programmed equations occurred often. Derivatives of formulas can be checked by comparing the results of the derivative to the numerical calculation of the derivative with small step sizes. Validating other equations is more difficult, as the results are not known in general as such all equations were methodically broken up into smaller elements checked and rebuilt up to make the corresponding equation with each part checked meticulously and individually. However due to the complexity and size of some of the equations used in this thesis it is not reasonable to calculate exact solution. Rather bounds needed to be calculated which are based on the individual components of each equation. When checking the calculated derivatives of equations, the MATLAB Symbolic toolbox was used to compare the calculated symbolic derivatives with those calculated by MATLAB and when discrepancies were found the partial derivatives were checked and solutions found. Due to the control design chosen the only unbounded states that appear are in the design of the control torques resulting from the Coriolis and centripetal effect. None of the previous designs states have any unbounded terms.

7.2 Conclusion

In this thesis an in-depth model of a quadrotor aircraft has been presented taking into account both deterministic and stochastic conditions. In addition, the Modified Rodrigues Parameters have been used for attitude representation to reduce the effects of singularities in the angular system equations of motions. Thus we have meet the first point of the project aims depicted in section 1.3.

In total four control algorithms have been designed and tested. Firstly, two control algorithms have been presented in Chapter 3 for attitude stabilisation. Both controllers are based on the backstepping design method, one controller is designed about a deterministic model and the other is based around a stochastic model. Both control methods employ disturbance estimation and rejection measures. Thus we have meet the third point of the project aims depicted in section 1.3. While both control strategies take into account external disturbances acting on the system, it has been shown through simulation that when faced with stochastic disturbances acting on the system the stochastic controller out performs the deterministic controller. Moreover when tested on a physical quadrotor the stochastic controller out preformed the deterministic controller. Thus we have meet the fourth and fifth point of the project aims depicted in section 1.3.

The second set of two controllers presented in Chapter 4, are both based on the one-step ahead backstepping and backstepping control design methods to allow the quadrotor to track a linear position trajectory and yaw position trajectory in three dimensional space. While both controllers take into account disturbances acting on the system, one is based on a deterministic model of the aircraft, and the latter is based on a stochastic model of the aircraft. The stochastic controller takes into account air currents flowing over the aircraft during flight, and wind ghosts experienced during flight. Both controllers use linear velocity estimators to estimate the linear velocity of the aircraft, as full state feedback is not possible, only output state feedback is possible. Thus we have meet the sixth point of the project aims depicted in section 1.3.

Numerical simulation results in Chapter 5 show that each controller when tested under each of their deterministic and stochastic models respectively, perform well ensuring tracking of a reference signal in 3D space. However, when the deterministic controller is tested under the same stochastic conditions as the stochastic controller, it is shown that not taking into account the stochastic disturbances acting on the aircraft has an adverse effect on the aircrafts performance. Thus we have meet the last point of the project aims depicted in section 1.3.

In summary the proposed control law is attractive for applications where exact tracking is required in disturbance filled environments, such as an open field with wind gusts. The effectiveness of the control law regarding trajectory tracking and disturbance rejection, has been validated by simulation and through limited practical testing. However, a comprehensive tethered and untethered field testing program is required to fully verify the aircrafts behaviour under the proposed control law in an open turbulent environment. Both a tethered and untethered test program will require appropriate approvals from the Civil Aviation Safety Authority Certification.

7.3 Future work

It needs to be pointed out that complete autonomous flight and GPS way point tracking has not been conducted at any point during this project. This is because in accordance with the Civil Aviation Authority (CASA) If you are piloting an unmanned aircraft, you will need general aviation knowledge in line with that required for a private pilot's license. More specifically an UAV controller's certificate and an unmanned operator's certificate (UOC), is required[43]. as such future work would be to first obtain GPS controlled autonomies flight. Once this has been achieved the aircraft use can move into these areas:

1. The aircraft could be used to follow a small Unmanned Marine Vessel (UMV) travelling up and down the swan river for a verity of differing research purposes such as testing a new control algorithm on a UMV.
2. The aircraft could be fitted with a sensor such as LADAR or an RGBD sensor used in the Microsoft Kinect for either security patrols or 3dD mapping of hard to reach places.
3. A new neural network controller could be developed for the aircraft and tested to see how a neural network controller would compare to the proposed stochastic one-step ahead backstepping controller in this thesis.
4. A set of small jet engines could be built and equipped to the aircraft instead of the current propellers and motors used. Then examine the power to weight ratio of the aircraft, flight duration and noise produced when compared to the current motor propeller system to determine if the current system is in fact the best means of producing lift.
5. Attach actuators to each motor to allow each motor to rotate and change the direction of thrust relative to the quadrotor aircrafts body.
6. Add more motors to the aircraft allowing motor failure to be managed and introduce the ability to detect motor failure into the control law.
7. Perform a feasibility study of developing a mended sized aircraft.
8. Water proof the aircraft so that it is capable of operating underwater.
9. Produce a number of these quadrotor aircraft and implement a swarm of quadrotors using the proposed one-step ahead stochastic backstepping method.

These three points are pieces of work that the author of this thesis would be interested in pursuing in a future project.

8 Bibliography

[1] K. D. Do and J. Pan, Control of Ships and Underwater Vehicles: Design for Underactuated and Nonlinear Marine Systems, London: Springer, 2009.

[2] J. Hauser, S. Sastry and G. Meyer, “Nonlinear control design for slightly non-minimum phase systems: application to V/STOL aircraft,” *Automatica,* vol. 28, no. 4, p. 665–679, 1993.

[3] C.-S. Huang and K. Yuan, “Output tracking of a nonlinear non-minimum phase PVTOL aircraft based on nonlinear state feedback,” *International Journal of Control,* vol. 75, no. 6, p. 466–473, 2002.

[4] F. Lin, W. Zhang and R. Brandt, “Robust hovering control of a PVTOL aircraft,” *IEEE Transactions on Control Systems Technology,* vol. 7, no. 3, p. 343 – 351, 1999.

[5] P. Martin, S. Devasia and B. Paden, “A different look at output tracking control of a VTOL aircraft,” *Automatica,* vol. 32, no. 1, p. 101–107, 1996.

[6] R. Olfati-Saber, “Global configuration stabilization for the VTOL aircraft with strong input coupling,” *IEEE Transactions on Automatic Control,* vol. 47, no. 11, p. 1949–1952, 2002.

[7] K. D. Do, Z. P. Jiang and J. Pan, “Global tracking control of a VTOL aircraft without velocity measurements,” *IEEE Transactions on Automatic Control,* vol. 48, no. 12, p. 2212–2217, 2003.

[8] P. Setlur, D. Dawson, Y. Fang and B. Costic, “Nonlinear tracking control of the VTOL aircraft,” in *Proceedings of the 40th IEEE Conference on Decision and Control*, Florida USA, 2001.

[9] S. M. Joshi, A. G. Kelkar and J. T.-Y. Wen, “Robust attitude stabilization of spacecraft using nonlinear quaternion feedback,” *IEEE Transactions on Automatic Control,* vol. 40, no. 10, p. 1800–1803, 1995.

[10] A. Tayebi and S. McGilvray, “Attitude stabilization of a VTOL quadrotor aircraft,” *IEEE Transactions on Control Systems Technology,* vol. 14, no. 3, pp. 562-571, 2006.

[11] P. Castillo, A. Dzul and R. Lozano, “Real-time stabilization and tracking of a four-rotor mini rotorcraft,” *IEEE Transactions on Control Systems Technology,* vol. 12, no. 4, p. 510–516, 2004.

[12] T. Madani and A. Benallegue, “Control of a quadrotor mini-helicopter via full state backstepping technique,” in *Proceedings of the 45th IEEE Conference on Decision & Control*, San Diego, 2006.

[13] Z. Zuo, “Trajectory tracking control design with command-filtered compensation for a quadrotor,” *IET Control Theory and Applications,* vol. 4, no. 11, pp. 1515-1520, 2010.

[14] A. Abdessameud and A. Tayebi, “Global trajectory tracking control of VTOL-UAVs without linear

velocity measurements," *Automatica,* vol. 46, no. 6, pp. 1053-1059, 2010.

[15] A. Roberts and A. Tayebi, "Adaptive position tracking of VTOL UAVs," *IEEE Transactions on Robotics,* vol. 27, no. 1, pp. 129-142, 2011.

[16] Y. C. Choi and H. S. Ahn, "Nonlinear control of quadrotor for point tracking: Actual implementation and experimental tests," *IEEE/ASME Transactions on Mechatronics,* vol. 20, no. 3, pp. 1179-1192, 2015.

[17] X. Ding and Y. Yu, "Motion planning and stabilization control of a multipropeller multifunction aerial robot," *IEEE/ASME Transactions on Mechatronics,* vol. 18, no. 3, pp. 645-656, 2013.

[18] D. Lee, A. Franchi, H. I. Son, C. Ha, H. H. Bulthoff and P. R. Giordano, "Semiautonomous haptic teleoperation control architecture of multiple unmanned aerial vehicles," *IEEE/ASME Transactions on Mechatronics,* vol. 18, no. 4, p. 1334–1345, 2013.

[19] M. Xin, Y. Xu and R. Hopkins, "Trajectory control of miniature helicopters using a unified nonlinear optimal control technique," *Journal of Dynamic Systems, Measurement, and Control,* vol. 13, no. 6, p. 061001–1:061001–14, 2011.

[20] A. Zavala-Rio, I. Fantoni and R. Lozano, "Global stabilization of a pvtol aircraft model with bounded inputs," *International Journal of Control,* vol. 76, no. 18, pp. 1833-1844, 2003.

[21] P. Castillo, A. Dzul and R. Lozano, "Real-time stabilization and tracking of a four-rotor mini rotorcraft," *IEEE Transactions on Control Systems Technology,* vol. 12, no. 4, pp. 510-516, 2004.

[22] A. Ailon, "Simple tracking controllers for autonomous VTOL aircraft with bounded inputs," *IEEE Transactions on Automatic Control,* vol. 55, no. 3, p. 737–743, 2010.

[23] A. Zavala-Rio, I. Fantoni and G. Sanahuja, "Finite-time observer-based output-feedback control for the global stabilisation of the pvtol aircraft with bounded inputs," *International Journal of Systems Science,* vol. 47, no. 7, p. 1543–1562, 2014.

[24] A. R. Teel, "Global stabilization and restricted tracking for multiple integrators with bounded control," *Systems and Control Letters,* vol. 18, no. 3, p. 165–171, 1992.

[25] K. D. Do and J. Paxman, "Global Tracking Control of Quadrotor VTOL Aircraft in Three Dimensional Space," *Control Conference (AUCC), 2013 3rd Australian,* p. 26–33, 2013.

[26] W. C. Lindsey , Synchronization in Communication and Control, New Jersey: Prentice Hall, 1972.

[27] H. k. Khalil, Non Linear Systems, 3rd ed., A. Dworkin, Ed., Upper Saddle River, New Jersey: Prentice-Hall, inc, 2002, pp. 150 - 617.

[28] X. Mao, Stochastic Differential Equations and Applications, Second Eddition ed., Glasgow: Woodhead Publishing, 2008.

[29] H. Deng , M. Krstic and R. J. Williams, “Stabilization of Stochastic Nonlinear Systems Driven by Noise of Unknown Covariance,” *IEEE Transactions on Automatic Control,* vol. 46, no. 8, pp. 1237 - 1253, 2001.

[30] K. D. Do, “Bounded controllers for global path tracking control of unicycle-type mobile robots,” *Robotics and Autonomous Systems,* vol. 61, no. 8, p. 775–784, 2013.

[31] T. J. Koo and S. Sastry, “Output tracking control design of a helicopter model based on approximate linearization,” *Proceedings of the 37th IEEE Conference on Decision and Control,* vol. 4, pp. 3635 - 3640 , 1998.

[32] D. M. Boskovic and M. Krstic, “Global attitude/position regulation for underwater vehicles,” *Proceedings of the 1999 IEEE International Conference on Control Applications,* vol. 2, pp. 1768 - 1773, 1990.

[33] Z. Zuo and P. Ru, “Augmented L1 adaptive tracking control of quad-rotor unmanned aircrafts,” *IEEE Transactions on Aerospace and Electronic Systems,* vol. 50, no. 4, p. 3090 – 3101, 2014.

[34] M. Corless and G. Leitmann, “Adaptive control of systems containing uncertain functions and unknown functions with uncertain bounds,” *Journal of Optimization Theory and Applications,* vol. 41, no. 1, pp. 155 - 168, 1983.

[35] M. Krstic, I. Kanellakopoulos and P. Kokotovic, Nonlinear and adaptive control design, New York: Wiley, 1995.

[36] K. D. Do and J. Pan, “Nonlinear control of an active heave compensation system,” *Ocean Engineering,* vol. 35, no. 5 - 6, pp. 558 - 571, 2007.

[37] H. Deng and M. Krstic, “Output-feedback stabilization of stochastic nonlinear systems driven by noise of unknown covariance,” *Systems & Control Letters,* vol. 39, no. 3, p. 173{182, 2000.

[38] H.-B. Ji and H.-S. Xi, “Adaptive Output-Feedback Tracking of Stochastic,” *IEEE Transactions on Automatic Control,* vol. 51, no. 2, pp. 355 - 360, 2006.

[39] J. B. Pomet and L. Praly, “Adaptive nonlinear regulation: estimation from the Lyapunov equation,” *Automatic Control, IEEE Transactions on,* vol. 37, no. 6, pp. 729-740, 1992.

[40] K. D. Do and F. DeBoer, “Smooth projection robust adaptive nonlinear control of uncertain time varying nonlinear systems,” *ISA,* vol. 2, p. 988–994, 2000.

[41] K. D. Do and J. Paxman, “Global Tracking Control of Quadrotor VTOL Aircraft in Three Dimensional Space,” in *Control Conference (AUCC), 2013 3rd Australian* , Perth, 2013.

[42] N/A, “ptrbrtz/razor-9dof-ahrs,” GitHub, 1 April 2014. [Online]. Available: https://github.com/ptrbrtz/razor-9dof-ahrs. [Accessed 1 April 2014].

[43] C. A. S. Authority, “Civil Aviation Safety Authority,” Civil Aviation Safety Authority, 2016. [Online]. Available: https://www.casa.gov.au/operations/standard-page/model-aircraft-and-rpa. [Accessed 04 Febuary 2016].

[44] J. G. Leishman, Principles of Helicopter Aerodynamics, Second ed., New York: Cambridge University Press, 2006, pp. 183 - 186.

[45] H. Huang , G. M. Hoffmann, S. L. Waslander and C. J. Tomlin, “Aerodynamics and Control of Autonomous Quadrotor Helicopters in agaggressive,” in *EEE Int. Conf. on Robotics and Automation, 2009. ICRA ’09.*, Kobe, 2009.

[46] L. Wu, “Large and moderate deviations and exponential convergence for stochastic damping Hamiltonian systems,” *Stochastic Processes and their Applications,* vol. 91, no. 2, pp. 205 - 238, 2011.

[47] K. D. Do, “Control of fully actuated ocean vehicles under stochastic environmental loads in three dimensional space,” *Ocean Engineering,* vol. 99, pp. 34 - 43, 2015.

Appendix A –Code for Trig Backstepping Examples

```
clc
clear all
close all
% Simulation parameters
End_Time                =10
Delta_Time              = 1/10000;
Time_steps              = End_Time / Delta_Time;
alpha_1 = 1.1
Loop_Count = 1;
%initialize state variables
x_1_d = 0;
x_1 =-0.5;
x_2 =0.93;
x_2 =0;
results             = zeros(13, Time_steps);
results_2               = zeros(6, Time_steps);
alpha_1 = 0;
x_1_d_dot =1.3705;
x_1_d_dot =1.255;
%x_1_d_dot =-0.158;
x_1_d_double_dot = 0;
x_1_d_old = x_1_d - x_1_d*Delta_Time;
x_1_d_old = x_1_d;
% controller gains
  d_dot = 0;
 x_1_dot = 0;
  x_2_dot = 0;
i=0;
u=0;
z_1 = 0 ;
z_2 = 0;

m=1
x_1_d_old= x_1_d ;
k1=5.2
k1=5
k1=5.5
%k1=2
k2=2.5
%k2=2.5
sigma_1= 1 * 10^-3;
k_d =200
epsilon_1 = 2.4
epsilon_1 = 2.4/2.5*k2
epsilon_2 = 0.10;
gamma = 2 * (sigma_1 * k_d - 1 / (4 * epsilon_1)  - sigma_1 * epsilon_2)
a = [k1,(k2-epsilon_1 ), gamma/2];
b= [ 0.5,0.5, 0.5 *sigma_1 ];
c =  min(a) /max(b)

 lambda = sigma_1/(4* epsilon_1) *       16.5180 *      16.5180%18.1667^2%29.9570^2
 zeta_0 = -k_d*(x_2+atan(k1*(x_1-x_1_d)-x_1_d_dot));
zeta = zeta_0;
```

```
zeta=0;
  lambda/c

  K_1 =(tan   (   ((pi/2-(lambda/c)^0.5)^2-sigma_1*(18.3)^2-0.5^2)^0.5   )   )/(0-x_1 )

for  t = 0: Delta_Time: End_Time
   % --------------------  disturbance generator  -------------------------%
   d=0;
   d_hat = 0;
   d_dot=0;
%    x_1_d =  1.255;
%    x_1_d_dot  = 0;
   for i  = 1 :15
      d = d+ sin(i*t) + sin(x_2)*sin(i*t/2);
      d_dot = d_dot+ cos(i*t) + sin(x_2)*cos(i*t/2)+ x_2_dot * cos(x_2)*sin(i*t/2);
   end
   %------------------- Cordiant Transformation ---------------------%
   z_1 = x_1 - x_1_d;
   alpha_1 = atan(-k1*z_1+x_1_d_dot) ;
   z_2 = x_2 - alpha_1;
   delta =tan(z_2)*(1+(tan(alpha_1))^2)/(1-tan(z_2)*tan(alpha_1)) ;
   alpha_1_dot = (-k1*delta + k1*k1*z_1+x_1_d_double_dot ) /( 1+(-k1*z_1+x_1_d_dot)^2) ;
   z_1_dot = tan(alpha_1) +  delta  - x_1_d_dot;
   z_1_dot = -k1*z_1 +  delta ;
   if z_2 == 0;
      delta_dividied_z_2 = 1 + (tan(alpha_1))^2 ;
   else
      delta_dividied_z_2 = delta/z_2 ;
   end
   %---------------------disturbance observer --------------------------%
   zeta_dot = k_d*  k2*z_2;
   d_hat = zeta + k_d * z_2;
   if m ==2
      alpha_1
      m=3
   end
   if m==1
      zeta = -k_d * z_2;
      d_hat = 0;
      m=2;
      z_1_0 = z_1;
      z_2_0 = z_2;
      de_0 = d;
      alpha_1_dot = (-k1*delta + k1*k1*z_1+x_1_d_double_dot ) /( 1+(-k1*z_1+x_1_d_dot)^2) ;
   end
   zeta_dot =- k_d*  (u+ d_hat)  ;
   zeta_dot = k_d* ( k2*z_2+ z_1 * delta_dividied_z_2);
   %-------------------- control signal --------------------------%
   u = (alpha_1_dot - k2*z_2 - z_1 * delta_dividied_z_2  -  d_hat );

   %--------------------- Model -------------------------%
   x_2 = x_2 + Delta_Time * ( u +d );
   x_1_dot = tan(x_2);
   x_1 = x_1 + Delta_Time * tan(x_2);
   x_1_d_old =  x_1_d;
   x_1_d_dot_old = x_1_d_dot;
   x_1_d_double_dot = ( x_1_d_dot - x_1_d_dot_old)/Delta_Time;
   x_1_d = x_1_d_dot  *t;
   zeta = zeta + zeta_dot * Delta_Time;
   de = (d - d_hat);
```

```
    V_sum_bound = (z_1_0^2 + z_2_0^2 +sigma_1* de_0^2 - lambda / c)* exp(-c * t) +   lambda / c;
    V_sum =  (z_1^2 + z_2^2 +2* sigma_1 * de^2);

    results(1,Loop_Count) = x_1;
    results(2,Loop_Count) = x_2;
    results(3,Loop_Count) = z_2;
    results(4,Loop_Count) = u;
    results(5,Loop_Count) = alpha_1;
    results(7,Loop_Count) =V_sum_bound ;
    results(8,Loop_Count) = V_sum ;
    results(9,Loop_Count) = t;
    results(10,Loop_Count) = x_1_d ;
    results(11,Loop_Count) = z_1 ;
    results(12,Loop_Count) = x_1_dot;
    results(13,Loop_Count) =pi/2;%z_1_dot;
    results_2(:,Loop_Count) = [ x_1 ;
                                d ;
                                d_hat ;
                                t ;
                                 d_dot;
                                 de ] ;
    Loop_Count = Loop_Count +1;
end

figure
get(0,'Factory')
set(0,'defaultfigurecolor',[1 1 1])
plot(results(9,:) ,results(13,:), ...
        'LineWidth',2)
set(title({'Backstepping controller Trig example'}, 'FontSize',18))
 hold on
plot(results(9,:) ,results(1,:),'k', ...
        'LineWidth',2)
plot(results(9,:) ,results(2,:),'g', ...
        'LineWidth',2)
hold on
plot(results(9,:) ,results(5,:),'m', ...
        'LineWidth',2)
hold on
plot(results(9,:) ,-results(13,:),'r', ...
        'LineWidth',2)
hold off
axis([0,End_Time,-15 ,15])
set(legend({'$pi/2$','$x_1$','$x_2$','$alpha_1$','$-pi/2$'}, 'FontSize',20),'interpreter','latex')
set(xlabel({'Time(s)'}, 'FontSize',18))
set(ylabel({'Radians'}, 'FontSize',18))

figure
subplot(2,1,1); % top subplot
plot(results(9,:) ,results(11,:),'b', ...
        'LineWidth',2)
      hold on
plot(results(9,:) ,results(3,:),'g', ...
        'LineWidth',2)
      hold on
set(title({'State errors'}, 'FontSize',18))
set(legend({'$z_1$','$z_2$'}, 'FontSize',20),'interpreter','latex')
set(xlabel({'Time(s)'}, 'FontSize',14))
set(ylabel({'Error Magnitude'}, 'FontSize',14))
axis([0,End_Time, -25, 5])
```

```
subplot(2,1,2);
plot(results_2(4,:) ,results_2(6,:),'r', ...
            'LineWidth',2)
set(title({'Disturbance estimate error'}, 'FontSize',18))
set(legend({'$de$'}, 'FontSize',20),'interpreter','latex')
axis([0,End_Time, -5, 5])
set(xlabel({'Time(s)'}, 'FontSize',14))
set(ylabel({'Error Magnitude'}, 'FontSize',14))
hold off

figure
plot(results(9,:) ,results(7,:),'b', ...
            'LineWidth',2)
hold on
plot(results(9,:) ,results(8,:),'g', ...
            'LineWidth',2)
set(title({'V _3 and V sum bound'}, 'FontSize',18))
set(legend({'$V sum bound $','$V sum$'}, 'FontSize',20),'interpreter','latex')
set(xlabel({'Time(s)'}, 'FontSize',14))
set(ylabel({'V_3 / V sum bound'}, 'FontSize',14))
axis([0,End_Time, 0, 450])
hold off

figure
plot(results(9,:) ,results(4,:),'b', ...
            'LineWidth',2)
set(title({'Control Input U'}, 'FontSize',18))
set(legend({'$u$'}, 'FontSize',20),'interpreter','latex')
set(xlabel({'Time(s)'}, 'FontSize',14))
set(ylabel({'u'}, 'FontSize',14))
axis([0,End_Time, -100, 100])
hold off

figure
subplot(2,1,1); % top subplot
plot(results_2(4,:) ,results_2(2,:),'g', ...
            'LineWidth',2)
set(title({'Distrubance Vs Time'}, 'FontSize',18))
set(xlabel({'Time(s)'}, 'FontSize',14))
set(ylabel({'Magnitude'}, 'FontSize',14))
hold on
subplot(2,1,2);
plot(results_2(4,:) ,results_2(3,:),'r', ...
            'LineWidth',2)
hold off
set(title({'Disturbance Estimate Vs Time'}, 'FontSize',18))
set(xlabel({'Time(s)'}, 'FontSize',14))
set(ylabel({'Magnitude'}, 'FontSize',14))
axis([0,End_Time, -40, 40])
```

Appendix B –Code for Trig One-steep Ahead Backstepping Example

```
close all
clc
clear all
% Simulation paramaters
End_Time                =20;
Delta_Time              = 1/10000;
Time_steps              = End_Time / Delta_Time;
x_1_d = 0;
Loop_Count = 1;
%initialize state varables
x_1 = 5;
x_2 = 2;
x_2_dot  = 0;
results             = zeros(12, Time_steps);
alpha_1 = 0;
x_1_d_dot =0;
x_1_d_double_dot = 0;
x_1_d_old = x_1_d - x_1_d*Delta_Time;
x_1_d_old = x_1_d;
% controller gains
k1= 3

k_gamma=1.5;
k2=2.9;
k_d = 200;
D_1 = 0.2;
c_1 = 2
 zeta = 0;
 de = 0;
 u = 0;
 delta_1_constant = 1/1000;
for  t = 0: Delta_Time: End_Time
    % --------------------  disturbance generator  -------------------------%
    d=0;
    d_hat = 0;
    d_dot=0;
%    x_1_d =  1.255;
%    x_1_d_dot  = 0;
    for i  = 1 :15
        d = d+ sin(i*t) + sin(x_2)*sin(i*t/2);
        d_dot = d_dot+ cos(i*t) + sin(x_2)*cos(i*t/2)+ x_2_dot * cos(x_2)*sin(i*t/2);
    end
    %------------------- Cordianant Transofrmation ---------------------%
    z_1 = x_1 - x_1_d;
     x_2
     sigma_1 = z_1 / (1+z_1^2)^0.5 ;
    delta_1   = 1 + 0.5 * x_2^2;
    alpha_1 = asin( (-k1*sigma_1) /delta_1 + x_1_d_dot);
    z_2 = x_2 - alpha_1
    sigma_2 = z_2 / (1+z_2^2)^0.5 ;
    delta = sin(z_2)*cos(alpha_1) + (cos(z_2) - 1)*sin(alpha_1) ;
```

```
   if z_2 == 0;
      delta_dividied_z_2 = cos(alpha_1);
   else
      delta_dividied_z_2 = delta/z_2 ;
   end

   z_1_dot = sin(alpha_1) + sin(z_2)*cos(alpha_1) + (cos(z_2) - 1)*sin(alpha_1) - x_1_d_dot;
   %--------------------disturbance observer -------------------------%
   zeta_dot = k_d*  k2*z_2;


   %zeta_dot = k_d* ( k2*z_2+ z_1 * delta_dividied_z_2);
   gamma_1 = k_gamma;
   z_2_dot = -D_1*z_2 - k2*sigma_2 - c_1*(z_2+2*alpha_1)/(2*delta_1) - gamma_1*sigma_1 *
delta_dividied_z_2 ;
   zeta_dot = k_d* ( z_2_dot );
   zeta_dot = -k_d*zeta  - k_d*(-    z_2_dot  + k_d *z_2) ;
   %{
   zeta_dot = -k_d*zeta  - G * (-D_1*x_2 +u) + k1 * sigma_1_dash/delta_1 * z_1_dot

   (-k1*x_2*z_2/delta_1^2*sigma_1_dash*z_1_dot -k1*sigma_1*z_2*(1-2*x_2*x_2/delta_1)/delta_1^2
*)
   %}
      G = 1 - k1*sigma_1*x_2/delta_1^2;
   sigma_1_dash = 1 / (1+z_1^2)^1.5 ;
%    zeta_dot = -G*k_d*zeta  -k_d*( G * (-D_1*x_2 +u+k_d*z_2) + k1 * sigma_1_dash/delta_1 *
z_1_dot );
  % zeta_dot = -G*k_d*zeta  -k_d*( G * (-D_1*x_2 +u+k_d*z_2) + k1 * sigma_1_dash/delta_1 *
z_1_dot );
    zeta_dot = -G*k_d*zeta*0  -k_d*(     -c_1*(z_2+2*alpha_1)/(2*delta_1)-D_1*z_2 - k2*sigma_2 -
gamma_1*sigma_1 * delta_dividied_z_2);

%    k_d k_d* ( z_2_dot );
%    d_hat = zeta_dot +K_d*z_2_dot

   %------------------- control signal -------------------------%
    if Loop_Count ==1
        z_1_0 = z_1;
        z_2_0 = z_2;
        de_0 = de;
        lambda = delta_1_constant * 16.2*(1+0.5*k1*k1);
       a = [(gamma_1*k1-lambda )/(1+z_1_0*z_1_0)^0.5,D_1,];
       b= [ 1,0.5, ];
       c =  min(a) /max(b)
       alpha_1_0 = alpha_1;
       zeta = -k_d * z_2;
      % zeta_dot = -G*k_d*zeta  -k_d*( G * (-D_1*x_2 +u+k_d*z_2) + k1 * sigma_1_dash/delta_1 *
z_1_dot );
       zeta_dot = 0;
       d_hat = 0;
       de_0 = d;
       zeta
    end
   d_hat = zeta + k_d * z_2;
   de = d - d_hat;
    if Loop_Count ==1
       de_0 = de;
    end
```

```
    %alpha_1_dot  = A/B
    B = (1-(-k1*sigma_1 / delta_1 + x_1_d_dot)^2)^0.5;
    G = 1 - k1*sigma_1*x_2/delta_1^2;
    sigma_1_dash = 1 / (1+z_1^2)^1.5 ;
    u = (G^(-1)) * (-c_1*(z_2+2*alpha_1)/(2*delta_1)+D_1*alpha_1 -
k1*sigma_1*x_2*D_1*x_2/delta_1^2- k2*sigma_2 - gamma_1*sigma_1 * delta_dividied_z_2 -
k1*sigma_1_dash/delta_1 * z_1_dot ) - d_hat;
%    u = (-D_1*alpha_1 - B*k2*sigma_2 - B*k_gamma*(sigma_1 * delta_dividied_z_2) -
B*(k1/delta_1)*(z_1_dot/(1+z_1^2)^1.5)  - x_1_d_double_dot) / (B - k1*sigma_1*x_2/(delta_1^2));
    %-------------------- Model ------------------------%

    x_2_dot =  u + d;
    x_2 = x_2 + Delta_Time * x_2_dot  ;
    x_1_dot = sin(x_2);
    x_1 = x_1 + Delta_Time * sin(x_2);
    zeta = zeta + zeta_dot * Delta_Time

    x_1_d_dot = (x_1_d - x_1_d_old)/Delta_Time;
     x_1_d_old =  x_1_d;
     x_1_d_double_dot =0;
%{
    results(1,Loop_Count) = x_1;
    results(2,Loop_Count) = x_2;
    results(3,Loop_Count) = z_2;
    results(4,Loop_Count) = u;
    results(5,Loop_Count) = alpha_1;
    results(7,Loop_Count) = sigma_1;
    results(8,Loop_Count) = sigma_2;
    results(9,Loop_Count) = t;
    results(10,Loop_Count) = x_1_d ;
    results(11,Loop_Count) = z_1 ;
    results(12,Loop_Count) = x_1_dot;
    %}


%    V_sum_bound = (z_1_0^2 + z_2_0^2 +sigma_1* de_0^2 - lambda / c)* exp(-c * t) +   lambda / c;
    V_sum_bound = ((1+z_1_0^2)^0.5 -1 + 0.5*z_2_0^2 + 0.5*delta_1_constant * de_0^2- lambda / c)*
exp(-c * t) +   lambda / c
    V_sum =  ((1+z_1^2)^0.5 -1+ 0.5*z_2^2 +0.5* delta_1_constant * de^2)

    results(1,Loop_Count) = x_1;
    results(2,Loop_Count) = x_2;
    results(3,Loop_Count) = z_2;
    results(4,Loop_Count) = u;
    results(5,Loop_Count) = alpha_1;
    results(7,Loop_Count) =V_sum_bound ;
    results(8,Loop_Count) = V_sum ;
    results(9,Loop_Count) = t;
    results(10,Loop_Count) = x_1_d ;
    results(11,Loop_Count) = z_1 ;
    results(12,Loop_Count) = x_1_dot;
    results(13,Loop_Count) =pi/2;%z_1_dot;
    results_2(:,Loop_Count) = [ x_1 ;
                               d ;
                               d_hat ;
                               t ;
                               d_dot;
                               de ] ;

    Loop_Count = Loop_Count +1;
```

```
end


figure
get(0,'Factory')
set(0,'defaultfigurecolor',[1 1 1]);
%plot(results(9,:) ,results(13,:), ...
%               'LineWidth',2)
%set(title({'One-step ahead backstepping controller example'}, 'FontSize',18))
% hold on
plot(results(9,:) ,results(1,:),'k', ...
             'LineWidth',2)
set(title({'One-step ahead backstepping controller example'}, 'FontSize',18))
hold on
plot(results(9,:) ,results(2,:),'g', ...
             'LineWidth',2)
hold on
plot(results(9,:) ,results(5,:),'m', ...
             'LineWidth',2)
hold on
%plot(results(9,:) ,-results(13,:),'r', ...
%               'LineWidth',2)
%hold off
axis([0,End_Time,-5 ,10])
%set(legend({'$pi/2$','$x_1$','$x_2$','$alpha_1$','$-pi/2$'}, 'FontSize',20),'interpreter','latex')
set(legend({'$x_1$','$x_2$','$alpha_1$'}, 'FontSize',20),'interpreter','latex')
set(xlabel({'Time(s)'}, 'FontSize',18))
set(ylabel({'Radians'}, 'FontSize',18))

figure
subplot(2,1,1); % top subplot
plot(results(9,:) ,results(11,:),'b', ...
             'LineWidth',2)
         hold on
plot(results(9,:) ,results(3,:),'g', ...
             'LineWidth',2)
         hold on
set(title({'State errors'}, 'FontSize',18))
set(legend({'$z_1$','$z_2$'}, 'FontSize',20),'interpreter','latex')
set(xlabel({'Time(s)'}, 'FontSize',14))
set(ylabel({'Error Magnitude'}, 'FontSize',14))
axis([0,End_Time, -10, 10])
subplot(2,1,2);
plot(results_2(4,:) ,results_2(6,:),'r', ...
             'LineWidth',2)
set(title({'Disturbance estimate error'}, 'FontSize',18))
set(legend({'$de$'}, 'FontSize',20),'interpreter','latex')
axis([0,End_Time, -5, 5])
set(xlabel({'Time(s)'}, 'FontSize',14))
set(ylabel({'Error Magnitude'}, 'FontSize',14))
hold off

figure
plot(results(9,:) ,results(7,:),'b', ...
             'LineWidth',2)
hold on
plot(results(9,:) ,results(8,:),'g', ...
             'LineWidth',2)
set(title({'V _3 and V_{3,bound}'}, 'FontSize',18))
set(legend({'$V_{3,bound}$','$V_3$'}, 'FontSize',20),'interpreter','latex')
```

```
set(xlabel({'Time(s)'}, 'FontSize',14))
set(ylabel({'V_3 / V_{3,bound}'}, 'FontSize',14))
axis([0,End_Time, 0, 10])
hold off

figure
plot(results(9,:) ,results(4,:),'b', ...
            'LineWidth',2)
set(title({'Control Input U'}, 'FontSize',18))
set(legend({'$u$'}, 'FontSize',20),'interpreter','latex')
set(xlabel({'Time(s)'}, 'FontSize',14))
set(ylabel({'u'}, 'FontSize',14))
axis([0,End_Time, -100, 100])
hold off

figure
subplot(2,1,1); % top subplot
plot(results_2(4,:) ,results_2(2,:),'g', ...
            'LineWidth',2)
set(title({'Distrubance Vs Time'}, 'FontSize',18))
hold on
plot(results_2(4,:) ,results_2(3,:),'b', ...
            'LineWidth',2)
set(xlabel({'Time(s)'}, 'FontSize',14))
set(ylabel({'Magnitude'}, 'FontSize',14))
hold on
subplot(2,1,2);
plot(results_2(4,:) ,results_2(2,:)-results_2(3,:),'r', ...
            'LineWidth',2)
hold off
set(title({'Distrubance Estimate Vs Time'}, 'FontSize',18))
set(xlabel({'Time(s)'}, 'FontSize',14))
set(ylabel({'Magnitude'}, 'FontSize',14))
axis([0,End_Time, -40, 40])
```

Appendix C –L Matrix for MRP Angle Representation

$$\mathbf{L} = \frac{8}{(1+\|\boldsymbol{q}\|^2)^2}\begin{bmatrix} \boldsymbol{q}_3 & 1 & -\boldsymbol{q}_1 - \boldsymbol{\alpha}_{q,1} \\ -1 & \boldsymbol{q}_3 & -\boldsymbol{q}_2 - \boldsymbol{\alpha}_{q,2} \\ \boldsymbol{\alpha}_{q,1} & \boldsymbol{\alpha}_{q,2} & 0 \end{bmatrix} + 4\left(\frac{\mathbf{I}_{3\times3}}{(1+\|\boldsymbol{q}\|^2)} - \frac{(\boldsymbol{q}+\boldsymbol{\alpha}_q)\boldsymbol{\alpha}_q^{\mathrm{T}}}{(1+\|\boldsymbol{q}\|^2)\left(1+\|\boldsymbol{\alpha}_q\|^2\right)}\right)$$
$$\times\begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} - 8\left(\frac{\boldsymbol{q}_e\left(\|\boldsymbol{q}_e\|^2 + 2 + 2\|\boldsymbol{\alpha}_q\|^2\right) + 4\boldsymbol{\alpha}_q\left(1+\|\boldsymbol{q}\|^2 - \boldsymbol{q}_e^{\mathrm{T}}\boldsymbol{\alpha}_q\right)}{(1+\|\boldsymbol{q}\|^2)^2\left(1+\|\boldsymbol{\alpha}_q\|^2\right)^2}\right)$$
$$\times\left(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^{\mathrm{T}}\left(\boldsymbol{\alpha}_q\boldsymbol{\alpha}_q^{\mathrm{T}} - \|\boldsymbol{\alpha}_q\|^2\mathbf{I}_{3\times3}\right) + \boldsymbol{\alpha}_q^{\mathrm{T}}\begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}\right). \tag{C-1}$$

The derivative of $\dot{\mathbf{L}}$ times an arbitrary vector $\boldsymbol{\vartheta}$ is defined as follows:

$$\dot{\mathbf{L}}\boldsymbol{\vartheta} = \frac{\partial\mathbf{L}}{\partial\boldsymbol{q}}\begin{bmatrix} \boldsymbol{\vartheta}_1 & 0 & 0 \\ 0 & \boldsymbol{\vartheta}_2 & 0 \\ 0 & 0 & \boldsymbol{\vartheta}_3 \end{bmatrix}\dot{\boldsymbol{q}} + \frac{\partial\mathbf{L}}{\partial\boldsymbol{\alpha}_q}\begin{bmatrix} \boldsymbol{\vartheta}_1 & 0 & 0 \\ 0 & \boldsymbol{\vartheta}_2 & 0 \\ 0 & 0 & \boldsymbol{\vartheta}_3 \end{bmatrix}\dot{\boldsymbol{\alpha}}_q, \tag{C-2}$$

$$\mathbf{L}\boldsymbol{\vartheta} = -8\frac{\left(\|\boldsymbol{q}_e\|^2 + 2 + 2\|\boldsymbol{\alpha}_q\|^2\right)\mathbf{I}_{3\times3} + 2\boldsymbol{q}_e\boldsymbol{q}_e^{\mathrm{T}}}{(1+\|\boldsymbol{q}\|^2)^2\left(1+\|\boldsymbol{\alpha}_q\|^2\right)^2} \tag{C-3}$$
$$\times\left(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^{\mathrm{T}}\boldsymbol{\alpha}_q\boldsymbol{\alpha}_q^{\mathrm{T}} - \|\boldsymbol{\alpha}_q\|^2\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^{\mathrm{T}} + \boldsymbol{\alpha}_q^{\mathrm{T}}\begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}\right)\boldsymbol{\vartheta},$$

$$\frac{\partial \mathbf{L}}{\partial \boldsymbol{q}}\begin{bmatrix}\boldsymbol{\vartheta}_1 & 0 & 0\\ 0 & \boldsymbol{\vartheta}_2 & 0\\ 0 & 0 & \boldsymbol{\vartheta}_3\end{bmatrix} = -\frac{32}{(1+\|\boldsymbol{q}\|^2)^3}\begin{bmatrix}\boldsymbol{q}_3 & 1 & -\boldsymbol{q}_1-\boldsymbol{\alpha}_{q,1}\\ -1 & \boldsymbol{q}_3 & -\boldsymbol{q}_2-\boldsymbol{\alpha}_{q,2}\\ \boldsymbol{\alpha}_{q,1} & \boldsymbol{\alpha}_{q,2} & 0\end{bmatrix}\boldsymbol{\vartheta}\boldsymbol{q}^{\mathrm{T}}$$
$$+\frac{8}{(1+\|\boldsymbol{q}\|^2)^2}\left(\boldsymbol{\vartheta}\begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}} - \begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}}\boldsymbol{\vartheta}\mathbf{I}_{3\times3}\right) - 8\begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\frac{\boldsymbol{\vartheta}\boldsymbol{q}^{\mathrm{T}}}{(1+\|\boldsymbol{q}\|^2)^2}$$
$$+\frac{4\boldsymbol{\alpha}_q^{\mathrm{T}}}{(1+\|\boldsymbol{q}\|^2)\left(1+\|\boldsymbol{\alpha}_q\|^2\right)}\begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\boldsymbol{\vartheta}\left(-\mathbf{I}_{3\times3} + 2\frac{\left(\boldsymbol{q}+\boldsymbol{\alpha}_q\right)}{(1+\|\boldsymbol{q}\|^2)}\boldsymbol{q}^{\mathrm{T}}\right)$$
$$-8\left(\frac{4\boldsymbol{\alpha}_q}{\left(1+\|\boldsymbol{\alpha}_q\|^2\right)^2}\left(\begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}}\left(\boldsymbol{\alpha}_q\boldsymbol{\alpha}_q^{\mathrm{T}} - \|\boldsymbol{\alpha}_q\|^2\mathbf{I}_{3\times3}\right) + \boldsymbol{\alpha}_q^{\mathrm{T}}\begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\right)\boldsymbol{\vartheta}\right.$$
$$\times\frac{\left(2\boldsymbol{q}^{\mathrm{T}} - \boldsymbol{\alpha}_q^{\mathrm{T}}\right)}{(1+\|\boldsymbol{q}\|^2)^2} - 2\frac{\boldsymbol{q}_e\left(\|\boldsymbol{q}_e\|^2 + 2 + 2\|\boldsymbol{\alpha}_q\|^2\right) + 4\boldsymbol{\alpha}_q\left(1+\|\boldsymbol{q}\|^2 - \boldsymbol{q}_e^{\mathrm{T}}\boldsymbol{\alpha}_q\right)}{(1+\|\boldsymbol{q}\|^2)^2\left(1+\|\boldsymbol{\alpha}_q\|^2\right)^2}$$
$$\left.\times\left(\begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}}\boldsymbol{\alpha}_q\boldsymbol{\alpha}_q^{\mathrm{T}} - \|\boldsymbol{\alpha}_q\|^2\begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}} + \boldsymbol{\alpha}_q^{\mathrm{T}}\begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\right)\boldsymbol{\vartheta}\left(\frac{2\boldsymbol{q}^{\mathrm{T}}}{1+\|\boldsymbol{q}\|^2}\right)\right) + \frac{\partial \mathbf{L}}{\partial \boldsymbol{q}_e}, \tag{C-4}$$

$$\frac{\partial \mathbf{L}}{\partial \boldsymbol{\alpha}_q}\begin{bmatrix}\boldsymbol{\vartheta}_1 & 0 & 0\\ 0 & \boldsymbol{\vartheta}_2 & 0\\ 0 & 0 & \boldsymbol{\vartheta}_3\end{bmatrix} = \frac{8}{(1+\|\boldsymbol{q}\|^2)^2}\left(\begin{bmatrix}0\\0\\1\end{bmatrix}\boldsymbol{\vartheta}^{\mathrm{T}} - \begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}}\boldsymbol{\vartheta}\mathbf{I}_{3\times 3}\right) - \frac{8}{(1+\|\boldsymbol{q}\|^2)^2}\begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\boldsymbol{\vartheta}\boldsymbol{q}^{\mathrm{T}} \tag{C-5}$$

$$- \frac{4(\boldsymbol{q}+\boldsymbol{\alpha}_q)}{(1+\|\boldsymbol{q}\|^2)\left(1+\|\boldsymbol{\alpha}_q\|^2\right)}\left(\begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\boldsymbol{\vartheta}\right)^{\mathrm{T}}$$

$$+ \frac{4\boldsymbol{\alpha}_q^{\mathrm{T}}}{(1+\|\boldsymbol{q}\|^2)\left(1+\|\boldsymbol{\alpha}_q\|^2\right)}\begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\boldsymbol{\vartheta}\left(-\mathbf{I}_{3\times 3}\right.$$

$$\left. + 2\frac{(\boldsymbol{q}+\boldsymbol{\alpha}_q)}{\left(1+\|\boldsymbol{\alpha}_q\|^2\right)}\boldsymbol{\alpha}_q^{\mathrm{T}}\right)$$

$$-8\left(\frac{4\boldsymbol{q}_e\boldsymbol{\alpha}_q^{\mathrm{T}}}{(1+\|\boldsymbol{q}\|^2)^2\left(1+\|\boldsymbol{\alpha}_q\|^2\right)^2}\left(\begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}}\left(\boldsymbol{\alpha}_q\boldsymbol{\alpha}_q^{\mathrm{T}} - \|\boldsymbol{\alpha}_q\|^2\mathbf{I}_{3\times 3}\right)\right.\right.$$

$$\left.\left. +\boldsymbol{\alpha}_q^{\mathrm{T}}\begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\right)\right)\boldsymbol{\vartheta} + \frac{4d\boldsymbol{\alpha}_q\left(1+\|\boldsymbol{q}\|^2 - \boldsymbol{q}_e^{\mathrm{T}}\boldsymbol{\alpha}_q\right) + 4\boldsymbol{\alpha}_q\left(2\boldsymbol{\alpha}_q^{\mathrm{T}} - \boldsymbol{q}^{\mathrm{T}}\right)}{(1+\|\boldsymbol{q}\|^2)^2\left(1+\|\boldsymbol{\alpha}_q\|^2\right)^2}$$

$$\times\left(\begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}}\left(\boldsymbol{\alpha}_q\boldsymbol{\alpha}_q^{\mathrm{T}} - \|\boldsymbol{\alpha}_q\|^2\mathbf{I}_{3\times 3}\right) + \boldsymbol{\alpha}_q^{\mathrm{T}}\begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\right)\boldsymbol{\vartheta}$$

$$-2\frac{\boldsymbol{q}_e\left(\|\boldsymbol{q}_e\|^2 + 2 + 2\|\boldsymbol{\alpha}_q\|^2\right) + 4\boldsymbol{\alpha}_q\left(1+\|\boldsymbol{q}\|^2 - \boldsymbol{q}_e^{\mathrm{T}}\boldsymbol{\alpha}_q\right)}{(1+\|\boldsymbol{q}\|^2)^2\left(1+\|\boldsymbol{\alpha}_q\|^2\right)^2}\left(\begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}}\left(\boldsymbol{\alpha}_q\boldsymbol{\alpha}_q^{\mathrm{T}}\right.\right.$$

$$\left.\left. -\|\boldsymbol{\alpha}_q\|^2\mathbf{I}_{3\times 3}\right) + \boldsymbol{\alpha}_q^{\mathrm{T}}\begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\right)\boldsymbol{\vartheta}\frac{2\boldsymbol{\alpha}_q^{\mathrm{T}}}{\left(1+\|\boldsymbol{\alpha}_q\|^2\right)}\Bigg)$$

$$-8\left(\frac{\boldsymbol{q}_e\left(\|\boldsymbol{q}_e\|^2 + 2 + 2\|\boldsymbol{\alpha}_q\|^2\right) + 49\boldsymbol{\alpha}_q\left(\|\boldsymbol{\alpha}_q\|^2 + \|\boldsymbol{q}_e\|^2 + 1 + \boldsymbol{q}_e^{\mathrm{T}}\boldsymbol{\alpha}_q\right)}{\left(1+\|\boldsymbol{q}_e\|^2 + 2\boldsymbol{q}_e^{\mathrm{T}}\boldsymbol{\alpha}_q + \|\boldsymbol{\alpha}_q\|^2\right)^2\left(1+\|\boldsymbol{\alpha}_q\|^2\right)^2}\right)$$

$$\times\left(+\begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}}\left(\boldsymbol{\alpha}_q^{\mathrm{T}}\boldsymbol{\vartheta}\mathbf{I}_{3\times 3} + \boldsymbol{\alpha}_q\boldsymbol{\vartheta}^{\mathrm{T}} - 2\boldsymbol{\vartheta}\boldsymbol{\alpha}_q^{\mathrm{T}}\right) - \boldsymbol{\vartheta}^{\mathrm{T}}\begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\right) - \frac{\partial \mathbf{L}}{\partial \boldsymbol{q}_e}.$$

Appendix D –Partial Derivative of Omega

$$\boldsymbol{\Omega} = m\mathbf{G}_1^{-1}\left(\mathbf{D}_1\boldsymbol{\alpha}_1 - \mathbf{K}_2\sigma(\boldsymbol{\nu}_{1e}) - \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\left(\sigma(\boldsymbol{\eta}_{1e})\frac{2+\|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\sigma(\boldsymbol{\eta}_{1e})\right) - \frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\mathbf{D}_1\hat{\boldsymbol{\nu}}_1 - \left(c + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{2\boldsymbol{\alpha}_1 + \boldsymbol{\nu}_{1e}}{8\Delta_1(\hat{\boldsymbol{\nu}}_1)}\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2} + \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \ddot{\boldsymbol{\eta}}_{1d}\right) + m\left(g\boldsymbol{e}_3 - \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\frac{2\gamma_1\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) \quad \text{(D-1)}$$

We will now define the following:

$$\mathbf{N} = \mathbf{G}_1^{-1} = \mathbf{I}_{3\times3} + \frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2 - \sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{\nu}}_1}, \quad \text{(D-2)}$$

$$\mathbf{M}_1 = \mathbf{D}_1\boldsymbol{\alpha}_1 - \mathbf{K}_2\sigma(\boldsymbol{\nu}_{1e}), \quad \text{(D-3)}$$

$$\mathbf{M}_2 = -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma(\boldsymbol{\eta}_{1e})\frac{2+\|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}, \quad \text{(D-4)}$$

$$\mathbf{M}_3 = \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\sigma(\boldsymbol{\eta}_{1e}), \quad \text{(D-5)}$$

$$\mathbf{M}_4 = -\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\mathbf{D}_1\hat{\boldsymbol{\nu}}_1, \quad \text{(D-6)}$$

$$\mathbf{M}_5 = -\left(c + \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{2\boldsymbol{\alpha}_1 + \boldsymbol{\nu}_{1e}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}, \quad \text{(D-7)}$$

$$\mathbf{M}_6 = \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \ddot{\boldsymbol{\eta}}_{1d}, \quad \text{(D-8)}$$

$$\mathbf{M}_7 = m\left(g\boldsymbol{e}_3 - \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\frac{2\gamma_1\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right). \quad \text{(D-9)}$$

And we get

$$\boldsymbol{\Omega} = m\mathbf{N}(\mathbf{M}_1 + \mathbf{M}_2 + \mathbf{M}_3 + \mathbf{M}_4 + \mathbf{M}_5 + \mathbf{M}_6) + \mathbf{M}_7 \quad \text{(D-10)}$$

Taking the derivative

$$\dot{\boldsymbol{\Omega}} = m\dot{\mathbf{N}}(\mathbf{M}_1 + \mathbf{M}_2 + \mathbf{M}_3 + \mathbf{M}_4 + \mathbf{M}_5 + \mathbf{M}_6) + m\mathbf{N}(\dot{\mathbf{M}}_1 + \dot{\mathbf{M}}_2 + \dot{\mathbf{M}}_3 + \dot{\mathbf{M}}_4 + \dot{\mathbf{M}}_5 + \dot{\mathbf{M}}_6) + \dot{\mathbf{M}}_7 \tag{D-11}$$

$$\dot{\boldsymbol{\Omega}} = \left(\frac{\partial \mathbf{N}}{\partial \hat{\boldsymbol{v}}_1}\dot{\hat{\boldsymbol{v}}}_1 + \frac{\partial \mathbf{N}}{\partial \boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e}\right)\left(\sum_{i=1}^{6}\mathbf{M}_i\right) + \left(m\mathbf{N}\left(\sum_{i=1}^{6}\frac{\partial \mathbf{M}_i}{\partial \hat{\boldsymbol{v}}_1}\right) + \frac{\partial \mathbf{M}_7}{\partial \hat{\boldsymbol{v}}_1}\right)\dot{\hat{\boldsymbol{v}}}_1 + \left(m\mathbf{N}\left(\sum_{i=1}^{6}\frac{\partial \mathbf{M}_i}{\partial \boldsymbol{\eta}_{1e}}\right) + \frac{\partial \mathbf{M}_7}{\partial \boldsymbol{\eta}_{1e}}\right)\dot{\boldsymbol{\eta}}_{1e} + \left(m\mathbf{N}\left(\sum_{i=1}^{6}\frac{\partial \mathbf{M}_i}{\partial \dot{\boldsymbol{\eta}}_{1d}}\right) + \frac{\partial \mathbf{M}_7}{\partial \dot{\boldsymbol{\eta}}_{1d}}\right)\ddot{\boldsymbol{\eta}}_{1d} + m\mathbf{N}\frac{\partial \mathbf{M}_6}{\partial \ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d} \tag{D-12}$$

N1 dot

$$\dot{\mathbf{N}}(\mathbf{M}_1 + \mathbf{M}_2 + \mathbf{M}_3 + \mathbf{M}_4 + \mathbf{M}_5 + \mathbf{M}_6) = \frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{M}_1 + \mathbf{M}_2 + \mathbf{M}_3 + \mathbf{M}_4 + \mathbf{M}_5 + \mathbf{M}_6)}{\left(1 - \sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\right)}\mathbf{N}\left(\mathbf{K}_1\frac{\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\right)\dot{\boldsymbol{\eta}}_{1e} + \frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})(\mathbf{M}_1 + \mathbf{M}_2 + \mathbf{M}_3 + \mathbf{M}_4 + \mathbf{M}_5 + \mathbf{M}_6)^{\mathbf{T}}\mathbf{N}^{\mathrm{T}}}{\left(1 - \sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\right)}\left(\frac{\mathbf{I}_{3x3}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} - 2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^3}\right)\dot{\hat{\boldsymbol{v}}}_1 \tag{D-13}$$

$$\left(\frac{\partial \mathbf{N}}{\partial \boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e}\right)\left(\sum_{i=1}^{6}\mathbf{M}_i\right) = \hat{\boldsymbol{v}}_1^{\mathrm{T}}\left(\sum_{i=1}^{6}\mathbf{M}_i\right)\frac{\mathbf{N}\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{(\Delta_1(\hat{\boldsymbol{v}}_1)^2 - \sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1)}\dot{\boldsymbol{\eta}}_{1e}, \tag{D-14}$$

$$\left(\frac{\partial \mathbf{N}}{\partial \hat{\boldsymbol{v}}_1}\dot{\hat{\boldsymbol{v}}}_1\right)\left(\sum_{i=1}^{6}\mathbf{M}_i\right) = \frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})(\mathbf{M}_1 + \mathbf{M}_2 + \mathbf{M}_3 + \mathbf{M}_4 + \mathbf{M}_5 + \mathbf{M}_6)^{\mathbf{T}}\mathbf{N}^{\mathrm{T}}}{(\Delta_1(\hat{\boldsymbol{v}}_1)^2 - \sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1)}\left(\mathbf{I}_{3x3} - 2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\dot{\hat{\boldsymbol{v}}}_1. \tag{D-15}$$

M1 Dot

$$\mathbf{M}_1 = \mathbf{D}_1\boldsymbol{\alpha}_1 - \mathbf{K}_2\sigma(\boldsymbol{\nu}_{1e}) \tag{D-16}$$

$$\frac{\partial \mathbf{M}_1}{\partial \hat{\boldsymbol{v}}_1}\dot{\hat{\boldsymbol{v}}}_1 = \left(\mathbf{D}_1\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} - \mathbf{K}_2\sigma'(\boldsymbol{\nu}_{1e})\mathbf{G}_1\right)\dot{\hat{\boldsymbol{v}}}_1 \tag{D-17}$$

$$\frac{\partial \mathbf{M}_1}{\partial \boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e} = -\big(\mathbf{D}_1 + \mathbf{K}_2\sigma'(\boldsymbol{\nu}_{1e})\big)\mathbf{K}_1\frac{\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\dot{\boldsymbol{\eta}}_{1e} \tag{D-18}$$

$$\frac{\partial \mathbf{M}_1}{\partial \dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} = \big(\mathbf{D}_1 + \mathbf{K}_2\sigma'(\boldsymbol{\nu}_{1e})\big)\ddot{\boldsymbol{\eta}}_{1d} \tag{D-19}$$

M2 Dot

$$\mathbf{M}_2 = -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma(\boldsymbol{\eta}_{1e})\frac{2 + \|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{v}}_1)}, \tag{D-20}$$

$$\frac{\partial \mathbf{M}_2}{\partial \hat{\boldsymbol{v}}_1}\dot{\hat{\boldsymbol{v}}}_1 = \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma(\boldsymbol{\eta}_{1e})\left(2 + \|\boldsymbol{\alpha}_1\|^2 - \frac{2\boldsymbol{\alpha}_1^{\mathrm{T}}\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{2\big(\Delta_1(\hat{\boldsymbol{v}}_1)\big)^2}\dot{\hat{\boldsymbol{v}}}_1, \tag{D-21}$$

$$\frac{\partial \mathbf{M}_2}{\partial \boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e} = -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\left((2 + \|\boldsymbol{\alpha}_1\|^2)\mathrm{I}_{3\times 3} - \sigma(\boldsymbol{\eta}_{1e})\frac{2\boldsymbol{\alpha}_1^{\mathrm{T}}\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\frac{\sigma'(\boldsymbol{\eta}_{1e})}{2\Delta_1(\hat{\boldsymbol{v}}_1)}\dot{\boldsymbol{\eta}}_{1e}, \tag{D-22}$$

$$\frac{\partial \mathbf{M}_2}{\partial \dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} = -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma(\boldsymbol{\eta}_{1e})\frac{2\boldsymbol{\alpha}_1^{\mathrm{T}}}{2\Delta_1(\hat{\boldsymbol{v}}_1)}\ddot{\boldsymbol{\eta}}_{1d}, \tag{D-23}$$

M3 Dot

$$\mathbf{M}_3 = \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} \sigma(\boldsymbol{\eta}_{1e}), \tag{D-24}$$

$$\frac{\partial \mathbf{M}_3}{\partial \hat{\boldsymbol{v}}_1} \dot{\hat{\boldsymbol{v}}}_1 = \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \frac{\mathbf{K}_1 \sigma(\boldsymbol{\eta}_{1e}) \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} \left(\mathbf{I}_{3x3} - 2 \frac{\hat{\boldsymbol{v}}_1 \hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)} \right) \dot{\hat{\boldsymbol{v}}}_1 \tag{D-25}$$

$$\frac{\partial \mathbf{M}_3}{\partial \boldsymbol{\eta}_{1e}} \dot{\boldsymbol{\eta}}_{1e} = \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \frac{\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} \left(\sigma(\boldsymbol{\eta}_{1e}) \hat{\boldsymbol{v}}_1^{\mathrm{T}} + \hat{\boldsymbol{v}}_1^{\mathrm{T}} \sigma(\boldsymbol{\eta}_{1e}) \mathbf{I}_{3\times 3} \right) \sigma'(\boldsymbol{\eta}_{1e}) \dot{\boldsymbol{\eta}}_{1e} \tag{D-26}$$

$$\frac{\partial \mathbf{M}_3}{\partial \dot{\boldsymbol{\eta}}_{1d}} \ddot{\boldsymbol{\eta}}_{1d} = 0 \mathbf{I}_{3x3} \ddot{\boldsymbol{\eta}}_{1d} \tag{D-27}$$

M4 dot

$$\mathbf{M}_4 = -\frac{\mathbf{K}_1 \sigma(\boldsymbol{\eta}_{1e}) \hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} \mathbf{D}_1 \hat{\boldsymbol{v}}_1 + \mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e}) \hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^3} \left(\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e}) \right)^{\mathrm{T}} \hat{\boldsymbol{v}}_1 \tag{D-28}$$

$$\begin{aligned} \frac{\partial \mathbf{M}_4}{\partial \hat{\boldsymbol{v}}_1} \dot{\hat{\boldsymbol{v}}}_1 = \Bigg(& -2 \frac{\mathbf{K}_1 \sigma(\boldsymbol{\eta}_{1e}) \hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} \mathbf{D}_1 + 2 \frac{\mathbf{K}_1 \sigma(\boldsymbol{\eta}_{1e}) \hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^3} \mathbf{D}_1 \hat{\boldsymbol{v}}_1 \hat{\boldsymbol{v}}_1^{\mathrm{T}} + 2 \mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e}) \hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^3} \mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e}) \\ & - 3 \mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e}) \hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^4} \mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e}) \hat{\boldsymbol{v}}_1 \hat{\boldsymbol{v}}_1^{\mathrm{T}} \Bigg) \dot{\hat{\boldsymbol{v}}}_1 \end{aligned} \tag{D-29}$$

$$\begin{aligned} \frac{\partial \mathbf{M}_4}{\partial \boldsymbol{\eta}_{1e}} \dot{\boldsymbol{\eta}}_{1e} = \Bigg(& -\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}} \mathbf{D}_1 \hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} \mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e}) + \frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}} \mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e}) \hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^3} \mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e}) \\ & + \mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e}) \hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^3} \mathbf{K}_1 \sigma''(\boldsymbol{\eta}_{1e}) \begin{bmatrix} \hat{\boldsymbol{v}}_{1,1} & 0 & 0 \\ 0 & \hat{\boldsymbol{v}}_{1,2} & 0 \\ 0 & 0 & \hat{\boldsymbol{v}}_{1,3} \end{bmatrix} \Bigg) \dot{\boldsymbol{\eta}}_{1e} \end{aligned} \tag{D-30}$$

M5 Dot

$$\mathbf{M}_5 = -\left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{2\boldsymbol{\alpha}_1 + \boldsymbol{\nu}_{1e}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} \sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2} \tag{D-31}$$

$$\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\nu}_{1e}} = -\left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} \mathbf{I}_{3\times3} - \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{2\boldsymbol{\alpha}_1 + \boldsymbol{\nu}_{1e}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} \frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}} \tag{D-32}$$

$$\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\nu}_1} = \frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\nu}_{1e}} \mathbf{G}_1 + \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \left(-\frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2} + \frac{(2\boldsymbol{\alpha}_1 + \boldsymbol{\nu}_{1e})}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} \hat{\boldsymbol{\nu}}_1^{\mathrm{T}}\right) \tag{D-33}$$

$$\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\eta}_{1e}} = \frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\nu}_{1e}} \left(\mathbf{K}_1 \frac{\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right) - \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \left(-\mathbf{K}_1 \frac{\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right) \tag{D-34}$$

$$\frac{\partial \mathbf{M}_5}{\partial \dot{\boldsymbol{\eta}}_{1d}} = -\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\nu}_{1e}} - \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \mathbf{I}_{3\times3} \tag{D-35}$$

M6 Dot

$$\mathbf{M}_6 = \frac{\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \frac{\mathbf{K}_1 \sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \ddot{\boldsymbol{\eta}}_{1d}, \tag{D-36}$$

$$\frac{\partial \mathbf{M}_6}{\partial \hat{\boldsymbol{\nu}}_1} \dot{\hat{\boldsymbol{\nu}}}_1 = -2 \frac{\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})}{\left(\Delta_1(\hat{\boldsymbol{\nu}}_1)\right)^3} \mathbf{K}_1 \sigma(\boldsymbol{\eta}_{1e}) \hat{\boldsymbol{\nu}}_1^{\mathrm{T}} \dot{\hat{\boldsymbol{\nu}}}_1, \tag{D-37}$$

$$\frac{\partial \mathbf{M}_6}{\partial \boldsymbol{\eta}_{1e}} \dot{\boldsymbol{\eta}}_{1e} = \left(\frac{\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \frac{\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \frac{\mathbf{K}_1 \sigma''(\boldsymbol{\eta}_{1e})}{\left(\Delta_1(\hat{\boldsymbol{\nu}}_1)\right)^2} \mathbf{K}_1 \begin{bmatrix} \sigma(\boldsymbol{\eta}_{1e,1}) & 0 & 0 \\ 0 & \sigma(\boldsymbol{\eta}_{1e,2}) & 0 \\ 0 & 0 & \sigma(\boldsymbol{\eta}_{1e,3}) \end{bmatrix} \right) \dot{\boldsymbol{\eta}}_{1e}, \tag{D-38}$$

$$\frac{\partial \mathbf{M}_6}{\partial \dot{\boldsymbol{\eta}}_{1d}} \ddot{\boldsymbol{\eta}}_{1d} = 0\mathbf{I}_{3x3} \ddot{\boldsymbol{\eta}}_{1d} \tag{D-39}$$

$$\frac{\partial \mathbf{M}_6}{\partial \ddot{\boldsymbol{\eta}}_{1d}} \dddot{\boldsymbol{\eta}}_{1d} = \mathbf{I}_{3\times3} \dddot{\boldsymbol{\eta}}_{1d}. \tag{D-40}$$

M7 Dot

$$\mathbf{M}_7 = m\left(g\boldsymbol{e}_3 - \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\frac{2\gamma_1\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\right), \tag{D-41}$$

$$\frac{\partial\mathbf{M}_7}{\partial\boldsymbol{v}_{1e}}\dot{\boldsymbol{v}}_{1e} = -m\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\frac{2\gamma_1}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{v}_{1e}\boldsymbol{v}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{v}_{1e}\|^2}\right), \tag{D-42}$$

$$\frac{\partial\mathbf{M}_7}{\partial\hat{\boldsymbol{v}}_1}\dot{\hat{\boldsymbol{v}}}_1 = \left(\frac{\partial\mathbf{M}_7}{\partial\boldsymbol{v}_{1e}}\mathbf{G}_1 + m\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\left(\frac{2\gamma_1\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\right)\dot{\hat{\boldsymbol{v}}}_1, \tag{D-43}$$

$$\frac{\partial\mathbf{M}_7}{\partial\boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e} = \left(\frac{\partial\mathbf{M}_7}{\partial\boldsymbol{v}_{1e}}\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} - m\frac{2\gamma_1}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\frac{\mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\begin{bmatrix}\boldsymbol{v}_{1e,1} & 0 & 0\\ 0 & \boldsymbol{v}_{1e,2} & 0\\ 0 & 0 & \boldsymbol{v}_{1e,3}\end{bmatrix}\right)\dot{\boldsymbol{\eta}}_{1e}, \tag{D-44}$$

$$\frac{\partial\mathbf{M}_7}{\partial\dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} = -\frac{\partial\mathbf{M}_7}{\partial\boldsymbol{v}_{1e}}\ddot{\boldsymbol{\eta}}_{1d}. \tag{D-45}$$

Appendix E –Derivative of α_ω

$$\boldsymbol{\alpha}_\omega = \mathbf{R}_2^{-1}(\boldsymbol{q})\left(\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) + \left(\mathbf{A}_{1,d}\boldsymbol{\Omega} + \boldsymbol{e}_3\right)\dot{\alpha}_\psi\right) - \mathbf{K}_3\boldsymbol{q}_e - \frac{2\gamma_1\|\boldsymbol{\Omega}\|\mathbf{L}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\gamma_2 m\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) \tag{E-1}$$

$$\begin{aligned}\dot{\boldsymbol{\alpha}}_\omega = {} & \mathbf{P}_1\dot{\boldsymbol{q}} + \mathbf{P}_2\dot{\boldsymbol{q}} \\ & + \mathbf{R}_2^{-1}(\boldsymbol{q})\Bigg(\mathbf{P}_3\dot{\boldsymbol{\alpha}}_q + \mathbf{P}_4\dot{\boldsymbol{\Omega}} + \mathbf{P}_9\dot{\boldsymbol{\Omega}} + +\mathbf{P}_5\dot{\boldsymbol{\alpha}}_q + \mathbf{P}_6\dot{\boldsymbol{\alpha}}_q + \mathbf{P}_7\dot{\boldsymbol{\alpha}}_2 + \mathbf{P}_8\dot{\boldsymbol{\alpha}}_2 + \mathbf{P}_{10}\dot{\alpha}_\psi \\ & + \mathbf{P}_{11}\left(\frac{\partial\boldsymbol{\Omega}_d}{\partial\boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e} + \frac{\partial\boldsymbol{\Omega}_d}{\partial\hat{\boldsymbol{\nu}}_1}\dot{\hat{\boldsymbol{\nu}}}_1 + \frac{\partial\boldsymbol{\Omega}_d}{\partial\dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial\boldsymbol{\Omega}_d}{\partial\ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d} + \frac{\partial\boldsymbol{\Omega}_d}{\partial\dddot{\boldsymbol{\eta}}_{1d}}\ddddot{\boldsymbol{\eta}}_{1d}\right) \\ & + \frac{\partial}{\partial\boldsymbol{\eta}_{1e}}\left(\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\dot{\boldsymbol{\eta}}_{1e} + \frac{\partial}{\partial\hat{\boldsymbol{\nu}}_1}\left(\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\dot{\hat{\boldsymbol{\nu}}}_1 \\ & + \frac{\partial}{\partial\dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial}{\partial\ddot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\dddot{\boldsymbol{\eta}}_{1d}\Bigg) + \mathbf{P}_{12}\dot{\boldsymbol{\Omega}} \\ & + \mathbf{P}_{13}\dot{\boldsymbol{\alpha}}_2 + \mathbf{P}_{14}\dot{\alpha}_\psi + \mathbf{P}_{15}\ddot{\alpha}_\psi\Bigg)\end{aligned} \tag{E-2}$$

We will define the partial derivatives of $\boldsymbol{\alpha}_\omega$ as follows:

$$\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\alpha}_q} = \mathbf{R}_2^{-1}(\boldsymbol{q})(\mathbf{P}_3 + \mathbf{P}_5 + \mathbf{P}_6) \tag{E-3}$$

$$\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\alpha}_2} = \mathbf{R}_2^{-1}(\boldsymbol{q})\left(\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}(\mathbf{P}_7 + \mathbf{P}_8 + \mathbf{P}_{13}) + \mathbf{P}_{16}\right) + \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\alpha}_q}\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1} \tag{E-4}$$

$$\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\Omega}} = \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\alpha}_2}\mathbf{A}_{2,d} + \mathbf{R}_2^{-1}(\boldsymbol{q})(\mathbf{P}_4 + \mathbf{P}_9 + \mathbf{P}_{12}) \tag{E-5}$$

$$\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{q}} = \mathbf{P}_1 + \mathbf{P}_2 \tag{E-6}$$

$$\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\alpha_\psi} = \mathbf{R}_2^{-1}(\boldsymbol{q})(\mathbf{P}_{10} + \mathbf{P}_{14}) + \frac{\partial\boldsymbol{\alpha}_\omega}{\partial\boldsymbol{\alpha}_2}\left(\begin{bmatrix}0\\0\\1\end{bmatrix} + \mathbf{A}_{1,d}\boldsymbol{\Omega}\right) \tag{E-7}$$

$$\frac{\partial\boldsymbol{\alpha}_\omega}{\partial\dot{\alpha}_\psi} = \mathbf{R}_2^{-1}(\boldsymbol{q})\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{P}_{15} \tag{E-8}$$

$$\frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\eta}_{1e}} = \mathbf{R}_2^{-1}(\boldsymbol{q})\mathbf{P}_{11}\left(\frac{\partial \boldsymbol{\Omega}_d}{\partial \boldsymbol{\eta}_{1e}} + \frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\frac{2\gamma_1 \mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\right) + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\Omega}}\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}} \tag{E-9}$$

$$\frac{\partial \boldsymbol{\alpha}_\omega}{\partial \hat{\boldsymbol{\nu}}_1} = \mathbf{R}_2^{-1}(\boldsymbol{q})\mathbf{P}_{11}\left(\frac{\partial \boldsymbol{\Omega}_d}{\partial \hat{\boldsymbol{\nu}}_1} + \frac{\partial}{\partial \hat{\boldsymbol{\nu}}_1}\left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\frac{2\gamma_1 \mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\right) + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\Omega}}\frac{\partial \boldsymbol{\Omega}}{\partial \hat{\boldsymbol{\nu}}_1} \tag{E-10}$$

$$\frac{\partial \boldsymbol{\alpha}_\omega}{\partial \dot{\boldsymbol{\eta}}_{1d}} = \mathbf{R}_2^{-1}(\boldsymbol{q})\mathbf{P}_{11}\left(\frac{\partial \boldsymbol{\Omega}_d}{\partial \dot{\boldsymbol{\eta}}_{1d}} + \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\frac{2\gamma_1 \mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\right) + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\Omega}}\frac{\partial \boldsymbol{\Omega}}{\partial \dot{\boldsymbol{\eta}}_{1d}} \tag{E-11}$$

$$\frac{\partial \boldsymbol{\alpha}_\omega}{\partial \ddot{\boldsymbol{\eta}}_{1d}} = \mathbf{R}_2^{-1}(\boldsymbol{q})\mathbf{P}_{11}\left(\frac{\partial \boldsymbol{\Omega}_d}{\partial \ddot{\boldsymbol{\eta}}_{1d}} + \frac{\partial}{\partial \ddot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\frac{2\gamma_1 \mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\right) + \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{\Omega}}\frac{\partial \boldsymbol{\Omega}}{\partial \ddot{\boldsymbol{\eta}}_{1d}} \tag{E-12}$$

$$\frac{\partial \boldsymbol{\alpha}_\omega}{\partial \dddot{\boldsymbol{\eta}}_{1d}} = \mathbf{R}_2^{-1}(\boldsymbol{q})\mathbf{P}_{11}\frac{\partial \boldsymbol{\Omega}_d}{\partial \dddot{\boldsymbol{\eta}}_{1d}} \tag{E-13}$$

$$\mathbf{P}_1 = \frac{\partial \boldsymbol{\alpha}_\omega}{\partial \boldsymbol{q}} = \frac{64\boldsymbol{\alpha}_\omega \boldsymbol{q}^{\mathrm{T}}}{(1+\|\boldsymbol{q}\|^2)^3} + \frac{8}{(1+\|\boldsymbol{q}\|^2)^2} * \Bigg(\boldsymbol{q}^{\mathrm{T}}\bigg(\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\bigg(\mathbf{A}_{2,d}\bigg(\boldsymbol{\Omega}_d + \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\frac{2\gamma_1 \mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\bigg) + (\mathbf{A}_{1,d}\boldsymbol{\Omega} + \boldsymbol{e}_3)\dot{\alpha}_\psi\bigg) - \mathbf{K}_3\boldsymbol{q}_e - \frac{2\gamma_1\|\boldsymbol{\Omega}\|\mathbf{L}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\gamma_2 m\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\bigg)\mathbf{I}_{3\times 3}$$

$$+ \boldsymbol{q}\bigg(\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\bigg(\mathbf{A}_{2,d}\bigg(\boldsymbol{\Omega}_d + \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\frac{2\gamma_1 \mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\bigg) + (\mathbf{A}_{1,d}\boldsymbol{\Omega} + \boldsymbol{e}_3)\dot{\alpha}_\psi\bigg) - \mathbf{K}_3\boldsymbol{q}_e - \frac{2\gamma_1\|\boldsymbol{\Omega}\|\mathbf{L}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\gamma_2 m\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\bigg)^{\mathrm{T}}$$

$$- \bigg(\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\bigg(\mathbf{A}_{2,d}\bigg(\boldsymbol{\Omega}_d + \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\frac{2\gamma_1 \mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\bigg) + (\mathbf{A}_{1,d}\boldsymbol{\Omega} + \boldsymbol{e}_3)\dot{\alpha}_\psi\bigg) - \mathbf{K}_3\boldsymbol{q}_e - \frac{2\gamma_1\|\boldsymbol{\Omega}\|\mathbf{L}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\gamma_2 m\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\bigg)\boldsymbol{q}^{\mathrm{T}}$$

$$- \Bigg(\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}\Bigg(\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}^T \bigg(\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\bigg(\mathbf{A}_{2,d}\bigg(\boldsymbol{\Omega}_d + \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\frac{2\gamma_1 \mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\bigg) + (\mathbf{A}_{1,d}\boldsymbol{\Omega} + \boldsymbol{e}_3)\dot{\alpha}_\psi\bigg) - \mathbf{K}_3\boldsymbol{q}_e - \frac{2\gamma_1\|\boldsymbol{\Omega}\|\mathbf{L}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\gamma_2 m\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\bigg)\Bigg)$$

$$+ \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}\Bigg(\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}^T \bigg(\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\bigg(\mathbf{A}_{2,d}\bigg(\boldsymbol{\Omega}_d + \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\frac{2\gamma_1 \mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\bigg) + (\mathbf{A}_{1,d}\boldsymbol{\Omega} + \boldsymbol{e}_3)\dot{\alpha}_\psi\bigg) - \mathbf{K}_3\boldsymbol{q}_e - \frac{2\gamma_1\|\boldsymbol{\Omega}\|\mathbf{L}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\gamma_2 m\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\bigg)\Bigg)$$

$$+ \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}\Bigg(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^T \bigg(\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\bigg(\mathbf{A}_{2,d}\bigg(\boldsymbol{\Omega}_d + \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\frac{2\gamma_1 \mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\bigg) + (\mathbf{A}_{1,d}\boldsymbol{\Omega} + \boldsymbol{e}_3)\dot{\alpha}_\psi\bigg) - \mathbf{K}_3\boldsymbol{q}_e - \frac{2\gamma_1\|\boldsymbol{\Omega}\|\mathbf{L}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\gamma_2 m\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\bigg)\Bigg)\Bigg)\Bigg), \quad \text{(E-14)}$$

$$\mathbf{P}_2\dot{\boldsymbol{q}} = \mathbf{R}_2^{-1}(\boldsymbol{q})\left(-\mathbf{K}_3 - \frac{2\gamma_1\|\boldsymbol{\Omega}\|}{\gamma_2 m\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\frac{\partial\left(\mathbf{L}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}\right)}{\partial \boldsymbol{q}}\right)\dot{\boldsymbol{q}}, \quad \text{(E-15)}$$

where the last partial derivative in the equation above is deffiend in (C-4).

$$\mathbf{P}_3\dot{\boldsymbol{\alpha}}_q = \mathbf{K}_3\dot{\boldsymbol{\alpha}}_q, \tag{E-16}$$

$$\mathbf{P}_4\dot{\boldsymbol{\Omega}} = -\frac{2\gamma_1\mathbf{L}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\gamma_2 m\|\boldsymbol{\Omega}\|\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\boldsymbol{\Omega}^{\mathrm{T}}\left(\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\tilde{\boldsymbol{\nu}}_1 + \frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{\nu}}_1}\mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1\right), \tag{E-17}$$

where $\boldsymbol{\Omega}_d$ is defined in (4-291).

$$\mathbf{P}_5\dot{\boldsymbol{\alpha}}_q = -\frac{2\gamma_1\|\boldsymbol{\Omega}\|}{\gamma_2 m\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\frac{\partial\left(\mathbf{L}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}\right)}{\partial\boldsymbol{\alpha}_q}\dot{\boldsymbol{\alpha}}_q, \tag{E-18}$$

where the last partial derivative in the equation above is deffiend in (C-5).

$$\begin{aligned}
\mathbf{P}_6\dot{\boldsymbol{\alpha}}_q &= \left(\frac{\partial\mathbf{R}_2(\boldsymbol{\alpha}_q)}{\partial\boldsymbol{\alpha}_q}\dot{\boldsymbol{\alpha}}_q\right)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) + \left(\mathbf{A}_{1,d}\boldsymbol{\Omega} + \begin{bmatrix}0\\0\\1\end{bmatrix}\right)\dot{\alpha}_\psi\right) \\
&= \frac{1}{2}\Bigg(\Bigg(\boldsymbol{\alpha}_q^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\right. \\
&\quad \left.+ \left(\mathbf{A}_{1,d}\boldsymbol{\Omega} + \begin{bmatrix}0\\0\\1\end{bmatrix}\right)\dot{\alpha}_\psi\right)\mathbf{I}_{3x3} \\
&\quad + \boldsymbol{\alpha}_q\left(\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) + \left(\mathbf{A}_{1,d}\boldsymbol{\Omega} + \begin{bmatrix}0\\0\\1\end{bmatrix}\right)\dot{\alpha}_\psi\right)\right)^{\mathrm{T}} \\
&\quad - \mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) + \left(\mathbf{A}_{1,d}\boldsymbol{\Omega} + \begin{bmatrix}0\\0\\1\end{bmatrix}\right)\dot{\alpha}_\psi\right)\boldsymbol{\alpha}_q^{\mathrm{T}}\Bigg) \\
&\quad + \Bigg(\begin{bmatrix}0&0&0\\0&0&1\\0&-1&0\end{bmatrix}\left(\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\right.\right. \\
&\quad \left.\left.+ \left(\mathbf{A}_{1,d}\boldsymbol{\Omega} + \begin{bmatrix}0\\0\\1\end{bmatrix}\right)\dot{\alpha}_\psi\right)\begin{bmatrix}1&0&0\end{bmatrix}\right) \\
&\quad + \begin{bmatrix}0&0&-1\\0&0&0\\1&0&0\end{bmatrix}\left(\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\right.\right. \\
&\quad \left.\left.+ \left(\mathbf{A}_{1,d}\boldsymbol{\Omega} + \begin{bmatrix}0\\0\\1\end{bmatrix}\right)\dot{\alpha}_\psi\right)\begin{bmatrix}0&1&0\end{bmatrix}\right) \\
&\quad + \begin{bmatrix}0&1&0\\-1&0&0\\0&0&0\end{bmatrix}\left(\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\right.\right. \\
&\quad \left.\left.+ \left(\mathbf{A}_{1,d}\boldsymbol{\Omega} + \begin{bmatrix}0\\0\\1\end{bmatrix}\right)\dot{\alpha}_\psi\right)\begin{bmatrix}0&0&1\end{bmatrix}\right)\Bigg)\Bigg)\dot{\boldsymbol{\alpha}}_q
\end{aligned} \tag{E-19}$$

$$\mathbf{P}_7\dot{\boldsymbol{\alpha}}_2 = \mathbf{R}_2(\boldsymbol{\alpha}_q)\left(\frac{\partial \mathbf{T}(\boldsymbol{\alpha}_2)^{-1}}{\partial \boldsymbol{\alpha}_2}\dot{\boldsymbol{\alpha}}_q\right)\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d + \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) + \left(\mathbf{A}_{1,d}\boldsymbol{\Omega} + \begin{bmatrix}0\\0\\1\end{bmatrix}\right)\dot{\alpha}_\psi\right) \tag{E-20}$$

$$= \mathbf{R}_2(\boldsymbol{\alpha}_q)\left(\begin{bmatrix}0 & 0 & 0\\ 0 & -\sin(\alpha_\phi) & \cos(\alpha_\theta)\cos(\alpha_\phi)\\ 0 & -\cos(\alpha_\phi) & -\cos(\alpha_\theta)\sin(\alpha_\phi)\end{bmatrix}\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d + \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) + \left(\mathbf{A}_{1,d}\boldsymbol{\Omega} + \begin{bmatrix}0\\0\\1\end{bmatrix}\right)\dot{\alpha}_\psi\right)\begin{bmatrix}1\\0\\0\end{bmatrix}^{\mathrm{T}} + \begin{bmatrix}0 & 0 & -\cos(\alpha_\theta)\\ 0 & 0 & -\sin(\alpha_\theta)\sin(\alpha_\phi)\\ 0 & 0 & -\sin(\alpha_\theta)\cos(\alpha_\phi)\end{bmatrix}\left(\mathbf{A}_{2,d}\left(\boldsymbol{\Omega}_d + \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) + \left(\mathbf{A}_{1,d}\boldsymbol{\Omega} + \begin{bmatrix}0\\0\\1\end{bmatrix}\right)\dot{\alpha}_\psi\right)\begin{bmatrix}0\\1\\0\end{bmatrix}^{\mathrm{T}}\right)\dot{\boldsymbol{\alpha}}_2$$

$$\mathbf{P}_8\dot{\boldsymbol{\alpha}}_2 = \mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\left(\frac{\partial \mathbf{A}_{2,dd}}{\partial \boldsymbol{\alpha}_2}\dot{\boldsymbol{\alpha}}_q\right)\left(\boldsymbol{\Omega}_d + \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) \tag{E-21}$$

$$= \left(\begin{bmatrix}s_{\alpha_\psi}\dfrac{\sin(\alpha_\phi)}{(\cos(\alpha_\phi))^2\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}} & -c_{\alpha_\psi}\dfrac{\sin(\alpha_\phi)}{(\cos(\alpha_\phi))^2\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}} & 0\\ 0 & 0 & 0\\ 0 & 0 & 0\end{bmatrix} * \left(\boldsymbol{\Omega}_d + \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) * \begin{bmatrix}1\\0\\0\end{bmatrix}^{\mathrm{T}} + \begin{bmatrix}0 & 0 & 0\\ -c_{\alpha_\psi}\dfrac{\sin(2\alpha_\theta)}{\Omega_3} & -s_{\alpha_\psi}\dfrac{\sin(2\alpha_\theta)}{\Omega_3} & -\dfrac{\cos(2\alpha_\theta)}{\Omega_3}\\ 0 & 0 & 0\end{bmatrix} * \left(\boldsymbol{\Omega}_d + \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}\right) * \begin{bmatrix}0\\1\\0\end{bmatrix}^{\mathrm{T}} - \begin{bmatrix}\dfrac{1}{(\cos(\alpha_\phi))^2\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}\\ 0\\ 0\end{bmatrix} * \boldsymbol{\Omega}^{\mathrm{T}}\left(\boldsymbol{\Omega}_d + \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) * \begin{bmatrix}1\\0\\0\end{bmatrix}^{\mathrm{T}}\right)\dot{\boldsymbol{\alpha}}_2$$

$$
\begin{aligned}
\mathbf{P}_9\dot{\boldsymbol{\Omega}} = \mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\Bigg(&\begin{bmatrix} \frac{-1}{(\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega})^{1.5}}\frac{\mathrm{s}_{\alpha_\psi}}{\cos(\alpha_\phi)} & \frac{\mathrm{c}_{\alpha_\psi}}{\cos(\alpha_\phi)}\frac{1}{(\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega})^{1.5}} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \\
&* \left(\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\boldsymbol{\Omega}^{\mathrm{T}} \\
&+ \begin{bmatrix} 0 & 0 & 0 \\ -\cos^2(\alpha_\theta)\frac{\mathrm{c}_{\alpha_\psi}}{\Omega_3^2} & -\cos^2(\alpha_\theta)\frac{\mathrm{s}_{\alpha_\psi}}{\Omega_3^2} & \cos^2(\alpha_\theta)\frac{\tan(\alpha_\theta)}{\Omega_3^2} \\ 0 & 0 & 0 \end{bmatrix} \\
&* \left(\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) * \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^{\mathrm{T}} - \begin{bmatrix} \frac{\tan(\alpha_\phi)}{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}} \\ 0 \\ 0 \end{bmatrix} \\
&* \left(\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)^{\mathrm{T}} + 2\begin{bmatrix} \frac{\tan(\alpha_\phi)}{(\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega})^2} \\ 0 \\ 0 \end{bmatrix} \\
&* \boldsymbol{\Omega}^{\mathrm{T}}\left(\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\boldsymbol{\Omega}^{\mathrm{T}}\Bigg)\dot{\boldsymbol{\Omega}}
\end{aligned}
\tag{E-22}
$$

$$
\begin{aligned}
\mathbf{P}_{10}\dot{\alpha}_\psi = \mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}&\begin{bmatrix} \frac{\mathrm{c}_{\alpha_\psi}}{\cos(\alpha_\phi)\,\|\boldsymbol{\Omega}\|} & \frac{\mathrm{s}_{\alpha_\psi}}{\cos(\alpha_\phi)\,\|\boldsymbol{\Omega}\|} & 0 \\ -\cos^2(\alpha_\theta)\frac{\mathrm{s}_{\alpha_\psi}}{\Omega_3} & \cos^2(\alpha_\theta)\frac{c_{\alpha_\psi}}{\Omega_3} & 0 \\ 0 & 0 & 0 \end{bmatrix}\Bigg(\boldsymbol{\Omega}_d \\
&+ \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\Bigg)\dot{\alpha}_\psi
\end{aligned}
\tag{E-23}
$$

$$
\begin{aligned}
\mathbf{P}_{11}\Bigg(&\frac{\partial\boldsymbol{\Omega}_d}{\partial\boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e} + \frac{\partial\boldsymbol{\Omega}_d}{\partial\hat{\boldsymbol{v}}_1}\dot{\hat{\boldsymbol{v}}}_1 + \frac{\partial\boldsymbol{\Omega}_d}{\partial\dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial\boldsymbol{\Omega}_d}{\partial\ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d} + \frac{\partial\boldsymbol{\Omega}_d}{\partial\dddot{\boldsymbol{\eta}}_{1d}}\ddddot{\boldsymbol{\eta}}_{1d} \\
&+ \frac{\partial}{\partial\boldsymbol{\eta}_{1e}}\left(\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\dot{\boldsymbol{\eta}}_{1e} + \frac{\partial}{\partial\hat{\boldsymbol{v}}_1}\left(\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\dot{\hat{\boldsymbol{v}}}_1 \\
&+ \frac{\partial}{\partial\dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial}{\partial\ddot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\dddot{\boldsymbol{\eta}}_{1d}\Bigg) \\
= \mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\mathbf{A}_{2,d}\Bigg(&\frac{\partial\boldsymbol{\Omega}_d}{\partial\boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e} + \frac{\partial\boldsymbol{\Omega}_d}{\partial\hat{\boldsymbol{v}}_1}\dot{\hat{\boldsymbol{v}}}_1 + \frac{\partial\boldsymbol{\Omega}_d}{\partial\dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial\boldsymbol{\Omega}_d}{\partial\ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d} \\
&+ \frac{\partial\boldsymbol{\Omega}_d}{\partial\dddot{\boldsymbol{\eta}}_{1d}}\ddddot{\boldsymbol{\eta}}_{1d} + \frac{\partial}{\partial\boldsymbol{\eta}_{1e}}\left(\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\dot{\boldsymbol{\eta}}_{1e} + \frac{\partial}{\partial\hat{\boldsymbol{v}}_1}\left(\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\dot{\hat{\boldsymbol{v}}}_1 \\
&+ \frac{\partial}{\partial\dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial}{\partial\ddot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\frac{2\gamma_1\mathbf{G}_1^{\mathrm{T}}\boldsymbol{\nu}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)\dddot{\boldsymbol{\eta}}_{1d}\Bigg)
\end{aligned}
\tag{E-24}
$$

For the deffinitions of the partial derivatives in the equation above please refer to Appendix F.

$$\mathbf{P}_{12}\dot{\boldsymbol{\Omega}} = \mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\dot{\alpha}_\psi\left(\begin{bmatrix} \frac{-c_{\alpha_\psi}}{\cos(\alpha_\phi)\,(\boldsymbol{\Omega}^\mathrm{T}\boldsymbol{\Omega})^{1.5}} & \frac{-s_{\alpha_\psi}}{\cos(\alpha_\phi)\,(\boldsymbol{\Omega}^\mathrm{T}\boldsymbol{\Omega})^{1.5}} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} * \boldsymbol{\Omega}\boldsymbol{\Omega}^\mathrm{T} + \begin{bmatrix} 0 & 0 & 0 \\ s_{\alpha_\psi}\frac{\cos^2(\alpha_\theta)}{\Omega_3^2} & -c_{\alpha_\psi}\frac{\cos^2(\alpha_\theta)}{\Omega_3^2} & 0 \\ 0 & 0 & 0 \end{bmatrix}\boldsymbol{\Omega}\begin{bmatrix}0\\0\\1\end{bmatrix}^\mathrm{T} + \mathbf{A}_{1,d}\right)\dot{\boldsymbol{\Omega}} \tag{E-25}$$

$$\mathbf{P}_{13}\dot{\boldsymbol{\alpha}}_2 = \mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\dot{\alpha}_\psi\left(\begin{bmatrix} c_{\alpha_\psi}\frac{\sin(\alpha_\phi)}{(\cos(\alpha_\phi))^2\sqrt{\boldsymbol{\Omega}^\mathrm{T}\boldsymbol{\Omega}}} & s_{\alpha_\psi}\frac{\sin(\alpha_\phi)}{(\cos(\alpha_\phi))^2\sqrt{\boldsymbol{\Omega}^\mathrm{T}\boldsymbol{\Omega}}} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} * \boldsymbol{\Omega} * \begin{bmatrix}1\\0\\0\end{bmatrix}^\mathrm{T} + \begin{bmatrix} 0 & 0 & 0 \\ s_{\alpha_\psi}\frac{\sin(2\alpha_\theta)}{\Omega_3} & -c_{\alpha_\psi}\frac{\sin(2\alpha_\theta)}{\Omega_3} & 0 \\ 0 & 0 & 0 \end{bmatrix} * \boldsymbol{\Omega} * \begin{bmatrix}0\\1\\0\end{bmatrix}^\mathrm{T}\right)\dot{\boldsymbol{\alpha}}_2 \tag{E-26}$$

$$\mathbf{P}_{14}\dot{\alpha}_\psi^2 = \dot{\alpha}_\psi^2\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1} * \begin{bmatrix} -s_{\alpha_\psi}\frac{1}{\cos(\alpha_\phi)\sqrt{\boldsymbol{\Omega}^\mathrm{T}\boldsymbol{\Omega}}} & c_{\alpha_\psi}\frac{1}{\cos(\alpha_\phi)\sqrt{\boldsymbol{\Omega}^\mathrm{T}\boldsymbol{\Omega}}} & 0 \\ -c_{\alpha_\psi}\frac{\cos^2(\alpha_\theta)}{\Omega_3} & -s_{\alpha_\psi}\frac{\cos^2(\alpha_\theta)}{\Omega_3} & 0 \\ 0 & 0 & 0 \end{bmatrix} * \boldsymbol{\Omega} \tag{E-27}$$

$$\mathbf{P}_{15}\ddot{\alpha}_\psi = \mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\left(\mathbf{A}_{1,d}\boldsymbol{\Omega} + \begin{bmatrix}0\\0\\1\end{bmatrix}\right)\ddot{\alpha}_\psi, \tag{E-28}$$

Appendix F –Partial Derivatives of $\boldsymbol{\Omega}_d$

$$\boldsymbol{\eta}_{1e\_dot} = \tilde{\boldsymbol{\nu}}_1 - \dot{\boldsymbol{\eta}}_{1d} \tag{F-1}$$

$$\boldsymbol{\eta}_{1e\_dot} = \tilde{\boldsymbol{\nu}}_1 - \dot{\boldsymbol{\eta}}_{1d} \tag{F-2}$$

$$d\boldsymbol{\eta}_{1e} = \dot{\hat{\boldsymbol{\nu}}}_1 - \ddot{\boldsymbol{\eta}}_{1d} \tag{F-3}$$

$$d\hat{\boldsymbol{\nu}}_1 = -\mathbf{D}_1\hat{\boldsymbol{\nu}}_1 - g\boldsymbol{e}_3 + \frac{f}{m}\mathbf{R}(\boldsymbol{\eta}_2)\boldsymbol{e}_3 + \mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1 + \gamma_1\sigma(\boldsymbol{\eta}_{1e}) + \left(\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)^{\mathrm{T}}\boldsymbol{\nu}_{1e} + \boldsymbol{h}_4 \tag{F-4}$$

$$\begin{aligned} d\hat{\boldsymbol{\nu}}_1 = &-\left(\mathbf{D}_1 + \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\boldsymbol{\nu}_{1e}\hat{\boldsymbol{\nu}}_1^{\mathrm{T}} - \left(\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)^{\mathrm{T}}\mathbf{G}_1 + \frac{\mathbf{R}(\boldsymbol{\eta}_2)\boldsymbol{e}_3\boldsymbol{\Omega}^{\mathrm{T}}}{m\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{\nu}}_1} + \frac{\partial\boldsymbol{h}_4}{\partial\boldsymbol{\eta}_{1e}}\right)\dot{\hat{\boldsymbol{\nu}}}_1 \\ &+ \left(\gamma_1\sigma'(\boldsymbol{\eta}_{1e}) + \frac{\mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\begin{bmatrix}\boldsymbol{\nu}_{1e,1} & 0 & 0\\ 0 & \boldsymbol{\nu}_{1e,2} & 0\\ 0 & 0 & \boldsymbol{\nu}_{1e,3}\end{bmatrix} + \frac{\mathbf{R}(\boldsymbol{\eta}_2)\boldsymbol{e}_3\boldsymbol{\Omega}^{\mathrm{T}}}{m\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}}\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\right. \\ &\left. + \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \frac{\partial\boldsymbol{h}_4}{\partial\boldsymbol{\eta}_{1e}}\right)\dot{\boldsymbol{\eta}}_{1e} + \left(\frac{\mathbf{R}(\boldsymbol{\eta}_2)\boldsymbol{e}_3\boldsymbol{\Omega}^{\mathrm{T}}}{m\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}}\frac{\partial\boldsymbol{\Omega}}{\partial\dot{\boldsymbol{\eta}}_{1d}} + \frac{\partial\boldsymbol{h}_4}{\partial\dot{\boldsymbol{\eta}}_{1d}}\right)\ddot{\boldsymbol{\eta}}_{1d} \\ &+ \left(\frac{\mathbf{R}(\boldsymbol{\eta}_2)\boldsymbol{e}_3\boldsymbol{\Omega}^{\mathrm{T}}}{m\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}}\frac{\partial\boldsymbol{\Omega}}{\partial\ddot{\boldsymbol{\eta}}_{1d}} + \frac{\partial\boldsymbol{h}_4}{\partial\ddot{\boldsymbol{\eta}}_{1d}}\right)\dddot{\boldsymbol{\eta}}_{1d} + \frac{\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}}{m}\frac{\partial\mathbf{R}(\boldsymbol{\eta}_2)\boldsymbol{e}_3}{\partial\boldsymbol{\eta}_2}\dot{\boldsymbol{\eta}}_2 + \frac{\partial\boldsymbol{h}_4}{\partial\alpha_\psi}\dot{\alpha}_\psi + \frac{\partial\boldsymbol{h}_4}{\partial\boldsymbol{q}}\dot{\boldsymbol{q}} \end{aligned} \tag{F-5}$$

$$\begin{aligned} d\hat{\boldsymbol{\nu}}_1 = &\frac{\partial\hat{\boldsymbol{\nu}}_{1\_dot}}{\partial\hat{\boldsymbol{\nu}}_1}\dot{\hat{\boldsymbol{\nu}}}_1 + \frac{\partial\hat{\boldsymbol{\nu}}_{1\_dot}}{\partial\boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e} + \frac{\partial\hat{\boldsymbol{\nu}}_{1\_dot}}{\partial\dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial\hat{\boldsymbol{\nu}}_{1\_dot}}{\partial\ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d} + \frac{\partial\hat{\boldsymbol{\nu}}_{1\_dot}}{\partial\boldsymbol{\eta}_2}\dot{\boldsymbol{\eta}}_2 + \frac{\partial\hat{\boldsymbol{\nu}}_{1\_dot}}{\partial\alpha_\psi}\dot{\alpha}_\psi \\ &+ \frac{\partial\hat{\boldsymbol{\nu}}_{1\_dot}}{\partial\boldsymbol{q}}\dot{\boldsymbol{q}} \end{aligned} \tag{F-6}$$

In the following equation the partial derivatives of the term $\boldsymbol{h}_4$ are defined in Appendix G.

$$\frac{\partial d\hat{\boldsymbol{\nu}}_1}{\partial\hat{\boldsymbol{\nu}}_1} = -\left(\mathbf{D}_1 + \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\boldsymbol{\nu}_{1e}\hat{\boldsymbol{\nu}}_1^{\mathrm{T}} - \left(\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)^{\mathrm{T}}\mathbf{G}_1 + \frac{\mathbf{R}(\boldsymbol{\eta}_2)\boldsymbol{e}_3\boldsymbol{\Omega}^{\mathrm{T}}}{m\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}}\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{\nu}}_1} + \frac{\partial\boldsymbol{h}_4}{\partial\boldsymbol{\eta}_{1e}}\right) \tag{F-7}$$

$$\begin{aligned} \frac{\partial d\hat{\boldsymbol{\nu}}_1}{\partial\boldsymbol{\eta}_{1e}} = &\gamma_1\sigma'(\boldsymbol{\eta}_{1e}) + \frac{\mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\begin{bmatrix}\boldsymbol{\nu}_{1e,1} & 0 & 0\\ 0 & \boldsymbol{\nu}_{1e,2} & 0\\ 0 & 0 & \boldsymbol{\nu}_{1e,3}\end{bmatrix} + \frac{\mathbf{R}(\boldsymbol{\eta}_2)\boldsymbol{e}_3\boldsymbol{\Omega}^{\mathrm{T}}}{m\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}}\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}} \\ &+ \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \frac{\partial\boldsymbol{h}_4}{\partial\boldsymbol{\eta}_{1e}} \end{aligned} \tag{F-8}$$

$$\frac{\partial d\hat{\boldsymbol{v}}_1}{\partial \dot{\boldsymbol{\eta}}_{1d}} = \frac{\mathbf{R}(\boldsymbol{\eta}_2)\boldsymbol{e}_3\boldsymbol{\Omega}^{\mathrm{T}}}{m\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}}\frac{\partial \boldsymbol{\Omega}}{\partial \dot{\boldsymbol{\eta}}_{1d}} + \frac{\partial \boldsymbol{h}_4}{\partial \dot{\boldsymbol{\eta}}_{1d}} \quad \text{(F-9)}$$

$$\frac{\partial d\hat{\boldsymbol{v}}_1}{\partial \ddot{\boldsymbol{\eta}}_{1d}} = \frac{\mathbf{R}(\boldsymbol{\eta}_2)\boldsymbol{e}_3\boldsymbol{\Omega}^{\mathrm{T}}}{m\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}}\frac{\partial \boldsymbol{\Omega}}{\partial \ddot{\boldsymbol{\eta}}_{1d}} + \frac{\partial \boldsymbol{h}_4}{\partial \ddot{\boldsymbol{\eta}}_{1d}} \quad \text{(F-10)}$$

$$\frac{\partial d\hat{\boldsymbol{v}}_1}{\partial \boldsymbol{q}} = \frac{\sqrt{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}}{m}\frac{\partial \mathbf{R}(\boldsymbol{q})\boldsymbol{e}_3}{\partial \boldsymbol{q}} + \frac{\partial \boldsymbol{h}_4}{\partial \boldsymbol{q}} \quad \text{(F-11)}$$

$$\frac{\partial d\hat{\boldsymbol{v}}_1}{\partial \alpha_{\psi}} = \frac{\partial \boldsymbol{h}_4}{\partial \alpha_{\psi}} \quad \text{(F-12)}$$

The partial derivatives of the term $\boldsymbol{h}_4$ are defined in Appendix G.

$$\boldsymbol{\Omega}_d = \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\boldsymbol{\eta}_{1e\_dot} + \frac{\partial \boldsymbol{\Omega}}{\partial \hat{\boldsymbol{v}}_1}\hat{\boldsymbol{v}}_{1\_dot} + \frac{\partial \boldsymbol{\Omega}}{\partial \dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial \boldsymbol{\Omega}}{\partial \ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d} + \frac{\partial \boldsymbol{\Omega}_d}{\partial \boldsymbol{q}}\dot{\boldsymbol{q}} + \frac{\partial \boldsymbol{\Omega}_d}{\partial \boldsymbol{\eta}_2}\dot{\boldsymbol{\eta}}_2 \quad \text{(F-13)}$$

$$\dot{\hat{\boldsymbol{v}}}_1 = -\mathbf{D}_1\hat{\boldsymbol{v}}_1 - g\boldsymbol{e}_3 + \frac{f}{m}\mathbf{R}(\boldsymbol{\eta}_2)\boldsymbol{e}_3 + \mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1 + \gamma_1\sigma(\boldsymbol{\eta}_{1e}) + \left(\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)^{\mathrm{T}}\boldsymbol{v}_{1e} + \boldsymbol{h}_4 + \boldsymbol{h}_5 \quad \text{(F-14)}$$

$$\boldsymbol{\Omega}_d = \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\boldsymbol{\eta}_{1e\_dot} + \frac{\partial \boldsymbol{\Omega}}{\partial \hat{\boldsymbol{v}}_1}\hat{\boldsymbol{v}}_{1\_dot} + \frac{\partial \boldsymbol{\Omega}}{\partial \dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial \boldsymbol{\Omega}}{\partial \ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d} + \frac{\partial \boldsymbol{\Omega}_d}{\partial \boldsymbol{q}}\dot{\boldsymbol{q}} + \frac{\partial \boldsymbol{\Omega}_d}{\partial \boldsymbol{\eta}_2}\dot{\boldsymbol{\eta}}_2 \quad \text{(F-15)}$$

$$\frac{\partial \boldsymbol{\Omega}_d}{\partial \hat{\boldsymbol{v}}_1} = m\sum_{j=1}^{3}\mathbf{Y}_{1,j,1} + m\mathbf{N}_1\sum_{i=2}^{7}\sum_{j=1}^{3}\mathbf{Y}_{i,j,1} + \sum_{j=1}^{3}\mathbf{Y}_{8,j,1} + \mathbf{Y}_{9,1} + \frac{\partial \boldsymbol{\Omega}}{\partial \hat{\boldsymbol{v}}_1}\frac{\partial d\hat{\boldsymbol{v}}_1}{\partial \hat{\boldsymbol{v}}_1} + \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\frac{\partial d\boldsymbol{\eta}_{1e}}{\partial \hat{\boldsymbol{v}}_1} \quad \text{(F-16)}$$

$$\frac{\partial \boldsymbol{\Omega}_d}{\partial \boldsymbol{\eta}_{1e}} = m\sum_{j=1}^{3}\mathbf{Y}_{1,j,2} + m\mathbf{N}_1\sum_{i=2}^{7}\sum_{j=1}^{3}\mathbf{Y}_{i,j,2} + \sum_{j=1}^{3}\mathbf{Y}_{8,j,2} + \mathbf{Y}_{9,2} + \frac{\partial \boldsymbol{\Omega}}{\partial \hat{\boldsymbol{v}}_1}\frac{\partial d\hat{\boldsymbol{v}}_1}{\partial \boldsymbol{\eta}_{1e}} \quad \text{(F-17)}$$

$$\frac{\partial \boldsymbol{\Omega}_d}{\partial \dot{\boldsymbol{\eta}}_{1d}} = m\sum_{j=1}^{3}\mathbf{Y}_{1,j,3} + m\mathbf{N}_1\sum_{i=2}^{7}\sum_{j=1}^{3}\mathbf{Y}_{i,j,3} + \sum_{j=1}^{3}\mathbf{Y}_{8,j,3} + \frac{\partial \boldsymbol{\Omega}}{\partial \hat{\boldsymbol{v}}_1}\frac{\partial d\hat{\boldsymbol{v}}_1}{\partial \dot{\boldsymbol{\eta}}_{1d}} \quad \text{(F-18)}$$

$$\frac{\partial \boldsymbol{\Omega}_d}{\partial \ddot{\boldsymbol{\eta}}_{1d}} = m\sum_{j=1}^{3}\mathbf{Y}_{,j,4} + m\mathbf{N}_1\sum_{i=2}^{7}\sum_{j=1}^{3}\mathbf{Y}_{i,j,4} + \frac{\partial \boldsymbol{\Omega}}{\partial \hat{\boldsymbol{v}}_1}\frac{\partial d\hat{\boldsymbol{v}}_1}{\partial \ddot{\boldsymbol{\eta}}_{1d}} \quad \text{(F-19)}$$

$$\frac{\partial \boldsymbol{\Omega}_d}{\partial \dddot{\boldsymbol{\eta}}_{1d}} = \frac{\partial \boldsymbol{\Omega}}{\partial \ddot{\boldsymbol{\eta}}_{1d}}\frac{\partial \ddot{\boldsymbol{\eta}}_{1d}}{\partial \dddot{\boldsymbol{\eta}}_{1d}} \quad \text{(F-20)}$$

$$\frac{\partial \boldsymbol{\Omega}_d}{\partial \boldsymbol{q}} = \frac{\partial \boldsymbol{\Omega}}{\partial \hat{\boldsymbol{v}}_1} \frac{\partial d\hat{\boldsymbol{v}}_1}{\partial \boldsymbol{q}} \quad \text{(F-21)}$$

$$\frac{\partial \boldsymbol{\Omega}_d}{\partial \widetilde{\boldsymbol{\eta}}_1} = \frac{\partial \boldsymbol{\Omega}}{\partial \hat{\boldsymbol{v}}_1} \frac{\partial d\hat{\boldsymbol{v}}_1}{\partial \widetilde{\boldsymbol{\eta}}_1} \quad \text{(F-22)}$$

N1

$$\left(\frac{\partial \mathbf{N}}{\partial \hat{\boldsymbol{v}}_1}\dot{\hat{\boldsymbol{v}}}_1\right)(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6) = \frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}})^2-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1}\mathbf{N}_1^{\mathrm{T}} * \left(\mathbf{I}_{3x3}-2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\dot{\hat{\boldsymbol{v}}}_1 \tag{F-23}$$

$$\left(\frac{\partial \mathbf{N}}{\partial \boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e}\right)(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6) = \frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)}{\Delta_1(\hat{\boldsymbol{v}}_1)^2-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1}\mathbf{N}\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\dot{\boldsymbol{\eta}}_{1e} \tag{F-24}$$

$$\begin{aligned}\mathbf{Y}_{1,1,1}\dot{\hat{\boldsymbol{v}}}_1 &= \left(\frac{\partial}{\partial \hat{\boldsymbol{v}}_1}\left(\left(\frac{\partial \mathbf{N}}{\partial \hat{\boldsymbol{v}}_1}\right)(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)\right)\dot{\hat{\boldsymbol{v}}}_1\right)\dot{\hat{\boldsymbol{v}}}_1 \\
&= \left(-\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})d\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{(\Delta_1(\hat{\boldsymbol{v}})^2-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1)^2}\left(\mathbf{I}_{3x3}-2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\mathbf{N}_1\right. \\
&* (\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)\left(2*\Delta_1(\hat{\boldsymbol{v}})\hat{\boldsymbol{v}}_1^{\mathrm{T}}-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\right) \\
&+ \frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}})^2-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1}\mathbf{N}_1^{\mathrm{T}} \\
&* \left(-\frac{2}{\Delta_1(\hat{\boldsymbol{v}}_1)}\left(\hat{\boldsymbol{v}}_1^{\mathrm{T}}d\hat{\boldsymbol{v}}_1\mathbf{I}_{3x3}++\hat{\boldsymbol{v}}_1 d\hat{\boldsymbol{v}}_1^{\mathrm{T}}\right)+2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^2}d\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}\right) \\
&+ \frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})d\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}})^2-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1} * \left(\mathbf{I}_{3x3}-2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right) \\
&* \frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}})^2-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1}\mathbf{N}_1^{\mathrm{T}} \\
&* \left(\mathbf{I}_{3x3}-2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right) + \frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})d\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}})^2-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1} * \left(\mathbf{I}_{3x3}-2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\mathbf{N}_1 \\
&\left.* \left(\frac{\partial \mathbf{M}_1}{\partial \hat{\boldsymbol{v}}_1}+\frac{\partial \mathbf{M}_2}{\partial \hat{\boldsymbol{v}}_1}+\frac{\partial \mathbf{M}_3}{\partial \hat{\boldsymbol{v}}_1}+\frac{\partial \mathbf{M}_4}{\partial \hat{\boldsymbol{v}}_1}+\frac{\partial \mathbf{M}_5}{\partial \hat{\boldsymbol{v}}_1}+\frac{\partial \mathbf{M}_6}{\partial \hat{\boldsymbol{v}}_1}\right)\right)\dot{\hat{\boldsymbol{v}}}_1\end{aligned} \tag{F-25}$$

$$\mathbf{Y}_{1,1,2}\dot{\boldsymbol{\eta}}_{1e} = \left(\frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{\partial \mathbf{N}}{\partial \hat{\boldsymbol{v}}_1}\dot{\hat{\boldsymbol{v}}}_1\right)\dot{\boldsymbol{\eta}}_{1e}\right)(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)$$
$$= \left(\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})d\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{(\Delta_1(\hat{\boldsymbol{v}})^2-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1)^2}\left(\mathbf{I}_{3x3}-2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\mathbf{N}_1\right.$$
$$* (\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)\hat{\boldsymbol{v}}_1^{\mathrm{T}}\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})$$
$$+\frac{d\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}})^2-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1} * \left(\mathbf{I}_{3x3}-2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\mathbf{N}_1$$
$$* (\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})$$
$$+\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})d\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}})^2-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1}\left(\mathbf{I}_{3x3}-2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)$$
$$*\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)}{\Delta_1(\hat{\boldsymbol{v}}_1)^2-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1}\mathbf{N}\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})$$
$$+\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})d\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}})^2-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1} * \left(\mathbf{I}_{3x3}-2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\mathbf{N}_1$$
$$\left.*\left(\frac{\partial\mathbf{M}_1}{\partial\boldsymbol{\eta}_{1e}}+\frac{\partial\mathbf{M}_2}{\partial\boldsymbol{\eta}_{1e}}+\frac{\partial\mathbf{M}_3}{\partial\boldsymbol{\eta}_{1e}}+\frac{\partial\mathbf{M}_4}{\partial\boldsymbol{\eta}_{1e}}+\frac{\partial\mathbf{M}_5}{\partial\boldsymbol{\eta}_{1e}}+\frac{\partial\mathbf{M}_6}{\partial\boldsymbol{\eta}_{1e}}\right)\right)\dot{\boldsymbol{\eta}}_{1e} \tag{F-26}$$

$$\mathbf{Y}_{1,1,3}\ddot{\boldsymbol{\eta}}_{1d} = \left(\frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{N}}{\partial \hat{\boldsymbol{v}}_1}\dot{\hat{\boldsymbol{v}}}_1\right)\ddot{\boldsymbol{\eta}}_{1d}\right)(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)$$
$$=\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})d\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}})^2-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1} * \left(\mathbf{I}_{3x3}-2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\mathbf{N}_1$$
$$*\left(\frac{\partial\mathbf{M}_1}{\partial\dot{\boldsymbol{\eta}}_{1d}}+\frac{\partial\mathbf{M}_2}{\partial\dot{\boldsymbol{\eta}}_{1d}}+\frac{\partial\mathbf{M}_3}{\partial\dot{\boldsymbol{\eta}}_{1d}}+\frac{\partial\mathbf{M}_4}{\partial\dot{\boldsymbol{\eta}}_{1d}}+\frac{\partial\mathbf{M}_5}{\partial\dot{\boldsymbol{\eta}}_{1d}}+\frac{\partial\mathbf{M}_6}{\partial\dot{\boldsymbol{\eta}}_{1d}}\right)\ddot{\boldsymbol{\eta}}_{1d} \tag{F-27}$$

$$\mathbf{Y}_{1,1,4}\dddot{\boldsymbol{\eta}}_{1d} = \left(\frac{\partial}{\partial \ddot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{N}}{\partial \hat{\boldsymbol{v}}_1}\dot{\hat{\boldsymbol{v}}}_1\right)\dddot{\boldsymbol{\eta}}_{1d}\right)(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)$$
$$=\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})d\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}})^2-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1} * \left(\mathbf{I}_{3x3}-2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\mathbf{N}_1 * \frac{\partial\mathbf{M}_6}{\partial\ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d} \tag{F-28}$$

$$\mathbf{Y}_{1,2,1}\dot{\hat{\boldsymbol{v}}}_1 = \left(\frac{\partial}{\partial \hat{\boldsymbol{v}}_1}\left(\frac{\partial \mathbf{N}}{\partial \boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e}\right)\dot{\hat{\boldsymbol{v}}}_1\right)(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)$$
$$=\left(-\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)}{(\Delta_1(\hat{\boldsymbol{v}}_1)^2-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1)^2}\mathbf{N}\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e}\right.$$
$$*\left(2*\Delta_1(\hat{\boldsymbol{v}})\hat{\boldsymbol{v}}_1^{\mathrm{T}}-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\right)+\frac{\mathbf{N}\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1}$$
$$*(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)^{\mathrm{T}}+\frac{\mathbf{N}_1\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e}\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1}$$
$$*\left(\frac{\partial\mathbf{M}_1}{\partial\hat{\boldsymbol{v}}_1}+\frac{\partial\mathbf{M}_2}{\partial\hat{\boldsymbol{v}}_1}+\frac{\partial\mathbf{M}_3}{\partial\hat{\boldsymbol{v}}_1}+\frac{\partial\mathbf{M}_4}{\partial\hat{\boldsymbol{v}}_1}+\frac{\partial\mathbf{M}_5}{\partial\hat{\boldsymbol{v}}_1}+\frac{\partial\mathbf{M}_6}{\partial\hat{\boldsymbol{v}}_1}\right)$$
$$+\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)}{\Delta_1(\hat{\boldsymbol{v}}_1)^2-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1}\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})(\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e})^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}})^2-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1}\mathbf{N}_1^{\mathrm{T}}$$
$$\left.*\left(\mathbf{I}_{3x3}-2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\right)\dot{\hat{\boldsymbol{v}}}_1 \tag{F-29}$$

$$\mathbf{Y}_{1,2,2}\dot{\boldsymbol{\eta}}_{1e} = \left(\frac{\partial}{\partial\boldsymbol{\eta}_{1e}}\left(\frac{\partial\mathbf{N}}{\partial\boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e}\right)\dot{\boldsymbol{\eta}}_{1e}\right)(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)$$
$$= \frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)}{(\Delta_1(\hat{\boldsymbol{v}}_1)^2-\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1)^2}\mathbf{N}_1\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e}\hat{\boldsymbol{v}}_1^{\mathrm{T}}\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})$$
$$+\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)}{\Delta_1(\hat{\boldsymbol{v}}_1)^2-\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1}\mathbf{N}_1\mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e})*\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0\\ 0 & d\boldsymbol{\eta}_{1e,2} & 0\\ 0 & 0 & d\boldsymbol{\eta}_{1e,3}\end{bmatrix}$$
$$+\frac{\mathbf{N}_1\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e}\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2-\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1}*\left(\frac{\partial\mathbf{M}_1}{\partial\boldsymbol{\eta}_{1e}}+\frac{\partial\mathbf{M}_2}{\partial\boldsymbol{\eta}_{1e}}+\frac{\partial\mathbf{M}_3}{\partial\boldsymbol{\eta}_{1e}}+\frac{\partial\mathbf{M}_4}{\partial\boldsymbol{\eta}_{1e}}+\frac{\partial\mathbf{M}_5}{\partial\boldsymbol{\eta}_{1e}}+\frac{\partial\mathbf{M}_6}{\partial\boldsymbol{\eta}_{1e}}\right)$$
$$+\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)}{(\Delta_1(\hat{\boldsymbol{v}}_1)^2-\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1)^2}\left(\hat{\boldsymbol{v}}_1^{\mathrm{T}}\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e}\right)\mathbf{N}_1\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e}) \tag{F-30}$$

$$\mathbf{Y}_{1,2,3}\ddot{\boldsymbol{\eta}}_{1d} = \left(\frac{\partial}{\partial\dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial\mathbf{N}}{\partial\boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e}\right)\ddot{\boldsymbol{\eta}}_{1d}\right)(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)$$
$$=\frac{\mathbf{N}_1\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e}\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2-\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1}$$
$$*\left(\frac{\partial\mathbf{M}_1}{\partial\dot{\boldsymbol{\eta}}_{1d}}+\frac{\partial\mathbf{M}_2}{\partial\dot{\boldsymbol{\eta}}_{1d}}+\frac{\partial\mathbf{M}_3}{\partial\dot{\boldsymbol{\eta}}_{1d}}+\frac{\partial\mathbf{M}_4}{\partial\dot{\boldsymbol{\eta}}_{1d}}+\frac{\partial\mathbf{M}_5}{\partial\dot{\boldsymbol{\eta}}_{1d}}+\frac{\partial\mathbf{M}_6}{\partial\dot{\boldsymbol{\eta}}_{1d}}\right)\ddot{\boldsymbol{\eta}}_{1d} \tag{F-31}$$

$$\mathbf{Y}_{1,2,4}\dddot{\boldsymbol{\eta}}_{1d} = \left(\frac{\partial}{\partial\ddot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial\mathbf{N}}{\partial\boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e}\right)\dddot{\boldsymbol{\eta}}_{1d}\right)(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)$$
$$=\frac{\mathbf{N}_1\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e}\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2-\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1}\frac{\partial\mathbf{M}_6}{\partial\ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d} \tag{F-32}$$

M1

$$\mathbf{M}_1 = \mathbf{D}_1\boldsymbol{\alpha}_1 - \mathbf{K}_2\sigma(\boldsymbol{\nu}_{1e}) \tag{F-33}$$

$$\frac{\partial\mathbf{M}_1}{\partial\hat{\boldsymbol{v}}_1}\dot{\hat{\boldsymbol{v}}}_1 = \left(\mathbf{D}_1\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} - \mathbf{K}_2\sigma'(\boldsymbol{\nu}_{1e})\mathbf{G}_1\right)\dot{\hat{\boldsymbol{v}}}_1 \tag{F-34}$$

$$\mathbf{Y}_{2,1,1} = \frac{\partial}{\partial\hat{\boldsymbol{v}}_1}\left(\frac{\partial\mathbf{M}_1}{\partial\hat{\boldsymbol{v}}_1}\right)\begin{bmatrix} d\hat{\boldsymbol{v}}_{1,1} & 0 & 0 \\ 0 & d\hat{\boldsymbol{v}}_{1,2} & 0 \\ 0 & 0 & d\hat{\boldsymbol{v}}_{1,3}\end{bmatrix} \tag{F-35}$$

$$= \mathbf{D}_1\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})\left(\frac{d\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} - 2\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}d\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^3}\hat{\boldsymbol{v}}_1^{\mathrm{T}}\right) - \mathbf{K}_2\sigma''(\boldsymbol{\nu}_{1e})\left(\begin{bmatrix}1&0&0\\0&0&0\\0&0&0\end{bmatrix}\left(\begin{bmatrix}1\\0\\0\end{bmatrix}^{\mathrm{T}}\mathbf{G}_1 d\hat{\boldsymbol{v}}_{1,1}\right)\right.$$

$$\left. + \begin{bmatrix}0&0&0\\0&1&0\\0&0&0\end{bmatrix}\left(\begin{bmatrix}0\\1\\0\end{bmatrix}^{\mathrm{T}}\mathbf{G}_1 d\hat{\boldsymbol{v}}_{1,1}\right) + \begin{bmatrix}0&0&0\\0&0&0\\0&0&1\end{bmatrix}\left(\begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}}\mathbf{G}_1 d\hat{\boldsymbol{v}}_{1,1}\right)\right)\mathbf{G}_1$$

$$-\mathbf{K}_2\sigma'(\boldsymbol{\nu}_{1e})\left(-\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})d\hat{\boldsymbol{v}}_1^{\mathrm{T}}\left(\frac{\mathbf{I}_{3x3}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} - 2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^3}\right)\right)$$

$$\mathbf{Y}_{2,1,2} = \frac{\partial}{\partial\boldsymbol{\eta}_{1e}}\left(\frac{\partial\mathbf{M}_1}{\partial\hat{\boldsymbol{v}}_1}\right)\begin{bmatrix} d\hat{\boldsymbol{v}}_{1,1} & 0 & 0 \\ 0 & d\hat{\boldsymbol{v}}_{1,2} & 0 \\ 0 & 0 & d\hat{\boldsymbol{v}}_{1,3}\end{bmatrix} \tag{F-36}$$

$$= \mathbf{D}_1\mathbf{K}_1\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}d\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\sigma'(\boldsymbol{\eta}_{1e})\dot{\boldsymbol{\eta}}_{1e} + \mathbf{K}_2\sigma'(\boldsymbol{\nu}_{1e})\left(\mathbf{K}_1\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}d\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\sigma'(\boldsymbol{\eta}_{1e})\right)$$

$$-\mathbf{K}_2\sigma''(\boldsymbol{\nu}_{1e})\left(\begin{bmatrix}1&0&0\\0&0&0\\0&0&0\end{bmatrix}\left(\begin{bmatrix}1\\0\\0\end{bmatrix}^{\mathrm{T}}\mathbf{G}_1 d\hat{\boldsymbol{v}}_{1,1}\right) + \begin{bmatrix}0&0&0\\0&1&0\\0&0&0\end{bmatrix}\left(\begin{bmatrix}0\\1\\0\end{bmatrix}^{\mathrm{T}}\mathbf{G}_1 d\hat{\boldsymbol{v}}_{1,1}\right)\right.$$

$$\left. + \begin{bmatrix}0&0&0\\0&0&0\\0&0&1\end{bmatrix}\left(\begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}}\mathbf{G}_1 d\hat{\boldsymbol{v}}_{1,1}\right)\right)\left(\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)$$

$$\mathbf{Y}_{2,1,3}\ddot{\boldsymbol{\eta}}_{1d} = \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{M}_1}{\partial \hat{\boldsymbol{v}}_1}\right)\begin{bmatrix} d\hat{\boldsymbol{v}}_{1,1} & 0 & 0 \\ 0 & d\hat{\boldsymbol{v}}_{1,2} & 0 \\ 0 & 0 & d\hat{\boldsymbol{v}}_{1,3} \end{bmatrix}$$
$$= \mathbf{K}_2\sigma''(\boldsymbol{v}_{1e})\left(\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}\left(\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}^{\mathrm{T}}\mathbf{G}_1 d\hat{\boldsymbol{v}}_{1,1}\right) + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}\left(\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}^{\mathrm{T}}\mathbf{G}_1 d\hat{\boldsymbol{v}}_{1,1}\right)\right.$$
$$\left. + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}\left(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^{\mathrm{T}}\mathbf{G}_1 d\hat{\boldsymbol{v}}_{1,1}\right)\right) \quad \text{(F-37)}$$

$$\frac{\partial \mathbf{M}_1}{\partial \boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e} = -\left(\mathbf{D}_1 + \mathbf{K}_2\sigma'(\boldsymbol{v}_{1e})\right)\mathbf{K}_1\frac{\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\dot{\boldsymbol{\eta}}_{1e} \quad \text{(F-38)}$$

$$\mathbf{Y}_{2,2,1} = \frac{\partial}{\partial \hat{\boldsymbol{v}}_1}\left(\frac{\partial \mathbf{M}_1}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3} \end{bmatrix}$$
$$= -\left(\mathbf{K}_2\sigma''(\boldsymbol{v}_{1e})\right)\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3} \end{bmatrix}\mathbf{G}_1$$
$$+\left(\mathbf{D}_1 + \mathbf{K}_2\sigma'(\boldsymbol{v}_{1e})\right)\mathbf{K}_1\frac{\sigma'(\boldsymbol{\eta}_{1e})}{\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^2}d\boldsymbol{\eta}_{1e}\hat{\boldsymbol{v}}_1^{\mathrm{T}} \quad \text{(F-39)}$$

$$\mathbf{Y}_{2,2,2}\dot{\boldsymbol{\eta}}_{1e} = \frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{\partial \mathbf{M}_1}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3} \end{bmatrix}$$
$$= -\left(\mathbf{K}_2\sigma''(\boldsymbol{v}_{1e})\right)\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3} \end{bmatrix}\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}$$
$$-\left(\mathbf{D}_1 + \mathbf{K}_2\sigma'(\boldsymbol{v}_{1e})\right)\mathbf{K}_1\frac{\sigma''(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3} \end{bmatrix} \quad \text{(F-40)}$$

$$\mathbf{Y}_{2,2,3}\ddot{\boldsymbol{\eta}}_{1d} = \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{M}_1}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3} \end{bmatrix}$$
$$= \left(\mathbf{K}_2\sigma''(\boldsymbol{v}_{1e})\right)\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3} \end{bmatrix} \quad \text{(F-41)}$$

$$\frac{\partial \mathbf{M}_1}{\partial \dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} = \left(\mathbf{D}_1 + \mathbf{K}_2\sigma'(\boldsymbol{v}_{1e})\right)\ddot{\boldsymbol{\eta}}_{1d} \quad \text{(F-42)}$$

$$\mathbf{Y}_{2,3,1}\dot{\hat{\boldsymbol{\nu}}}_1 = \frac{\partial}{\partial \hat{\boldsymbol{\nu}}_1}\left(\frac{\partial \mathbf{M}_1}{\partial \dot{\boldsymbol{\eta}}_{1d}}\right)\begin{bmatrix} \ddot{\boldsymbol{\eta}}_{1d,1} & 0 & 0 \\ 0 & \ddot{\boldsymbol{\eta}}_{1d,2} & 0 \\ 0 & 0 & \ddot{\boldsymbol{\eta}}_{1d,3} \end{bmatrix} = \mathbf{K}_2\sigma''(\boldsymbol{\nu}_{1e})\begin{bmatrix} \ddot{\boldsymbol{\eta}}_{1d,1} & 0 & 0 \\ 0 & \ddot{\boldsymbol{\eta}}_{1d,2} & 0 \\ 0 & 0 & \ddot{\boldsymbol{\eta}}_{1d,3} \end{bmatrix}\mathbf{G}_1 \tag{F-43}$$

$$\begin{aligned}\mathbf{Y}_{2,3,2}\dot{\boldsymbol{\eta}}_{1e} &= \frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{\partial \mathbf{M}_1}{\partial \dot{\boldsymbol{\eta}}_{1d}}\right)\begin{bmatrix} \ddot{\boldsymbol{\eta}}_{1d,1} & 0 & 0 \\ 0 & \ddot{\boldsymbol{\eta}}_{1d,2} & 0 \\ 0 & 0 & \ddot{\boldsymbol{\eta}}_{1d,3} \end{bmatrix} \\ &= \mathbf{K}_2\sigma''(\boldsymbol{\nu}_{1e})\begin{bmatrix} \ddot{\boldsymbol{\eta}}_{1d,1} & 0 & 0 \\ 0 & \ddot{\boldsymbol{\eta}}_{1d,2} & 0 \\ 0 & 0 & \ddot{\boldsymbol{\eta}}_{1d,3} \end{bmatrix}\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\end{aligned} \tag{F-44}$$

$$\mathbf{Y}_{2,3,3} = \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{M}_1}{\partial \dot{\boldsymbol{\eta}}_{1d}}\right)\begin{bmatrix} \ddot{\boldsymbol{\eta}}_{1d,1} & 0 & 0 \\ 0 & \ddot{\boldsymbol{\eta}}_{1d,2} & 0 \\ 0 & 0 & \ddot{\boldsymbol{\eta}}_{1d,3} \end{bmatrix} = -\mathbf{K}_2\sigma''(\boldsymbol{\nu}_{1e})\begin{bmatrix} \ddot{\boldsymbol{\eta}}_{1d,1} & 0 & 0 \\ 0 & \ddot{\boldsymbol{\eta}}_{1d,2} & 0 \\ 0 & 0 & \ddot{\boldsymbol{\eta}}_{1d,3} \end{bmatrix} \tag{F-45}$$

M2

$$\mathbf{M}_2 = -\gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \sigma(\boldsymbol{\eta}_{1e}) \frac{2 + \|\boldsymbol{\alpha}_1\|^2}{2\Delta_1(\hat{\boldsymbol{v}}_1)}, \tag{F-46}$$

$$\frac{\partial \mathbf{M}_2}{\partial \hat{\boldsymbol{v}}_1} \dot{\hat{\boldsymbol{v}}}_1 = \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \sigma(\boldsymbol{\eta}_{1e}) \left(2 + \|\boldsymbol{\alpha}_1\|^2 - \frac{2\boldsymbol{\alpha}_1^{\mathrm{T}} \mathbf{K}_1 \sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right) \frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{2\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^2} \dot{\hat{\boldsymbol{v}}}_1, \tag{F-47}$$

$$\begin{aligned} \mathbf{Y}_{3,1,1} &= \frac{\partial}{\partial \hat{\boldsymbol{v}}_1} \left(\frac{\partial \mathbf{M}_2}{\partial \hat{\boldsymbol{v}}_1}\right) \begin{bmatrix} d\hat{\boldsymbol{v}}_{1,1} & 0 & 0 \\ 0 & d\hat{\boldsymbol{v}}_{1,2} & 0 \\ 0 & 0 & d\hat{\boldsymbol{v}}_{1,3} \end{bmatrix} \\ &= \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \sigma(\boldsymbol{\eta}_{1e}) \left(\boldsymbol{\alpha}_1^{\mathrm{T}} \mathbf{K}_1 \sigma(\boldsymbol{\eta}_{1e}) - \|\mathbf{K}_1 \sigma(\boldsymbol{\eta}_{1e})\|^2 + \boldsymbol{\alpha}_1^{\mathrm{T}} \mathbf{K}_1 \sigma(\boldsymbol{\eta}_{1e})\right) \frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}} d\hat{\boldsymbol{v}}_1 \hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^4} \\ &\quad \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \sigma(\boldsymbol{\eta}_{1e}) \left(2 + \|\boldsymbol{\alpha}_1\|^2 - \frac{2\boldsymbol{\alpha}_1^{\mathrm{T}} \mathbf{K}_1 \sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right) \frac{d\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{2\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^2} \left(\mathbf{I}_{3x3} - 2\frac{\hat{\boldsymbol{v}}_1 \hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right) \end{aligned} \tag{F-48}$$

$$\begin{aligned} \mathbf{Y}_{3,1,2} &= \frac{\partial}{\partial \boldsymbol{\eta}_{1e}} \left(\frac{\partial \mathbf{M}_2}{\partial \hat{\boldsymbol{v}}_1}\right) \begin{bmatrix} d\hat{\boldsymbol{v}}_{1,1} & 0 & 0 \\ 0 & d\hat{\boldsymbol{v}}_{1,2} & 0 \\ 0 & 0 & d\hat{\boldsymbol{v}}_{1,3} \end{bmatrix} \\ &= \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \left(\frac{\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} (\dot{\boldsymbol{\eta}}_{1d}^{\mathrm{T}} - 3\boldsymbol{\alpha}_1^{\mathrm{T}}) \mathbf{K}_1 + 1 + \frac{\|\boldsymbol{\alpha}_1\|^2}{2} - \frac{\boldsymbol{\alpha}_1^{\mathrm{T}} \mathbf{K}_1 \sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right) \frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}} d\hat{\boldsymbol{v}}_1 \sigma'(\boldsymbol{\eta}_{1e})}{\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^2} \end{aligned} \tag{F-49}$$

$$\begin{aligned} \mathbf{Y}_{3,1,3} &= \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}} \left(\frac{\partial \mathbf{M}_2}{\partial \hat{\boldsymbol{v}}_1}\right) \begin{bmatrix} d\hat{\boldsymbol{v}}_{1,1} & 0 & 0 \\ 0 & d\hat{\boldsymbol{v}}_{1,2} & 0 \\ 0 & 0 & d\hat{\boldsymbol{v}}_{1,3} \end{bmatrix} \\ &= \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \sigma(\boldsymbol{\eta}_{1e}) \left(2\boldsymbol{\alpha}_1^{\mathrm{T}} - \frac{2\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e}) \mathbf{K}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right) \frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}} d\hat{\boldsymbol{v}}_1}{2\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^2} \end{aligned} \tag{F-50}$$

$$\frac{\partial \mathbf{M}_2}{\partial \boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e} = -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\left((2+\|\boldsymbol{\alpha}_1\|^2)\mathbf{I}_{3\times3} - \sigma(\boldsymbol{\eta}_{1e})\frac{2\boldsymbol{\alpha}_1^{\mathrm{T}}\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)\frac{\sigma'(\boldsymbol{\eta}_{1e})}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\dot{\boldsymbol{\eta}}_{1e}, \tag{F-51}$$

$$\begin{aligned}\mathbf{Y}_{3,2,1} &= \frac{\partial}{\partial \hat{\boldsymbol{\nu}}_1}\left(\frac{\partial \mathbf{M}_2}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3}\end{bmatrix} \\ &= \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\left((2+\|\boldsymbol{\alpha}_1\|^2)\mathbf{I}_{3\times3} - \sigma(\boldsymbol{\eta}_{1e})\frac{2\boldsymbol{\alpha}_1^{\mathrm{T}}\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)\frac{\sigma'(\boldsymbol{\nu}_{1e})d\boldsymbol{\eta}_{1e}}{2\left(\Delta_1(\hat{\boldsymbol{\nu}}_1)\right)^2}\hat{\boldsymbol{\nu}}_1^{\mathrm{T}} \\ &\quad -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\Bigg(-\boldsymbol{\alpha}_1^{\mathrm{T}}\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e} + \sigma(\boldsymbol{\eta}_{1e})\boldsymbol{\alpha}_1^{\mathrm{T}}\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e} \\ &\qquad - \sigma(\boldsymbol{\eta}_{1e})\frac{(\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e})^{\mathrm{T}}\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\Bigg)\frac{\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\left(\Delta_1(\hat{\boldsymbol{\nu}}_1)\right)^3}\end{aligned} \tag{F-52}$$

$$\begin{aligned}\mathbf{Y}_{3,2,2} &= \frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{\partial \mathbf{M}_2}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3}\end{bmatrix} \\ &= -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\left(\frac{(2+\|\boldsymbol{\alpha}_1\|^2)}{2}\mathbf{I}_{3\times3} - \sigma(\boldsymbol{\eta}_{1e})\frac{\boldsymbol{\alpha}_1^{\mathrm{T}}\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)\frac{\sigma''(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \\ &\qquad * \begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3}\end{bmatrix} \\ &\quad -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\Bigg(-\sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e}\boldsymbol{\alpha}_1^{\mathrm{T}}\mathbf{K}_1 - \boldsymbol{\alpha}_1^{\mathrm{T}}\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e}\mathbf{I}_{3\times3} \\ &\qquad + \sigma(\boldsymbol{\eta}_{1e})\frac{(\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e})^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\mathbf{K}_1\Bigg)\frac{\sigma'(\boldsymbol{\eta}_{1e})}{\left(\Delta_1(\hat{\boldsymbol{\nu}}_1)\right)^2}\end{aligned} \tag{F-53}$$

$$\begin{aligned}\mathbf{Y}_{3,2,3} &= \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{M}_2}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3}\end{bmatrix} \\ &= -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\left(\frac{\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}d\boldsymbol{\eta}_{1e}\boldsymbol{\alpha}_1^{\mathrm{T}} - \sigma(\boldsymbol{\eta}_{1e})\frac{(\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e})^{\mathrm{T}}}{\left(\Delta_1(\hat{\boldsymbol{\nu}}_1)\right)^2}\right)\end{aligned} \tag{F-54}$$

$$\frac{\partial \mathbf{M}_2}{\partial \dot{\boldsymbol{\eta}}_{1d}} \ddot{\boldsymbol{\eta}}_{1d} = -\gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \sigma(\boldsymbol{\eta}_{1e}) \frac{2\boldsymbol{\alpha}_1^{\mathrm{T}}}{2\Delta_1(\hat{\boldsymbol{v}}_1)} \ddot{\boldsymbol{\eta}}_{1d}, \tag{F-55}$$

$$\begin{aligned} \mathbf{Y}_{3,3,1} &= \frac{\partial}{\partial \hat{\boldsymbol{v}}_1} \left(\frac{\partial \mathbf{M}_2}{\partial \dot{\boldsymbol{\eta}}_{1d}} \right) \begin{bmatrix} \ddot{\boldsymbol{\eta}}_{1d,1} & 0 & 0 \\ 0 & \ddot{\boldsymbol{\eta}}_{1d,2} & 0 \\ 0 & 0 & \ddot{\boldsymbol{\eta}}_{1d,3} \end{bmatrix} \\ &= \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \sigma(\boldsymbol{\eta}_{1e}) \left(\boldsymbol{\alpha}_1^{\mathrm{T}} - \left(\mathbf{K}_1 \sigma(\boldsymbol{\eta}_{1e}) \right)^{\mathrm{T}} \right) \frac{\ddot{\boldsymbol{\eta}}_{1d} \hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\left(\Delta_1(\hat{\boldsymbol{v}}_1) \right)^2} \end{aligned} \tag{F-56}$$

$$\begin{aligned} \mathbf{Y}_{3,3,2} &= \frac{\partial}{\partial \boldsymbol{\eta}_{1e}} \left(\frac{\partial \mathbf{M}_2}{\partial \dot{\boldsymbol{\eta}}_{1d}} \right) \begin{bmatrix} \ddot{\boldsymbol{\eta}}_{1d,1} & 0 & 0 \\ 0 & \ddot{\boldsymbol{\eta}}_{1d,2} & 0 \\ 0 & 0 & \ddot{\boldsymbol{\eta}}_{1d,3} \end{bmatrix} \\ &= -\gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \left(\boldsymbol{\alpha}_1^{\mathrm{T}} \ddot{\boldsymbol{\eta}}_{1d} \mathbf{I}_{3\times 3} - \sigma(\boldsymbol{\eta}_{1e}) \frac{\ddot{\boldsymbol{\eta}}_{1d}^{\mathrm{T}} \mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)} \right) \frac{\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} \end{aligned} \tag{F-57}$$

$$\mathbf{Y}_{3,3,3} = \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}} \left(\frac{\partial \mathbf{M}_2}{\partial \dot{\boldsymbol{\eta}}_{1d}} \right) \begin{bmatrix} \ddot{\boldsymbol{\eta}}_{1d,1} & 0 & 0 \\ 0 & \ddot{\boldsymbol{\eta}}_{1d,2} & 0 \\ 0 & 0 & \ddot{\boldsymbol{\eta}}_{1d,3} \end{bmatrix} = -\gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \frac{\sigma(\boldsymbol{\eta}_{1e}) \ddot{\boldsymbol{\eta}}_{1d}^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)} \tag{F-58}$$

M3

$$\mathbf{M}_3 = \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} \sigma(\boldsymbol{\eta}_{1e}), \tag{F-59}$$

$$\frac{\partial \mathbf{M}_3}{\partial \hat{\boldsymbol{v}}_1} \dot{\hat{\boldsymbol{v}}}_1 = \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \frac{\mathbf{K}_1 \sigma(\boldsymbol{\eta}_{1e}) \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} \left(\mathbf{I}_{3x3} - 2 \frac{\hat{\boldsymbol{v}}_1 \hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)} \right) \dot{\hat{\boldsymbol{v}}}_1 \tag{F-60}$$

$$\begin{aligned} \mathbf{Y}_{\mathbf{4},1,1} &= \frac{\partial}{\partial \hat{\boldsymbol{v}}_1} \left(\frac{\partial \mathbf{M}_3}{\partial \hat{\boldsymbol{v}}_1} \right) \begin{bmatrix} d\hat{\boldsymbol{v}}_{1,1} & 0 & 0 \\ 0 & d\hat{\boldsymbol{v}}_{1,2} & 0 \\ 0 & 0 & d\hat{\boldsymbol{v}}_{1,3} \end{bmatrix} \\ &= -\gamma_1 \lambda_m(\mathbf{D}_1) \frac{\mathbf{K}_1 \sigma(\boldsymbol{\eta}_{1e}) \sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)^3} \left(\left(\mathbf{I}_{3x3} - 2 \frac{\hat{\boldsymbol{v}}_1 \hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)} \right) \left(d\hat{\boldsymbol{v}}_1 \hat{\boldsymbol{v}}_1^{\mathrm{T}} + \frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}} d\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)} \mathbf{I}_{3x3} \right) \right. \\ &\qquad \left. + \hat{\boldsymbol{v}}_1 d\hat{\boldsymbol{v}}_1^{\mathrm{T}} \right) \end{aligned} \tag{F-61}$$

$$\begin{aligned} \mathbf{Y}_{4,1,2} &= \frac{\partial}{\partial \boldsymbol{\eta}_{1e}} \left(\frac{\partial \mathbf{M}_3}{\partial \hat{\boldsymbol{v}}_1} \right) \begin{bmatrix} d\hat{\boldsymbol{v}}_{1,1} & 0 & 0 \\ 0 & d\hat{\boldsymbol{v}}_{1,2} & 0 \\ 0 & 0 & d\hat{\boldsymbol{v}}_{1,3} \end{bmatrix} \\ &== \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \left(\frac{\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} \left(\mathbf{I}_{3x3} - 2 \frac{\hat{\boldsymbol{v}}_1 \hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)} \right) d\hat{\boldsymbol{v}}_1 \mathbf{K}_1 \right. \\ &\qquad \left. + \frac{\mathbf{K}_1 \sigma(\boldsymbol{\eta}_{1e}) d\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} \left(\mathbf{I}_{3x3} - 2 \frac{\hat{\boldsymbol{v}}_1 \hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)} \right) \right) \sigma'(\boldsymbol{\eta}_{1e}) \end{aligned} \tag{F-62}$$

$$\mathbf{Y}_{4,1,3} = \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}} \left(\frac{\partial \mathbf{M}_3}{\partial \hat{\boldsymbol{v}}_1} \right) \begin{bmatrix} d\hat{\boldsymbol{v}}_{1,1} & 0 & 0 \\ 0 & d\hat{\boldsymbol{v}}_{1,2} & 0 \\ 0 & 0 & d\hat{\boldsymbol{v}}_{1,3} \end{bmatrix} = \mathbf{0}_{3x3} \tag{F-63}$$

$$\frac{\partial \mathbf{M}_3}{\partial \boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e} = \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\frac{\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\left(\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}} + \hat{\boldsymbol{v}}_1^{\mathrm{T}}\sigma(\boldsymbol{\eta}_{1e})\mathbf{I}_{3\times3}\right)\sigma'(\boldsymbol{\eta}_{1e})\dot{\boldsymbol{\eta}}_{1e} \tag{F-64}$$

$$\begin{aligned}\mathbf{Y}_{4,2,1} &= \frac{\partial}{\partial \hat{\boldsymbol{v}}_1}\left(\frac{\partial \mathbf{M}_3}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3}\end{bmatrix} \\ &= \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\frac{\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\left(\sigma(\boldsymbol{\eta}_{1e})(\sigma'(\boldsymbol{\eta}_{1e})\dot{\boldsymbol{\eta}}_{1e})^{\mathrm{T}} + \sigma'(\boldsymbol{\eta}_{1e})\dot{\boldsymbol{\eta}}_{1e}\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\right)\left(\mathbf{I}_{3x3} - 2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\end{aligned} \tag{F-65}$$

$$\begin{aligned}\mathbf{Y}_{4,2,2} &= \frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{\partial \mathbf{M}_3}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3}\end{bmatrix} \\ &= \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\frac{\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\left(\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}}\right. \\ &\qquad \left. + \hat{\boldsymbol{v}}_1^{\mathrm{T}}\sigma(\boldsymbol{\eta}_{1e})\mathbf{I}_{3\times3}\right)\sigma''(\boldsymbol{\eta}_{1e})\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3}\end{bmatrix} \\ &\quad + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\frac{\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\left(\hat{\boldsymbol{v}}_1^{\mathrm{T}}\sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e}\mathbf{I}_{3\times3} + \sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e}\hat{\boldsymbol{v}}_1^{\mathrm{T}}\right)\sigma'(\boldsymbol{\eta}_{1e})\end{aligned} \tag{F-66}$$

$$\mathbf{Y}_{4,2,3} = \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{M}_3}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} d\hat{\boldsymbol{v}}_{1,1} & 0 & 0 \\ 0 & d\hat{\boldsymbol{v}}_{1,2} & 0 \\ 0 & 0 & d\hat{\boldsymbol{v}}_{1,3}\end{bmatrix} = \mathbf{0}_{3x3} \tag{F-67}$$

$$\frac{\partial \mathbf{M}_3}{\partial \dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} = \mathbf{0}_{3x3}\ddot{\boldsymbol{\eta}}_{1d} \tag{F-68}$$

$$\mathbf{Y}_{4,3,1} = \frac{\partial}{\partial \hat{\boldsymbol{v}}_1}\left(\frac{\partial \mathbf{M}_3}{\partial \dot{\boldsymbol{\eta}}_{1d}}\right)\begin{bmatrix} \ddot{\boldsymbol{\eta}}_{1d,1} & 0 & 0 \\ 0 & \ddot{\boldsymbol{\eta}}_{1d,2} & 0 \\ 0 & 0 & \ddot{\boldsymbol{\eta}}_{1d,3}\end{bmatrix} = \mathbf{0}_{3x3} \tag{F-69}$$

$$\mathbf{Y}_{4,3,2} = \frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{\partial \mathbf{M}_3}{\partial \dot{\boldsymbol{\eta}}_{1d}}\right)\begin{bmatrix} \ddot{\boldsymbol{\eta}}_{1d,1} & 0 & 0 \\ 0 & \ddot{\boldsymbol{\eta}}_{1d,2} & 0 \\ 0 & 0 & \ddot{\boldsymbol{\eta}}_{1d,3}\end{bmatrix} = \mathbf{0}_{3x3} \tag{F-70}$$

$$\mathbf{Y}_{4,3,3} = \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{M}_3}{\partial \dot{\boldsymbol{\eta}}_{1d}}\right)\begin{bmatrix} \ddot{\boldsymbol{\eta}}_{1d,1} & 0 & 0 \\ 0 & \ddot{\boldsymbol{\eta}}_{1d,2} & 0 \\ 0 & 0 & \ddot{\boldsymbol{\eta}}_{1d,3}\end{bmatrix} = \mathbf{0}_{3x3} \tag{F-71}$$

M4

$$\mathbf{M}_4 = -\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\mathbf{D}_1\hat{\boldsymbol{v}}_1 \tag{F-72}$$

$$\frac{\partial\mathbf{M}_4}{\partial\hat{\boldsymbol{v}}_1}\dot{\hat{\boldsymbol{v}}}_1 = \left(-2\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\mathbf{D}_1 + 2\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^3}\mathbf{D}_1\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}\right)\dot{\hat{\boldsymbol{v}}}_1 \tag{F-73}$$

$$\begin{aligned}\mathbf{Y}_{5,1,1} &= \frac{\partial}{\partial\hat{\boldsymbol{v}}_1}\left(\frac{\partial\mathbf{M}_4}{\partial\hat{\boldsymbol{v}}_1}\right)\begin{bmatrix} d\hat{\boldsymbol{v}}_{1,1} & 0 & 0 \\ 0 & d\hat{\boldsymbol{v}}_{1,2} & 0 \\ 0 & 0 & d\hat{\boldsymbol{v}}_{1,3}\end{bmatrix} \\ &= \mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})\left(\left(-2\frac{d\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\mathbf{D}_1^{\mathrm{T}} + 4\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}\mathbf{D}_1 d\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^3}\hat{\boldsymbol{v}}_1^{\mathrm{T}} + 2\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}\mathbf{D}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^3}\hat{\boldsymbol{v}}_1^{\mathrm{T}} d\hat{\boldsymbol{v}}_1\right.\right. \\ &\quad \left.\left. - \left(-4\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^3}\mathbf{D}_1 + 6\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}\mathbf{D}_1\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^4}\right)d\hat{\boldsymbol{v}}_1\right)\hat{\boldsymbol{v}}_1^{\mathrm{T}}\right)\end{aligned} \tag{F-74}$$

$$\mathbf{Y}_{5,1,2}\frac{\partial}{\partial\boldsymbol{\eta}_{1e}}\left(\frac{\partial\mathbf{M}_4}{\partial\hat{\boldsymbol{v}}_1}\right)\begin{bmatrix} d\hat{\boldsymbol{v}}_{1,1} & 0 & 0 \\ 0 & d\hat{\boldsymbol{v}}_{1,2} & 0 \\ 0 & 0 & d\hat{\boldsymbol{v}}_{1,3}\end{bmatrix} = \left(\left(-2\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}\mathbf{D}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} + 2\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}\mathbf{D}_1\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^3}\right)d\hat{\boldsymbol{v}}_1\right)\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e}) \tag{F-75}$$

$$\mathbf{Y}_{5,1,3}\frac{\partial}{\partial\dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial\mathbf{M}_4}{\partial\hat{\boldsymbol{v}}_1}\right)\begin{bmatrix} d\hat{\boldsymbol{v}}_{1,1} & 0 & 0 \\ 0 & d\hat{\boldsymbol{v}}_{1,2} & 0 \\ 0 & 0 & d\hat{\boldsymbol{v}}_{1,3}\end{bmatrix} = \mathbf{0}_{3x3} \tag{F-76}$$

$$\frac{\partial\mathbf{M}_4}{\partial\boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e} = -\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}\mathbf{D}_1\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\dot{\boldsymbol{\eta}}_{1e} \tag{F-77}$$

$$\begin{aligned}\mathbf{Y}_{5,2,1} &= \frac{\partial}{\partial\hat{\boldsymbol{v}}_1}\left(\frac{\partial\mathbf{M}_4}{\partial\boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3}\end{bmatrix} \\ &= \mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e}\left(-2\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\mathbf{D}_1 + 2\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^3}\mathbf{D}_1\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}\right)\end{aligned} \tag{F-78}$$

$$\begin{aligned}\mathbf{Y}_{5,2,2} &= \frac{\partial}{\partial\boldsymbol{\eta}_{1e}}\left(\frac{\partial\mathbf{M}_4}{\partial\boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3}\end{bmatrix} \\ &= -\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}\mathbf{D}_1\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e})\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3}\end{bmatrix}\end{aligned} \tag{F-79}$$

$$\mathbf{Y}_{5,2,3} = \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}} \left(\frac{\partial \mathbf{M}_4}{\partial \boldsymbol{\eta}_{1e}} \right) \begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3} \end{bmatrix} = \mathbf{0}_{3x3} \quad \text{(F-80)}$$

M5

$$\mathbf{M}_5 = -\left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{2\boldsymbol{\alpha}_1 + \boldsymbol{\nu}_{1e}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} \sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2} \tag{F-81}$$

$$\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\nu}_{1e}} = -\left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} \mathbf{I}_{3\times3} - \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{2\boldsymbol{\alpha}_1 + \boldsymbol{\nu}_{1e}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} \frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}} \tag{F-82}$$

$$\frac{\partial \mathbf{M}_5}{\partial \hat{\boldsymbol{\nu}}_1} = \frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\nu}_{1e}} \mathbf{G}_1 + \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \left(-\frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2} + \frac{(2\boldsymbol{\alpha}_1 + \boldsymbol{\nu}_{1e})}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} \hat{\boldsymbol{\nu}}_1^{\mathrm{T}}\right) \tag{F-83}$$

$$\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\eta}_{1e}} = \frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\nu}_{1e}} \left(\mathbf{K}_1 \frac{\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right) - \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \left(-\mathbf{K}_1 \frac{\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right) \tag{F-84}$$

$$\frac{\partial \mathbf{M}_5}{\partial \dot{\boldsymbol{\eta}}_{1d}} = -\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\nu}_{1e}} - \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \mathbf{I}_{3\times3} \tag{F-85}$$

$$\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\nu}_{1e}} = -\left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \left(\frac{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} \mathbf{I}_{3\times3} + \frac{\boldsymbol{\alpha}_1 + \hat{\boldsymbol{\nu}}_1}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} \frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}\right) \tag{F-86}$$

$$\frac{\partial \mathbf{M}_5}{\partial \hat{\boldsymbol{\nu}}_1} = \frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\nu}_{1e}} \mathbf{G}_1 + \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \left(-\frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2} + \frac{\boldsymbol{\alpha}_1 + \hat{\boldsymbol{\nu}}_1}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} \hat{\boldsymbol{\nu}}_1^{\mathrm{T}}\right) \tag{F-87}$$

$$\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\nu}_{1e}} = -\left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{(\boldsymbol{\alpha}_1 + \hat{\boldsymbol{\nu}}_1)}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} \left(\frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}\right) \tag{F-88}$$

$$\frac{\partial \mathbf{M}_5}{\partial \hat{\boldsymbol{\nu}}} = \frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\nu}_{1e}} \mathbf{G}_1 - \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} \left(\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2} + \mathbf{I}_{3\times3} - \frac{2(\boldsymbol{\alpha}_1 + \hat{\boldsymbol{\nu}}_1)\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right) \tag{F-89}$$

$$\begin{aligned} \mathrm{Y}_{6,0,0} = {} & \frac{\partial}{\partial \boldsymbol{\nu}_{1e}} \left(\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\nu}_{1e}}\right) \left(\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \left(\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}^{\mathrm{T}} \mathbf{G}_1 d\hat{\boldsymbol{\nu}}_1 \right) + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \left(\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}^{\mathrm{T}} \mathbf{G}_1 d\hat{\boldsymbol{\nu}}_1 \right) \right. \\ & \left. + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \left(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^{\mathrm{T}} \mathbf{G}_1 d\hat{\boldsymbol{\nu}}_1 \right) \right) \\ = {} & -\left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{(\boldsymbol{\alpha}_1 + \hat{\boldsymbol{\nu}}_1)(\mathbf{G}_1 d\hat{\boldsymbol{\nu}}_1)^{\mathrm{T}}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}} \left(\mathbf{I}_{3\times3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1 + \|\boldsymbol{\nu}_{1e}\|^2}\right) \end{aligned} \tag{F-90}$$

$$\Upsilon_{6,0,1} = \frac{\partial}{\partial \hat{\boldsymbol{\nu}}_1}\left(\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\nu}_{1e}}\right)\mathbf{G}_1 d\hat{\boldsymbol{\nu}}_1$$
$$= \Upsilon_{3\_0\_0}\mathbf{G}_1 - \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{(\boldsymbol{\alpha}_1 + \hat{\boldsymbol{\nu}}_1)}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{G}_1 d\hat{\boldsymbol{\nu}}_1}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}$$
$$* \left(\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2} + \mathbf{I}_{3\times3} - \frac{2(\boldsymbol{\alpha}_1 + \hat{\boldsymbol{\nu}}_1)\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right) \tag{F-91}$$

$$\Upsilon_{6,0,2} = \frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\nu}_{1e}}\right)\mathbf{G}_1 d\boldsymbol{\nu}_1$$
$$= \Upsilon_{6\_0\_0}\mathbf{K}_1 \frac{\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{G}_1 d\hat{\boldsymbol{\nu}}_1}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{2\big(\Delta_1(\hat{\boldsymbol{\nu}}_1)\big)^2} \tag{F-92}$$

$$\Upsilon_{6,0,3} = \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\nu}_{1e}}\right)\mathbf{G}_1 d\hat{\boldsymbol{\nu}}_1 = -\Upsilon_{6\_0\_0} - \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{1}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{G}_1 d\hat{\boldsymbol{\nu}}_1}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}\mathbf{I}_{3\times3} \tag{F-93}$$

$$\Upsilon_{6,1,1} = \frac{\partial}{\partial \hat{\boldsymbol{\nu}}_1}\left(\frac{\partial \mathbf{M}_5}{\partial \hat{\boldsymbol{\nu}}_1}\right)\begin{bmatrix} d\hat{\boldsymbol{\nu}}_{1,1} & 0 & 0 \\ 0 & d\hat{\boldsymbol{\nu}}_{1,2} & 0 \\ 0 & 0 & d\hat{\boldsymbol{\nu}}_{1,3}\end{bmatrix}$$
$$= \Upsilon_{6,0,1} - \frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\nu}_{1e}}\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})d\boldsymbol{\nu}_1^{\mathrm{T}}}{\big(\Delta_1(\hat{\boldsymbol{\nu}}_1)\big)^2}\left(\mathbf{I}_{3x3} - 2\frac{\hat{\boldsymbol{\nu}}_1\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)$$
$$+ \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\left(\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2} + \mathbf{I}_{3\times3} - \frac{2(\boldsymbol{\alpha}_1 + \hat{\boldsymbol{\nu}}_1)\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)d\hat{\boldsymbol{\nu}}_1\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}$$
$$- \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\left(\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})d\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\left(\mathbf{I}_{3x3} - 2\frac{\hat{\boldsymbol{\nu}}_1\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)\right.$$
$$\left. - \frac{2(\boldsymbol{\alpha}_1 + \hat{\boldsymbol{\nu}}_1)d\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\left(\mathbf{I}_{3x3} - 2\frac{\hat{\boldsymbol{\nu}}_1\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right) - \frac{2\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}d\hat{\boldsymbol{\nu}}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\left(\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2} + \mathbf{I}_{3\times3}\right)\right)$$
$$- \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{1}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}\left(\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2} + \mathbf{I}_{3\times3}\right.$$
$$\left. - \frac{2(\boldsymbol{\alpha}_1 + \hat{\boldsymbol{\nu}}_1)\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)d\hat{\boldsymbol{\nu}}_1\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{G}_1 \tag{F-94}$$

$$\Upsilon_{6,1,2} = \frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{\partial \mathbf{M}_5}{\partial \hat{\boldsymbol{\nu}}_1}\right)\begin{bmatrix} d\hat{\boldsymbol{\nu}}_{1,1} & 0 & 0 \\ 0 & d\hat{\boldsymbol{\nu}}_{1,2} & 0 \\ 0 & 0 & d\hat{\boldsymbol{\nu}}_{1,3}\end{bmatrix}$$
$$= \Upsilon_{6,0,2} - \frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\nu}_{1e}}\mathbf{K}_1 \frac{\sigma'(\boldsymbol{\eta}_{1e})d\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}\hat{\boldsymbol{\nu}}_1}{\big(\Delta_1(\hat{\boldsymbol{\nu}}_1)\big)^2}\sigma'(\boldsymbol{\eta}_{1e})$$
$$- \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\left(\mathbf{K}_1 \frac{\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}d\hat{\boldsymbol{\nu}}_{1_1}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\sigma'(\boldsymbol{\eta}_{1e}) - \frac{2\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}d\hat{\boldsymbol{\nu}}_{1_1}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\left(-\mathbf{K}_1 \frac{\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)\right)$$
$$- \frac{2c + \gamma_1\lambda_m(\mathbf{D}_1)}{4\Delta_1(\hat{\boldsymbol{\nu}}_1)\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}\left(\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2} + \mathbf{I}_{3\times3} - \frac{2(\boldsymbol{\alpha}_1 + \hat{\boldsymbol{\nu}}_1)\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)d\hat{\boldsymbol{\nu}}_1\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{K}_1 \frac{\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \tag{F-95}$$

$$\mathrm{Y}_{6,1,3} = \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{M}_5}{\partial \hat{\boldsymbol{v}}_1}\right)\begin{bmatrix} d\boldsymbol{v}_{1,1} & 0 & 0 \\ 0 & d\boldsymbol{v}_{1,2} & 0 \\ 0 & 0 & d\boldsymbol{v}_{1,3} \end{bmatrix}$$
$$= \mathrm{Y}_{6,0,3} - \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}{2\Delta_1(\hat{\boldsymbol{v}}_1)}\left(-\frac{2\hat{\boldsymbol{v}}_1^{\mathrm{T}} d\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\mathbf{I}_{3\times 3}$$
$$+ \frac{2c + \gamma_1 \lambda_m(\mathbf{D}_1)}{4\Delta_1(\hat{\boldsymbol{v}}_1)\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\left(\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} + \mathbf{I}_{3\times 3} - \frac{2(\boldsymbol{\alpha}_1 + \hat{\boldsymbol{v}}_1)\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right) d\hat{\boldsymbol{v}}_1 \boldsymbol{v}_{1e}^{\mathrm{T}} \quad \text{(F-96)}$$

$$\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\eta}_{1e}} = -\left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right)\left(\frac{(\boldsymbol{\alpha}_1 + \hat{\boldsymbol{v}}_1)\boldsymbol{v}_{1e}^{\mathrm{T}}}{2\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}} - \sqrt{1+\|\boldsymbol{v}_{1e}\|^2}\mathbf{I}_{3\times 3}\right)\frac{\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})}{\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^2} \quad \text{(F-97)}$$

$$\mathrm{Y}_{6,2,0} = \frac{\partial}{\partial \boldsymbol{v}_{1e}}\left(\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3} \end{bmatrix}$$
$$= -\left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right)\left(\frac{(\boldsymbol{\alpha}_1 + \hat{\boldsymbol{v}}_1)(\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e}) d\boldsymbol{\eta}_{1e})^{\mathrm{T}}}{2\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^2\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\left(\mathbf{I}_{3\times 3} - \frac{\boldsymbol{v}_{1e}\boldsymbol{v}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{v}_{1e}\|^2}\right)\right.$$
$$\left. - \frac{\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e}) d\boldsymbol{\eta}_{1e} \boldsymbol{v}_{1e}^{\mathrm{T}}}{\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^2\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\right) \quad \text{(F-98)}$$

$$\mathrm{Y}_{6,2,1} = \frac{\partial}{\partial \hat{\boldsymbol{v}}_1}\left(\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3} \end{bmatrix}$$
$$= \mathrm{Y}_{6,2,0}\mathbf{G}_1 - \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right)\left(\frac{\boldsymbol{v}_{1e}^{\mathrm{T}} \mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e}) d\boldsymbol{\eta}_{1e}}{2\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^2\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\right)$$
$$* \left(\mathbf{K}_1 \frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} + \mathbf{I}_{3\times 3} - \frac{2(\boldsymbol{\alpha}_1 + \hat{\boldsymbol{v}}_1)\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right) - 2c\frac{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}{\Delta_1(\hat{\boldsymbol{v}}_1)^3}\left(\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})\right)\dot{\boldsymbol{\eta}}_{1e}\hat{\boldsymbol{v}}_1^{\mathrm{T}} \quad \text{(F-99)}$$

$$\mathrm{Y}_{6,2,2}=\frac{\partial}{\partial\boldsymbol{\eta}_{1e}}\left(\frac{\partial\mathbf{M}_5}{\partial\boldsymbol{\eta}_{1e}}\right)\begin{bmatrix}d\boldsymbol{\eta}_{1e,1}&0&0\\0&d\boldsymbol{\eta}_{1e,2}&0\\0&0&d\boldsymbol{\eta}_{1e,3}\end{bmatrix}$$
$$=\mathrm{Y}_{6,2,0}\mathbf{K}_1\frac{\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}-\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\left(\frac{(\boldsymbol{\alpha}_1+\hat{\boldsymbol{\nu}}_1)\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{2(1+\|\boldsymbol{\nu}_{1e}\|^2)}-\mathbf{I}_{3\times3}\right)$$
$$*\mathbf{K}_1\frac{\sigma''(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\begin{bmatrix}d\boldsymbol{\eta}_{1e,1}&0&0\\0&d\boldsymbol{\eta}_{1e,2}&0\\0&0&d\boldsymbol{\eta}_{1e,3}\end{bmatrix}+\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)\left(\frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)$$
$$*\mathbf{K}_1\frac{\sigma'(\boldsymbol{\eta}_{1e})}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)^3} \tag{F-100}$$

$$\mathrm{Y}_{6,2,3}=\frac{\partial}{\partial\dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial\mathbf{M}_5}{\partial\boldsymbol{\eta}_{1e}}\right)\begin{bmatrix}d\boldsymbol{\eta}_{1e,1}&0&0\\0&d\boldsymbol{\eta}_{1e,2}&0\\0&0&d\boldsymbol{\eta}_{1e,3}\end{bmatrix}$$
$$=-\mathrm{Y}_{6\_2\_0}-\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{1}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}}\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})d\boldsymbol{\eta}_{1e}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\mathbf{I}_{3\times3} \tag{F-101}$$

$$\frac{\partial\mathbf{M}_5}{\partial\dot{\boldsymbol{\eta}}_{1d}}=\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{(\boldsymbol{\alpha}_1+\hat{\boldsymbol{\nu}}_1)}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\left(\frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)-\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\mathbf{I}_{3\times3} \tag{F-102}$$

$$\mathrm{Y}_{6,3,0}=\frac{\partial}{\partial\boldsymbol{\nu}_{1e}}\left(\frac{\partial\mathbf{M}_5}{\partial\dot{\boldsymbol{\eta}}_{1d}}\right)\begin{bmatrix}\ddot{\boldsymbol{\eta}}_{1d,1}&0&0\\0&\ddot{\boldsymbol{\eta}}_{1d,2}&0\\0&0&\ddot{\boldsymbol{\eta}}_{1d,3}\end{bmatrix}$$
$$=\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{(\boldsymbol{\alpha}_1+\hat{\boldsymbol{\nu}}_1)}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\frac{\ddot{\boldsymbol{\eta}}_{1d}^{\mathrm{T}}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\left(\mathbf{I}_{3\times3}-\frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)$$
$$-\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{\ddot{\boldsymbol{\eta}}_{1d}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \tag{F-103}$$

$$\mathrm{Y}_{6,3,1}=\frac{\partial}{\partial\hat{\boldsymbol{\nu}}_1}\left(\frac{\partial\mathbf{M}_2}{\partial\dot{\boldsymbol{\eta}}_{1d}}\right)\begin{bmatrix}\ddot{\boldsymbol{\eta}}_{1d,1}&0&0\\0&\ddot{\boldsymbol{\eta}}_{1d,2}&0\\0&0&\ddot{\boldsymbol{\eta}}_{1d,3}\end{bmatrix}$$
$$=\mathrm{Y}_{6,3,0}\mathbf{G}_1+\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{1}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)}\left(\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}+\mathbf{I}_{3\times3}-\frac{2(\boldsymbol{\alpha}_1+\hat{\boldsymbol{\nu}}_1)\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)$$
$$*\left(\frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}}\ddot{\boldsymbol{\eta}}_{1d}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right)+\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\ddot{\boldsymbol{\eta}}_{1d}\hat{\boldsymbol{\nu}}_1^{\mathrm{T}} \tag{F-104}$$

$$\mathrm{Y}_{6,3,2}=\frac{\partial}{\partial\boldsymbol{\eta}_{1e}}\left(\frac{\partial\mathbf{M}_5}{\partial\dot{\boldsymbol{\eta}}_{1d}}\right)\begin{bmatrix}\ddot{\boldsymbol{\eta}}_{1d,1}&0&0\\0&\ddot{\boldsymbol{\eta}}_{1d,2}&0\\0&0&\ddot{\boldsymbol{\eta}}_{1d,3}\end{bmatrix}$$
$$=\mathrm{Y}_{6,3,0}\mathbf{K}_1\frac{\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}+\left(c+\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{(-\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e}))}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\left(\frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}}\ddot{\boldsymbol{\eta}}_{1d}}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\right) \tag{F-105}$$

$$\mathrm{Y}_{6,3,3} = \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{M}_5}{\partial \dot{\boldsymbol{\eta}}_{1d}}\right)\begin{bmatrix} \ddot{\boldsymbol{\eta}}_{1d,1} & 0 & 0 \\ 0 & \ddot{\boldsymbol{\eta}}_{1d,2} & 0 \\ 0 & 0 & \ddot{\boldsymbol{\eta}}_{1d,3} \end{bmatrix}$$
$$= -\mathrm{Y}_{6,3,0} + \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right)\frac{1}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\left(\frac{\boldsymbol{\nu}_{1e}^{\mathrm{T}}\ddot{\boldsymbol{\eta}}_{1d}}{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}\right)\mathbf{I}_{3\times 3} \quad \text{(F-106)}$$

M6

$$\mathbf{M}_6 = \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} + \ddot{\boldsymbol{\eta}}_{1d}, \tag{F-107}$$

$$\frac{\partial \mathbf{M}_6}{\partial \hat{\boldsymbol{v}}_1}\dot{\hat{\boldsymbol{v}}}_1 = -2\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^3}\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}} d\hat{\boldsymbol{v}}_1, \tag{F-108}$$

$$\begin{aligned} \mathrm{Y}_{7,1,1} &= \frac{\partial}{\partial \hat{\boldsymbol{v}}_1}\left(\frac{\partial \mathbf{M}_6}{\partial \boldsymbol{v}_1}\right)\begin{bmatrix} d\hat{\boldsymbol{v}}_{1,1} & 0 & 0 \\ 0 & d\hat{\boldsymbol{v}}_{1,2} & 0 \\ 0 & 0 & d\hat{\boldsymbol{v}}_{1,3} \end{bmatrix} \\ &= -2\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^3}\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e}) d\hat{\boldsymbol{v}}_1^{\mathrm{T}}\left(\mathbf{I}_{3\times 3} - 3\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right) \end{aligned} \tag{F-109}$$

$$\begin{aligned} \mathrm{Y}_{7,1,2} &= \frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{\partial \mathbf{M}_6}{\partial \boldsymbol{v}_1}\right)\begin{bmatrix} d\hat{\boldsymbol{v}}_{1,1} & 0 & 0 \\ 0 & d\hat{\boldsymbol{v}}_{1,2} & 0 \\ 0 & 0 & d\hat{\boldsymbol{v}}_{1,3} \end{bmatrix} \\ &= -2\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^3}\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}} d\hat{\boldsymbol{v}}_1 \\ &\quad -2\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}} d\hat{\boldsymbol{v}}_1}{\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^3}\mathbf{K}_1\mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e})\begin{bmatrix} \sigma(\boldsymbol{\eta}_{1e,1}) & 0 & 0 \\ 0 & \sigma(\boldsymbol{\eta}_{1e,2}) & 0 \\ 0 & 0 & \sigma(\boldsymbol{\eta}_{1e,3}) \end{bmatrix} \end{aligned} \tag{F-110}$$

$$\begin{aligned} \mathrm{Y}_{7,1,3} &= \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{M}_6}{\partial \boldsymbol{v}_1}\right)\begin{bmatrix} d\boldsymbol{v}_{1,1} & 0 & 0 \\ 0 & d\boldsymbol{v}_{1,2} & 0 \\ 0 & 0 & d\boldsymbol{v}_{1,3} \end{bmatrix} \\ &= \mathbf{0}_{3\times 3} \end{aligned} \tag{F-111}$$

$$\frac{\partial \mathbf{M}_6}{\partial \boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e} = \left(\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} + \frac{\mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e})}{\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^2}\mathbf{K}_1\begin{bmatrix} \sigma(\boldsymbol{\eta}_{1e,1}) & 0 & 0 \\ 0 & \sigma(\boldsymbol{\eta}_{1e,2}) & 0 \\ 0 & 0 & \sigma(\boldsymbol{\eta}_{1e,3}) \end{bmatrix}\right)\dot{\boldsymbol{\eta}}_{1e}, \tag{F-112}$$

$$\begin{aligned} \mathrm{Y}_{7,2,1} &= \frac{\partial}{\partial \hat{\boldsymbol{v}}_1}\left(\frac{\partial \mathbf{M}_6}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3} \end{bmatrix} \\ &= -\mathbf{K}_1\left(\sigma'(\boldsymbol{\eta}_{1e})\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e}) + \sigma''(\boldsymbol{\eta}_{1e})\mathbf{K}_1\begin{bmatrix} \sigma(\boldsymbol{\eta}_{1e,1}) & 0 & 0 \\ 0 & \sigma(\boldsymbol{\eta}_{1e,2}) & 0 \\ 0 & 0 & \sigma(\boldsymbol{\eta}_{1e,3}) \end{bmatrix}\right)\frac{d\boldsymbol{\eta}_{1e}\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^3} \end{aligned} \tag{F-113}$$

$$\mathrm{Y}_{7,2,2} = \frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{\partial \mathbf{M}_6}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3} \end{bmatrix}$$
$$= \left(3\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\frac{\mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e})}{\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^2} + \frac{\mathbf{K}_1\sigma'''(\boldsymbol{\eta}_{1e})}{\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^2}\mathbf{K}_1\begin{bmatrix} \sigma(\boldsymbol{\eta}_{1e,1}) & 0 & 0 \\ 0 & \sigma(\boldsymbol{\eta}_{1e,2}) & 0 \\ 0 & 0 & \sigma(\boldsymbol{\eta}_{1e,3}) \end{bmatrix}\right) * \begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3} \end{bmatrix} \quad \text{(F-114)}$$

$$\mathrm{Y}_{7,2,3} = \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{M}_6}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3} \end{bmatrix} = \mathbf{0}_{3x3} \quad \text{(F-115)}$$

$$\frac{\partial \mathbf{M}_6}{\partial \dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} = 0\mathbf{I}_{3x3}\ddot{\boldsymbol{\eta}}_{1d} \quad \text{(F-116)}$$

$$\mathbf{Y}_{7,3,1} = \frac{\partial}{\partial \hat{\boldsymbol{v}}_1}\left(\frac{\partial \mathbf{M}_6}{\partial \ddot{\boldsymbol{\eta}}_{1d}}\right)\begin{bmatrix} \ddot{\boldsymbol{\eta}}_{1d,1} & 0 & 0 \\ 0 & \ddot{\boldsymbol{\eta}}_{1d,2} & 0 \\ 0 & 0 & \ddot{\boldsymbol{\eta}}_{1d,3} \end{bmatrix} = \mathbf{0}_{3x3} \quad \text{(F-117)}$$

$$\mathbf{Y}_{7,3,2} = \frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{\partial \mathbf{M}_6}{\partial \ddot{\boldsymbol{\eta}}_{1d}}\right)\begin{bmatrix} \ddot{\boldsymbol{\eta}}_{1d,1} & 0 & 0 \\ 0 & \ddot{\boldsymbol{\eta}}_{1d,2} & 0 \\ 0 & 0 & \ddot{\boldsymbol{\eta}}_{1d,3} \end{bmatrix} = \mathbf{0}_{3x3} \quad \text{(F-118)}$$

$$\mathbf{Y}_{7,3,3} = \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{M}_6}{\partial \ddot{\boldsymbol{\eta}}_{1d}}\right)\begin{bmatrix} \ddot{\boldsymbol{\eta}}_{1d,1} & 0 & 0 \\ 0 & \ddot{\boldsymbol{\eta}}_{1d,2} & 0 \\ 0 & 0 & \ddot{\boldsymbol{\eta}}_{1d,3} \end{bmatrix} = \mathbf{0}_{3x3} \quad \text{(F-119)}$$

$$\frac{\partial \mathbf{M}_6}{\partial \ddot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} = \mathbf{I}_{3\times 3}\ddot{\boldsymbol{\eta}}_{1d}. \quad \text{(F-120)}$$

M7

$$\mathbf{M}_7 = m\left(g\boldsymbol{e}_3 - \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\frac{2\gamma_1\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\right). \tag{F-121}$$

$$\frac{\partial\mathbf{M}_7}{\partial\boldsymbol{v}_{1e}}\dot{\boldsymbol{v}}_{1e} = -m\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\frac{2\gamma_1}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{v}_{1e}\boldsymbol{v}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{v}_{1e}\|^2}\right), \tag{F-122}$$

$$\frac{\partial\mathbf{M}_7}{\partial\hat{\boldsymbol{v}}_1}\dot{\hat{\boldsymbol{v}}}_1 = \left(\frac{\partial\mathbf{M}_7}{\partial\boldsymbol{v}_{1e}}\mathbf{G}_1 + m\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\left(\frac{2\gamma_1\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\right)\dot{\hat{\boldsymbol{v}}}_1, \tag{F-123}$$

$$\begin{aligned}
\mathrm{Y}_{8,1,0} &= \frac{\partial}{\partial\boldsymbol{v}_{1e}}\left(\frac{\partial\mathbf{M}_7}{\partial\boldsymbol{v}_1}\right)\begin{bmatrix} d\hat{\boldsymbol{v}}_{1,1} & 0 & 0 \\ 0 & d\hat{\boldsymbol{v}}_{1,2} & 0 \\ 0 & 0 & d\hat{\boldsymbol{v}}_{1,3}\end{bmatrix} \\
&= m\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\frac{\gamma_1}{(1+\|\boldsymbol{v}_{1e}\|^2)^{1.5}}\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{v}_{1e}\boldsymbol{v}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{v}_{1e}\|^2}\right)\mathbf{G}_1 d\hat{\boldsymbol{v}}_1\boldsymbol{v}_{1e}^{\mathrm{T}} \\
&\quad - m\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\frac{2\gamma_1}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{v}_{1e}(\mathbf{G}_1 d\hat{\boldsymbol{v}}_1)^{\mathrm{T}}}{1+\|\boldsymbol{v}_{1e}\|^2}\right. \\
&\qquad \left. - \frac{\boldsymbol{v}_{1e}^{\mathrm{T}}\mathbf{G}_1 d\hat{\boldsymbol{v}}_1}{1+\|\boldsymbol{v}_{1e}\|^2}\left(\mathbf{I}_{3x3} - 2\frac{\boldsymbol{v}_{1e}\boldsymbol{v}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{v}_{1e}\|^2}\right)\right) \\
&\quad + 2\gamma_1 m\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}} d\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)}\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{v}_{1e}\boldsymbol{v}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{v}_{1e}\|^2}\right)
\end{aligned} \tag{F-124}$$

$$\begin{aligned}
\mathrm{Y}_{8,1,1} &= \frac{\partial}{\partial\hat{\boldsymbol{v}}_1}\left(\frac{\partial\mathbf{M}_7}{\partial\hat{\boldsymbol{v}}_1}\right)\begin{bmatrix} d\hat{\boldsymbol{v}}_{1,1} & 0 & 0 \\ 0 & d\hat{\boldsymbol{v}}_{1,2} & 0 \\ 0 & 0 & d\hat{\boldsymbol{v}}_{1,3}\end{bmatrix} \\
&= \mathrm{Y}_{8,1,0}\mathbf{G}_1 \\
&\qquad + \left(-\frac{\partial\mathbf{M}_7}{\partial\boldsymbol{v}_{1e}}\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})d\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^3}\left(\mathbf{I}_{3\times3} - 3\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\right. \\
&\qquad \left. + m\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\left(\frac{2\gamma_1\boldsymbol{v}_{1e}}{\sqrt{1+\|\boldsymbol{v}_{1e}\|^2}}\frac{d\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^2}\left(\mathbf{I}_{3\times3} - 2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\right)\right)
\end{aligned} \tag{F-125}$$

$$\mathrm{Y}_{8,1,2} = \frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{\partial \mathbf{M}_7}{\partial \boldsymbol{\nu}_1}\right)\begin{bmatrix} d\hat{\boldsymbol{\nu}}_{1,1} & 0 & 0 \\ 0 & d\hat{\boldsymbol{\nu}}_{1,2} & 0 \\ 0 & 0 & d\hat{\boldsymbol{\nu}}_{1,3} \end{bmatrix}$$
$$= \mathrm{Y}_{8,1,0}\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} - \frac{2\gamma_1 m}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\frac{\mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}$$
$$* \left(\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}\left(\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}^{\mathrm{T}}\left(\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\mathbf{G}_1 d\hat{\boldsymbol{\nu}}_1 - \left(\boldsymbol{\nu}_{1e}\frac{\hat{\boldsymbol{\nu}}_1^{\mathrm{T}} d\hat{\boldsymbol{\nu}}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)\right)\right)\right.$$
$$+ \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}\left(\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}^{\mathrm{T}}\left(\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\mathbf{G}_1 d\hat{\boldsymbol{\nu}}_1 - \left(\boldsymbol{\nu}_{1e}\frac{\hat{\boldsymbol{\nu}}_1^{\mathrm{T}} d\hat{\boldsymbol{\nu}}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)\right)\right)$$
$$\left. + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} * \left(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^{\mathrm{T}}\left(\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\mathbf{G}_1 d\hat{\boldsymbol{\nu}}_1 - \left(\boldsymbol{\nu}_{1e}\frac{\hat{\boldsymbol{\nu}}_1^{\mathrm{T}} d\hat{\boldsymbol{\nu}}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right)\right)\right)\right) \tag{F-126}$$

$$\mathrm{Y}_{7,1,3} = \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{M}_7}{\partial \boldsymbol{\nu}_1}\right)\begin{bmatrix} d\boldsymbol{\nu}_{1,1} & 0 & 0 \\ 0 & d\boldsymbol{\nu}_{1,2} & 0 \\ 0 & 0 & d\boldsymbol{\nu}_{1,3} \end{bmatrix}$$
$$= -\mathrm{Y}_{8,1,0} \tag{F-127}$$

$$\frac{\partial \mathbf{M}_7}{\partial \boldsymbol{\eta}_{1e}} d\boldsymbol{\eta}_{1e} = \left(-m\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\frac{2\gamma_1}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right.$$
$$\left. - m\frac{2\gamma_1}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\frac{\mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\begin{bmatrix} \boldsymbol{\nu}_{1e,1} & 0 & 0 \\ 0 & \boldsymbol{\nu}_{1e,2} & 0 \\ 0 & 0 & \boldsymbol{\nu}_{1e,3} \end{bmatrix}\right) d\boldsymbol{\eta}_{1e}, \tag{F-128}$$

$$\frac{\partial \mathbf{M}_7}{\partial \boldsymbol{\eta}_{1e}} \dot{\boldsymbol{\eta}}_{1e} = -m\frac{2\gamma_1\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\left(\sigma'(\boldsymbol{\eta}_{1e})\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right.$$
$$\left. + \sigma''(\boldsymbol{\eta}_{1e})\begin{bmatrix} \boldsymbol{\nu}_{1e,1} & 0 & 0 \\ 0 & \boldsymbol{\nu}_{1e,2} & 0 \\ 0 & 0 & \boldsymbol{\nu}_{1e,3} \end{bmatrix}\right) d\boldsymbol{\eta}_{1e}, \tag{F-129}$$

$$Y_{8,2,0} = \frac{\partial}{\partial \boldsymbol{\nu}_{1e}}\left(\frac{\partial \mathbf{M}_7}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3} \end{bmatrix}$$

$$= m\frac{2\gamma_1\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)(1+\|\boldsymbol{\nu}_{1e}\|^2)^{1.5}}\left(\sigma'(\boldsymbol{\eta}_{1e})\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \frac{\sigma''(\boldsymbol{\eta}_{1e})}{2}\begin{bmatrix} \boldsymbol{\nu}_{1e,1} & 0 & 0 \\ 0 & \boldsymbol{\nu}_{1e,2} & 0 \\ 0 & 0 & \boldsymbol{\nu}_{1e,3} \end{bmatrix}\right) d\boldsymbol{\eta}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}$$
$$+m\frac{2\gamma_1\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)(1+\|\boldsymbol{\nu}_{1e}\|^2)^{1.5}}\left(\sigma'(\boldsymbol{\eta}_{1e})\left(\boldsymbol{\nu}_{1e}\left(\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}d\boldsymbol{\eta}_{1e}\right)^{\mathrm{T}} + \left(\boldsymbol{\nu}_{1e}^{\mathrm{T}}\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}d\boldsymbol{\eta}_{1e}\right)\left(\mathbf{I}_{3x3} - 2\frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\right) - (1+\|\boldsymbol{\nu}_{1e}\|^2)\sigma''(\boldsymbol{\eta}_{1e})\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3} \end{bmatrix}\right), \tag{F-130}$$

$$Y_{8,2,1} = \frac{\partial}{\partial \hat{\boldsymbol{\nu}}_1}\left(\frac{\partial \mathbf{M}_7}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3} \end{bmatrix}$$
$$= Y_{8,2,0}\mathbf{G}_1 + m\frac{2\gamma_1\mathbf{K}_1}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\left(\sigma'(\boldsymbol{\eta}_{1e})\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \sigma'(\boldsymbol{\eta}_{1e})\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e}) + \sigma''(\boldsymbol{\eta}_{1e})\begin{bmatrix} \boldsymbol{\nu}_{1e,1} & 0 & 0 \\ 0 & \boldsymbol{\nu}_{1e,2} & 0 \\ 0 & 0 & \boldsymbol{\nu}_{1e,3} \end{bmatrix}\right)\frac{d\boldsymbol{\eta}_{1e}\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\left(\Delta_1(\hat{\boldsymbol{\nu}}_1)\right)^2} \tag{F-131}$$

$$\frac{\partial \mathbf{M}_7}{\partial \boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e} = -m\frac{2\gamma_1\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\left(\sigma'(\boldsymbol{\eta}_{1e})\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \sigma''(\boldsymbol{\eta}_{1e})\begin{bmatrix} \boldsymbol{\nu}_{1e,1} & 0 & 0 \\ 0 & \boldsymbol{\nu}_{1e,2} & 0 \\ 0 & 0 & \boldsymbol{\nu}_{1e,3} \end{bmatrix}\right) d\boldsymbol{\eta}_{1e}, \tag{F-132}$$

$$\begin{aligned}\mathrm{Y}_{8,2,2} &= \frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{\partial \mathbf{M}_7}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3}\end{bmatrix} \\ &= \mathrm{Y}_{8,2,0}\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} - m\frac{2\gamma_1\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \\ &\quad * \left(\sigma'(\boldsymbol{\eta}_{1e})\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\frac{\mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \sigma'''(\boldsymbol{\eta}_{1e})\begin{bmatrix} \boldsymbol{\nu}_{1e,1} & 0 & 0 \\ 0 & \boldsymbol{\nu}_{1e,2} & 0 \\ 0 & 0 & \boldsymbol{\nu}_{1e,3}\end{bmatrix}\right) \\ &\quad * \begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3}\end{bmatrix} + m\frac{2\gamma_1\mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)(1+\|\boldsymbol{\nu}_{1e}\|^2)^{1.5}}\left(\boldsymbol{\nu}_{1e}^{\mathrm{T}}\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}d\boldsymbol{\eta}_{1e}\right) \\ &\quad * \begin{bmatrix} \boldsymbol{\nu}_{1e,1} & 0 & 0 \\ 0 & \boldsymbol{\nu}_{1e,2} & 0 \\ 0 & 0 & \boldsymbol{\nu}_{1e,3}\end{bmatrix}\end{aligned} \tag{F-133}$$

$$\begin{aligned}\mathrm{Y}_{8,2,3} &= \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{M}_7}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} d\boldsymbol{\eta}_{1e,1} & 0 & 0 \\ 0 & d\boldsymbol{\eta}_{1e,2} & 0 \\ 0 & 0 & d\boldsymbol{\eta}_{1e,3}\end{bmatrix} \\ &= -\mathrm{Y}_{8,2,0}\end{aligned} \tag{F-134}$$

$$\frac{\partial \mathbf{M}_7}{\partial \dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} = -\frac{\partial \mathbf{M}_7}{\partial \boldsymbol{\nu}_{1e}}\ddot{\boldsymbol{\eta}}_{1d}. \tag{F-135}$$

$$\frac{\partial \mathbf{M}_7}{\partial \dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} = m\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\frac{2\gamma_1}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\ddot{\boldsymbol{\eta}}_{1d}. \tag{F-136}$$

$$\begin{aligned}\mathrm{Y}_{8,3,0} &= \frac{\partial}{\partial \boldsymbol{\nu}_{1e}}\left(\frac{\partial \mathbf{M}_7}{\partial \dot{\boldsymbol{\eta}}_{1d}}\right)\begin{bmatrix} \ddot{\boldsymbol{\eta}}_{1d,1} & 0 & 0 \\ 0 & \ddot{\boldsymbol{\eta}}_{1d,2} & 0 \\ 0 & 0 & \ddot{\boldsymbol{\eta}}_{1d,3}\end{bmatrix} \\ &= -m\frac{2\gamma_1\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)(1+\|\boldsymbol{\nu}_{1e}\|^2)^{1.5}}\left(\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\left(\ddot{\boldsymbol{\eta}}_{1d}\boldsymbol{\nu}_{1e}^{\mathrm{T}} + \boldsymbol{\nu}_{1e}^{\mathrm{T}}\ddot{\boldsymbol{\eta}}_{1d}\mathbf{I}_{3x3}\right)\right. \\ &\qquad \left. + \boldsymbol{\nu}_{1e}\ddot{\boldsymbol{\eta}}_{1d}^{\mathrm{T}}\right),\end{aligned} \tag{F-137}$$

$$\begin{aligned}\mathrm{Y}_{8,3,1} &= \frac{\partial}{\partial \hat{\boldsymbol{\nu}}_1}\left(\frac{\partial \mathbf{M}_7}{\partial \dot{\boldsymbol{\eta}}_{1d}}\right)\begin{bmatrix} \ddot{\boldsymbol{\eta}}_{1d,1} & 0 & 0 \\ 0 & \ddot{\boldsymbol{\eta}}_{1d,2} & 0 \\ 0 & 0 & \ddot{\boldsymbol{\eta}}_{1d,3}\end{bmatrix} \\ &= \mathrm{Y}_{8,3,0}\mathbf{G}_1 - m\frac{2\gamma_1\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\frac{\ddot{\boldsymbol{\eta}}_{1d}\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\left(\Delta_1(\hat{\boldsymbol{\nu}}_1)\right)^2}\end{aligned} \tag{F-138}$$

$$\begin{aligned}
\mathrm{Y}_{8,3,2} &= \frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{\partial \mathbf{M}_7}{\partial \dot{\boldsymbol{\eta}}_{1d}}\right)\begin{bmatrix} \ddot{\boldsymbol{\eta}}_{1d,1} & 0 & 0 \\ 0 & \ddot{\boldsymbol{\eta}}_{1d,2} & 0 \\ 0 & 0 & \ddot{\boldsymbol{\eta}}_{1d,3} \end{bmatrix} \\
&= \mathrm{Y}_{8,3,0}\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} + m\frac{2\gamma_1\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\frac{\sigma''(\boldsymbol{\eta}_{1e})}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}} \\
&\quad * \left(\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}\left(\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}^{\mathrm{T}}\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\ddot{\boldsymbol{\eta}}_{1d}\right)\right. \\
&\qquad + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}\left(\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}^{\mathrm{T}}\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\ddot{\boldsymbol{\eta}}_{1d}\right) \\
&\qquad \left. + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}\left(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^{\mathrm{T}}\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\ddot{\boldsymbol{\eta}}_{1d}\right)\right)
\end{aligned} \tag{F-139}$$

$$\begin{aligned}
\mathrm{Y}_{8,3,3} &= \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{M}_7}{\partial \dot{\boldsymbol{\eta}}_{1d}}\right)\begin{bmatrix} \ddot{\boldsymbol{\eta}}_{1d,1} & 0 & 0 \\ 0 & \ddot{\boldsymbol{\eta}}_{1d,2} & 0 \\ 0 & 0 & \ddot{\boldsymbol{\eta}}_{1d,3} \end{bmatrix} \\
&= -\mathrm{Y}_{8,3,0}
\end{aligned} \tag{F-140}$$

M8

$$\mathbf{M}_8 = m\mathbf{N}\left(\left(\sum_{i=1}^{6}\frac{\partial \mathbf{M}_i}{\partial \hat{\boldsymbol{v}}_1}\right)d\hat{\boldsymbol{v}}_1 + \left(\sum_{i=1}^{6}\frac{\partial \mathbf{M}_i}{\partial \boldsymbol{\eta}_{1e}}\right)d\boldsymbol{\eta}_{1e} + \left(\sum_{i=1}^{6}\frac{\partial \mathbf{M}_i}{\partial \dot{\boldsymbol{\eta}}_{1d}}\right)\ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial \mathbf{M}_6}{\partial \ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d}\right) \tag{F-141}$$

$$\begin{aligned}\mathrm{Y}_{9,1} &= \left(\frac{\partial \mathbf{N}}{\partial \hat{\boldsymbol{v}}_1}\dot{\hat{\boldsymbol{v}}}_1\right)\left(\left(\sum_{i=1}^{6}\frac{\partial \mathbf{M}_i}{\partial \hat{\boldsymbol{v}}_1}\right)d\hat{\boldsymbol{v}}_1 + \left(\sum_{i=1}^{6}\frac{\partial \mathbf{M}_i}{\partial \boldsymbol{\eta}_{1e}}\right)d\boldsymbol{\eta}_{1e} + \left(\sum_{i=1}^{6}\frac{\partial \mathbf{M}_i}{\partial \dot{\boldsymbol{\eta}}_{1d}}\right)\ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial \mathbf{M}_6}{\partial \ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d}\right) \\ &= \frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})}{(\Delta_1(\hat{\boldsymbol{v}}_1))^2 - \sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1)}\left(\left(\sum_{i=1}^{6}\frac{\partial \mathbf{M}_i}{\partial \hat{\boldsymbol{v}}_1}\right)d\hat{\boldsymbol{v}}_1 + \left(\sum_{i=1}^{6}\frac{\partial \mathbf{M}_i}{\partial \boldsymbol{\eta}_{1e}}\right)d\boldsymbol{\eta}_{1e}\right. \\ &\quad \left. + \left(\sum_{i=1}^{6}\frac{\partial \mathbf{M}_i}{\partial \dot{\boldsymbol{\eta}}_{1d}}\right)\ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial \mathbf{M}_6}{\partial \ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d}\right)^{\mathrm{T}}\mathbf{N}^{\mathrm{T}}\left(\mathbf{I}_{3x3} - 2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\dot{\hat{\boldsymbol{v}}}_1.\end{aligned} \tag{F-142}$$

$$\begin{aligned}\mathrm{Y}_{9,2} &= \left(\frac{\partial \mathbf{N}}{\partial \boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e}\right)\left(\left(\sum_{i=1}^{6}\frac{\partial \mathbf{M}_i}{\partial \hat{\boldsymbol{v}}_1}\right)d\hat{\boldsymbol{v}}_1 + \left(\sum_{i=1}^{6}\frac{\partial \mathbf{M}_i}{\partial \boldsymbol{\eta}_{1e}}\right)d\boldsymbol{\eta}_{1e} + \left(\sum_{i=1}^{6}\frac{\partial \mathbf{M}_i}{\partial \dot{\boldsymbol{\eta}}_{1d}}\right)\ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial \mathbf{M}_6}{\partial \ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d}\right) \\ &= \hat{\boldsymbol{v}}_1^{\mathrm{T}}\left(\left(\sum_{i=1}^{6}\frac{\partial \mathbf{M}_i}{\partial \hat{\boldsymbol{v}}_1}\right)d\hat{\boldsymbol{v}}_1 + \left(\sum_{i=1}^{6}\frac{\partial \mathbf{M}_i}{\partial \boldsymbol{\eta}_{1e}}\right)d\boldsymbol{\eta}_{1e} + \left(\sum_{i=1}^{6}\frac{\partial \mathbf{M}_i}{\partial \dot{\boldsymbol{\eta}}_{1d}}\right)\ddot{\boldsymbol{\eta}}_{1d}\right. \\ &\quad \left. + \frac{\partial \mathbf{M}_6}{\partial \ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d}\right)\frac{\mathbf{N}\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{(\Delta_1(\hat{\boldsymbol{v}}_1))^2 - \sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1)}\dot{\boldsymbol{\eta}}_{1e},\end{aligned} \tag{F-143}$$

Appendix G –Partil Derivitives of Interlace Term h4

$$\boldsymbol{h}_4 = -\gamma_2 \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q) \boldsymbol{q_e} \tag{G-1}$$

$$\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}} = m \left(\frac{\partial \mathbf{N}}{\partial \boldsymbol{\eta}_{1e}}\right) (\mathbf{M}_1 + \mathbf{M}_2 + \mathbf{M}_3 + \mathbf{M}_4 + \mathbf{M}_5 + \mathbf{M}_6) + m\mathbf{N} \left(\frac{\partial \mathbf{M}_1}{\partial \boldsymbol{\eta}_{1e}} + \frac{\partial \mathbf{M}_2}{\partial \boldsymbol{\eta}_{1e}} + \frac{\partial \mathbf{M}_3}{\partial \boldsymbol{\eta}_{1e}} + \frac{\partial \mathbf{M}_4}{\partial \boldsymbol{\eta}_{1e}} + \frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\eta}_{1e}} + \frac{\partial \mathbf{M}_6}{\partial \boldsymbol{\eta}_{1e}}\right) \tag{G-2}$$

$$\begin{aligned}\dot{\boldsymbol{h}}_4 = -\gamma_2 \Bigg(& \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q) \dot{\boldsymbol{q}}_{\boldsymbol{e}} + \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \dot{\mathbf{R}}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q) \boldsymbol{q_e} \\ & + \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \dot{\mathbf{T}}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_2) \boldsymbol{q_e} + \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \dot{\mathbf{A}}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q) \boldsymbol{q_e} \\ & + \left(\frac{\partial}{\partial \boldsymbol{\eta}_{1e}} \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \dot{\boldsymbol{\eta}}_{1e} + \frac{\partial}{\partial \hat{\boldsymbol{v}}_1} \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \dot{\hat{\boldsymbol{v}}}_1 + \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}} \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \ddot{\boldsymbol{\eta}}_{1d} \right. \\ & \left. + \frac{\partial}{\partial \ddot{\boldsymbol{\eta}}_{1d}} \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \dddot{\boldsymbol{\eta}}_{1d} \right) \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q) \boldsymbol{q_e} \Bigg)\end{aligned} \tag{G-3}$$

$$\begin{aligned}\dot{\boldsymbol{h}}_4 = -\gamma_2 \Bigg(& \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q) \dot{\boldsymbol{q}}_{\boldsymbol{e}} + \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \dot{\mathbf{R}}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q) \boldsymbol{q_e} \\ & + \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \dot{\mathbf{T}}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_2) \boldsymbol{q_e} + \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \dot{\mathbf{A}}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q) \boldsymbol{q_e} \\ & + \left(\frac{\partial}{\partial \boldsymbol{\eta}_{1e}} \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \dot{\boldsymbol{\eta}}_{1e} + \frac{\partial}{\partial \hat{\boldsymbol{v}}_1} \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \dot{\hat{\boldsymbol{v}}}_1 + \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}} \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \ddot{\boldsymbol{\eta}}_{1d} \right. \\ & \left. + \frac{\partial}{\partial \ddot{\boldsymbol{\eta}}_{1d}} \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \dddot{\boldsymbol{\eta}}_{1d} \right) \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q) \boldsymbol{q_e} \Bigg)\end{aligned} \tag{G-4}$$

$$\begin{aligned}\dot{\boldsymbol{h}}_4 = -\gamma_2 \Bigg(& \left(\frac{\partial \dot{\boldsymbol{\Omega}}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q) \boldsymbol{q_e} + \mathbf{Q}_1 \dot{\boldsymbol{\Omega}} + \mathbf{Q}_2 \dot{\boldsymbol{\alpha}}_2 + \mathbf{Q}_3 \dot{\alpha}_\psi + \mathbf{Q}_4 \dot{\boldsymbol{\alpha}}_2 + \mathbf{Q}_5 \dot{\boldsymbol{\alpha}}_2 + \mathbf{Q}_8 \dot{\boldsymbol{q}} \\ & + \mathbf{Q}_9 \dot{\boldsymbol{\alpha}}_2 \Bigg)\end{aligned} \tag{G-5}$$

$$\begin{aligned}\dot{\boldsymbol{h}}_4 = -\gamma_2 \Bigg(& \left(\frac{\partial \dot{\boldsymbol{\Omega}}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q) \boldsymbol{q_e} + \mathbf{Q}_1 \dot{\boldsymbol{\Omega}} + \mathbf{Q}_2 \left(\mathbf{A}_{2,d} \dot{\boldsymbol{\Omega}} + \mathbf{A}_{1,d} \boldsymbol{\Omega} \dot{\alpha}_\psi\right) + \mathbf{Q}_3 \dot{\alpha}_\psi \\ & + \mathbf{Q}_4 \dot{\boldsymbol{\alpha}}_2 + \mathbf{Q}_5 \dot{\boldsymbol{\alpha}}_2 + \mathbf{Q}_8 \dot{\boldsymbol{q}} + \mathbf{Q}_9 \dot{\boldsymbol{\alpha}}_2 \Bigg)\end{aligned} \tag{G-6}$$

$$\dot{\boldsymbol{h}}_3 = -\gamma_2 \left(\left(\frac{\partial \dot{\boldsymbol{\Omega}}}{\partial \boldsymbol{\eta}_{1e}} \right)^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q) \boldsymbol{q}_{\boldsymbol{e}} + \mathbf{Q}_1 \dot{\boldsymbol{\Omega}} + \mathbf{Q}_3 \dot{\alpha}_\psi + (\mathbf{Q}_2 + \mathbf{Q}_4 + \mathbf{Q}_5 + \mathbf{Q}_9) \left(\mathbf{A}_{2,d} \dot{\boldsymbol{\Omega}} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \dot{\alpha}_\psi + \mathbf{A}_{1,d} \boldsymbol{\Omega} \dot{\alpha}_\psi, \right) + (\mathbf{Q}_8) \dot{\boldsymbol{q}} \right) \tag{G-7}$$

$$\dot{\boldsymbol{\alpha}}_2 = \mathbf{A}_{2,d} \boldsymbol{\Omega}_d + \mathbf{A}_{2,d} \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}} \tilde{\boldsymbol{v}}_1 + \mathbf{A}_{2,d} \frac{\partial \boldsymbol{\Omega}}{\partial \hat{\boldsymbol{v}}_1} \boldsymbol{h}_5 + \mathbf{A}_{2,d} \frac{\partial \boldsymbol{\Omega}}{\partial \hat{\boldsymbol{v}}_1} \mathbf{K}_{02} \tilde{\boldsymbol{\eta}}_1 + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \dot{\alpha}_\psi + \mathbf{A}_{1,d} \boldsymbol{\Omega} \dot{\alpha}_\psi, \tag{G-8}$$

$$\dot{\boldsymbol{h}}_4 = -\gamma_2 \left(\left(\frac{\partial \dot{\boldsymbol{\Omega}}}{\partial \boldsymbol{\eta}_{1e}} \right)^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q) \boldsymbol{q}_{\boldsymbol{e}} + \big(\mathbf{Q}_1 + (\mathbf{Q}_2 + \mathbf{Q}_4 + \mathbf{Q}_5 + \mathbf{Q}_9) \mathbf{A}_{2,d}\big) \dot{\boldsymbol{\Omega}} + \left(\mathbf{Q}_3 + (\mathbf{Q}_2 + \mathbf{Q}_4 + \mathbf{Q}_5 + \mathbf{Q}_9) \left(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \mathbf{A}_{1,d} \boldsymbol{\Omega}, \right) \right) \dot{\alpha}_\psi + (\mathbf{Q}_8) \dot{\boldsymbol{q}} \right) \tag{G-9}$$

$$\dot{\boldsymbol{h}}_4 = -\gamma_2 \left(\left(\frac{\partial \dot{\boldsymbol{\Omega}}}{\partial \boldsymbol{\eta}_{1e}} \right)^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q) \boldsymbol{q}_{\boldsymbol{e}} + \big(\mathbf{Q}_1 + (\mathbf{Q}_2 + \mathbf{Q}_4 + \mathbf{Q}_5 + \mathbf{Q}_9) \mathbf{A}_{2,d}\big) \left(\boldsymbol{\Omega}_d + \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}} \tilde{\boldsymbol{v}}_1 + \mathbf{A}_{2,d} \frac{\partial \boldsymbol{\Omega}}{\partial \hat{\boldsymbol{v}}_1} \mathbf{K}_{02} \tilde{\boldsymbol{\eta}}_1 \right) + \left(\mathbf{Q}_3 + (\mathbf{Q}_2 + \mathbf{Q}_4 + \mathbf{Q}_5 + \mathbf{Q}_9) \left(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \mathbf{A}_{1,d} \boldsymbol{\Omega}, \right) \right) \dot{\alpha}_\psi + (\mathbf{Q}_8) \dot{\boldsymbol{q}} \right) \tag{G-10}$$

$$\dot{\boldsymbol{h}}_4 = -\gamma_2 \left(\left(\frac{\partial \dot{\boldsymbol{\Omega}}}{\partial \boldsymbol{\eta}_{1e}} \right)^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q) \boldsymbol{q}_{\boldsymbol{e}} + \sum_{i=1}^{9} \mathbf{Z}_{i,1} \dot{\hat{\boldsymbol{v}}}_1 + \sum_{i=1}^{9} \mathbf{Z}_{i,2} \ddot{\boldsymbol{\eta}}_{1d} + \sum_{i=1}^{9} \mathbf{Z}_{i,3} \dot{\boldsymbol{\eta}}_{1e} + \sum_{i=1}^{9} \mathbf{Z}_{i,4} \dddot{\boldsymbol{\eta}}_{1d} + \big(\mathbf{Q}_1 + (\mathbf{Q}_2 + \mathbf{Q}_4 + \mathbf{Q}_5 + \mathbf{Q}_9) \mathbf{A}_{2,d}\big) \left(\boldsymbol{\Omega}_d + \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}} \tilde{\boldsymbol{v}}_1 + \frac{\partial \boldsymbol{\Omega}}{\partial \hat{\boldsymbol{v}}_1} \mathbf{K}_{02} \tilde{\boldsymbol{\eta}}_1 \right) + \left(\mathbf{Q}_3 + (\mathbf{Q}_2 + \mathbf{Q}_4 + \mathbf{Q}_5 + \mathbf{Q}_9) \left(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \mathbf{A}_{1,d} \boldsymbol{\Omega}, \right) \right) \dot{\alpha}_\psi + (\mathbf{Q}_8) \dot{\boldsymbol{q}} \right) \tag{G-11}$$

$$\dot{\boldsymbol{h}}_4 = -\gamma_2\left(\left(\frac{\partial\dot{\boldsymbol{\Omega}}}{\partial\boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}}\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}} + \sum_{i=1}^{9}\mathbf{Z}_{i,1}\,\dot{\hat{\boldsymbol{v}}}_1 + \sum_{i=1}^{9}\mathbf{Z}_{i,2}\,\dot{\boldsymbol{\eta}}_{1e} + \sum_{i=1}^{9}\mathbf{Z}_{i,3}\,\ddot{\boldsymbol{\eta}}_{1d} + \sum_{i=1}^{9}\mathbf{Z}_{i,4}\,\dddot{\boldsymbol{\eta}}_{1d} + \sum_{i=1}^{9}\mathbf{Z}_{i,1}\,\mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1 + \sum_{i=1}^{9}\mathbf{Z}_{i,2}\,\tilde{\boldsymbol{v}}_1 + \left(\mathbf{Q}_1 + (\mathbf{Q}_2+\mathbf{Q}_4+\mathbf{Q}_5+\mathbf{Q}_9)\mathbf{A}_{2,d}\right)\left(\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\tilde{\boldsymbol{v}}_1 + \frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1\right) + \left(\mathbf{Q}_3 + (\mathbf{Q}_2+\mathbf{Q}_4+\mathbf{Q}_5+\mathbf{Q}_9)\left(\begin{bmatrix}0\\0\\1\end{bmatrix} + \mathbf{A}_{1,d}\boldsymbol{\Omega},\right)\right)\dot{\alpha}_\psi + (\mathbf{Q}_8)\dot{\boldsymbol{q}}\right) \tag{G-12}$$

$$\dot{\boldsymbol{h}}_4 = -\gamma_2\left(\sum_{i=1}^{9}\mathbf{Z}_{i,1}\,\dot{\hat{\boldsymbol{v}}}_1 + \sum_{i=1}^{9}\mathbf{Z}_{i,2}\,\dot{\boldsymbol{\eta}}_{1e} + \sum_{i=1}^{9}\mathbf{Z}_{i,3}\,\ddot{\boldsymbol{\eta}}_{1d} + \sum_{i=1}^{9}\mathbf{Z}_{i,4}\,\dddot{\boldsymbol{\eta}}_{1d} + \sum_{i=1}^{9}\mathbf{Z}_{i,1}\,\mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1 + \sum_{i=1}^{9}\mathbf{Z}_{i,2}\,\tilde{\boldsymbol{v}}_1 + \left(\mathbf{Q}_1 + (\mathbf{Q}_2+\mathbf{Q}_4+\mathbf{Q}_5+\mathbf{Q}_9)\mathbf{A}_{2,d}\right)\left(\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\tilde{\boldsymbol{v}}_1 + \frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1\right) + \left(\mathbf{Q}_3 + (\mathbf{Q}_2+\mathbf{Q}_4+\mathbf{Q}_5+\mathbf{Q}_9)\left(\begin{bmatrix}0\\0\\1\end{bmatrix} + \mathbf{A}_{1,d}\boldsymbol{\Omega}\right)\right)\dot{\alpha}_\psi + (\mathbf{Q}_8)\dot{\boldsymbol{q}}\right) \tag{G-13}$$

$$\begin{aligned}\dot{\boldsymbol{h}}_4 &= \frac{\partial\boldsymbol{h}_4}{\partial\boldsymbol{\eta}_{1e}}\dot{\hat{\boldsymbol{v}}}_1 + \frac{\partial\boldsymbol{h}_4}{\partial\boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e} + \frac{\partial\boldsymbol{h}_4}{\partial\dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} + \frac{\partial\boldsymbol{h}_4}{\partial\ddot{\boldsymbol{\eta}}_{1d}}\dddot{\boldsymbol{\eta}}_{1d} + \frac{\partial\boldsymbol{h}_4}{\partial\alpha_\psi}\dot{\alpha}_\psi + \frac{\partial\boldsymbol{h}_4}{\partial\boldsymbol{q}}\dot{\boldsymbol{q}} \\ &= -\gamma_2\left(\sum_{i=1}^{9}\mathbf{Z}_{i,1}\,\dot{\hat{\boldsymbol{v}}}_1 + \sum_{i=1}^{9}\mathbf{Z}_{i,2}\,\dot{\boldsymbol{\eta}}_{1e} + \sum_{i=1}^{9}\mathbf{Z}_{i,3}\,\ddot{\boldsymbol{\eta}}_{1d} + \sum_{i=1}^{9}\mathbf{Z}_{i,4}\,\dddot{\boldsymbol{\eta}}_{1d} + \sum_{i=1}^{9}\mathbf{Z}_{i,1}\,\mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1 + \sum_{i=1}^{9}\mathbf{Z}_{i,2}\,\tilde{\boldsymbol{v}}_1 \right.\\ &\quad + \left(\mathbf{Q}_1 + (\mathbf{Q}_2+\mathbf{Q}_4+\mathbf{Q}_5+\mathbf{Q}_9)\mathbf{A}_{2,d}\right)\left(\boldsymbol{\Omega}_d + \frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\tilde{\boldsymbol{v}}_1 + \frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\mathbf{K}_{02}\tilde{\boldsymbol{\eta}}_1\right) \\ &\quad \left. + \left(\mathbf{Q}_3 + (\mathbf{Q}_2+\mathbf{Q}_4+\mathbf{Q}_5+\mathbf{Q}_9)\left(\begin{bmatrix}0\\0\\1\end{bmatrix} + \mathbf{A}_{1,d}\boldsymbol{\Omega}\right)\right)\dot{\alpha}_\psi + (\mathbf{Q}_8)\dot{\boldsymbol{q}}\right)\end{aligned} \tag{G-14}$$

$$\frac{\partial\boldsymbol{h}_4}{\partial\boldsymbol{\eta}_{1e}}\dot{\hat{\boldsymbol{v}}}_1 = -\gamma_2\left(\sum_{i=1}^{9}\mathbf{Z}_{i,1} + \left(\mathbf{Q}_1 + (\mathbf{Q}_2+\mathbf{Q}_4+\mathbf{Q}_5+\mathbf{Q}_9)\mathbf{A}_{2,d}\right)\frac{\partial\boldsymbol{\Omega}}{\partial\hat{\boldsymbol{v}}_1}\right)\dot{\hat{\boldsymbol{v}}}_1 \tag{G-15}$$

$$\frac{\partial\boldsymbol{h}_4}{\partial\boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e} = -\gamma_2\left(\sum_{i=1}^{9}\mathbf{Z}_{i,2} + \left(\mathbf{Q}_1 + (\mathbf{Q}_2+\mathbf{Q}_4+\mathbf{Q}_5+\mathbf{Q}_9)\mathbf{A}_{2,d}\right)\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\right)\dot{\boldsymbol{\eta}}_{1e} \tag{G-16}$$

$$\frac{\partial\boldsymbol{h}_4}{\partial\dot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} = -\gamma_2\left(\sum_{i=1}^{9}\mathbf{Z}_{i,3}\,\ddot{\boldsymbol{\eta}}_{1d} + \left(\mathbf{Q}_1 + (\mathbf{Q}_2+\mathbf{Q}_4+\mathbf{Q}_5+\mathbf{Q}_9)\mathbf{A}_{2,d}\right)\frac{\partial\boldsymbol{\Omega}}{\partial\dot{\boldsymbol{\eta}}_{1d}}\right)\ddot{\boldsymbol{\eta}}_{1d} \tag{G-17}$$

$$\frac{\partial \boldsymbol{h}_4}{\partial \ddot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d} = -\gamma_2\left(\sum_{i=1}^{9}\mathbf{Z}_{i,4}\,\ddot{\boldsymbol{\eta}}_{1d} + \left(\mathbf{Q}_1 + (\mathbf{Q}_2 + \mathbf{Q}_4 + \mathbf{Q}_5 + \mathbf{Q}_9)\mathbf{A}_{2,d}\right)\frac{\partial \boldsymbol{\Omega}}{\partial \ddot{\boldsymbol{\eta}}_{1d}}\ddot{\boldsymbol{\eta}}_{1d}\right) \tag{G-18}$$

$$\frac{\partial \boldsymbol{h}_4}{\partial \alpha_\psi}\dot{\alpha}_\psi = -\gamma_2\left(\mathbf{Q}_3 + (\mathbf{Q}_2 + \mathbf{Q}_4 + \mathbf{Q}_5 + \mathbf{Q}_9)\left(\begin{bmatrix}0\\0\\1\end{bmatrix} + \mathbf{A}_{1,d}\boldsymbol{\Omega}\right)\right)\dot{\alpha}_\psi \tag{G-19}$$

$$\frac{\partial \boldsymbol{h}_4}{\partial \boldsymbol{q}}\dot{\boldsymbol{q}} = -\gamma_2(\mathbf{Q}_8)\dot{\boldsymbol{q}} \tag{G-20}$$

$$\begin{aligned}\mathbf{Q}_1\dot{\boldsymbol{\Omega}} = \Bigg(&\begin{bmatrix}-\frac{s_{\alpha_\psi}}{\cos(\alpha_\phi)} & 0 & 0\\ \frac{c_{\alpha_\psi}}{\cos(\alpha_\phi)} & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\frac{\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\boldsymbol{\Omega}^{\mathrm{T}}}{(\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega})^{1.5}} + \begin{bmatrix}0 & -\cos^2(\alpha_\theta)\frac{c_{\alpha_\psi}}{\Omega_3^2} & 0\\ 0 & -\cos^2(\alpha_\theta)\frac{s_{\alpha_\psi}}{\Omega_3^2} & 0\\ 0 & \cos^2(\alpha_\theta)\frac{\tan(\alpha_\theta)}{\Omega_3^2} & 0\end{bmatrix}\\ &* \frac{\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}}{(\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega})^{1.5}}\begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}} - \mathbf{I}_{3x3}\begin{bmatrix}\frac{\tan(\alpha_\phi)}{\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega}}\\0\\0\end{bmatrix}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\\ &+ 2\boldsymbol{\Omega}\begin{bmatrix}\frac{\tan(\alpha_\phi)}{(\boldsymbol{\Omega}^{\mathrm{T}}\boldsymbol{\Omega})^2}\\0\\0\end{bmatrix}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\boldsymbol{\Omega}^{\mathrm{T}}\Bigg)\dot{\boldsymbol{\Omega}}\end{aligned} \tag{G-21}$$

$$\mathbf{Q}_2\left(\mathbf{A}_{2,d}\dot{\boldsymbol{\Omega}} + \mathbf{A}_{1,d}\boldsymbol{\Omega}\dot{\alpha}_\psi\right) = \left(\begin{bmatrix} s_{\alpha_\psi}\dfrac{\sin(\alpha_\phi)}{(\cos(\alpha_\phi))^2\sqrt{\boldsymbol{\Omega}^\mathrm{T}\boldsymbol{\Omega}}} & 0 & 0 \\ -c_{\alpha_\psi}\dfrac{\sin(\alpha_\phi)}{(\cos(\alpha_\phi))^2\sqrt{\boldsymbol{\Omega}^\mathrm{T}\boldsymbol{\Omega}}} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^\mathrm{T}(\boldsymbol{\alpha}_q)\boldsymbol{q}_e\begin{bmatrix}1\\0\\0\end{bmatrix}^\mathrm{T} + \begin{bmatrix} 0 & -c_{\alpha_\psi}\dfrac{\sin(2\alpha_\theta)}{\Omega_3} & 0 \\ 0 & -s_{\alpha_\psi}\dfrac{\sin(2\alpha_\theta)}{\Omega_3} & 0 \\ 0 & -\dfrac{\cos(2\alpha_\theta)}{\Omega_3} & 0 \end{bmatrix}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^\mathrm{T}(\boldsymbol{\alpha}_q)\boldsymbol{q}_e\begin{bmatrix}0\\1\\0\end{bmatrix}^\mathrm{T} - \boldsymbol{\Omega}\begin{bmatrix}\dfrac{1}{(\cos(\alpha_\phi))^2\boldsymbol{\Omega}^\mathrm{T}\boldsymbol{\Omega}}\\0\\0\end{bmatrix}^\mathrm{T}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^\mathrm{T}(\boldsymbol{\alpha}_q)\boldsymbol{q}_e\begin{bmatrix}1\\0\\0\end{bmatrix}^\mathrm{T}\right)\left(\mathbf{A}_{2,d}\dot{\boldsymbol{\Omega}} + \mathbf{A}_{1,d}\boldsymbol{\Omega}\dot{\alpha}_\psi\right) \tag{G-22}$$

$$\mathbf{Q}_3\dot{\alpha}_\psi = \dot{\alpha}_\psi\begin{bmatrix} \dfrac{c_{\alpha_\psi}}{\cos(\alpha_\phi)\sqrt{\boldsymbol{\Omega}^\mathrm{T}\boldsymbol{\Omega}}} & -\cos^2(\alpha_\theta)\dfrac{s_{\alpha_\psi}}{\Omega_3} & 0 \\ \dfrac{s_{\alpha_\psi}}{\cos(\alpha_\phi)\sqrt{\boldsymbol{\Omega}^\mathrm{T}\boldsymbol{\Omega}}} & \cos^2(\alpha_\theta)\dfrac{c_{\alpha_\psi}}{\Omega_3} & 0 \\ 0 & 0 & 0 \end{bmatrix}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^\mathrm{T}(\boldsymbol{\alpha}_q)\boldsymbol{q}_e \tag{G-23}$$

$$\mathbf{Q}_4\dot{\boldsymbol{\alpha}}_2 = \left(\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\right)^\mathrm{T}\mathbf{A}_{2,d}^\mathrm{T}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^\mathrm{T}(\boldsymbol{\alpha}_q)\boldsymbol{q}_e = \left(\frac{\partial\boldsymbol{\Omega}}{\partial\boldsymbol{\eta}_{1e}}\right)^\mathrm{T}\mathbf{A}_{2,d}^\mathrm{T}\left(\begin{bmatrix} 0 & 0 & 0 \\ 0 & -\sin(\alpha_\phi) & \cos(\alpha_\theta)\cos(\alpha_\phi) \\ 0 & -\cos(\alpha_\phi) & -\cos(\alpha_\theta)\sin(\alpha_\phi) \end{bmatrix}^\mathrm{T}\mathbf{R}_2^\mathrm{T}(\boldsymbol{\alpha}_q)\boldsymbol{q}_e\begin{bmatrix}1\\0\\0\end{bmatrix}^\mathrm{T} + \begin{bmatrix} 0 & 0 & -\cos(\alpha_\theta) \\ 0 & 0 & -\sin(\alpha_\theta)\sin(\alpha_\phi) \\ 0 & 0 & -\sin(\alpha_\theta)\cos(\alpha_\phi) \end{bmatrix}\mathbf{R}_2^\mathrm{T}(\boldsymbol{\alpha}_q)\boldsymbol{q}_e\begin{bmatrix}0\\1\\0\end{bmatrix}^\mathrm{T}\right)\dot{\boldsymbol{\alpha}}_2 \tag{G-24}$$

$$\mathbf{Q}_5\dot{\boldsymbol{\alpha}}_q = \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\dot{\mathbf{R}}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_e \tag{G-25}$$

$$= \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\Bigg(\frac{1}{2}\Bigg(\boldsymbol{\alpha}_q^{\mathrm{T}}\boldsymbol{q}_e\mathbf{I}_{3x3} + \boldsymbol{\alpha}_q\boldsymbol{q}_e^{\mathrm{T}} - \left(\begin{bmatrix}0&0&0\\0&0&1\\0&-1&0\end{bmatrix}([1\quad 0\quad 0]\boldsymbol{q}_e) + \begin{bmatrix}0&0&-1\\0&0&0\\1&0&0\end{bmatrix}([0\quad 1\quad 0]\boldsymbol{q}_e) + \begin{bmatrix}0&1&0\\-1&0&0\\0&0&0\end{bmatrix}([0\quad 0\quad 1]\boldsymbol{q}_e)\right) - \boldsymbol{q}_e\boldsymbol{\alpha}_q^{\mathrm{T}}\Bigg)\Bigg)\mathbf{R}_2(\boldsymbol{\alpha}_q)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\dot{\boldsymbol{\alpha}}_2$$

$$\mathbf{Q}_8\dot{\boldsymbol{q}} = \left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\mathbf{R}_2(\boldsymbol{q})\boldsymbol{\omega} \tag{G-26}$$

$$\mathbf{Q}_9\dot{\boldsymbol{\alpha}}_2 = -\left(\frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\mathbf{R}_2(\boldsymbol{\alpha}_2)\mathbf{T}(\boldsymbol{\alpha}_2)^{-1}\dot{\boldsymbol{\alpha}}_2 \tag{G-27}$$

$$\left(\frac{\partial \mathbf{N}}{\partial \boldsymbol{\eta}_{1e}}\right)(\mathbf{M}_1 + \mathbf{M}_2 + \mathbf{M}_3 + \mathbf{M}_4 + \mathbf{M}_5 + \mathbf{M}_6) = \frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{M}_1 + \mathbf{M}_2 + \mathbf{M}_3 + \mathbf{M}_4 + \mathbf{M}_5 + \mathbf{M}_6)}{\left(1 - \sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\right)}\left(\mathbf{I}_{3x3} + \frac{\mathbf{K}_1\frac{\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}}{1 - \sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}}\right)\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} \tag{G-28}$$

$$\left(\frac{\partial \mathbf{N}}{\partial \boldsymbol{\eta}_{1e}}\right)(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6) = \frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)}{1-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}}\mathbf{N}\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} \tag{G-29}$$

$$\left(\left(\frac{\partial \mathbf{N}}{\partial \boldsymbol{\eta}_{1e}}\right)(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)\right)^{\mathrm{T}} = \frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)}{\Delta_1(\hat{\boldsymbol{v}}_1)^2-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\hat{\boldsymbol{v}}_1}\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\,\mathbf{N}^{\mathbf{T}} \tag{G-30}$$

$$\begin{aligned}
\boldsymbol{Z}_{1,1}\dot{\hat{\boldsymbol{v}}}_1 &= \left(\frac{\partial}{\partial \hat{\boldsymbol{v}}_1}\left(\left(\frac{\partial \mathbf{N}}{\partial \boldsymbol{\eta}_{1e}}\right)(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5 \right.\right.\\
&\quad \left.\left.+\mathbf{M}_6)\right)^{\mathrm{T}}\dot{\hat{\boldsymbol{v}}}_1\right)\gamma_2 m\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\\
&= \left(\left(\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\gamma_2 m\mathbf{N}^{\mathrm{T}}\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}}{1-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}}(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4\right.\right.\\
&\quad +\mathbf{M}_5+\mathbf{M}_6)^{\mathrm{T}}+\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{M}_1+\mathbf{M}_2+\mathbf{M}_3+\mathbf{M}_4+\mathbf{M}_5+\mathbf{M}_6)}{\left(1-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\right)^2}\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\\
&\quad *\left(\gamma_2 m\mathbf{N}^{\mathrm{T}}\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\right.\\
&\quad \left.+\left(\gamma_2 m\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\right)^{\mathrm{T}}\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})\mathbf{N}^{\mathrm{T}}\right)*\left(\frac{\mathbf{I}_{3x3}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}-2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^3}\right)\\
&\quad +\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\mathbf{N}^{\mathrm{T}}\frac{\gamma_2 m\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{1-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}}\\
&\quad \left.*\left(\frac{\partial \mathbf{M}_1}{\partial \hat{\boldsymbol{v}}_1}+\frac{\partial \mathbf{M}_2}{\partial \hat{\boldsymbol{v}}_1}+\frac{\partial \mathbf{M}_3}{\partial \hat{\boldsymbol{v}}_1}+\frac{\partial \mathbf{M}_4}{\partial \hat{\boldsymbol{v}}_1}+\frac{\partial \mathbf{M}_5}{\partial \hat{\boldsymbol{v}}_1}+\frac{\partial \mathbf{M}_6}{\partial \hat{\boldsymbol{v}}_1}\right)\right)\dot{\hat{\boldsymbol{v}}}_1
\end{aligned} \tag{G-31}$$

(G-32)

$$\boldsymbol{Z}_{12}\dot{\boldsymbol{\eta}}_{1e} = \left(\frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\left(\frac{\partial \mathbf{N}}{\partial \boldsymbol{\eta}_{1e}}\right)(\mathbf{M}_1 + \mathbf{M}_2 + \mathbf{M}_3 + \mathbf{M}_4 + \mathbf{M}_5 + \mathbf{M}_6)\right)^{\mathrm{T}} \dot{\boldsymbol{\eta}}_{1e}\right)\gamma_2 m \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}$$

$$= \left(\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{M}_1 + \mathbf{M}_2 + \mathbf{M}_3 + \mathbf{M}_4 + \mathbf{M}_5 + \mathbf{M}_6)}{1 - \sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1 \frac{\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}} \frac{\mathbf{K}_1 \sigma''(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\right.$$

$$* \left(\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}\left(\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}^{\mathrm{T}} \gamma_2 m \mathbf{N}^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\right)\right.$$

$$+ \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}\left(\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}^{\mathrm{T}} \gamma_2 m \mathbf{N}^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\right) + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\left.\left. * \left(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^{\mathrm{T}} \gamma_2 m \mathbf{N}^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\right)\right)\right)$$

$$+ \frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{M}_1 + \mathbf{M}_2 + \mathbf{M}_3 + \mathbf{M}_4 + \mathbf{M}_5 + \mathbf{M}_6)}{\left(1 - \sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1 \frac{\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\right)^2} \frac{\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}$$

$$* \mathbf{N}^{\mathrm{T}} \|\boldsymbol{q}_e\|^2 \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}} \left(\frac{\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\right)^{\mathrm{T}} \mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})$$

$$+ \frac{\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} \frac{\gamma_2 m \mathbf{N}^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}} \hat{\boldsymbol{v}}_1^{\mathrm{T}}}{1 - \sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1 \frac{\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}} \left(\frac{\partial \mathbf{M}_1}{\partial \boldsymbol{\eta}_{1e}} + \frac{\partial \mathbf{M}_2}{\partial \boldsymbol{\eta}_{1e}} + \frac{\partial \mathbf{M}_3}{\partial \boldsymbol{\eta}_{1e}} + \frac{\partial \mathbf{M}_4}{\partial \boldsymbol{\eta}_{1e}}\right.$$

$$\left. + \frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\eta}_{1e}} + \frac{\partial \mathbf{M}_6}{\partial \boldsymbol{\eta}_{1e}}\right) + \frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}(\mathbf{M}_1 + \mathbf{M}_2 + \mathbf{M}_3 + \mathbf{M}_4 + \mathbf{M}_5 + \mathbf{M}_6)}{1 - \sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1 \frac{\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}}$$

$$\left. * \frac{\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} \left(\frac{\frac{\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} \left(\gamma_2 m \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\right)^{\mathrm{T}}}{1 - \sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1 \frac{\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}} \mathbf{N}\right) \mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})\right) \dot{\boldsymbol{\eta}}_{1e}$$

$$\boldsymbol{Z}_{13}\ddot{\boldsymbol{\eta}}_{1d} = \left(\frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\left(\frac{\partial \mathbf{N}}{\partial \boldsymbol{\eta}_{1e}}\right)(\mathbf{M}_1 + \mathbf{M}_2 + \mathbf{M}_3 + \mathbf{M}_4 + \mathbf{M}_5 + \mathbf{M}_6)\right)^{\mathrm{T}} \ddot{\boldsymbol{\eta}}_{1d}\right)\|\boldsymbol{q}_e\|^2 \mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}} = \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\frac{\gamma_2 m \mathbf{N}^{\mathrm{T}}\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}}{1 - \sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\hat{\boldsymbol{\nu}}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}} * \hat{\boldsymbol{\nu}}_1^{\mathrm{T}}\left(\frac{\partial \mathbf{M}_1}{\partial \dot{\boldsymbol{\eta}}_{1d}} + \frac{\partial \mathbf{M}_2}{\partial \dot{\boldsymbol{\eta}}_{1d}} + \frac{\partial \mathbf{M}_3}{\partial \dot{\boldsymbol{\eta}}_{1d}} + \frac{\partial \mathbf{M}_5}{\partial \dot{\boldsymbol{\eta}}_{1d}} + \frac{\partial \mathbf{M}_6}{\partial \dot{\boldsymbol{\eta}}_{1d}}\right)\ddot{\boldsymbol{\eta}}_{1d} \quad \text{(G-33)}$$

$$\boldsymbol{Z}_{14}\dddot{\boldsymbol{\eta}}_{1d} = \left(\frac{\partial}{\partial \ddot{\boldsymbol{\eta}}_{1d}}\left(\left(\frac{\partial \mathbf{N}}{\partial \boldsymbol{\eta}_{1e}}\right)(\mathbf{M}_1 + \mathbf{M}_2 + \mathbf{M}_3 + \mathbf{M}_4 + \mathbf{M}_5 + \mathbf{M}_6)\right)^{\mathrm{T}} \dddot{\boldsymbol{\eta}}_{1d}\right) = \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\frac{\gamma_2 m \mathbf{N}^{\mathrm{T}}\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}}{1 - \sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\hat{\boldsymbol{\nu}}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}}\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}\left(\frac{\partial \mathbf{M}_6}{\partial \ddot{\boldsymbol{\eta}}_{1d}}\right)\mathbf{N}^{\mathrm{T}}\dddot{\boldsymbol{\eta}}_{1d} \quad \text{(G-34)}$$

$$\frac{\partial \mathbf{M}_1}{\partial \boldsymbol{\eta}_{1e}} = -\left(\mathbf{D}_1 + \mathbf{K}_2 \sigma'(\boldsymbol{\nu}_{1e})\right)\mathbf{K}_1 \frac{\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \tag{G-35}$$

$$\left(\frac{\partial \mathbf{M}_1}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} = -\left(\mathbf{D}_1 + \mathbf{K}_2 \sigma'(\boldsymbol{\nu}_{1e})\right)\mathbf{K}_1 \frac{\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \gamma_2 m \mathbf{N}^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}} \tag{G-36}$$

$$\begin{aligned}
\boldsymbol{Z}_{2,1}\dot{\hat{\boldsymbol{\nu}}}_1 &= \left(\frac{\partial}{\partial \hat{\boldsymbol{\nu}}_1}\left(\frac{\partial \mathbf{M}_1}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \gamma_2 m \mathbf{N}^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\right)\dot{\hat{\boldsymbol{\nu}}}_1 \\
&= \Bigg(-\left(\mathbf{K}_2 \sigma''(\boldsymbol{\nu}_{1e})\right)\frac{\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\Bigg(\begin{bmatrix}1&0&0\\0&0&0\\0&0&0\end{bmatrix}\left(\begin{bmatrix}1\\0\\0\end{bmatrix}^{\mathrm{T}} \gamma_2 m \mathbf{N}^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\right) \\
&+ \begin{bmatrix}0&0&0\\0&1&0\\0&0&0\end{bmatrix}\left(\begin{bmatrix}0\\1\\0\end{bmatrix}^{\mathrm{T}} \gamma_2 m \mathbf{N}^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\right) \\
&+ \begin{bmatrix}0&0&0\\0&0&0\\0&0&1\end{bmatrix}\left(\begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}} \gamma_2 m \mathbf{N}^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\right)\Bigg)\mathbf{G}_1 \\
&+ \left(\mathbf{D}_1 + \mathbf{K}_2 \sigma'(\boldsymbol{\nu}_{1e})\right)\mathbf{K}_1 \frac{\sigma'(\boldsymbol{\eta}_{1e})}{\left(\Delta_1(\hat{\boldsymbol{\nu}}_1)\right)^2}\left(\gamma_2 m \mathbf{N}^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\right)\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}\Bigg)\dot{\hat{\boldsymbol{\nu}}}_1
\end{aligned} \tag{G-37}$$

$$\begin{aligned}
\boldsymbol{Z}_{2,2}\dot{\boldsymbol{\eta}}_{1e} &= \left(\frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{\partial \mathbf{M}_1}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \gamma_2 m \mathbf{N}^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\right)\dot{\boldsymbol{\eta}}_{1e} \\
&= -\left(\mathbf{K}_2 \sigma''(\boldsymbol{\nu}_{1e})\frac{\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\frac{\mathbf{K}_1 \sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \left(\mathbf{D}_1 + \mathbf{K}_2 \sigma'(\boldsymbol{\nu}_{1e})\right)\mathbf{K}_1 \frac{\sigma''(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\right) \\
&* \Bigg(\begin{bmatrix}1&0&0\\0&0&0\\0&0&0\end{bmatrix}\left(\begin{bmatrix}1\\0\\0\end{bmatrix}^{\mathrm{T}} \gamma_2 m \mathbf{N}^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\right) \\
&+ \begin{bmatrix}0&0&0\\0&1&0\\0&0&0\end{bmatrix}\left(\begin{bmatrix}0\\1\\0\end{bmatrix}^{\mathrm{T}} \gamma_2 m \mathbf{N}^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\right) \\
&+ \begin{bmatrix}0&0&0\\0&0&0\\0&0&1\end{bmatrix}\left(\begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}} \gamma_2 m \mathbf{N}^{\mathrm{T}} \mathbf{A}_{2,d}^{\mathrm{T}} \mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}} \mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\right)\Bigg)\dot{\boldsymbol{\eta}}_{1e}
\end{aligned} \tag{G-38}$$

$$\boldsymbol{Z}_{2,3}\ddot{\boldsymbol{\eta}}_{1d} = \left(\frac{\partial}{\partial\dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial\mathbf{M}_1}{\partial\boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}}\gamma_2 m\mathbf{N}^{\mathrm{T}}\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\right)\ddot{\boldsymbol{\eta}}_{1d}$$
$$= \mathbf{K}_2\sigma''(\boldsymbol{\nu}_{1e})\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}$$
$$*\left(\begin{bmatrix}1&0&0\\0&0&0\\0&0&0\end{bmatrix}\left(\begin{bmatrix}1\\0\\0\end{bmatrix}^{\mathrm{T}}\gamma_2 m\mathbf{N}^{\mathrm{T}}\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\right)\right.$$
$$+\begin{bmatrix}0&0&0\\0&1&0\\0&0&0\end{bmatrix}\left(\begin{bmatrix}0\\1\\0\end{bmatrix}^{\mathrm{T}}\gamma_2 m\mathbf{N}^{\mathrm{T}}\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\right)$$
$$\left.+\begin{bmatrix}0&0&0\\0&0&0\\0&0&1\end{bmatrix}\left(\begin{bmatrix}0\\0\\1\end{bmatrix}^{\mathrm{T}}\gamma_2 m\mathbf{N}^{\mathrm{T}}\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}\right)\right)\ddot{\boldsymbol{\eta}}_{1d} \quad \text{(G-39)}$$

$$\frac{\partial \mathbf{M}_2}{\partial \boldsymbol{\eta}_{1e}} = -\gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\left((2+\|\boldsymbol{\alpha}_1\|^2)\mathrm{I}_{3\times 3} - \sigma(\boldsymbol{\eta}_{1e})\frac{2\boldsymbol{\alpha}_1^{\mathrm{T}}\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\frac{\sigma'(\boldsymbol{\eta}_{1e})}{2\Delta_1(\hat{\boldsymbol{v}}_1)}, \tag{G-40}$$

$$\boldsymbol{\vartheta}_1 = m\mathbf{N}^{\mathrm{T}}\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}, \tag{G-41}$$

$$\begin{aligned}\mathbf{Z}_{3,1} &= \frac{\partial}{\partial \hat{\boldsymbol{v}}_1}\left(\frac{\partial \mathbf{M}_2}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}}\begin{bmatrix}\boldsymbol{\vartheta}_{1,1} & 0 & 0\\ 0 & \boldsymbol{\vartheta}_{1,2} & 0\\ 0 & 0 & \boldsymbol{\vartheta}_{1,3}\end{bmatrix}\\ &= \gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma'(\boldsymbol{\eta}_{1e})\left((2+\|\boldsymbol{\alpha}_1\|^2)\mathbf{I}_{3\times 3} - \sigma(\boldsymbol{\eta}_{1e})\frac{2\boldsymbol{\alpha}_1^{\mathrm{T}}\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\frac{\boldsymbol{\vartheta}_1}{2\big(\Delta_1(\hat{\boldsymbol{v}}_1)\big)^2}\hat{\boldsymbol{v}}_1^{\mathrm{T}}\\ &\quad -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma'(\boldsymbol{\eta}_{1e})\Bigg(-\boldsymbol{\alpha}_1^{\mathrm{T}}\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})\boldsymbol{\vartheta}_1 + \mathbf{K}_1\boldsymbol{\alpha}_1\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\boldsymbol{\vartheta}_1\\ &\qquad - \mathbf{K}_1\frac{\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\boldsymbol{\vartheta}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)}\mathbf{K}_1\sigma(\boldsymbol{v}_{1e})\Bigg)\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\big(\Delta_1(\hat{\boldsymbol{v}}_1)\big)^3}\end{aligned} \tag{G-42}$$

$$\begin{aligned}\mathbf{Z}_{3,2} &= \frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{\partial \mathbf{M}_2}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix}d\boldsymbol{\eta}_{1e,1} & 0 & 0\\ 0 & d\boldsymbol{\eta}_{1e,2} & 0\\ 0 & 0 & d\boldsymbol{\eta}_{1e,3}\end{bmatrix}\\ &= -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{\sigma''(\boldsymbol{\eta}_{1e})}{2\Delta_1(\hat{\boldsymbol{v}}_1)}\left((2+\|\boldsymbol{\alpha}_1\|^2)\begin{bmatrix}\boldsymbol{\vartheta}_{1,1} & 0 & 0\\ 0 & \boldsymbol{\vartheta}_{1,2} & 0\\ 0 & 0 & \boldsymbol{\vartheta}_{1,3}\end{bmatrix}\right.\\ &\qquad \left. - 2\frac{\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\boldsymbol{\vartheta}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)}\mathbf{K}_1\begin{bmatrix}\boldsymbol{\alpha}_{1,1} & 0 & 0\\ 0 & \boldsymbol{\alpha}_{1,2} & 0\\ 0 & 0 & \boldsymbol{\alpha}_{1,3}\end{bmatrix}\right)\\ &\quad -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\sigma'(\boldsymbol{v}_{1e})\Bigg(-\boldsymbol{\vartheta}_1\boldsymbol{\alpha}_1^{\mathrm{T}}\mathbf{K}_1 - -\mathbf{K}_1\boldsymbol{\alpha}_1\boldsymbol{\vartheta}_1^{\mathrm{T}}\\ &\qquad + \frac{\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})\boldsymbol{\vartheta}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)}\mathbf{K}_1\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\Bigg)\frac{\sigma'(\boldsymbol{\eta}_{1e})}{\big(\Delta_1(\hat{\boldsymbol{v}}_1)\big)^2}\end{aligned} \tag{G-43}$$

$$\begin{aligned}\mathbf{Z}_{3,3} &= \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{M}_2}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}}\begin{bmatrix}\boldsymbol{\vartheta}_{1,1} & 0 & 0\\ 0 & \boldsymbol{\vartheta}_{1,2} & 0\\ 0 & 0 & \boldsymbol{\vartheta}_{1,3}\end{bmatrix}\\ &= -\gamma_1\frac{\lambda_m(\mathbf{D}_1)}{2}\frac{\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\left(\boldsymbol{\vartheta}_1\boldsymbol{\alpha}_1^{\mathrm{T}} - 2\mathbf{K}_1\frac{\sigma^{\mathrm{T}}(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\boldsymbol{\vartheta}_1\right)\end{aligned} \tag{G-44}$$

$$\frac{\partial \mathbf{M}_3}{\partial \boldsymbol{\eta}_{1e}} = \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2} \frac{\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2} \left(\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}} + \hat{\boldsymbol{\nu}}_1^{\mathrm{T}}\sigma(\boldsymbol{\eta}_{1e})\mathbf{I}_{3\times 3}\right) \sigma'(\boldsymbol{\eta}_{1e}) \tag{G-45}$$

$$\boldsymbol{\vartheta}_1 = m\mathbf{N}^{\mathrm{T}}\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}, \tag{G-46}$$

$$\begin{aligned} \mathbf{Z}_{4,1} &= \frac{\partial}{\partial \hat{\boldsymbol{\nu}}_1}\left(\frac{\partial \mathbf{M}_3}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \begin{bmatrix} \boldsymbol{\vartheta}_{1,1} & 0 & 0 \\ 0 & \boldsymbol{\vartheta}_{1,2} & 0 \\ 0 & 0 & \boldsymbol{\vartheta}_{1,3} \end{bmatrix} \\ &= \mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\boldsymbol{\vartheta}_1\left(-2\frac{\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\mathbf{D}_1 + 2\frac{\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^3}\mathbf{D}_1\hat{\boldsymbol{\nu}}_1\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}\right) \end{aligned} \tag{G-47}$$

$$\mathbf{Z}_{4,2} = \frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{\partial \mathbf{M}_3}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \begin{bmatrix} \boldsymbol{\vartheta}_{1,1} & 0 & 0 \\ 0 & \boldsymbol{\vartheta}_{1,2} & 0 \\ 0 & 0 & \boldsymbol{\vartheta}_{1,3} \end{bmatrix} = -\frac{\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}\mathbf{D}_1\hat{\boldsymbol{\nu}}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)^2}\mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e}) \begin{bmatrix} \boldsymbol{\vartheta}_{1,1} & 0 & 0 \\ 0 & \boldsymbol{\vartheta}_{1,2} & 0 \\ 0 & 0 & \boldsymbol{\vartheta}_{1,3} \end{bmatrix} \tag{G-48}$$

$$\begin{aligned} \mathbf{Z}_{4,3} &= \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{M}_3}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \begin{bmatrix} \boldsymbol{\vartheta}_{1,1} & 0 & 0 \\ 0 & \boldsymbol{\vartheta}_{1,2} & 0 \\ 0 & 0 & \boldsymbol{\vartheta}_{1,3} \end{bmatrix} \\ &= \mathbf{0}_{3x3} \end{aligned} \tag{G-49}$$

$$\frac{\partial \mathbf{M}_4}{\partial \boldsymbol{\eta}_{1e}} = \frac{\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\left(-\hat{\boldsymbol{v}}_1^{\mathrm{T}}\mathbf{D}_1\hat{\boldsymbol{v}}_1\right)\sigma'(\boldsymbol{\eta}_{1e}) \tag{G-50}$$

$$\boldsymbol{\vartheta}_1 = m\mathbf{N}^{\mathrm{T}}\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_e, \tag{G-51}$$

$$\begin{aligned}\boldsymbol{Z}_{5,1} &= \left(\frac{\partial}{\partial\hat{\boldsymbol{v}}_1}\left(\frac{\partial \mathbf{M}_4}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}}\right)\begin{bmatrix}\boldsymbol{\vartheta}_{1,1} & 0 & 0\\ 0 & \boldsymbol{\vartheta}_{1,2} & 0\\ 0 & 0 & \boldsymbol{\vartheta}_{1,3}\end{bmatrix}\\ &= \Bigg(\Bigg(\Bigg(\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)^3}\boldsymbol{\vartheta}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e}) + \boldsymbol{\vartheta}_1^{\mathrm{T}}\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})\frac{\mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)^3}\\ &\quad * \begin{bmatrix}\hat{\boldsymbol{v}}_{1,1} & 0 & 0\\ 0 & \hat{\boldsymbol{v}}_{1,2} & 0\\ 0 & 0 & \hat{\boldsymbol{v}}_{1,3}\end{bmatrix}\Bigg)\left(2 * \mathbf{I}_{3x3} - 3\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right) - 2\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\boldsymbol{\vartheta}_1\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}\mathbf{D}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\\ &\quad * \left(\mathbf{I}_{3x3} - \frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)}\right)\Bigg)\end{aligned} \tag{G-52}$$

$$\begin{aligned}\boldsymbol{Z}_{5,2} &= \left(\frac{\partial}{\partial\boldsymbol{\eta}_{1e}}\left(\frac{\partial \mathbf{M}_4}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}}\right)\boldsymbol{\vartheta}_1\\ &= \Bigg(\left(-\frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}\mathbf{D}_1\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2} + \frac{\hat{\boldsymbol{v}}_1^{\mathrm{T}}\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^3}\right)\mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e})\begin{bmatrix}\boldsymbol{\vartheta}_{1,1} & 0 & 0\\ 0 & \boldsymbol{\vartheta}_{1,2} & 0\\ 0 & 0 & \boldsymbol{\vartheta}_{1,3}\end{bmatrix}\\ &\quad + \mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e}) * \frac{\boldsymbol{\vartheta}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^3}\hat{\boldsymbol{v}}_1^{\mathrm{T}}\mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e}) * \begin{bmatrix}\hat{\boldsymbol{v}}_{1,1} & 0 & 0\\ 0 & \hat{\boldsymbol{v}}_{1,2} & 0\\ 0 & 0 & \hat{\boldsymbol{v}}_{1,3}\end{bmatrix}\\ &\quad + \frac{\boldsymbol{\vartheta}_1^{\mathrm{T}}\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)^3}\mathbf{K}_1\sigma'''(\boldsymbol{\eta}_{1e}) * \begin{bmatrix}\hat{\boldsymbol{v}}_{1,1} & 0 & 0\\ 0 & \hat{\boldsymbol{v}}_{1,2} & 0\\ 0 & 0 & \hat{\boldsymbol{v}}_{1,3}\end{bmatrix} * \begin{bmatrix}\hat{\boldsymbol{v}}_{1,1} & 0 & 0\\ 0 & \hat{\boldsymbol{v}}_{1,2} & 0\\ 0 & 0 & \hat{\boldsymbol{v}}_{1,3}\end{bmatrix}\\ &\quad + \begin{bmatrix}\hat{\boldsymbol{v}}_{1,1} & 0 & 0\\ 0 & \hat{\boldsymbol{v}}_{1,2} & 0\\ 0 & 0 & \hat{\boldsymbol{v}}_{1,3}\end{bmatrix} * \mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e})\frac{\hat{\boldsymbol{v}}_1\boldsymbol{\vartheta}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^3}\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\Bigg)\end{aligned} \tag{G-53}$$

$$\boldsymbol{Z}_{5,3} = \begin{bmatrix}\boldsymbol{\vartheta}_{1,1} & 0 & 0\\ 0 & \boldsymbol{\vartheta}_{1,2} & 0\\ 0 & 0 & \boldsymbol{\vartheta}_{1,3}\end{bmatrix} \tag{G-54}$$

$$\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\eta}_{1e}} = -\left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} \left(\frac{(2\boldsymbol{\alpha}_1 + \boldsymbol{\nu}_{1e})\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1 + \|\boldsymbol{\nu}_{1e}\|^2} - \mathbf{I}_{3\times3}\right) \mathbf{K}_1 \frac{\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} \tag{G-55}$$

$$\boldsymbol{\vartheta}_1 = m\mathbf{N}^{\mathrm{T}}\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_e, \tag{G-56}$$

$$\begin{aligned}
\mathbf{Z}_{6,0} &= \frac{\partial}{\partial \boldsymbol{\nu}_{1e}} \left(\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \begin{bmatrix} \boldsymbol{\vartheta}_{1,1} & 0 & 0 \\ 0 & \boldsymbol{\vartheta}_{1,2} & 0 \\ 0 & 0 & \boldsymbol{\vartheta}_{1,3} \end{bmatrix} \\
&= \frac{(2c + \gamma_1\lambda_m(\mathbf{D}_1))\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{4(\Delta_1(\hat{\boldsymbol{\nu}}_1))^2\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}} \left(\boldsymbol{\vartheta}_1\boldsymbol{\nu}_{1e}^{\mathrm{T}} - (\boldsymbol{\alpha}_1^{\mathrm{T}} + \hat{\boldsymbol{\nu}}_1^{\mathrm{T}})\boldsymbol{\vartheta}_1\left(\mathbf{I}_{3\times3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1 + \|\boldsymbol{\nu}_{1e}\|^2}\right)\right)
\end{aligned} \tag{G-57}$$

$$\begin{aligned}
\mathbf{Z}_{6,1} &= \frac{\partial}{\partial \hat{\boldsymbol{\nu}}_1} \left(\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \begin{bmatrix} \boldsymbol{\vartheta}_{1,1} & 0 & 0 \\ 0 & \boldsymbol{\vartheta}_{1,2} & 0 \\ 0 & 0 & \boldsymbol{\vartheta}_{1,3} \end{bmatrix} \\
&= \mathbf{Z}_{6,0}\mathbf{G}_1 + \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{(\Delta_1(\hat{\boldsymbol{\nu}}_1))^2} \sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2} \\
&\quad * \left(\left(\frac{\boldsymbol{\nu}_{1e}(\boldsymbol{\alpha}_1^{\mathrm{T}} + \hat{\boldsymbol{\nu}}_1^{\mathrm{T}})}{1 + \|\boldsymbol{\nu}_{1e}\|^2} - \mathbf{I}_{3\times3}\right) \frac{\boldsymbol{\vartheta}_1\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\vartheta}_1^{\mathrm{T}}(2\mathbf{I}_{3\times3} - \mathbf{G}_1)}{1 + \|\boldsymbol{\nu}_{1e}\|^2}\right)
\end{aligned} \tag{G-58}$$

$$\begin{aligned}
\mathbf{Z}_{6,2} &= \frac{\partial}{\partial \boldsymbol{\eta}_{1e}} \left(\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \begin{bmatrix} \boldsymbol{\vartheta}_{1,1} & 0 & 0 \\ 0 & \boldsymbol{\vartheta}_{1,2} & 0 \\ 0 & 0 & \boldsymbol{\vartheta}_{1,3} \end{bmatrix} \\
&= \mathbf{Z}_{6,0} \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} - \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}{2(\Delta_1(\hat{\boldsymbol{\nu}}_1))^2} \mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e}) \\
&\quad * \left(\frac{(\boldsymbol{\alpha}_1^{\mathrm{T}} + \hat{\boldsymbol{\nu}}_1^{\mathrm{T}})\boldsymbol{\vartheta}_1}{1 + \|\boldsymbol{\nu}_{1e}\|^2} \begin{bmatrix} \boldsymbol{\nu}_{1e,1} & 0 & 0 \\ 0 & \boldsymbol{\nu}_{1e,2} & 0 \\ 0 & 0 & \boldsymbol{\nu}_{1e,3} \end{bmatrix} - \begin{bmatrix} \boldsymbol{\vartheta}_{1,1} & 0 & 0 \\ 0 & \boldsymbol{\vartheta}_{1,2} & 0 \\ 0 & 0 & \boldsymbol{\vartheta}_{1,3} \end{bmatrix}\right) \\
&\quad + \left(c + \gamma_1 \frac{\lambda_m(\mathbf{D}_1)}{2}\right) \frac{\sqrt{1 + \|\boldsymbol{\nu}_{1e}\|^2}}{2(\Delta_1(\hat{\boldsymbol{\nu}}_1))^3} \mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e}) \left(\frac{\boldsymbol{\nu}_{1e}\boldsymbol{\vartheta}_1^{\mathrm{T}}}{1 + \|\boldsymbol{\nu}_{1e}\|^2} - \mathbf{I}_{3\times3}\right) \mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})
\end{aligned} \tag{G-59}$$

$$\begin{aligned}
\mathbf{Z}_{6,3} &= \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}} \left(\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \begin{bmatrix} \boldsymbol{\vartheta}_{1,1} & 0 & 0 \\ 0 & \boldsymbol{\vartheta}_{1,2} & 0 \\ 0 & 0 & \boldsymbol{\vartheta}_{1,3} \end{bmatrix} \\
&= -\mathbf{Z}_{6,0}
\end{aligned} \tag{G-60}$$

$$\frac{\partial \mathbf{M}_6}{\partial \boldsymbol{\eta}_{1e}}\dot{\boldsymbol{\eta}}_{1e} = \frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)}\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)} + \frac{\mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e})}{\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^2}\mathbf{K}_1\begin{bmatrix}\sigma(\boldsymbol{\eta}_{1e,1}) & 0 & 0\\ 0 & \sigma(\boldsymbol{\eta}_{1e,2}) & 0\\ 0 & 0 & \sigma(\boldsymbol{\eta}_{1e,3})\end{bmatrix}, \tag{G-61}$$

$$\boldsymbol{\vartheta}_1 = \mathbf{N}^{\mathrm{T}}\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}, \tag{G-62}$$

$$\begin{aligned}\mathbf{Z}_{7,1} &= \frac{\partial}{\partial \hat{\boldsymbol{v}}_1}\left(\frac{\partial \mathbf{M}_6}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}}\begin{bmatrix}\boldsymbol{\vartheta}_{1,1} & 0 & 0\\ 0 & \boldsymbol{\vartheta}_{1,2} & 0\\ 0 & 0 & \boldsymbol{\vartheta}_{1,3}\end{bmatrix}\\ &= -\mathbf{K}_1\left(\sigma'(\boldsymbol{\eta}_{1e})\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e}) + \sigma''(\boldsymbol{\eta}_{1e})\mathbf{K}_1\begin{bmatrix}\sigma(\boldsymbol{\eta}_{1e,1}) & 0 & 0\\ 0 & \sigma(\boldsymbol{\eta}_{1e,2}) & 0\\ 0 & 0 & \sigma(\boldsymbol{\eta}_{1e,3})\end{bmatrix}\right)\frac{\boldsymbol{\vartheta}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^3}\end{aligned} \tag{G-63}$$

$$\begin{aligned}\mathbf{Z}_{7,2} &= \frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{\partial \mathbf{M}_6}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}}\begin{bmatrix}\boldsymbol{\vartheta}_{1,1} & 0 & 0\\ 0 & \boldsymbol{\vartheta}_{1,2} & 0\\ 0 & 0 & \boldsymbol{\vartheta}_{1,3}\end{bmatrix}\\ &= \left(3\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})\frac{\mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e})}{\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^2} + \frac{\mathbf{K}_1\sigma'''(\boldsymbol{\eta}_{1e})}{\left(\Delta_1(\hat{\boldsymbol{v}}_1)\right)^2}\mathbf{K}_1\begin{bmatrix}\sigma(\boldsymbol{\eta}_{1e,1}) & 0 & 0\\ 0 & \sigma(\boldsymbol{\eta}_{1e,2}) & 0\\ 0 & 0 & \sigma(\boldsymbol{\eta}_{1e,3})\end{bmatrix}\right)\\ &\quad * \begin{bmatrix}\boldsymbol{\vartheta}_{1,1} & 0 & 0\\ 0 & \boldsymbol{\vartheta}_{1,2} & 0\\ 0 & 0 & \boldsymbol{\vartheta}_{1,3}\end{bmatrix}\end{aligned} \tag{G-64}$$

$$\begin{aligned}\mathbf{Z}_{7,3} &= \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{M}_6}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}}\begin{bmatrix}\boldsymbol{\vartheta}_{1,1} & 0 & 0\\ 0 & \boldsymbol{\vartheta}_{1,2} & 0\\ 0 & 0 & \boldsymbol{\vartheta}_{1,3}\end{bmatrix}\\ &= \mathbf{0}_{3x3}\end{aligned} \tag{G-65}$$

$$\frac{\partial \mathbf{M}_7}{\partial \boldsymbol{\eta}_{1e}} = -\frac{2\gamma_1 m}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\frac{\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\left(\sigma'(\boldsymbol{\eta}_{1e})\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \sigma''(\boldsymbol{\eta}_{1e})\begin{bmatrix}\boldsymbol{\nu}_{1e,1} & 0 & 0\\ 0 & \boldsymbol{\nu}_{1e,2} & 0\\ 0 & 0 & \boldsymbol{\nu}_{1e,3}\end{bmatrix}\right) \quad \text{(G-66)}$$

$$\boldsymbol{\vartheta}_2 = \mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_e, \quad \text{(G-67)}$$

$$\begin{aligned} \mathrm{Z}_{8,0} &= \frac{\partial}{\partial \boldsymbol{\nu}_{1e}}\left(\frac{\partial \mathbf{M}_7}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}}\begin{bmatrix}\boldsymbol{\vartheta}_{2,1} & 0 & 0\\ 0 & \boldsymbol{\vartheta}_{2,2} & 0\\ 0 & 0 & \boldsymbol{\vartheta}_{2,3}\end{bmatrix} \\ &= m\frac{2\gamma_1\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)(1+\|\boldsymbol{\nu}_{1e}\|^2)^{1.5}}\left(\sigma'(\boldsymbol{\eta}_{1e})\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{2\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \frac{\sigma''(\boldsymbol{\eta}_{1e})}{2}\begin{bmatrix}\boldsymbol{\nu}_{1e,1} & 0 & 0\\ 0 & \boldsymbol{\nu}_{1e,2} & 0\\ 0 & 0 & \boldsymbol{\nu}_{1e,3}\end{bmatrix}\right)\boldsymbol{\vartheta}_2\boldsymbol{\nu}_{1e}^{\mathrm{T}} \\ &\quad + m\frac{2\gamma_1\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)(1+\|\boldsymbol{\nu}_{1e}\|^2)^{1.5}}\left(\sigma'(\boldsymbol{\eta}_{1e})\left(\boldsymbol{\nu}_{1e}\left(\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\boldsymbol{\vartheta}_2\right)^{\mathrm{T}} + \left(\boldsymbol{\nu}_{1e}^{\mathrm{T}}\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}d\boldsymbol{\eta}_{1e}\right)\left(\mathbf{I}_{3x3} - 2\frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\right) - (1+\|\boldsymbol{\nu}_{1e}\|^2)\sigma''(\boldsymbol{\eta}_{1e})\begin{bmatrix}\boldsymbol{\vartheta}_{2,1} & 0 & 0\\ 0 & \boldsymbol{\vartheta}_{2,2} & 0\\ 0 & 0 & \boldsymbol{\vartheta}_{2,3}\end{bmatrix}\right), \end{aligned} \quad \text{(G-68)}$$

$$\begin{aligned} \mathbf{Z}_{8,1} &= \frac{\partial}{\partial \hat{\boldsymbol{\nu}}_1}\left(\frac{\partial \mathbf{M}_7}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}}\begin{bmatrix}\boldsymbol{\vartheta}_{2,1} & 0 & 0\\ 0 & \boldsymbol{\vartheta}_{2,2} & 0\\ 0 & 0 & \boldsymbol{\vartheta}_{2,3}\end{bmatrix} \\ &= \mathbf{Z}_{8,0}\mathbf{G}_1 + m\frac{2\gamma_1\mathbf{K}_1}{\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}\left(\sigma'(\boldsymbol{\eta}_{1e})\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\left(\frac{1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}+1\right)\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e}) + \sigma''(\boldsymbol{\eta}_{1e})\begin{bmatrix}\boldsymbol{\nu}_{1e,1} & 0 & 0\\ 0 & \boldsymbol{\nu}_{1e,2} & 0\\ 0 & 0 & \boldsymbol{\nu}_{1e,3}\end{bmatrix}\right)\frac{\boldsymbol{\vartheta}_2\hat{\boldsymbol{\nu}}_1^{\mathrm{T}}}{\left(\Delta_1(\hat{\boldsymbol{\nu}}_1)\right)^2} \end{aligned} \quad \text{(G-69)}$$

$$\mathbf{Z}_{8,2} = \frac{\partial}{\partial \boldsymbol{\eta}_{1e}}\left(\frac{\partial \mathbf{M}_7}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix} \boldsymbol{\vartheta}_{2,1} & 0 & 0 \\ 0 & \boldsymbol{\vartheta}_{2,2} & 0 \\ 0 & 0 & \boldsymbol{\vartheta}_{2,3} \end{bmatrix}$$
$$= \mathbf{Z}_{8,0}\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} - m\frac{2\gamma_1\mathbf{K}_1}{\Delta_1(\hat{\boldsymbol{\nu}}_1)\sqrt{1+\|\boldsymbol{\nu}_{1e}\|^2}}$$
$$* \left(\sigma'(\boldsymbol{\eta}_{1e})\left(\mathbf{I}_{3x3} - \frac{\boldsymbol{\nu}_{1e}\boldsymbol{\nu}_{1e}^{\mathrm{T}}}{1+\|\boldsymbol{\nu}_{1e}\|^2}\right)\frac{\mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)} + \sigma'''(\boldsymbol{\eta}_{1e})\begin{bmatrix} \boldsymbol{\nu}_{1e,1} & 0 & 0 \\ 0 & \boldsymbol{\nu}_{1e,2} & 0 \\ 0 & 0 & \boldsymbol{\nu}_{1e,3} \end{bmatrix}\right)$$
$$* \begin{bmatrix} \boldsymbol{\vartheta}_{2,1} & 0 & 0 \\ 0 & \boldsymbol{\vartheta}_{2,2} & 0 \\ 0 & 0 & \boldsymbol{\vartheta}_{2,3} \end{bmatrix} + m\frac{2\gamma_1\mathbf{K}_1\sigma''(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)(1+\|\boldsymbol{\nu}_{1e}\|^2)^{1.5}}\left(\boldsymbol{\nu}_{1e}^{\mathrm{T}}\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{\nu}}_1)}\boldsymbol{\vartheta}_2\right)$$
$$* \begin{bmatrix} \boldsymbol{\nu}_{1e,1} & 0 & 0 \\ 0 & \boldsymbol{\nu}_{1e,2} & 0 \\ 0 & 0 & \boldsymbol{\nu}_{1e,3} \end{bmatrix} \tag{G-70}$$

$$\mathbf{Z}_{8,3} = \frac{\partial}{\partial \dot{\boldsymbol{\eta}}_{1d}}\left(\frac{\partial \mathbf{M}_7}{\partial \boldsymbol{\eta}_{1e}}\right)\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}$$
$$= -\mathbf{Z}_{8,0} \tag{G-71}$$

$$m\left(\frac{\partial \mathbf{M}_1}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_2}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_3}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_4}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_6}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}} \mathbf{N}^{\mathrm{T}}\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}} \tag{G-72}$$

$$\boldsymbol{\vartheta}_2 = m\mathbf{A}_{2,d}^{\mathrm{T}}\mathbf{T}(\boldsymbol{\alpha}_2)^{-\mathrm{T}}\mathbf{R}_2^{\mathrm{T}}(\boldsymbol{\alpha}_q)\boldsymbol{q}_{\boldsymbol{e}}, \tag{G-73}$$

$$\begin{aligned}\boldsymbol{Z}_{9,1}\dot{\hat{\boldsymbol{v}}}_1 &= m\left(\frac{\partial \mathbf{M}_1}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_2}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_3}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_4}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_6}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}}\left(\frac{\partial \mathbf{N}^{\mathrm{T}}}{\partial \hat{\boldsymbol{v}}_1}\right)\begin{bmatrix}\boldsymbol{\vartheta}_{1,1} & 0 & 0\\ 0 & \boldsymbol{\vartheta}_{1,2} & 0\\ 0 & 0 & \boldsymbol{\vartheta}_{1,3}\end{bmatrix}\dot{\hat{\boldsymbol{v}}}_1\\ &= \left(\frac{\partial \mathbf{M}_1}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_2}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_3}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_4}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_6}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}}\\ &\quad * \frac{\boldsymbol{\vartheta}_1^{\mathrm{T}}\mathbf{K}_1\sigma(\boldsymbol{\eta}_{1e})\mathbf{N}^{\mathrm{T}}}{1-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}}\left(\frac{\mathbf{I}_{3x3}}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}-2\frac{\hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^{\mathrm{T}}}{\Delta_1(\hat{\boldsymbol{v}}_1)^3}\right)\dot{\hat{\boldsymbol{v}}}_1\end{aligned} \tag{G-74}$$

$$\begin{aligned}\boldsymbol{Z}_{9,2}\dot{\boldsymbol{\eta}}_{1e} &= \left(\frac{\partial \mathbf{M}_1}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_2}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_3}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_4}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_6}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}}\left(\frac{\partial \mathbf{N}^{\mathrm{T}}}{\partial \boldsymbol{\eta}_{1e}}\right)\begin{bmatrix}\boldsymbol{\vartheta}_{1,1} & 0 & 0\\ 0 & \boldsymbol{\vartheta}_{1,2} & 0\\ 0 & 0 & \boldsymbol{\vartheta}_{1,3}\end{bmatrix}\dot{\boldsymbol{\eta}}_{1e}\\ &= \left(\frac{\partial \mathbf{M}_1}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_2}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_3}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_4}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_5}{\partial \boldsymbol{\eta}_{1e}}+\frac{\partial \mathbf{M}_6}{\partial \boldsymbol{\eta}_{1e}}\right)^{\mathrm{T}}\\ &\quad * \frac{\hat{\boldsymbol{v}}_1\boldsymbol{\vartheta}_1^{\mathrm{T}}\mathbf{N}}{1-\sigma^T(\boldsymbol{\eta}_{1e})\mathbf{K}_1\frac{\hat{\boldsymbol{v}}_1}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}}\frac{\mathbf{K}_1\sigma'(\boldsymbol{\eta}_{1e})}{\Delta_1(\hat{\boldsymbol{v}}_1)^2}\dot{\boldsymbol{\eta}}_{1e}\end{aligned} \tag{G-75}$$

$$\boldsymbol{Z}_{9,3}\ddot{\boldsymbol{\eta}}_{1d} = \begin{bmatrix}\boldsymbol{\vartheta}_{1,1} & 0 & 0\\ 0 & \boldsymbol{\vartheta}_{1,2} & 0\\ 0 & 0 & \boldsymbol{\vartheta}_{1,3}\end{bmatrix}\ddot{\boldsymbol{\eta}}_{1d} \tag{G-76}$$

Appendix H –Simulation Code, Deterministic Backstepping Controller for Quadrotor Attitude Control

```
clc
clear all
close all
End_Time                   = 10;
Delta_Time                 = 1/1000;
Controller_update_frequancy = 1/1000;
Controller_update_time = Controller_update_frequancy /Delta_Time;
Controller_time_steps = Controller_update_time ;
Delta_OMEGA                = 0;
%% model paramters physical
Copter_Radius              = 0.30;
Copter_Radius_Along_X_Axis  = 0.26;
Copter_Radius_Along_Y_Axis  = 0.15;
g   = 9.81;
IXX =     0.0823;
IYY =     0.0539;
IZZ =     0.2169;
m = 2.23;
JTP =  2.5172e-006;
Motor_radius = 20/1000                          ;
copterradius = Copter_Radius;
Two_pi = 2*pi;
I_H        = [ IXX, 0, 0;
               0, IYY, 0;
               0, 0, IZZ];
%% mathamtical deffinitions
T_inverse_Phi_derivative    = zeros(3, 3);
T_inverse_Theta_derivative  = zeros(3, 3);
T_inverse                   = zeros(3, 3);
T_inverse_inverse           = zeros(3, 3);
E_Frame_Linear_possition       = zeros(3, 1);
E_Frame_Linear_velocity        = zeros(3, 1);
angles                      = zeros(3, 1);
temp                        = zeros(2, 1);
Possition                   = zeros(6, 1);
 Zeta                        =  zeros(6, 1);
Time_steps                  = End_Time / Delta_Time;
results                  = zeros(22, Time_steps);
Increment_Refference_Time = 0;
Refference_Signal_Time_Steps =20;
Refference_Signal_Time_Steps_Inverse = 20* End_Time;
Input                    = zeros(4, Time_steps);
Input_rate               = zeros(4, Time_steps);
Refference_signal                     = zeros(6, Refference_Signal_Time_Steps );
E_Frame_angulare_velocity   = zeros(3,1);
```

```
B_Frame_angulare_velocity   = zeros(3,1);
Loop_Count              = 1;
loop_count_2 = 10;

xi_1 = [0; 0; 0];
K_1 = [ 10 0 0;
        0 10 0;
        0 0 2 ] ;
K_2 = [ 10 0 0;
        0 10 0;
        0 0 5] ;
K_d = 7*[ 10 0 0;
        0 10 0;
        0 0 10]
Epsilon_1 = 20
Epsilon_2 = 2
sigma_1 = 10^0
Tor_aero_dot_max = 20
gamma_1 = 2 * ( min(eig(K_d - 0.25*norm(inv(I_H )*inv(I_H))*eye(3)/Epsilon_2/sigma_1)) -
Epsilon_1)
a = [ 1,min(eig(K_1)), min(eig(K_2))-Epsilon_2, gamma_1/2];
b = [ 0.5 , 0.5*sigma_1 ];
c = min(a) /max(b)
lambda = sigma_1 * (Tor_aero_dot_max^2)/(4*Epsilon_1)

lambda / c

Possition = [0;0;0; 0;-pi/8;-pi/3]

for  t = 0: Delta_Time: End_Time
Desired_Possition(4,1) =(pi/18)* sin(1*t) ;
Desired_Possition(5,1) = -(pi/18)* sin(1*t) ;
Desired_Possition(6,1) =0;% 0.1*t ;
angles = [  Possition(4,1);
            Possition(5,1);
            Possition(6,1) ];
   phi     = Possition(4,1);
   theta   = Possition(5,1);
   R       = rotation(angles);
   E_Frame_angulare_velocity    =
B_FRAME_ANGULARE_Velocity_TO_E_FRAME_Angulare_velocity(B_Frame_angulare_velocity,
angles);
   Position_ERROR      = (Possition - Desired_Possition );
   eta_2_d_dot         = [ (pi/18)* cos(1*t) ;
                          -(pi/18)* cos(1*t) ;
                             0];

   eta_2_d             = [ Desired_Possition(4,1);
                           Desired_Possition(5,1);
                           Desired_Possition(6,1) ];
   eta_2_d_double_dot    = [ -(pi/18)* sin(1*t) ;
                          (pi/18)* sin(1*t) ;
                             0];
   %% Second stage set up converssions between frames of refference
   eta_2_error              =  [ Position_ERROR(4,1);
                                 Position_ERROR(5,1);
                                 Position_ERROR(6,1)];
   x_2                  = E_Frame_Linear_velocity;
   eta_2                = angles ;
   control_time = t ;
```

```
  if  (Controller_time_steps == Controller_update_time )
      U = Attitude_Backstepping_Controller_1(I_H,  K_1, K_2, K_d, eta_2,
B_Frame_angulare_velocity, eta_2_d, eta_2_d_dot, eta_2_d_double_dot,
Controller_update_frequancy, xi_1,Loop_Count);
      Controller_time_steps = 0;
      xi_1 = [U(1,1) ; U(2,1); U(3,1)];
      tor_aero_hat = [U(7,1) ; U(8,1); U(9,1)];
      alpha_w = [U(10,1) ; U(11,1); U(12,1)];
      u =[ 0 ; U(4,1); U(5,1); U(6,1) ];
      u(1,1)=0;
      U=u;
  end
  w_error                 =   B_Frame_angulare_velocity - alpha_w;
  Control_Input = u;
  loop_count_2 = -1;
%% disturbance dynamics
        Aero_Disturbance = [0;0;0;0.1*sin(t); 0.1*cos(t);0.1*tanh(t)];

        tau_error = [Aero_Disturbance(4,1); Aero_Disturbance(5,1); Aero_Disturbance(6,1)] -
tor_aero_hat;

        Collision_Disturbance =[0;0;0;0;0;0];
%%      // 6 dof dynamics
        Zeta_dot = SIX_DOF_Dynamics(R, E_Frame_angulare_velocity, angles,g,m,I_H,IXX,IYY,IZZ,
Aero_Disturbance, Collision_Disturbance, Control_Input, JTP, Zeta );
      B_Frame_angulare_acceleration       = [Zeta_dot(4,1); Zeta_dot(5,1); Zeta_dot(6,1)];
      E_Frame_Linear_acceleration         = [Zeta_dot(1,1); Zeta_dot(2,1); Zeta_dot(3,1)];
      E_Frame_Linear_acceleration(3,1)    = 0;
      B_Frame_angulare_velocity           = B_Frame_angulare_velocity + Delta_Time *
B_Frame_angulare_acceleration;
      E_Frame_angulare_velocity           =
B_FRAME_ANGULARE_Velocity_TO_E_FRAME_Angulare_velocity(B_Frame_angulare_velocity,
angles);
      angles                              = angles + Delta_Time * E_Frame_angulare_velocity;
      E_Frame_Linear_velocity             = E_Frame_Linear_velocity + Delta_Time *
E_Frame_Linear_acceleration;
      E_Frame_Linear_possition            = E_Frame_Linear_possition + Delta_Time *
E_Frame_Linear_velocity;
      eta_2_error = eta_2 - eta_2_d;
      if Loop_Count ==1
          eta_2_error_0 = eta_2_error;
      w_error_0 = w_error;
      tau_error_0 = tau_error;
      end

      V_sum_bound = ((norm(eta_2_error_0))^2 + norm(w_error_0)^2 + 0.5*sigma_1 *
norm(tau_error_0)^2 -  lambda/c)*exp(-c * t) +   lambda / c;
      V_sum = (norm(eta_2 - eta_2_d))^2 +   (norm(w_error))^2 + 0.5*sigma_1 * (norm(tau_error))^2;
       Lyapunov_bound(1,Loop_Count) = V_sum_bound;
       Lyapunov_bound(2,Loop_Count) = V_sum;

      if (E_Frame_Linear_possition(3,1) <= 0)
          E_Frame_Linear_possition(3,1) =0;
      end

      Zeta  = [ E_Frame_Linear_acceleration;
                B_Frame_angulare_velocity          ];

Possition_OLD   = Possition;
      Possition = [ E_Frame_Linear_possition(1);
```

```
                                E_Frame_Linear_possition(2);
                                E_Frame_Linear_possition(3);
                                angles(1);
                                angles(2);
                                angles(3) ];
%% function to plot the refference signal
   Input(:,Loop_Count) = Control_Input;
   if Loop_Count > 1
      Input_rate(:,Loop_Count)  = (Input(:,Loop_Count) - Input(:,(Loop_Count-1))) / Delta_Time;
   end
   if Loop_Count == 1
      Input_rate(:,Loop_Count)  = 0;
   end
   results(:,Loop_Count) = [ Possition;
                                   t ;
                                   B_Frame_angulare_velocity ;
                                   tor_aero_hat;
                                   w_error;
                                    eta_2_error;
                                    tau_error];
for i = 1:3
   Refference_signal(i,Loop_Count) = Desired_Possition(i,1) ;
end
Refference_signal(4,Loop_Count) = eta_2_d(1,1);
Refference_signal(5,Loop_Count) = eta_2_d(2,1);
Refference_signal(6,Loop_Count) = eta_2_d(3,1);
Loop_Count             = Loop_Count + 1;
Controller_time_steps = Controller_time_steps + 1;
%% END OF SIMULATION LOOP%%
t
loop_count_2 = loop_count_2 + 1;
end
Input(:,2)  =[0;0;0;0];
Input(:,1) =Input(:,2) ;

figure
get(0,'Factory');
set(0,'defaultfigurecolor',[1 1 1]);
 plot(results(7,:) , Lyapunov_bound(1,:),'b', ...
               'LineWidth',2)
 set(title({'V _3 and V_{3,bound} Vs Time'}, 'FontSize',18))
 hold on
plot(results(7,:) , Lyapunov_bound(2,:),'g', ...
               'LineWidth',2)
 hold on
plot(results(7,:) , Lyapunov_bound(1,:),'b', ...
               'LineWidth',2)
hold on
legend({'V_{3,bound}','V_3'}, 'FontSize',16)
set(xlabel({'Time(s)'}, 'FontSize',14))
set(ylabel({'Magnitude'}, 'FontSize',14))
hold off

Rad_to_Deg = 180/pi;
figure
plot(results(7,:) ,Rad_to_Deg *results(4,:), ...
               'LineWidth',3)
set(title({'Attitude  Vs Time'}, 'FontSize',18))
hold on
plot(results(7,:) ,Rad_to_Deg *results(5,:),'g', ...
```

```
            'LineWidth',3)
hold on
plot(results(7,:) ,Rad_to_Deg *results(6,:),'r', ...
            'LineWidth',3)

plot(results(7,:) ,Rad_to_Deg *Refference_signal(4,:),'-b' , ...
            'LineWidth',1)

         hold on
plot(results(7,:) ,Rad_to_Deg *Refference_signal(5,:),'-g' , ...
            'LineWidth',1)
         hold on
plot(results(7,:) ,Rad_to_Deg *Refference_signal(6,:),'-r' , ...
            'LineWidth',1)
axis([0, End_Time, - 60, 20])
legend({'\phi','\theta','\psi','\phi_d','\theta_d','\psi_d'}, 'FontSize',16)
set(xlabel({'Time(s)'}, 'FontSize',14))
set(ylabel({'Attitude(Degrees)'}, 'FontSize',14))
hold off


figure
plot(results(7,:) ,results(20,:), ...
            'LineWidth',2)
 set(title({'Disturbance observer error   Vs Time'}, 'FontSize',18))
hold on
plot(results(7,:) ,results(21,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(22,:),'r', ...
            'LineWidth',2)
 set(legend({'$\widetilde{\tau}_p$','$\widetilde{\tau}_q$','$\widetilde{\tau}_r$'},
'FontSize',20),'interpreter','latex')
set(xlabel({'Time(s)'}, 'FontSize',14))
set(ylabel({'Observer Error (Nm)'}, 'FontSize',14))
hold off


figure
plot(results(7,:) ,Rad_to_Deg*results(17,:),'b', ...
            'LineWidth',2)
set(title({'Attitude error  Vs Time'}, 'FontSize',18))

hold on
plot(results(7,:) ,Rad_to_Deg*results(18,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,Rad_to_Deg*results(19,:),'r', ...
            'LineWidth',2)
legend({'\phi _e','\theta_e','\psi_e'}, 'FontSize',16)
set(xlabel({'Time(s)'}, 'FontSize',14))
set(ylabel({'Attitude Error (Degrees)'}, 'FontSize',14))
hold off
```

```
figure
plot(results(7,:) ,Input(2,:) ,'b', ...
               'LineWidth',2)
set(title({'Control Signal Vs Time'}, 'FontSize',18))
hold on
plot(results(7,:) ,Input(3,:) ,'g', ...
               'LineWidth',2)
hold on
plot(results(7,:) ,Input(4,:) ,'r', ...
               'LineWidth',2)
legend({'U_2','U_3','U_4'}, 'FontSize',14)
set(xlabel({'Time(s)'}, 'FontSize',14))
set(ylabel({'Control input (Nm)'}, 'FontSize',14))
hold off
 figure
plot(results(7,:) ,results(11,:), ...
               'LineWidth',2)
set(title({'Disturbance observer   Vs Time'}, 'FontSize',14))
hold on
plot(results(7,:) ,results(12,:),'g', ...
               'LineWidth',2)
hold on
plot(results(7,:) ,results(13,:),'r', ...
               'LineWidth',2)
 set(legend({'$\hat{\tau}_p$','$\hat{\tau}_q$','$\hat{\tau}_r$'}, 'FontSize',20),'interpreter','latex')
set(xlabel({'Time(s)'}, 'FontSize',14))
set(ylabel({'Distrubance Torque Estimate (Nm)'}, 'FontSize',14))
hold off

%% check initial roll refference as it is 90 degrees


function U = Attitude_Backstepping_Controller_1(I_H,  K_1, K_2, K_d, eta_2,
B_Frame_angulare_velocity, eta_2_d, eta_2_d_dot, eta_2_d_double_dot, Delta_Time, xi_1,
Loop_Count)
angles                  = eta_2;
T_inv                   = Angular_velocity_cordinant_transform(angles);
T                       = inv(T_inv);
eta_2_dot                 = T * B_Frame_angulare_velocity    ;
eta_2_error              = eta_2 - eta_2_d;
alpha_w                  = T_inv * ( eta_2_d - K_1 * eta_2_error);
w_error                  =   B_Frame_angulare_velocity - alpha_w;
if Loop_Count ==1;
    xi_1=- K_d * I_H * w_error;
end
tor_aero_hat             = xi_1+ K_d * I_H * w_error;
eta_2_error_dot          = K_1 * eta_2_error + T * w_error;
alpha_w_dot              = Calculate_alpha_w_dot_Attitude_Only(K_1, eta_2, eta_2_error, eta_2_d_dot,
eta_2_dot, eta_2_error_dot, eta_2_d_double_dot);
tor                      = I_H*( -T * eta_2_error   -K_2 * w_error + alpha_w_dot ) - tor_aero_hat +
cross(B_Frame_angulare_velocity, (I_H*B_Frame_angulare_velocity));
xi_1_dot                 = -K_d *( -cross(B_Frame_angulare_velocity,
(I_H*B_Frame_angulare_velocity))+tor +tor_aero_hat);
xi_1                     = xi_1_dot * Delta_Time + xi_1;
U                        = [xi_1; tor(1,1); tor(2,1) ; tor(3,1) ; tor_aero_hat; alpha_w];
```

```
function alpha_w_dot             = Calculate_alpha_w_dot_Attitude_Only(K_1, eta_2, eta_2_error,
eta_2_d_dot, eta_2_dot, eta_2_error_dot, eta_2_d_double_dot)
phi = eta_2(1,1);
theta = eta_2(2,1);
PHI_DOT = eta_2_dot(1,1);
THETA_DOT = eta_2_dot(2,1);
angles                      = eta_2;
T_inv   = Angular_velocity_cordinant_transform(angles);

T_inverse_Theta_derivative  = [ 0,0 ,-cos(theta);
                                 0,0 , -sin(theta)*sin(phi);
                                 0,0 ,-sin(theta)*cos(phi) ];
T_inverse_Phi_derivative    = [ 0,0 ,0;
                               0,-sin(phi) , cos(theta)*cos(phi);
                               0,-cos(phi) ,-cos(theta)*sin(phi)   ];
T_inv_dot                  = (T_inverse_Phi_derivative*PHI_DOT +
T_inverse_Theta_derivative*THETA_DOT);

alpha_w_dot   = T_inv_dot * (eta_2_d_dot -K_1 * eta_2_error) + T_inv     * (eta_2_d_double_dot -K_1
* eta_2_error_dot


function T_inv          = Angular_velocity_cordinant_transform(angles)
    phi = angles(1);
    theta = angles(2);
    psi = angles(3);
        T_inv =[
        1, 0, -sin(theta)
        0, cos(phi), cos(theta)*sin(phi)
        0, -sin(phi), cos(theta)*cos(phi)
    ];
end
```

```
function  Zeta_dot =SIX_DOF_Dynamics(R, E_Frame_angulare_velocity,
angles,g,m,I_H,IXX,IYY,IZZ, Aero_Disturbance, Collision_Disturbance, Control_Input, JTP,Zeta );
        B_Frame_angulare_velocity =
E_FRAME_ANGULARE_Velocity_TO_B_FRAME_Angulare_velocity(E_Frame_angulare_velocity,
angles);
        p           = B_Frame_angulare_velocity(1,1);
        q           = B_Frame_angulare_velocity(2,1);
        r           = B_Frame_angulare_velocity(3,1);
     EH = [ R, zeros(3,3);
            zeros(3,3), eye(3)];
     MH = [ m * eye(3), zeros(3,3);
            zeros(3,3), I_H        ];
     GH = [0;0; m*g;0;0;0];
     Skew = [  0, IZZ*r, -IYY*q;
              -IZZ*r, 0, IXX*p;
              IYY*q, -IXX*p, 0];
     CH = [ zeros(3,3), zeros(3,3);
            zeros(3,3), Skew       ];
     OH = [ 0, 0, 0, 0 0, 0;
            0, 0, 0, 0 0, 0;
            0, 0, 0, 0 0, 0;
            -p, p, -p, p, -p, p;
            -q, q, -q, q, -q, q;
            0, 0, 0, 0 0, 0 ] ;
     Control_H =  [ 0;
                    0;
                    Control_Input ];
                  Motor_Speed = zeros(6,1);
     Zeta_dot = (inv(MH))*(-CH*Zeta - GH +JTP*OH * Motor_Speed  +EH*Control_H
+Aero_Disturbance + Collision_Disturbance);
 end
```

Appendix I – Simulation Code, Stochastic Backstepping Controller for Quadrotor Attitude Control

```
clc
clear
clear all
close all
%% sumulation time properties
End_Time                  =20;
Delta_Time                = 1/1000;
Time_steps                = End_Time / Delta_Time;
Controller_update_frequancy = 1/1000;
Controller_update_time = Controller_update_frequancy /Delta_Time;
Controller_time_steps = Controller_update_time ;
%% model paramters physical
Copter_Radius              = 0.45;
Copter_Radius_Along_X_Axis  = 0.45*sin(pi/4);
Copter_Radius_Along_Y_Axis  = 0.45*sin(pi/4);
g   = 9.81;
IXX =     0.0823;
IYY =     0.0539;
IZZ =     0.2169;
m = 2.23;
IXX = 0.016507;
IYY = 0.016507;
IZZ = 0.016284
m = 2.23;
JTP =  2.5172e-006;
copterradius = Copter_Radius;
pitch =     4;
Propeller_radius =        (7.8/2)*.75;
alpha = atan(pitch/(2*pi*Propeller_radius ));
alpha = alpha * 180/pi;
Blade_pitch_angle = alpha ;
INCH_to_meter          = 0.0254    ;
BLade_diametre         = 6         ;
L                      = BLade_diametre * INCH_to_meter/2  ;
 p=1.1839; %     wikipedia at 25 degrees C
Blade_leangth          = L;
Effective_blade_leangth = 0.75*Blade_leangth     ;
air_density     =1.1839;
Hex_copter_height   =   1;
I_H         = [ IXX, 0, 0;
                0, IYY, 0;
                0, 0, IZZ];
I_A     = 0.5 * I_H;
%% mathamtical deffinitions
T_inverse_Phi_derivative    = zeros(3, 3);
T_inverse_Theta_derivative  = zeros(3, 3);
```

```
Plot_info   = zeros(6, 1);
T_inverse                  = zeros(3, 3);
T_inverse_inverse          = zeros(3, 3);
%% state vatrable decleration
angles                     = zeros(3, 1);
T_Collision                 = zeros(3, 1);
Possition                  = zeros(6, 1);
Possition_OLD              =  zeros(6, 1);
Desired_Possition_OLD =  zeros(6, 1);
 Motor_Speed               =  zeros(4, 1);
Zeta                       =  zeros(6, 1);
E_Frame_Linear_possition       = zeros(3, 1);
E_Frame_Linear_velocity        = zeros(3, 1);
E_Frame_angulare_velocity   = zeros(3,1);
B_Frame_angulare_velocity   = zeros(3,1);
B_Frame_angulare_velocity_temp     = zeros(3,1);
last_coverience_1_dW                       = zeros(3,1);
last_coverience_1_dW_2                     = zeros(3,1);
Total_Torque_distrubance                  = zeros(3,1);
Stochastic_Torque_distrubance            = zeros(3,1);
coverience_1                              =(0.1)* eye(3);
coverience_2                              = eye(3);
coverience_1_time_transpose                = coverience_1  * coverience_1';
coverience_1_time_trans_inf_norm       = norm(coverience_1_time_transpose,'inf')^2
coverience_2_time_transpose                = coverience_2  * coverience_2';
coverience_2_time_trans_inf_norm       = norm(coverience_2_time_transpose,'inf')^2;
eta_2_d_old                     = [0;0;0];
eta_2_d_dot_old                 = [0;0;0];
%% results gathering deffinitions
Lyapunov_bound = zeros(2, Time_steps);
results                    = zeros(47, Time_steps);
Refference_Signal_Time_Steps =20;
Input                      = zeros(4, Time_steps);
Input_rate                 = zeros(4, Time_steps);
Refference_signal                      = zeros(6, Refference_Signal_Time_Steps );
%% control and observer gain declaration
xi_1 = [0; 0; 0];
K_1 = [ 10 0 0;
        0 10 0;
        0 0 2] ;
K_2 = [10 0 0;
        0 10 0;
        0 0 2] ;
K_d =0* [ 10 0 0;
        0 10 0;
        0 0 10];
Epsilon_1 = .010;
sigma_1 = 5*10^-4;
tau_aero_dot_max = 1
gamma_1 = 2 * (norm(0.5*inv(I_H )*inv(K_2)*inv(I_H)+sigma_1 * K_d) - sigma_1 * Epsilon_1);
a = [ 1,min(eig(K_1)), min(eig(K_2)), gamma_1/2];
b = [ 1 , sigma_1 ];
c = min(a) /max(b);
lambda = sigma_1 * (tau_aero_dot_max^2)/(4*Epsilon_1)
lambda / c;
Control_Input = [0;0;0;0];
Loop_Count              = 1;
```

```
tor_aero_hat_deterministic         = [0 ; 0; 0];
tor_aero_hat_deterministic_dot   = [0 ; 0; 0];
K_d = [ 0.9 0 0;
          0 0.9 0;
          0 0 1];%2];
Possition(4,1) = 0;%-30*pi/180;
Possition(5,1) = 0;%30*pi/180;
D_1 = 0*eye(3);
D_2 = 0*eye(3) ;
K_1 = 10*eye(3);
K_2 =5*eye(3);
ROE =0.15*[50 0 0; 0 50 0; 0 0 50];%K_d;
ROE =5*0.115* ROE;
mew_1 = 10;
mew_2 = 57000000;
tor_aero_deterministic_MAX=[1;1;0.1];
tor_aero_Stochastic_MAX_1=0.1^4;
tor_aero_Stochastic_MAX_2= 1;
xi_aero_deterministic=0.005*[10;10;1];
xi_aero_Stochastic_1=0.1*tor_aero_Stochastic_MAX_1;
xi_aero_Stochastic_2=0.8*tor_aero_Stochastic_MAX_2;
gamma_2 = 0.1;
gamma_1 = 100;
epsilon_1 = 9;
epsilon_2 = 10;
epsilon_3 = 10;
epsilon_4 = 10;
sigma_hat_Stochastic_1 = 0;
sigma_hat_Stochastic_2 = 0;
sigma_hat_Stochastic_dot_1 =0;
sigma_hat_Stochastic_dot_2 = 0;
Possition = [0;0;0; 0;-pi/8;-pi/3]


a = [ min(eig(K_1)) - epsilon_1, min(eig(K_2)), ];
b = [ 1];
c = min(a) /max(b);
tau_error  = tor_aero_deterministic_MAX;
lambda = 1/4/(epsilon_2)+1/4/epsilon_3+1/4/epsilon_4         +   4*tau_error'*(inv(ROE))*tau_error +
4*(0.5/mew_1)*( coverience_1_time_trans_inf_norm-sigma_hat_Stochastic_1)^2  + 4*(0.5/mew_2)*(
coverience_2_time_trans_inf_norm-sigma_hat_Stochastic_2)^2
```

```
for  t = 0: Delta_Time: End_Time

   Desired_Possition(4,1) =(pi/18)*tanh(t)  ;
   Desired_Possition(5,1) = -(pi/18) *tanh(t);
   Desired_Possition(6,1) =-(pi/0.50) *sin(2*pi*0.01*t);% 0.1*t ;
   angles = [  Possition(4,1);
               Possition(5,1);
               Possition(6,1) ];

   E_Frame_angulare_velocity    =
B_FRAME_ANGULARE_Velocity_TO_E_FRAME_Angulare_velocity(B_Frame_angulare_velocity,
angles);
   Velocity              = (Possition-Possition_OLD) / Delta_Time;
   Desired_Velocity       = (Desired_Possition-Desired_Possition_OLD)/Delta_Time;

   Position_ERROR      = (Possition - Desired_Possition );
   Velocity_ERRPR      = (Desired_Possition-Desired_Possition_OLD)/Delta_Time-Velocity;
   eta_2_d             = [ (pi/18)*tanh(t)  ;
                           -(pi/18) *tanh(t);
                           -0.5*(pi/0.50) *sin(2*pi*0.01*t)];
   eta_2_d_dot         = [ (pi/18)*(1 - tanh(t)^2)   ;
                           -(pi/18) *(1 - tanh(t)^2)   ;
                           -0.5*(pi/0.50) *2*pi*0.01*cos(2*pi*0.01*t)];
   eta_2_d_double_dot   =  [ (pi/18)*(0 - 2*tanh(t)*(1 - tanh(t)^2)  )   ;
                             -(pi/18)*(0 - 2*tanh(t)*(1 - tanh(t)^2)  )   ;
                             0.5*(pi/0.50) *2*pi*0.01* 2*pi*0.01 *sin(2*pi*0.01*t)];

   %% Second stage set up converssions between frames of refference
   eta_2                  = angles ;
   eta_2_error  = eta_2 - eta_2_d ;
   control_time = t ;
   q= Euler_to_Modified_rodrigues_paramater( eta_2 );
   alpha_q = Euler_to_Modified_rodrigues_paramater(eta_2_d);
   q_e = q - alpha_q;

   if  (Controller_time_steps == Controller_update_time )
     for i  = 1 :10
          sensor_noise = sin(50*i*t/pi) + sin(50/pi*i*t/2);
      end
      Delta_time = Controller_update_frequancy ;
  U = Attitude_Backstepping_Controller_Stochastic(I_H, I_A, K_1, K_2, K_d, gamma_1, gamma_2,
epsilon_1, epsilon_2, epsilon_3, epsilon_4, D_2, eta_2, B_Frame_angulare_velocity, eta_2_d,
eta_2_d_dot, eta_2_d_double_dot, Delta_time, tor_aero_hat_deterministic,
tor_aero_hat_deterministic_dot, sigma_hat_Stochastic_1,
sigma_hat_Stochastic_2,sigma_hat_Stochastic_dot_1  ,sigma_hat_Stochastic_dot_2, Loop_Count   ,
ROE, tor_aero_deterministic_MAX, tor_aero_Stochastic_MAX_1, tor_aero_Stochastic_MAX_2,
mew_1, mew_2, xi_aero_deterministic, xi_aero_Stochastic_1, xi_aero_Stochastic_2);
      Controller_time_steps = 0;
      tor_aero_hat_deterministic         = [U(10,1) ; U(11,1); U(12,1)];
      tor_aero_hat_deterministic_dot   = [U(13,1) ; U(14,1); U(15,1)];
      sigma_hat_Stochastic_1           = U(16,1) ;
      sigma_hat_Stochastic_dot         = U(17,1) ;
      sigma_hat_Stochastic_2           = U(18,1) ;
      tau_aero_hat  = tor_aero_hat_deterministic;
      alpha_w = [U(4,1) ; U(5,1); U(6,1)];
      alpha_w_dot  = [U(7,1) ; U(8,1); U(9,1)];
      u =[ m*9.81 ; U(1,1); U(2,1); U(3,1) ];
```

```
      U=u;
      w_error                 =   B_Frame_angulare_velocity - alpha_w;
   end

%% motor spped to propeller thrust force and rotation toruqe
   Control_Input  = U;

%% Disturbance dynamics
         Aero_Disturbance = [0;0;0;0;0;0];

   Torque_Disturbance = [ -1;
                          -1;
                          0.1];

          Collision_Disturbance= [ 0;
                                   0;
                                   0;
                                   Torque_Disturbance];
         tau_error = Torque_Disturbance - tor_aero_hat_deterministic;

%%      // 6 dof dynamics
R = 0*eye(3);
         Zeta_dot = SIX_DOF_Dynamics(R, E_Frame_angulare_velocity, angles,g,m,I_H, I_A,
IXX,IYY,IZZ, Aero_Disturbance, Collision_Disturbance, Control_Input, JTP,Zeta, D_1, D_2 );

         %% Advance system state.
      B_Frame_angulare_acceleration       = [Zeta_dot(4,1); Zeta_dot(5,1); Zeta_dot(6,1)];
      E_Frame_Linear_acceleration         = [Zeta_dot(1,1); Zeta_dot(2,1); Zeta_dot(3,1)];
      E_Frame_Linear_acceleration(3,1)    = 0;
       %dW_last =  dW;
      dW                                  = sqrt(Delta_Time)* [randn;randn;randn];
      dW_2                                = sqrt(Delta_Time)* [randn;randn;randn];
    I_XX_A = I_A(1,1);
    I_YY_A = I_A(2,2);
    I_ZZ_A = I_A(3,3);

      Coriolis = [ 0, I_ZZ_A * B_Frame_angulare_velocity(3,1), -I_YY_A *
B_Frame_angulare_velocity(2,1);
                   -I_ZZ_A * B_Frame_angulare_velocity(3,1), 0, I_XX_A *
B_Frame_angulare_velocity(1,1);
                   I_YY_A * B_Frame_angulare_velocity(2,1), -I_XX_A *
B_Frame_angulare_velocity(1,1), 0 ];
      skew_w = [ 0, -B_Frame_angulare_velocity(3,1),  B_Frame_angulare_velocity(2,1);
                 B_Frame_angulare_velocity(3,1), 0, -B_Frame_angulare_velocity(1,1);
                 -B_Frame_angulare_velocity(2,1), B_Frame_angulare_velocity(1,1), 0];
      last_coverience_1_dW_2                  = coverience_2*(D_2*eye(3)+Coriolis - skew_w *
I_A)*dW_2;%+last_coverience_1_dW;

      if Loop_Count == 1
         dW_initial    =  dW  ;
      end
      last_coverience_1_dW                    = coverience_1*dW+0*last_coverience_1_dW;
      B_Frame_angulare_velocity             =  B_Frame_angulare_velocity+ Delta_Time *
B_Frame_angulare_acceleration  + inv(I_H+I_A)* (last_coverience_1_dW +last_coverience_1_dW_2)
;
      E_Frame_angulare_velocity             =
B_FRAME_ANGULARE_Velocity_TO_E_FRAME_Angulare_velocity(B_Frame_angulare_velocity,
angles);
      angles                                = angles + Delta_Time * E_Frame_angulare_velocity;
```

```
    E_Frame_Linear_velocity             = E_Frame_Linear_velocity + Delta_Time *
E_Frame_Linear_acceleration;
    E_Frame_Linear_possition            = E_Frame_Linear_possition + Delta_Time *
E_Frame_Linear_velocity;
    Stochastic_Torque_distrubance = inv(I_H+I_A)* (last_coverience_1_dW
+last_coverience_1_dW_2) ;
    Total_Torque_distrubance =  Torque_Disturbance + Stochastic_Torque_distrubance;
    if Loop_Count == 1
      q_e_0 =q_e;
      w_error_0 =w_error;
    tau_error_0 = tau_error;
    V_sum_initial =(gamma_1*norm(q_e))^2 + gamma_2*((1+ (norm(w_error))^4)^0.25 -1) +
tau_error'*(inv(ROE))*tau_error+ (1/mew_1)*( coverience_1_time_trans_inf_norm-
sigma_hat_Stochastic_1)^2 + (1/mew_2)*( coverience_2_time_trans_inf_norm-
sigma_hat_Stochastic_2)^2;

    end

    %V_sum_bound = ((norm(q_e_0))^2 + (1+norm(w_error_0)^4)^0.25 -1+ sigma_1 *
norm(tau_error_0)^2 -  lambda/c)*exp(-c * t) +   lambda / c;
    %V_sum =(norm(q_e))^2 + (1+ (norm(w_error))^4)^0.25 + tau_error'*(inv(ROE))*tau_error+
(1/mew_1)*( coverience_1_time_trans_inf_norm-sigma_hat_Stochastic_1)^2 + (1/mew_2)*(
coverience_2_time_trans_inf_norm-sigma_hat_Stochastic_2)^2;

      V_sum_bound = (V_sum_initial -  lambda/c)*exp(-c * t) +   lambda / c;
    V_sum =(gamma_1*norm(q_e))^2 + gamma_2*((1+ (norm(w_error))^4)^0.25 -1) +
tau_error'*(inv(ROE))*tau_error+ (1/mew_1)*( coverience_1_time_trans_inf_norm-
sigma_hat_Stochastic_1)^2 + (1/mew_2)*( coverience_2_time_trans_inf_norm-
sigma_hat_Stochastic_2)^2;

     Lyapunov_bound(1,Loop_Count) = V_sum_bound;
     Lyapunov_bound(2,Loop_Count) = V_sum;


    if (E_Frame_Linear_possition(3,1) <= 0)
      E_Frame_Linear_possition(3,1) =0;
    end

    Zeta  = [ E_Frame_Linear_velocity;
              B_Frame_angulare_velocity          ];

Possition_OLD   = Possition;
    Possition = [ E_Frame_Linear_possition;
                  angles                         ];
    Desired_Possition_OLD =     Desired_Possition;

%% For Plotting system information
  Input(:,Loop_Count) = Control_Input;

    if Loop_Count > 1
     if  (Controller_time_steps ~= Controller_update_time )
      Input_rate(:,Loop_Count)  = (Input(:,Loop_Count) - Input(:,(Loop_Count-1))) /
Controller_update_time;
    end
    if  (Controller_time_steps == Controller_update_time )
      Input_rate(:,Loop_Count)  = (Input(:,Loop_Count) - Input(:,(Loop_Count-
Controller_time_steps))) / Controller_update_time;
      Controller_time_steps  =0;
    end
```

```
        end
    if Loop_Count == 1
        Input_rate(:,Loop_Count)  = 0;
    end
    results(:,Loop_Count) = [ Possition;
                                            t ;
                                            B_Frame_angulare_velocity ;%10
                                            tor_aero_hat_deterministic;
                                            w_error;
                                             eta_2_error;
                                              alpha_w_dot %22
                                              Torque_Disturbance
                                              tau_error;
                                              [0;0;0];%31
                                              inv(I_H)* last_coverience_1_dW;%34
                                              sigma_hat_Stochastic_1;
                                              coverience_1_time_trans_inf_norm;
                                              (norm(coverience_1))^4 - sigma_hat_Stochastic_1;
                                              Total_Torque_distrubance;
                                              Stochastic_Torque_distrubance
                                              q_e
                                              sigma_hat_Stochastic_2];

    eta_2_d_dot_old           = eta_2_d_dot;
for i = 1:3
    Refference_signal(i,Loop_Count) = Desired_Possition(i,1) ;
end
Refference_signal(4,Loop_Count) = eta_2_d(1,1);
Refference_signal(5,Loop_Count) = eta_2_d(2,1);
Refference_signal(6,Loop_Count) = eta_2_d(3,1);
Loop_Count              = Loop_Count + 1;
Controller_time_steps = Controller_time_steps + 1;
%% END OF SIMULATION LOOP%%
 t
end

figure
plot(results(7,:) ,Lyapunov_bound(2,:) ,'g', ...
                'LineWidth',2)
title('V_2 V_{2, bound}  Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,Lyapunov_bound(1,:) ,'b', ...
                'LineWidth',2)
 set(legend({'$V_2$','$V_{2, bound}$'}, 'FontSize',20),'interpreter','latex')
xlabel('Time(s)', 'FontSize',14)
ylabel('V_2, V_{2, bound}', 'FontSize',14)


figure
plot(results(7,:) ,(results(26,:)), ...
                'LineWidth',2)
title('disturbance error  Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,(results(27,:)),'g', ...
                'LineWidth',2)
hold on
plot(results(7,:) ,(results(28,:)),'r', ...
                'LineWidth',2)

set(legend({'${\tau}_{p error}$','${\tau}_{q error}$','${\tau}_{r error}$'}, 'FontSize',20),'interpreter','latex')
```

```
xlabel('Time(s)', 'FontSize',14)
ylabel('Distrubance Error (Nm)', 'FontSize',14)
axis([0, End_Time, - 0.4, 0.4])

 figure
get(0,'Factory');
set(0,'defaultfigurecolor',[1 1 1]);
subplot(2,1,1)
plot(results(7,:) ,results(47,:),'g', ...
               'LineWidth',2)
set(title({'$\hat{\delta}_1 $'}, 'FontSize',16),'interpreter','latex')
set(ylabel({'$\hat{\delta}_1 $'}, 'FontSize',16),'interpreter','latex')
xlabel('Time(s)', 'FontSize',14)
%axis([0, End_Time,-0.4,0.4])
hold on
subplot(2,1,2)
plot(results(7,:) ,results(35,:),'b', ...
               'LineWidth',2)
set(title({'$\hat{\delta}_2 $'}, 'FontSize',16),'interpreter','latex')
hold off
xlabel('Time(s)', 'FontSize',14)
set(ylabel({'$\hat{\delta}_2 $'}, 'FontSize',16),'interpreter','latex')
%axis([0, End_Time,-0.4,0.4])
hold off


figure
get(0,'Factory');
set(0,'defaultfigurecolor',[1 1 1]);
subplot(2,1,1)
plot(results(7,:) ,results(35,:), ...
               'LineWidth',2)
hold on
plot(results(7,:) ,results(47,:),'g', ...
               'LineWidth',2)
%plot(results(7,:) ,0.1^4,'r', ...
%               'LineWidth',2)
title('sigma_hat_stochasticVs Time', 'FontSize',14)
%axis([0, End_Time,-0.4,0.4])
hold on
subplot(2,1,2)
plot(results(7,:) ,results(37,:),'g', ...
               'LineWidth',2)
hold off
title('Sigma error Vs Time', 'FontSize',14)
%legend({'\it{\phi} \rm{(Pitch)}','\it{\theta} \rm{(Roll)}','\it{\psi} \rm{(Yaw)}','\it{\phi} \rm{(Sensor
reading)}','\it{\theta} \rm{(Roll Sensor reading)}','\it{\psi} \rm{(Yaw Sensor reading)}'}, 'FontSize',16)
set(legend({'${p}$ ($p torque$)'}, 'FontSize',16),'interpreter','latex')
xlabel('Time(s)', 'FontSize',14)
ylabel('Stochastic Torque Disturbance', 'FontSize',14)
%axis([0, End_Time,-0.4,0.4])
hold off
```

```
figure
%subplot(2,1,1)
plot(results(7,:) ,Input(2,:) ,'b', ...
            'LineWidth',2)
title('Control Signal Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,Input(3,:) ,'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,Input(4,:) ,'r', ...
            'LineWidth',2)
set(legend({'${U_2}$','${U_3}$','${U_4}$'}, 'FontSize',16),'interpreter','latex')
xlabel('Time(s)', 'FontSize',14)
ylabel('Control input (Nm)', 'FontSize',14)
axis([0, End_Time, - 100, 100])
hold off


figure
plot(results(7,:) ,results(32,:)/Delta_Time^0.5 , ...
            'LineWidth',2)
plot(results(7,:) ,results(33,:)/Delta_Time^0.5 ,'r', ...
            'LineWidth',2)
plot(results(7,:) ,results(34,:)/Delta_Time^0.5 ,'g', ...
            'LineWidth',2)
title('inv_I_G*cov*dw Vs Time', 'FontSize',14)
hold on


figure
plot(results(7,:) ,results(8,:), ...
            'LineWidth',2)
plot(results(7,:) ,results(9,:),'r', ...
            'LineWidth',2)
plot(results(7,:) ,results(10,:),'g', ...
            'LineWidth',2)
title('w_error Vs Time', 'FontSize',14)
hold on

figure
subplot(2,1,1)
plot(results(7,:) ,results(35,:), ...
            'LineWidth',2)
title('sigma_hat_stochasticVs Time', 'FontSize',14)
axis([0, End_Time,-0.4,0.4])
hold on
subplot(2,1,2)
plot(results(7,:) ,results(36,:),'r', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(37,:),'g', ...
            'LineWidth',2)
hold off
set(legend({'${p}$ ($p torque$)','${q}$ ($q torque$)','${r}$ ($r torque$)'},
'FontSize',16),'interpreter','latex')
xlabel('Time(s)', 'FontSize',14)
```

```
ylabel('Stochastic Torque Disturbance', 'FontSize',14)
axis([0, End_Time,-0.4,0.4])
hold off


RAD_to_DEG = 180/pi;
figure
subplot(2,1,1)
pos=get(gca,'position')
pos(3)=0.65; %#re-size axes to leave room for legend
pos(1)=0.1;
set(gca,'position',pos)

plot(results(7,:) ,RAD_to_DEG*results(4,:), ...
            'LineWidth',2)
title('Attitude  Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,RAD_to_DEG*results(5,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,RAD_to_DEG*results(6,:),'r', ...
            'LineWidth',2)
plot(results(7,:) ,RAD_to_DEG*Refference_signal(4,:),'--b' , ...
            'LineWidth',1)
hold on
plot(results(7,:) ,RAD_to_DEG*Refference_signal(5,:),'--g' , ...
            'LineWidth',1)
hold on
plot(results(7,:) ,RAD_to_DEG*Refference_signal(6,:),'--r' , ...
            'LineWidth',1)
axis([0, End_Time, - RAD_to_DEG*0.2*pi, RAD_to_DEG*0.2*pi])
%set(legend({'\phi (Pitch)','\theta (Roll)','\psi (Yaw)'}, 'FontSize',20),'interpreter','latex')
set(legend({'\it{\phi} \rm{(Pitch)}','\it{\theta} \rm{(Roll)}','\it{\psi} \rm{(Yaw)}','\it{\phi_d} \rm{(Pitch
desired)}','\it{\theta_d} \rm{(Roll desired)}','\it{\psi_d} \rm{(Yaw desired)}'},
'FontSize',14,'Position',[(pos(3)+pos(1)+0.02),(pos(2)+0.1),0.2,0.1]));%'location','bestoutside')) ;
xlabel('Time(s)', 'FontSize',14)
ylabel('Attitude (Deg)', 'FontSize',14)
hold off

subplot(2,1,2)
pos=get(gca,'position')
pos(3)=0.65; %#re-size axes to leave room for legend
pos(1)=0.1;
set(gca,'position',pos)
plot(results(7,:) ,RAD_to_DEG*results(17,:),'b', ...
            'LineWidth',2)
title('E frame Anguilar Possition error  Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,RAD_to_DEG*results(18,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,RAD_to_DEG*results(19,:),'r', ...
            'LineWidth',2)
set(legend({'\it{\phi_e} \rm{(Pitch error)}','\it{\theta_e} \rm{(Roll error)}','\it{\psi_e} \rm{(Yaw error)}'},
'FontSize',14,'Position',[(pos(3)+pos(1)+0.02),(pos(2)+0.1),0.2,0.1]));%'location','bestoutside')) ;
xlabel('Time(s)', 'FontSize',14)
ylabel('Attitude error(Deg)', 'FontSize',14)

hold off
```

```
figure
plot(results(7,:) ,results(35,:), ...
            'LineWidth',2)
title('Stochastic noise error  Vs Time', 'FontSize',14)
xlabel('Time(s)', 'FontSize',14)
ylabel('W', 'FontSize',14)
hold off
xlabel('Time(s)', 'FontSize',14)
ylabel('Stochastic distrubance', 'FontSize',14)
axis([0, End_Time, - 0.2, 0.2])


figure
plot(results(7,:) ,(results(44,:)), ...
            'LineWidth',2)
title('Attitude error MRP   Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,(results(45,:)),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,(results(46,:)),'r', ...
            'LineWidth',2)

set(legend({'${q}_{1 error}$','${\q}_{2 error}$','${\q}_{3 error}$'}, 'FontSize',20),'interpreter','latex')
xlabel('Time(s)', 'FontSize',14)
ylabel('Distrubance  (Nm)', 'FontSize',14)
axis([0, End_Time, - 1, 1])


figure
subplot(2,1,1)
plot(results(7,:) ,results(8,:), ...
            'LineWidth',2)
title('B frame Anguilar Velocity   Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,results(9,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(10,:),'r', ...
            'LineWidth',2)
 set(legend({'$p$','$q$','$r$'}, 'FontSize',20),'interpreter','latex')
xlabel('Time(s)', 'FontSize',14)
ylabel('Angular Velocity (rads^{-1})', 'FontSize',14)

hold off
subplot(2,1,2)
plot(results(7,:) ,results(14,:),'b', ...
            'LineWidth',2)
title('B frame Anguilar Velocity error  Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,results(15,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(16,:),'r', ...
            'LineWidth',2)
```

```
set(legend({'${p}_e$ ($p$ $error$)','${q}_e$ ($q$ $error$)','${r}_e$ ($r$ $error$)'},
'FontSize',20),'interpreter','latex')
 xlabel('Time(s)', 'FontSize',14)
ylabel('Angular Velocity (rads^{-1})', 'FontSize',14)
hold off

function U = Attitude_Backstepping_Controller_Stochastic(I_H, I_A, K_1, K_2, K_d, gamma_1,
gamma_2, epsilon_1, epsilon_2, epsilon_3, epsilon_4, D_2, eta_2, B_Frame_angulare_velocity,
eta_2_d, eta_2_d_dot, eta_2_d_double_dot, Delta_time, tor_aero_hat_deterministic,
tor_aero_hat_deterministic_dot, sigma_hat_Stochastic_1,
sigma_hat_Stochastic_2,sigma_hat_Stochastic_dot_1  ,sigma_hat_Stochastic_dot_2, Loop_Count ,
ROE, tor_aero_deterministic_MAX, tor_aero_Stochastic_MAX_1, tor_aero_Stochastic_MAX_2,
mew_1, mew_2, xi_aero_deterministic, xi_aero_Stochastic_1, xi_aero_Stochastic_2)

n_2                     =  eta_2;
q                       = Euler_to_Modified_rodrigues_paramater(n_2);
alpha_q                 = Euler_to_Modified_rodrigues_paramater(eta_2_d);
q_e                     = q - alpha_q;
R_2_alpha_q             = R_2_MRP_calc( alpha_q);
angles                  = eta_2;
T_inv                   = Angular_velocity_cordinant_transform(angles);
R_2_q                   = R_2_MRP_calc(q);
R_2_q_inverse           = R_2_q.'*(16/(1+q.'*q)^2);
w                       = B_Frame_angulare_velocity;
alpha_w                 = R_2_q_inverse*(-K_1 * q_e + R_2_alpha_q * T_inv * eta_2_d_dot );
w_e                     =   B_Frame_angulare_velocity - alpha_w;
[Partial_alpha_w_wrt_q, Partial_alpha_w_wrt_eta_2_d,Partial_alpha_w_wrt_eta_2_d_dot] =
Partial_derivatives_of_alpha_w( K_1, n_2, q, alpha_q, eta_2_d, eta_2_d_dot);
alpha_w_dot                   = Partial_alpha_w_wrt_q * R_2_q * w + Partial_alpha_w_wrt_eta_2_d *
eta_2_d_dot + Partial_alpha_w_wrt_eta_2_d_dot * eta_2_d_double_dot ;
INV_I_H_plus_I_A                        = inv(I_H+I_A);

tor_aero_deterministic_p          = Projection_algorithm(gamma_2*((norm(w_e))^2*([1 0 0
]*INV_I_H_plus_I_A)*w_e)/(1+(w_e.'*w_e)^2)^0.5,[1 0 0]*tor_aero_hat_deterministic,[1 0
0]*tor_aero_hat_deterministic_dot, [1 0 0]*xi_aero_deterministic, [1 0 0]*tor_aero_deterministic_MAX,
ROE(1,1), Delta_time);
tor_aero_deterministic_q          = Projection_algorithm(gamma_2*((norm(w_e))^2*([0 1 0
]*INV_I_H_plus_I_A)*w_e)/(1+(w_e.'*w_e)^2)^0.5,[0 1 0]*tor_aero_hat_deterministic,[0 1
0]*tor_aero_hat_deterministic_dot, [0 1 0]*xi_aero_deterministic, [0 1
0]*tor_aero_deterministic_MAX,ROE(2,2), Delta_time);
tor_aero_deterministic_r          = Projection_algorithm(gamma_2*((norm(w_e))^2*([0 0 1
]*INV_I_H_plus_I_A)*w_e)/(1+(w_e.'*w_e)^2)^0.5,[0 0 1]*tor_aero_hat_deterministic,[0 0
1]*tor_aero_hat_deterministic_dot, [0 0 1]*xi_aero_deterministic, [0 0 1]*tor_aero_deterministic_MAX,
ROE(3,3), Delta_time);
tor_aero_deterministic            = [ tor_aero_deterministic_p;
                                      tor_aero_deterministic_q;
                                      tor_aero_deterministic_r ];

tor_aero_hat_deterministic        =[ tor_aero_deterministic(1,1);
                                      tor_aero_deterministic(3,1);
                                      tor_aero_deterministic(5,1)];
tor_aero_hat_deterministic_dot = [ tor_aero_deterministic(2,1);
                                      tor_aero_deterministic(4,1);
                                      tor_aero_deterministic(6,1)];
if Loop_Count   == 1
   tor_aero_hat_deterministic  =[0;0;0];
end

sigma_Stochastic= Projection_algorithm(gamma_2^2*
(9*epsilon_3)*(norm(inv(I_H+I_A)))^4*(norm(w_e))^4/(1+(norm(w_e))^4)^1.5,sigma_hat_Stochastic_1
```

```
,sigma_hat_Stochastic_dot_1  , xi_aero_Stochastic_1, tor_aero_Stochastic_MAX_1, mew_1,
Delta_time);
sigma_hat_Stochastic        = sigma_Stochastic(1,1);

sigma_hat_Stochastic_1 = sigma_hat_Stochastic;

I_XX_A = I_A(1,1);
I_YY_A = I_A(2,2);
I_ZZ_A = I_A(3,3);

Coriolis_A = [ 0, I_ZZ_A * B_Frame_angulare_velocity(3,1), -I_YY_A *
B_Frame_angulare_velocity(2,1);
                       -I_ZZ_A * B_Frame_angulare_velocity(3,1), 0, I_XX_A *
B_Frame_angulare_velocity(1,1);
                       I_YY_A * B_Frame_angulare_velocity(2,1), -I_XX_A *
B_Frame_angulare_velocity(1,1), 0 ];
skew_w = [ 0, -B_Frame_angulare_velocity(3,1),  B_Frame_angulare_velocity(2,1);
                        B_Frame_angulare_velocity(3,1), 0, -B_Frame_angulare_velocity(1,1);
                        -B_Frame_angulare_velocity(2,1), B_Frame_angulare_velocity(1,1), 0];
                   Multiplicitiveand_aditive_noise = inv(I_H+I_A)*(D_2*eye(3)+0*0.05*eye(3)+ Coriolis_A
+skew_w *I_A);
sigma_Stochastic= Projection_algorithm(gamma_2^2*
(9*epsilon_4)*(norm(Multiplicitiveand_aditive_noise
))^4*(norm(w_e))^4/(1+(norm(w_e))^4)^1.0,sigma_hat_Stochastic_2,sigma_hat_Stochastic_dot_2  ,
xi_aero_Stochastic_2, tor_aero_Stochastic_MAX_2, mew_2, Delta_time);
sigma_hat_Stochastic        =1*sigma_Stochastic(1,1);
sigma_hat_Stochastic_dot  =  sigma_Stochastic(2,1);
sigma_hat_Stochastic_2  = sigma_hat_Stochastic;
 w_e_dot              = - K_2 * w_e/ (1+0*(w_e.'*w_e)^2)^0.25 ;
tor                          = (I_H+I_A)*(D_2 * alpha_w + w_e_dot -
gamma_2*((9*epsilon_3/4)*(norm(inv(I_H + I_A)))^4*sigma_hat_Stochastic_1)*w_e /
(1+1*(w_e.'*w_e)^2)^0.5 -
gamma_2*((9*epsilon_4/4)*(norm(Multiplicitiveand_aditive_noise))^4*sigma_hat_Stochastic_2)*w_e/(
1+1*(w_e .'*w_e )^2)^0.5 + alpha_w_dot -
(1/gamma_2)*epsilon_2*((gamma_1/(4*epsilon_1))^2)*norm(R_2_alpha_q)^4/4
*w_e/(1+(w_e.'*w_e)^2)^3.25-
gamma_1*(1/gamma_2)*(q_e.'*R_2_alpha_q*w_e)*w_e/(1+(w_e.'*w_e)^2)^0.25 ) -
0*tor_aero_hat_deterministic + cross(B_Frame_angulare_velocity, ((I_H +
I_A)*B_Frame_angulare_velocity));

U                            = [ tor(1,1); tor(2,1) ; tor(3,1); alpha_w;alpha_w_dot; tor_aero_hat_deterministic;
tor_aero_hat_deterministic_dot; sigma_hat_Stochastic_1; sigma_hat_Stochastic_dot;
sigma_hat_Stochastic_2  ];
```

```
function [Partial_alpha_w_wrt_q, Partial_alpha_w_wrt_eta_2_d,Partial_alpha_w_wrt_eta_2_d_dot] =
Partial_derivatives_of_alpha_w( K_1, n_2, q, alpha_q, eta_2_d, eta_2_d_dot)

T_inv                  = Angular_velocity_cordinant_transform(eta_2_d);
T_inverse_eta_2_d = T_inv;
T_eta_2_d_inverse = T_inverse_eta_2_d;
R_2_alpha_q = R_2_MRP_calc(alpha_q);

q_e = q - alpha_q;


vector_0 =( -K_1 *q_e  + R_2_alpha_q*T_eta_2_d_inverse*eta_2_d_dot);
vector_1 = T_eta_2_d_inverse*eta_2_d_dot;
vector_2 = eta_2_d_dot;

Partial_R_2_alpha_q_wrt_partial_alpha_q_times_vector_1 = R_2_derivitive_times_vector (alpha_q,
R_2_alpha_q, vector_1);
Partial_T_inverse_eta_2_d_wrt_eta_2_d_times_vector_2 =
Partial_T_eta_2_d_inverse_wrt_eta_2_times_vector(eta_2_d, eta_2_d_dot);

R_2_q = R_2_MRP_calc(q);
R_2_inverse_dot_times_x = R_2_inverse_derivative_times_vector(R_2_q,  q, vector_0);
R_2_q_inverse  = R_2_q.'*(16/(1+q.'*q)^2);
P_1 = R_2_inverse_dot_times_x ;

P_2 =R_2_q_inverse * (-(K_1  ));
P_3 =(K_1) ;

P_4 =Partial_R_2_alpha_q_wrt_partial_alpha_q_times_vector_1;
P_5 =R_2_alpha_q*Partial_T_inverse_eta_2_d_wrt_eta_2_d_times_vector_2;
P_6 =R_2_alpha_q*T_inverse_eta_2_d;

Partial_alpha_w_wrt_q = P_1+ R_2_q_inverse *P_2  ;
Partial_alpha_w_wrt_alpha_q = R_2_q_inverse *( P_3  + P_4);
Partial_alpha_w_wrt_eta_2_d = R_2_q_inverse * P_5 + Partial_alpha_w_wrt_alpha_q;
Partial_alpha_w_wrt_eta_2_d_dot = R_2_q_inverse * P_6;

End


function R_2_MRP = R_2_MRP_calc(q)
skew_q = [ 0, -q(3,1), q(2,1);
            q(3,1), 0, -q(1,1);
            -q(2,1), q(1,1), 0];
R_2_MRP = 0.5*(eye(3) + q*q.' + skew_q - ((1+(q.'*q))/2)*eye(3) );
end
```

```
function Modified_rodrigues_paramater = Euler_to_Modified_rodrigues_paramater(Euler_angle)

Phi    = Euler_angle(1,1);
Theta = Euler_angle(2,1);
Psi    = Euler_angle(3,1);
 q_0 = (1 + cos(Phi/2)*cos(Theta/2)*cos(Psi/2) + sin(Phi/2)*sin(Theta/2)*sin(Psi/2) );
q_1 = (sin(Phi/2)*cos(Theta/2)*cos(Psi/2) - cos(Phi/2)*sin(Theta/2)*sin(Psi/2))/q_0;
q_2 = (cos(Phi/2)*sin(Theta/2)*cos(Psi/2) + sin(Phi/2)*cos(Theta/2)*sin(Psi/2))/q_0;
q_3 = (cos(Phi/2)*cos(Theta/2)*sin(Psi/2) - sin(Phi/2)*sin(Theta/2)*cos(Psi/2))/q_0;

Modified_rodrigues_paramater = [ q_1;
                                 q_2;
                                 q_3];

end


function  Partial_R_2_alpha_q_wrt_partial_q_times_vector  = R_2_derivitive_times_vector (q, R_2,
vector)
Partial_R_2_alpha_q_wrt_partial_q_times_vector = 1/2*((q.'*vector*eye(3)+q*(vector).'-vector*q.')-
([0,0,0;0,0,1;0,-1,0]*(vector*[1,0,0])+[0,0,-1;0,0,0;1,0,0]*(vector*[0,1,0])+[0,1,0;-
1,0,0;0,0,0]*(vector*[0,0,1])));
End


function R_2_inverse_dot_times_vector = R_2_inverse_derivative_times_vector(R_2,  q, vector)

R_2_inverse_dot_times_vector=8/(1+q.'*q)^2 * ((q.'*vector*eye(3)+q*vector.'-
vector*q.')+([0,0,0;0,0,1;0,-1,0]*(vector*[1,0,0])+[0,0,-1;0,0,0;1,0,0]*(vector*[0,1,0])+[0,1,0;-
1,0,0;0,0,0]*(vector*[0,0,1])))-64/(1+q.'*q)^3  * R_2.'*vector*q.';

End


function  Partial_T_eta_2_d_inverse_wrt_eta_2_times_vector_x =
Partial_T_eta_2_d_inverse_wrt_eta_2_times_vector (alpha_2, vector)
   alpha_phi    = alpha_2(1,1);
   alpha_theta = alpha_2(2,1);
   Partial_T_eta_2_d_inverse_wrt_eta_2_times_vector_x = ([0,0,0;0,-
sin(alpha_phi),cos(alpha_theta)*cos(alpha_phi);0,-cos(alpha_phi),-
cos(alpha_theta)*sin(alpha_phi)]*vector*[1;0;0].'+[0,0,-cos(alpha_theta);0,0,-
sin(alpha_theta)*sin(alpha_phi);0,0,-sin(alpha_theta)*cos(alpha_phi)]*vector*[0;1;0].');
end
```

```
function  Projection = Projection_algorithm(omega_Macron,omega_hat, omega_hat_dot, xi,
omega_MAX, gamma, Delta_time)
   XI =( (norm(omega_hat))^2 - omega_MAX^2) / (xi^2 + 2*xi*omega_MAX);
   Derivative_XI = 2*omega_hat'/ (xi^2 + 2*xi*omega_MAX);
   if(XI < 0)
      omega_hat_dot = gamma * omega_Macron;
      Derivative_XI = 2*omega_hat'/ (xi^2 + 2*xi*omega_MAX);
   end
   if((XI >=  0) && ((Derivative_XI * omega_Macron)<=0))
      omega_hat_dot = gamma * omega_Macron;
   end
   if((XI >=  0) && ((Derivative_XI * omega_Macron)>0))
      omega_hat_dot = gamma * (1-XI)*omega_Macron;
   end
    omega_hat = omega_hat + Delta_time * omega_hat_dot ;
   Projection  = [omega_hat; omega_hat_dot ];
end
```

```
function  Zeta_dot =SIX_DOF_Dynamics(R, E_Frame_angulare_velocity, angles,g,m,I_H,
I_A,IXX,IYY,IZZ, Aero_Disturbance,      Collision_Disturbance, Control_Input, JTP,Zeta, D_1, D_2 )

   B_Frame_angulare_velocity =
E_FRAME_ANGULARE_Velocity_TO_B_FRAME_Angulare_velocity(E_Frame_angulare_velocity,
angles);
   p                                   = B_Frame_angulare_velocity(1,1);
   q                                   = B_Frame_angulare_velocity(2,1);
   r                                   = B_Frame_angulare_velocity(3,1);
   EH = [ R, zeros(3,3);
           zeros(3,3), eye(3)];
   MH = [ m * eye(3), zeros(3,3);
              zeros(3,3), I_H    + I_A      ];
   GH = [0;0; m*g;0;0;0];
   I_XX_A = I_A(1,1);
   I_YY_A = I_A(2,2);
   I_ZZ_A = I_A(3,3);
   Coriolis_A = [ 0, I_ZZ_A * r, -I_YY_A * q;
                  -I_ZZ_A * r, 0, I_XX_A * p;
                  I_YY_A * q, -I_XX_A * p, 0 ];
   coriolis = [  0, IZZ*r, -IYY*q;
                 -IZZ*r, 0, IXX*p;
                 IYY*q, -IXX*p, 0];
   CH = [ zeros(3,3), zeros(3,3);
              zeros(3,3), (coriolis +Coriolis_A)     ];
   OH = [ 0, 0, 0, 0 0, 0;
              0, 0, 0, 0 0, 0;
              0, 0, 0, 0 0, 0;
              -p, p, -p, p, -p, p;
              -q, q, -q, q, -q, q;
              0, 0, 0, 0 0, 0 ] ;
   Control_H =  [ 0;
                  0;
                  Control_Input ];
   Motor_Speed = zeros(6,1);
   Zeta_dot = (inv(MH))*(-CH*Zeta - GH +JTP*OH * Motor_Speed  +EH*Control_H
+Aero_Disturbance + Collision_Disturbance) - [D_1 0*eye(3); 0*eye(3) D_2] * Zeta;
end
```

Appendix J –Simulation Code, Deterministic One-Step Ahead backstepping Controller for Quadrotor Aircraft

```
%alpha_PSI_dot_old  = PSI_Desired_Possition/Delta_Time;

close all
clc
clear
alpha_w_old = [0;0;0];
counting_Loop  = 0

End_Time                  =100%398.44;
Delta_Time                = 1/1000;
controller_update_time=Delta_Time   ;
controller_time_steps = controller_update_time/Delta_Time;

K_01 =20*  [1 0 0;
            0 1 0;
            0 0 1 ] ;
K_01 =1* [ 1.5 0 0;
              0 1.5 0;
              0 0 0.8];
K_01 =1* eye(3);
gamma_4 =    1   ;

K_01 =1* [ 1.5 0 0;
              0 1.5 0;
              0 0 1];
gamma_4 =    1   ;

K_02 = gamma_4*eye(3);

K_2 = 2*eye(3);

K_3 = 10*eye(3);

K_4 =10*eye(3);
k_1d = 0.05*eye(3);
zeta_disturbance_observer = [0;0;0];

Collision_time            = -1;
Motor_spped                = zeros(6, 1);
```

```
Motor_Voltage              = zeros(6, 1);
MOTOR_Spped_Voltage_Ratio   = 570;
PROP_Thrust_Coef           =  5.83368*10^-6;  %0.583368*10^-6;
PROP_Rotation_Coef      = PROP_Thrust_Coef /20 ; %0.27 * 10^-6;%0.00623535 * 10^-6;
Delta_OMEGA            = 0;
%% model paramters physical
Copter_Radius              = 0.30;
Copter_Radius_Along_X_Axis  = 0.26;
Copter_Radius_Along_Y_Axis  = 0.15;
g   = 9.81;

m = 2.23;

JTP =  2.5172e-006;
Motor_radius = 20/1000                          ;
copterradius = Copter_Radius;
Two_pi = 2*pi;
pitch =     4;
Propeller_radius =        (7.8/2)*.75;
alpha = atan(pitch/(2*pi*Propeller_radius ));
alpha = alpha * 180/pi;
Blade_pitch_angle = alpha ;
INCH_to_meter          = 0.0254    ;
BLade_diametre         = 6         ;
L                      = BLade_diametre * INCH_to_meter/2  ;

p=1.1839; %     wikipedia at 25 degrees C
Blade_leangth          = L;
Effective_blade_leangth = 0.75*Blade_leangth     ;
total_Propeller_surface_area =    2.8873*10^-5;
air_density     =1.1839;
Hex_copter_height   =   1;

I_XX = 0.016507;
I_YY = 0.016507;
I_ZZ = 0.016284;

I_H     = [ I_XX, 0, 0;
             0, I_YY, 0;
             0, 0, I_ZZ];
I_A     = 0.5 * I_H;
%% mathamtical deffinitions
T_inverse_Phi_derivative    = zeros(3, 3);
T_inverse_Theta_derivative  = zeros(3, 3);
Plot_info   = zeros(6, 1);
T_inverse                 = zeros(3, 3);
T_inverse_inverse          = zeros(3, 3);
E_Frame_Linear_possition       = zeros(3, 1);
E_Frame_Linear_velocity        = zeros(3, 1);
angles                     = zeros(3, 1);
Control_Signal_in_F            = zeros(3, 1);

temp                       = zeros(2, 1);
T_Collision                  = zeros(3, 1);
Possition                   = zeros(6, 1);
```

```
Possition_OLD             =  zeros(6, 1);
Desired_Possition_OLD =  zeros(6, 1);
 Zeta                          =  zeros(6, 1);
 Motor_Speed              =  zeros(6, 1);

%% dessired possitions
Motor_Speed_to_Control_Input = [
PROP_Thrust_Coef,PROP_Thrust_Coef,PROP_Thrust_Coef,PROP_Thrust_Coef,PROP_T
hrust_Coef,PROP_Thrust_Coef;
                              -PROP_Thrust_Coef*copterradius, -
PROP_Thrust_Coef*Copter_Radius_Along_Y_Axis,
PROP_Thrust_Coef*Copter_Radius_Along_Y_Axis, PROP_Thrust_Coef*copterradius,
PROP_Thrust_Coef*Copter_Radius_Along_Y_Axis, -
PROP_Thrust_Coef*Copter_Radius_Along_Y_Axis;
                              0, -PROP_Thrust_Coef*Copter_Radius_Along_X_Axis, -
PROP_Thrust_Coef*Copter_Radius_Along_X_Axis, 0,
PROP_Thrust_Coef*Copter_Radius_Along_X_Axis,
PROP_Thrust_Coef*Copter_Radius_Along_X_Axis;
                              PROP_Rotation_Coef*copterradius,-
PROP_Rotation_Coef*copterradius,PROP_Rotation_Coef*copterradius,-
PROP_Rotation_Coef*copterradius,PROP_Rotation_Coef*copterradius,-
PROP_Rotation_Coef*copterradius ] ;
Control_Input_to_Motor_Speed = [ 1/(6*PROP_Thrust_Coef), -
1/(2*PROP_Thrust_Coef*(copterradius + Copter_Radius_Along_Y_Axis)), 0,
Copter_Radius_Along_Y_Axis/(2*PROP_Rotation_Coef*copterradius*(copterradius +
Copter_Radius_Along_Y_Axis)) ;
                              1/(6*PROP_Thrust_Coef), -
1/(4*PROP_Thrust_Coef*(copterradius + Copter_Radius_Along_Y_Axis)), -
1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef), -
copterradius/(4*PROP_Rotation_Coef*copterradius*(copterradius +
Copter_Radius_Along_Y_Axis));
                              1/(6*PROP_Thrust_Coef),
1/(4*PROP_Thrust_Coef*(copterradius + Copter_Radius_Along_Y_Axis)), -
1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef),
copterradius/(4*PROP_Rotation_Coef*copterradius*(copterradius +
Copter_Radius_Along_Y_Axis));
                              1/(6*PROP_Thrust_Coef),
1/(2*PROP_Thrust_Coef*(copterradius + Copter_Radius_Along_Y_Axis)), 0, -
Copter_Radius_Along_Y_Axis/(2*PROP_Rotation_Coef*copterradius*(copterradius +
Copter_Radius_Along_Y_Axis));
                              1/(6*PROP_Thrust_Coef),
1/(4*PROP_Thrust_Coef*(copterradius + Copter_Radius_Along_Y_Axis)),
1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef),
copterradius/(4*PROP_Rotation_Coef*copterradius*(copterradius +
Copter_Radius_Along_Y_Axis));
                              1/(6*PROP_Thrust_Coef), -
1/(4*PROP_Thrust_Coef*(copterradius + Copter_Radius_Along_Y_Axis)),
1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef), -
copterradius/(4*PROP_Rotation_Coef*copterradius*(copterradius +
Copter_Radius_Along_Y_Axis))] ;

Time_steps              = End_Time / Delta_Time;
Lyapunov_Signal = zeros(2, Time_steps);
 x_1_d_dot_plot = zeros(3, Time_steps);
MOTOR =  zeros(6, Time_steps);
```

```
%results                       = zeros(35, Time_steps);
results                  = zeros(100, Time_steps);
stochastic_disturbance_results  = zeros(9, Time_steps);
W_1 = [0;0;0];
W_2 = [0;0;0];
W_3 = [0;0;0];

Increment_Refference_Time = 0;
Refference_Signal_Time_Steps =20;
Refference_Signal_Time_Steps_Inverse = 20* End_Time;
Input                    = zeros(4, Time_steps);
Input_rate               = zeros(4, Time_steps);
Refference_signal                    = zeros(6, Refference_Signal_Time_Steps );
E_Frame_angulare_velocity   = zeros(3,1);
B_Frame_angulare_velocity   = zeros(3,1);

x_1_d_triple_dot_old =0;
 alpha_PSI_dot_old  = 0;
Loop_Count              = 1;
loop_count_2 = 10;
Controller_count = controller_time_steps;
x_1_d_dot_old               = [0;0;0];
x_1_d_double_dot_old   = [0;0;0];

n_1_hat_old = 0;
v_1_hat_old = 0;
ready_to_start =1;
if ready_to_start == 1

   E_Frame_Linear_possition = [0; 0;0];
   Possition = [E_Frame_Linear_possition;0;0;0];
end
simulation_start_time = clock;
v_1_hat_dot  = [0; 0; 0];
v_1_hat = [0; 0; 0];
n_1_tilda  = [0;0;0];
n_1_hat = E_Frame_Linear_possition ;
%dt = 0.02;
dt = Delta_Time;
%dt = 1/1000;
v_1_hat_TEMP  = [0;0;0];
alpha_w_old = [0;0;0];
L2_to_L3 = 1/100;
L2_to_L3 = 1/500;
L2_to_L3 = 1/500;
L2_to_L3 = 1/200;
gamma_2 = 1/L2_to_L3 ;
gamma_2 =200
L2_to_L3 = 1/gamma_2;
gamma_3 = 0.1;
%L2_to_L3 = 1/500
last_coverience_dW                       = zeros(3,1);
coverience                               =0.1* [ 1 0 0;
                                              0 1 0;
                                              0 0 1];
```

```
coverience_angulare_system           =1.25*0.1^0.5* [ 1 0 0;
                                         0 1 0;
                                         0 0 1];
coverience_angulare_system           =2* [ 1 0 0;
                                         0 1 0;
                                         0 0 1];
coverience_3                          =0.1*eye(3);

tor_aero_hat_deterministic         = [0 ; 0; 0];
tor_aero_hat_deterministic_dot   = [0 ; 0; 0];
sigma_hat_Stochastic_1            = 0;
sigma_hat_Stochastic_dot_1      = 0;
sigma_hat_Stochastic_2            = 0;
sigma_hat_Stochastic_dot_2      = 0;
Inverse_I_H = inv(I_H);
Inverse_I_H_plus_I_A = inv(I_H + I_A);
   f_aero_Deterministic = 0*[0.5 ; 0.3;0.1];
   r=0.5
      Ca = 0.05;
MOTOR_FORCE_CALC = [ 1/4,        0, -1/(2*r), -1/(4*Ca);
                                  1/4, -1/(2*r),        0,  1/(4*Ca);
                                  1/4,        0,  1/(2*r), -1/(4*Ca);
                                  1/4,  1/(2*r),        0,  1/(4*Ca)];
Possition=0*[-15;1;0; -0*pi/4;-0*pi/4;0];
Possition_initial = Possition;
E_Frame_Linear_possition = [ Possition(1,1);
         Possition(2,1);
         Possition(3,1) ];
      gamma_5 = 1;

n_1_old       =  [ Possition(1,1);
         Possition(2,1);
         Possition(3,1) ];

   epsilon_3=20;
   epsilon_4 =10;
   epsilon_5 =10;
   epsilon_7 = 10;
gamma_1 =5.12;

   min_eigen_value_D_1=1/8;
   D_1= [ 0.25 0 0;
            0 0.25 0;
            0 0 0.125];
   D_2= 10^(-3)*[ 2.5 0 0;
            0 2.5 0;
            0 0 0.03];
   c_2                 =                 ((norm(coverience/m))^2/2+0.025+0.025+f_aero_Deterministic
.'*f_aero_Deterministic *min_eigen_value_D_1/3.9 + 3*min_eigen_value_D_1/6);
   c_1=c_2/gamma_1/2+gamma_1*min_eigen_value_D_1*0.5;
   c=c_1;


      for  t = 0: Delta_Time: End_Time
```

```
  %    [Desired_Possition,   Plot_info]   =   Get_Refference_Signal(Desired_Possition_2,
Desired_Possition_Time, t, Plot_info );
    Desired_Possition(3,1) = 0.1*t;
        n_1 =  [ Possition(1,1);
            Possition(2,1);
            Possition(3,1) ];
    if Loop_Count    == 1
        n_1 =  [ Possition(1,1);
            Possition(2,1);
            Possition(3,1) ];
    end
if Desired_Possition(3,1) >= 0
    Desired_Possition(3,1) = 0.1*t;
    Desired_Possition(2,1)  = 10*sin((0.01*t));
    Desired_Possition(1,1)  = 10*(cos((0.01*t)))-20;
    Desired_Possition(6,1) = 0.01*t +0*pi/4;
    Desired_Possition(6,1) = 0;
    if Possition (3,1) > 0
        Wind_Velocity              = [0 ;0;0];
    end
    if Possition (3,1) <= 0
        Wind_Velocity              = [0 ;0;0];
    end
end

    angles = [  Possition(4,1);
                 Possition(5,1);
                 Possition(6,1) ];
    phi     = Possition(4,1);
    theta   = Possition(5,1);
    R       = rotation(angles);

q = Euler_to_Modified_rodrigues_paramater(angles);
R_1_MPR = R_MRP(q);

%R = R_1_MPR;
    E_Frame_angulare_velocity                                                    =
B_FRAME_ANGULARE_Velocity_TO_E_FRAME_Angulare_velocity(B_Frame_angulare_ve
locity, angles);


    Velocity                = (Possition-Possition_OLD) / Delta_Time;
    Position_ERROR       = (Possition - Desired_Possition );
    Velocity_ERRPR       = (Desired_Possition-Desired_Possition_OLD)/Delta_Time-Velocity;


     n_1d                      = [ 10*(cos((0.1*t)))-20;
                                   10*sin((0.1*t));
                                   0.1*t-0.1];
    n_1d_dot             =  [ -1*sin(0.1*t);
                               1*cos((0.1*t));
                               0.1];
    n_1d_double_dot =[ -0.1*(cos((0.1*t)));
                          -0.1*sin((0.1*t));
                          0];
```

```
    n_1d_triple_dot      = [ 0.01*(sin((0.1*t)));
                             -0.01*cos((0.1*t));
                             0];
    n_1d_quad_dot       = [ 0.001*(cos((0.1*t)));
                             0.001*sin((0.1*t));
                             0];
    %% Second stage set up converssions between frames of refference
    alpha_PSI              = 0*Desired_Possition(6,1);
    alpha_PSI_dot          = 0*Desired_Velocity(6,1);
    alpha_PSI_double_dot          = 0*( alpha_PSI_dot  - alpha_PSI_dot_old) / Delta_Time;
    alpha_psi = alpha_PSI;
    n_1e = n_1 - n_1d;
    %v_1_hat = x_2;


if Controller_count == controller_time_steps
    %% LINEAR POSSITION SYSTEM
counting_Loop  = counting_Loop  +1;
    %if gps_read_time >= gps_update_time
        delta_1_v_1_hat = 1 + 0.5 * (v_1_hat.')*v_1_hat;
        sigma_n_1e= [ n_1e(1,1)/(1+n_1e(1,1)^2)^0.5;
                    n_1e(2,1)/(1+n_1e(2,1)^2)^0.5;
                    n_1e(3,1)/(1+n_1e(3,1)^2)^0.5];
        sigma_dash_n_1e = [ 1/(1+n_1e(1,1)^2)^1.5, 0, 0;
                           0, 1/(1+n_1e(2,1)^2)^1.5, 0;
                           0, 0, 1/(1+n_1e(3,1)^2)^1.5];
        alpha_1                  =  -K_1* sigma_n_1e/delta_1_v_1_hat + n_1d_dot;
        v_1e =  v_1_hat - alpha_1  ;
        [Omega,    Partial_Omega_wrt_partial_n_1e,    Partial_Omega_wrt_partial_v_1_hat    ,
Partial_Omega_wrt_partial_n_1d_dot      ,      Partial_Omega_wrt_partial_n_1d_double_dot,
Partial_M_1_to_6_wrt_partial_n_1e,                         Partial_M_1_to_6_wrt_partial_v_1_hat,
Partial_M_1_to_6_wrt_partial_n_1d_dot,     Partial_M_1_to_6_wrt_partial_n_1d_double_dot,
Sum_M1_to_M6]        =        Omega_Partial_stochastic_one_step_ahead_backstepping(m,
epsilon_2,gamma_1,  epsilon_3,min_eigen_value_D_1,K_1,  K_2,  D_1,c  ,  n_1e,  v_1_hat,
v_1e, n_1d_dot, n_1d_double_dot, alpha_1);
        OMEGA = Omega;
        alpha_PHI                 = asin((sin(alpha_PSI)*OMEGA  (1,1) - cos(alpha_PSI)*OMEGA
(2,1))/(OMEGA' * OMEGA)^0.5);
        alpha_THETA              = atan((cos(alpha_PSI)*OMEGA  (1,1) + sin(alpha_PSI)*OMEGA
(2,1))/OMEGA  (3,1));
        alpha_2                  = [alpha_PHI ; alpha_THETA ; alpha_PSI ];
        h_2 =min_eigen_value_D_1*0.5* gamma_1*(sigma_n_1e.').';
        h_3                                                                              =2*
gamma_1*(v_1e.'*K_1*sigma_dash_n_1e/delta_1_v_1_hat).'/(1+v_1e.'*v_1e)^0.5;
        alpha_psi_dot  =  alpha_PSI_dot         ;
        alpha_psi_double_dot  =   alpha_PSI_double_dot ;
        G_1 = eye(3)-(K_1*sigma_n_1e*(v_1_hat.')/delta_1_v_1_hat^2);
        cos_alpha_psi = cos(alpha_psi);
        alpha_phi = alpha_PHI;
        alpha_theta = alpha_THETA;
        Omega_3 = Omega(3,1);
        sin_alpha_psi = sin(alpha_psi);
        sigma_v_1e = [ v_1e(1,1)/(1+v_1e(1,1)^2)^0.5;
                    v_1e(2,1)/(1+v_1e(2,1)^2)^0.5;
                    v_1e(3,1)/(1+v_1e(3,1)^2)^0.5];
```

```
        clac_v_1_hat = 1;

    %end

%% ANGULARE POSSITION SYSTEM

    n_2e = angles -alpha_2;
    n_1 =  [ Possition(1,1);
             Possition(2,1);
             Possition(3,1) ];
         epsilon_1  = 1;
          epsilon_6  = 1;
%{
         %not used
            epsilon_1
         epsilon_6
%}
          Delta_time = Delta_Time;

          [ tor, alpha_w, alpha_w_dot, tor_aero_hat_deterministic, tor_aero_hat_deterministic_dot, sigma_hat_Stochastic_1, sigma_hat_Stochastic_dot_1,sigma_hat_Stochastic_2, sigma_hat_Stochastic_dot_2, h_4, h_5, q_e, w_e, Partial_alpha_w_wrt_partial_v_1_hat, zeta_disturbance_observer] = Attitude_Backstepping_Controller_Complete_System_Stochastic(m, I_H, I_A, g, L2_to_L3, epsilon_1,epsilon_2, epsilon_3, epsilon_4, epsilon_5, epsilon_6,epsilon_7, D_1, D_2,min_eigen_value_D_1, gamma_1, K_1, K_2, K_3, K_4, c, n_1e, v_1_hat, v_1e, n_1d, n_1d_dot, n_1d_double_dot, n_1d_triple_dot, n_1d_quad_dot, angles, alpha_1, alpha_2, alpha_psi, alpha_psi_dot, alpha_psi_double_dot, R, B_Frame_angulare_velocity, Delta_time, tor_aero_hat_deterministic, sigma_hat_Stochastic_1, sigma_hat_Stochastic_dot_1,sigma_hat_Stochastic_2, sigma_hat_Stochastic_dot_2, Loop_Count, Sum_M1_to_M6, Partial_M_1_to_6_wrt_partial_n_1e, Partial_M_1_to_6_wrt_partial_v_1_hat, Partial_M_1_to_6_wrt_partial_n_1d_dot, Partial_M_1_to_6_wrt_partial_n_1d_double_dot, Omega, Partial_Omega_wrt_partial_n_1e, Partial_Omega_wrt_partial_v_1_hat , Partial_Omega_wrt_partial_n_1d_dot , Partial_Omega_wrt_partial_n_1d_double_dot, alpha_w_old, zeta_disturbance_observer, k_1d,n_1_old,K_02, n_1_tilda,n_1 );
        alpha_w_old = alpha_w;
          Controller_count=0;
    tau_aero_hat  = tor_aero_hat_deterministic;
       %}
    %%
   Control_Input = [(Omega.'*Omega)^0.5;tor];


    if clac_v_1_hat ==1
        alpha_q = Euler_to_Modified_rodrigues_paramater(alpha_2)
        R_2_alpha_q = R_2_MRP_calc( alpha_q)
        T_alpha_2_inverse                    = Angular_velocity_cordinant_transform(alpha_2);

        A_1d=[               cos_alpha_psi*1/(cos(alpha_phi)*(Omega.'*Omega)^0.5)               , sin_alpha_psi*1/(cos(alpha_phi)*(Omega.'*Omega)^0.5) , 0;
            -sin_alpha_psi*(cos(alpha_theta))^2/Omega_3                                          , cos_alpha_psi*(cos(alpha_theta))^2/Omega_3               , 0;
            0                                                                                    , 0 , 0];
```

```
    A_2d=[          sin_alpha_psi/(cos(alpha_phi)*(Omega.'*Omega)^0.5),          (-
cos_alpha_psi)/(cos(alpha_phi)*(Omega.'*Omega)^0.5),                              0
;
      (cos(alpha_theta))^2*cos_alpha_psi/Omega_3                                    ,
(cos(alpha_theta))^2*sin_alpha_psi/Omega_3                                       ,   -
(cos(alpha_theta))^2*tan(alpha_theta)/Omega_3;
      0                                                                            , 0
, 0                                         ] - [tan(alpha_phi)/(Omega.'*Omega); 0;
0]*Omega.';


q = Euler_to_Modified_rodrigues_paramater(angles);
R_1_MPR = R_MRP(q);


R = R_1_MPR;
    h_1    =    K_02*    (2*gamma_1*v_1e.'*G_1/(1+v_1e.'*v_1e)^0.5    -(1/L2_to_L3)*
q_e.'*R_2_alpha_q   *   T_alpha_2_inverse*A_2d*Partial_Omega_wrt_partial_v_1_hat   -
0.1*(w_e.'*w_e)*w_e.'*Partial_alpha_w_wrt_partial_v_1_hat/(gamma_5+(w_e.'*w_e)^2)^0.5)
.';
    v_1_hat_dot = -D_1*v_1_hat - g *[0;0;1] + (Omega.'*Omega)^0.5/m*R*[0;0;1] + K_02 *
n_1_tilda + h_2 + h_3 + h_4 + h_5;
    v_1_hat = v_1_hat_dot*dt + v_1_hat   ;
    %[v_1_hat ,E_Frame_Linear_velocity];
    v_1_hat_tilda =  E_Frame_Linear_velocity -v_1_hat;
    v_1_tilda =  E_Frame_Linear_velocity -v_1_hat;
%      v_1_hat =  E_Frame_Linear_velocity ;

%     [ v_1_hat_tilda.' t]
   %{
    K_01 =10*(2*eye(3)-0*[ abs(n_1e(1,1))/(1+n_1e(1,1)^2)^0.5 0 0;
                          0 abs(n_1e(2,1))/(1+n_1e(2,1)^2)^0.5 0;
                          0 0 abs(n_1e(3,1))/(1+n_1e(3,1)^2)^0.5]);
%}

    n_1_hat_dot = K_01*(n_1 - n_1_hat) + h_1/gamma_4  +v_1_hat;
    n_1_hat = n_1_hat  +  n_1_hat_dot*dt;
    n_1_tilda = (n_1 - n_1_hat) ;
    clac_v_1_hat =0;
  end

end

  control_time = t ;

  loop_count_2 = -1;



  q= Euler_to_Modified_rodrigues_paramater(angles );
alpha_2 * 180/pi;
alpha_q = Euler_to_Modified_rodrigues_paramater(alpha_2);
q_e = q - alpha_q;
```

```
[Partial_L_wrt_q_times_vector , Partial_L_wrt_alpha_q_times_vector,L_m] = calc_L_MRP_norm_squared (q, alpha_q,G_1.'*v_1e/(1+ v_1e.'*v_1e)^0.5);
temp_X = -2*gamma_1*L2_to_L3*(1/m)*(Omega.'*Omega)^0.5 * L_m*G_1.'*v_1e/(1+v_1e.'*v_1e)^0.5-K_3*q_e;
%% motor spped to propeller thrust force and rotation toruqe
%v_1_hat =E_Frame_Linear_velocity;

%%disturbance dynamics
        Air_Velocity  = E_Frame_Linear_velocity -Wind_Velocity ;
        Aero_Disturbance    = Aero_Disturbance_Calc(Air_Velocity, Air_Flow_Type, angles, t, Wind_Start_Time, Wind_frequancy, Hex_copter_height,copterradius);

        Collision_Disturbance  =Collision_Disturbance_calc(t, Collision_time, Collision_Point,F_Collision_x, F_Collision_y, F_Collision_z, E_Frame_Linear_possition) ;
        Torque_Disturbance =0*10* [0.1; 0.2; 0.1];
        %
        Collision_Disturbance = [ 0*Collision_Disturbance(1,1);
                                  0* Collision_Disturbance(2,1);
                                  0* Collision_Disturbance(3,1);
                                   Torque_Disturbance];
Torque_Disturbance = [-1;-1; 0.1];
 Collision_Disturbance = [ f_aero_Deterministic ;Torque_Disturbance];
        %}
                              tau_error = Torque_Disturbance - tau_aero_hat  ;
              %dW_last =  dW;
     dW                                     = sqrt(Delta_Time)*randn * [1;1;1];
     dW_2                                   = sqrt(Delta_Time)*randn * [1;1;1];
     dW_3                                   = sqrt(Delta_Time)*randn * [1;1;1];

     if Loop_Count == 1
        dW_initial    =  dW  ;
         %dW   =  dW   - dW_initial ;
     end
   I_XX_A = I_A(1,1);
   I_YY_A = I_A(2,2);
   I_ZZ_A = I_A(3,3);

     Coriolis = [ 0, I_ZZ_A * B_Frame_angulare_velocity(3,1), -I_YY_A * B_Frame_angulare_velocity(2,1);
                 -I_ZZ_A * B_Frame_angulare_velocity(3,1), 0, I_XX_A * B_Frame_angulare_velocity(1,1);
                 I_YY_A * B_Frame_angulare_velocity(2,1), -I_XX_A * B_Frame_angulare_velocity(1,1), 0 ];
     skew_w = [ 0, -B_Frame_angulare_velocity(3,1),  B_Frame_angulare_velocity(2,1);
                 B_Frame_angulare_velocity(3,1), 0, -B_Frame_angulare_velocity(1,1);
                -B_Frame_angulare_velocity(2,1), B_Frame_angulare_velocity(1,1), 0];
     last_coverience_dW                     = coverience*dW;%+last_coverience_dW;
     last_coverience_dW_2 = coverience_angulare_system*(D_2*eye(3)+0*0.05*eye(3)+Coriolis - skew_w * I_A)*dW_2;%+last_coverience_dW;
     last_coverience_dW_3                   = coverience_3*dW_3;
%%      // 6 dof dynamics
W_1 =W_1+ last_coverience_dW   ;
W_2 =W_2+ last_coverience_dW_2;
```

```
W_3 =W_3+ last_coverience_dW_3;

stochastic_disturbance_results(:,Loop_Count) =[W_1;W_2; W_3];
        Zeta_dot = SIX_DOF_Dynamics(R, E_Frame_angulare_velocity, angles,g,m,(I_H+
I_A),(I_XX+I_XX_A),(I_YY+I_YY_A),(I_ZZ+I_ZZ_A),                    Aero_Disturbance,
Collision_Disturbance,         Control_Input,         JTP,Zeta,Omega         ,         D_1,
E_Frame_Linear_velocity,D_2 );
n_1_old =n_1;
        % Advance system state.
      B_Frame_angulare_acceleration   = [Zeta_dot(4,1); Zeta_dot(5,1); Zeta_dot(6,1)];
      E_Frame_Linear_acceleration     = [Zeta_dot(1,1); Zeta_dot(2,1); Zeta_dot(3,1)];
      B_Frame_angulare_velocity   =   B_Frame_angulare_velocity   +   Delta_Time   *
B_Frame_angulare_acceleration  +  Inverse_I_H_plus_I_A*(last_coverience_dW_2  +
last_coverience_dW_3);
      E_Frame_angulare_velocity                                                    =
B_FRAME_ANGULARE_Velocity_TO_E_FRAME_Angulare_velocity(B_Frame_angulare_ve
locity, angles);

      angles = angles +  Delta_Time * E_Frame_angulare_velocity;
            E_Frame_Linear_velocity   =   E_Frame_Linear_velocity   +   Delta_Time   *
E_Frame_Linear_acceleration;
            E_Frame_Linear_velocity     =     E_Frame_Linear_velocity     +     (1/m)*
last_coverience_dW  ;
      E_Frame_Linear_possition   =   E_Frame_Linear_possition   +   Delta_Time   *
E_Frame_Linear_velocity;

      if (E_Frame_Linear_possition(3,1) <= 0)
         E_Frame_Linear_possition(3,1) =0;
      end


      Zeta  = [ E_Frame_Linear_acceleration;
                 B_Frame_angulare_velocity           ];
             psi =  angles(3);

angles = Check_Angular_Position( angles);
  angles(3) = psi;
Possition_OLD   = Possition;
      Possition = [ E_Frame_Linear_possition(1);
                              E_Frame_Linear_possition(2);
                              E_Frame_Linear_possition(3);
                              angles(1);
                              angles(2);
                              angles(3) ];
      Desired_Possition_OLD =    Desired_Possition;
%% function to plot the refference signal
   Input(:,Loop_Count) = Control_Input;
   if Loop_Count > 1
      Input_rate(:,Loop_Count)     =   (Input(:,Loop_Count)   -   Input(:,(Loop_Count-1)))   /
Delta_Time;
   end
   if Loop_Count == 1
      Input_rate(:,Loop_Count)  = 0;
   end
    x_1_d_dot_plot(:,Loop_Count) =  x_1_d_dot;
```

```
    n_2_error = n_2e;
              w_error = B_Frame_angulare_velocity - alpha_w;

      c_2          =          ((norm(coverience/m))^2/2+0.025+0.025+f_aero_Deterministic
.'*f_aero_Deterministic                    *min_eigen_value_D_1/3.9                +
sigma_n_1e.'*sigma_n_1e*min_eigen_value_D_1/6);
    infinite_generator_of_V_4                              =                       -
gamma_1*0.5*min_eigen_value_D_1*(1/delta_1_v_1_hat)*(sigma_n_1e.'*K_1*sigma_n_1e)
-2*gamma_1*sigma_v_1e.'*K_2*v_1e/(1+v_1e.'*v_1e)^0.5
+2*gamma_1*v_1e.'*K_1*sigma_dash_n_1e/delta_1_v_1_hat*v_1e/(1+v_1e.'*v_1e)^0.5    -
2*gamma_1*c_1*v_1e.'*(v_1e+2*alpha_1)/(2*delta_1_v_1_hat)  -2*gamma_1*v_1e.'*(D_1  -
(1/8)*min_eigen_value_D_1*eye(3)*(7.6-
sigma_n_1e.'*sigma_n_1e/7.6))*v_1e/(1+v_1e.'*v_1e)^0.5    -n_1_tilda.'*K_01*n_1_tilda   -
v_1_tilda.'*(D_1-eye(3)*min_eigen_value_D_1*3.9/4)*v_1_tilda    -gamma_2*q_e.'*(K_3    -
(5/5.1)*min(eig(K_3))                    *                   eye(3))*q_e               -
gamma_3*(norm(w_e.'*w_e))^2*w_e.'*(K_4+D_2)*w_e/(1+(norm(w_e))^4)^0.75                -
(norm(w_e.'*w_e))^2*w_e.'*2*gamma_1*c_1*w_e/(1+(norm(w_e))^4)                        +
c_2/(1+(norm(w_e))^4)  ;
    V_4     =     (gamma_1*0.5*min_eigen_value_D_1     *     ((1+n_1e(1,1)^2)^0.5      +
(1+n_1e(2,1)^2)^0.5  +  (1+n_1e(3,1)^2)^0.5  -  3)  +  2*gamma_1*(1+v_1e.'*v_1e)^0.5    +
gamma_2*0.5*q_e.'*q_e            +gamma_3*(1+(norm(w_e))^4)^0.25+0.5                  *
(n_1_tilda.'*n_1_tilda+v_1_tilda.'*v_1_tilda)  +  0.5*(norm(coverience_angulare_system)^4  -
sigma_hat_Stochastic_1        )^2/1000000        +        0.5*(norm(coverience_3)^4       -
sigma_hat_Stochastic_2 )^2/-2*gamma_1-gamma_3 );
%                                              infinite_generator_of_V_4        =       -
gamma_1*0.5*min_eigen_value_D_1*(1/delta_1_v_1_hat)*(sigma_n_1e.'*K_1*sigma_n_1e)
-2*gamma_1*sigma_v_1e.'*K_2*v_1e/(1+v_1e.'*v_1e)^0.5
+2*gamma_1*v_1e.'*K_1*sigma_dash_n_1e/delta_1_v_1_hat*v_1e/(1+v_1e.'*v_1e)^0.5    -
2*gamma_1*c_1*v_1e.'*(v_1e+2*alpha_1)/(2*delta_1_v_1_hat)  -2*gamma_1*v_1e.'*(D_1  -
(1/8)*min_eigen_value_D_1*eye(3)*(7.6-
sigma_n_1e.'*sigma_n_1e/7.6))*v_1e/(1+v_1e.'*v_1e)^0.5    -n_1_tilda.'*K_01*n_1_tilda   -
v_1_tilda.'*(D_1-eye(3)*min_eigen_value_D_1*3.9/4)*v_1_tilda    -gamma_2*q_e.'*(K_3    -
(5/5.1)*min(eig(K_3))                    *                   eye(3))*q_e               -
gamma_3*(norm(w_e.'*w_e))^2*w_e.'*(K_4+D_2)*w_e/(1+(norm(w_e))^4)^0.75                -
(norm(w_e.'*w_e))^2*w_e.'*2*gamma_1*c_1*w_e/(1+(norm(w_e))^4)                        +
c_2/(1+(norm(w_e))^4-   0.5*(norm(coverience_angulare_system)^4  -sigma_hat_Stochastic
)^2/1000000++ 0.5*(norm(coverience_angulare_system)^4 *2 )^2/1000000  )  ;
V_4_on_L_V_4                                          =                            (-
gamma_1*0.5*min_eigen_value_D_1*(1/delta_1_v_1_hat)*(sigma_n_1e.'*K_1*sigma_n_1e)
-2*gamma_1*sigma_v_1e.'*K_2*v_1e/(1+v_1e.'*v_1e)^0.5
+2*gamma_1*v_1e.'*K_1*sigma_dash_n_1e/delta_1_v_1_hat*v_1e/(1+v_1e.'*v_1e)^0.5    -
2*gamma_1*c_1*v_1e.'*(v_1e+2*alpha_1)/(2*delta_1_v_1_hat)  -2*gamma_1*v_1e.'*(D_1  -
(1/8)*min_eigen_value_D_1*eye(3)*(7.6-
sigma_n_1e.'*sigma_n_1e/7.6))*v_1e/(1+v_1e.'*v_1e)^0.5    -n_1_tilda.'*K_01*n_1_tilda   -
v_1_tilda.'*(D_1-eye(3)*min_eigen_value_D_1*3.9/4)*v_1_tilda    -gamma_2*q_e.'*(K_3    -
(5/5.1)*min(eig(K_3))                    *                   eye(3))*q_e               -
gamma_3*(norm(w_e.'*w_e))^2*w_e.'*(K_4+D_2)*w_e/(1+(norm(w_e))^4)^0.75                -
(norm(w_e.'*w_e))^2*w_e.'*2*gamma_1*c_1*w_e/(1+(norm(w_e))^4)                        +
c_2/(1+(norm(w_e))^4-              0.5*(norm(coverience_angulare_system)^4            -
sigma_hat_Stochastic_1 )^2/1000000- 0.5*(norm(coverience_3)^4 -sigma_hat_Stochastic_2
)^2+ 0.5*(norm(coverience_angulare_system)^4 *2 )^2/1000000  )  ) / V_4;


if Loop_Count ==1
```

```
    %V_sum_initial = ( ((1+n_1e(1,1)^2)^0.5 + (1+n_1e(2,1)^2)^0.5 + (1+n_1e(3,1)^2)^0.5 - 3)
+ (1+v_1e.'*v_1e)^0.5 + q_e.'*q_e +(1+(norm(w_e))^4)^0.25+
(n_1_tilda.'*n_1_tilda+v_1_tilda.'*v_1_tilda) + (norm(coverience_angulare_system)^4 -
sigma_hat_Stochastic )^2 -3)
    V_sum_initial = V_4;
    b_7 = (1/3)*((1+n_1e(1,1)^2)^-0.5 + (1+n_1e(2,1)^2)^-0.5 + (1+n_1e(3,1)^2)^-
0.5)/delta_1_v_1_hat * c_2/(1+(norm(w_e))^4) *(2+alpha_1.'*alpha_1)/2;
%    b_6 = min([(gamma_1*0.5*min_eigen_value_D_1*(1/delta_1_v_1_hat)*min(eig(K_1)) -
c_1*(1/delta_1_v_1_hat)/(1+(norm(w_e))^4)) ,min(eig(D_1-
eye(3)*min_eigen_value_D_1*3.9/4))])/max([gamma_1*0.5*min_eigen_value_D_1, 2 *
gamma_1, gamma_2/2, 0.5, gamma_3]);
    b_6 = min([
        (gamma_1*0.5*min_eigen_value_D_1*(min(eig(K_1)) -
c_1)*(1/delta_1_v_1_hat)*((1+n_1e(1,1)^2)^-0.5 + (1+n_1e(2,1)^2)^-0.5 + (1+n_1e(3,1)^2)^-
0.5))/(gamma_1*0.5);
        (2*gamma_1*min(eig(D_1-
eye(3)*min_eigen_value_D_1/8*(epsilon_1+3/epsilon_1)))/(2*gamma_1));
        (2*gamma_1*(min(eig(K_2))-min(eig(K_1)))/2*gamma_1);
        (min(eig(K_01))/0.5);
        (min(eig(D_1-eye(3)*min_eigen_value_D_1*3.9/4))/0.5);
        (gamma_2*min(eig(K_3))/(0.5*gamma_2));
        (gamma_3*min(eig(K_4))/(gamma_3))
        ((1+n_1e(1,1)^2)^-0.5 + (1+n_1e(2,1)^2)^-0.5 + (1+n_1e(3,1)^2)^-
0.5)/delta_1_v_1_hat/0.5])
    b_7/b_6
end

V_sum =     V_sum_initial *exp(-b_6*t) + b_7/b_6;
    results(:,Loop_Count) = [ Possition;%1-6
                              t;%time ;%7
                              B_Frame_angulare_velocity ;%8-10
                              tau_aero_hat;%111 - 13
                              w_error;%14-16
                              n_2_error;%17-19
                              alpha_w_dot ;%20-22
                             Torque_Disturbance;%23-25
                              tau_error;%26-28
                              alpha_2;%eta_2_d;%   ];%eta_2_noise];%29 - 31
                              0;%exponentialMA ;%32
                              Delta_time;%33
                              v_1_hat;%q%34 - 36
                              alpha_q;%37 - 39
                              0;%count_3; %40
                              0;%reread; %41
                              0;%Time_to_read_seriel;%42
                              sigma_hat_Stochastic_1;%43
                              OMEGA;%44 - 46
                              v_1_hat_tilda;% 47 - 49
                              h_4; % 50 - 52
                              h_5; % 53 - 55
                              alpha_w;
                              n_1_tilda;
                              E_Frame_Linear_velocity ;
                              v_1_hat;
                              v_1e;
```

```
                                    temp_X;
                                     infinite_generator_of_V_4 ;
                                     V_4;
                                     V_sum;
                                     last_coverience_dW_2/dt;
                                     MOTOR_FORCE_CALC*Control_Input;
                                     sigma_hat_Stochastic_2;
                                     last_coverience_dW/dt;
                                     last_coverience_dW_3/dt;
                                     norm(w_e)
                                     h_1;
                                     h_2;
                                     h_3;];



   x_1_d_dot_old            =x_1_d_dot;
   x_1_d_double_dot_old  = x_1_d_double_dot;
   x_1_d_triple_dot_old    =x_1_d_triple_dot ;
   alpha_PSI_dot_old       = alpha_PSI_dot ;

   %{
for i = 1:3
   Refference_signal(i,Loop_Count) = Desired_Possition(i,1) ;
end
%}
Refference_signal(:,Loop_Count) = [ n_1d ;
                                            alpha_2];
%alpha_2 = [0;0;0];
Refference_signal(4,Loop_Count) = alpha_2(1,1);
Refference_signal(5,Loop_Count) = alpha_2(2,1);
Loop_Count              = Loop_Count + 1;
%% END OF SIMULATION LOOP%%
t;
%      [ v_1_hat_tilda.' t]
     [ n_1e.' v_1_hat_tilda.'  (1+v_1e.'*v_1e)^0.5/delta_1_v_1_hat t]

loop_count_2 = loop_count_2 + 1;
Controller_count = Controller_count+1;
end
simulation_end_time = clock;
simulation_elapsed_time = 60*60*(simulation_end_time(4) - simulation_start_time(4)) + 60*(simulation_end_time(5)-simulation_start_time(5)) + (simulation_end_time(6)-simulation_start_time(6));


%% check initial roll refference as it is 90 degrees
```

```
function [ tor, alpha_w, alpha_w_dot, tor_aero_hat_deterministic, tor_aero_hat_deterministic_dot, sigma_hat_Stochastic_1, sigma_hat_Stochastic_dot_1,sigma_hat_Stochastic_2, sigma_hat_Stochastic_dot_2, h_4, h_5, q_e, w_e, Partial_alpha_w_wrt_partial_v_1_hat, zeta_disturbance_observer] = Attitude_Backstepping_Controller_Complete_System_Stochastic(m, I_H, I_A, g, L2_to_L3, epsilon_1,epsilon_2, epsilon_3, epsilon_4, epsilon_5, epsilon_6,epsilon_7, D_1, D_2,min_eigen_value_D_1, gamma_1, K_1, K_2, K_3, K_4, c, n_1e, v_1_hat, v_1e, n_1d, n_1d_dot, n_1d_double_dot, n_1d_triple_dot, n_1d_quad_dot, angles, alpha_1, alpha_2, alpha_psi, alpha_psi_dot, alpha_psi_double_dot, R, B_Frame_angulare_velocity, Delta_time, tor_aero_hat_deterministic, sigma_hat_Stochastic_1, sigma_hat_Stochastic_dot_1,sigma_hat_Stochastic_2, sigma_hat_Stochastic_dot_2, Loop_Count, Sum_M1_to_M6, Partial_M_1_to_6_wrt_partial_n_1e, Partial_M_1_to_6_wrt_partial_v_1_hat, Partial_M_1_to_6_wrt_partial_n_1d_dot, Partial_M_1_to_6_wrt_partial_n_1d_double_dot, Omega, Partial_Omega_wrt_partial_n_1e, Partial_Omega_wrt_partial_v_1_hat , Partial_Omega_wrt_partial_n_1d_dot , Partial_Omega_wrt_partial_n_1d_double_dot, alpha_w_old, zeta_disturbance_observer, K_d, n_1_old, K_02, n_1_tilda,n_1 )
constant = 1;
min_eigen_value_K_3 = (5/5.1)*min(eig(K_3));
gamma_3 = 0.1;
%function U = Attitude_Backstepping_Controller_Complete_System_Stochastic(m, I_H, g, L2_to_L3, epsilon_1,epsilon_2, epsilon_3, epsilon_4, epsilon_5, epsilon_6,epsilon_7, D_1, D_2,min_eigen_value_D_1, gamma_1, K_1, K_2, K_3, K_4, c, n_1e, v_1_hat, v_1e, n_1d, n_1d_dot, n_1d_double_dot, n_1d_triple_dot, n_1d_quad_dot, angles, alpha_1, alpha_2, alpha_psi, alpha_psi_dot, alpha_psi_double_dot, R, B_Frame_angulare_velocity, Delta_time, tor_aero_hat_deterministic, sigma_hat_Stochastic ,Loop_Count, Sum_M1_to_M6, Partial_M_1_to_6_wrt_partial_n_1e, Partial_M_1_to_6_wrt_partial_v_1_hat, Partial_M_1_to_6_wrt_partial_n_1d_dot, Partial_M_1_to_6_wrt_partial_n_1d_double_dot, Omega, Partial_Omega_wrt_partial_n_1e, Partial_Omega_wrt_partial_v_1_hat , Partial_Omega_wrt_partial_n_1d_dot , Partial_Omega_wrt_partial_n_1d_double_dot)
sigma_n_1e= [ n_1e(1,1)/(1+n_1e(1,1)^2)^0.5;
              n_1e(2,1)/(1+n_1e(2,1)^2)^0.5;
              n_1e(3,1)/(1+n_1e(3,1)^2)^0.5];
sigma_dash_n_1e = [ 1/(1+n_1e(1,1)^2)^1.5, 0, 0;
                    0, 1/(1+n_1e(2,1)^2)^1.5, 0;
                    0, 0, 1/(1+n_1e(3,1)^2)^1.5];
delta_1_v_1_hat = 1 + 0.5 * ((v_1_hat.'))*v_1_hat;
n_1e_dot = v_1_hat - n_1d_dot;
G_1 = eye(3)-(K_1*sigma_n_1e*(v_1_hat.')/delta_1_v_1_hat^2);
n_2 = angles;
(n_2-alpha_2)*180/pi;
%dt = Delta_time;
%n_2e = angles -alpha_2;
q= Euler_to_Modified_rodrigues_paramater(n_2);
alpha_2 * 180/pi;
alpha_q = Euler_to_Modified_rodrigues_paramater(alpha_2);
q_e = q - alpha_q
R_2_alpha_q = R_2_MRP_calc( alpha_q);
T_alpha_2_inverse = Angular_velocity_cordinant_transform(alpha_2);
h_2 =min_eigen_value_D_1*0.5* gamma_1*(sigma_n_1e.').';
h_3 = 2*gamma_1*(v_1e.'*K_1*sigma_dash_n_1e/delta_1_v_1_hat).'/(1+v_1e.'*v_1e)^0.5;


alpha_phi = alpha_2(1,1);%alpha_PHI;
```

```
alpha_theta = alpha_2(2,1);%alpha_THETA;
alpha_psi = alpha_2(3,1);%alpha_PSI;

Omega_3 = Omega(3,1);
cos_alpha_psi =cos(alpha_psi);
sin_alpha_psi = sin(alpha_psi);

R_alpha_q = R_MRP(alpha_q);
R_q = R_MRP(q);
%H_alpha_q_qe = R_q - R_alpha_q;
%H_alpha_q_q_e_norm_4 = (trace(H_alpha_q_qe.'*H_alpha_q_qe))^2;
R_2_q = R_2_MRP_calc(q);
R_2_q_inverse  = (16/(1+q.'*q)^2)*R_2_q.';
R_2_alpha_q=R_2_MRP_calc(alpha_q);
A_1d=[                    cos_alpha_psi*1/(cos(alpha_phi)*(Omega.'*Omega)^0.5)                    , sin_alpha_psi*1/(cos(alpha_phi)*(Omega.'*Omega)^0.5) , 0;
          -sin_alpha_psi*(cos(alpha_theta))^2/Omega_3                                                      , cos_alpha_psi*(cos(alpha_theta))^2/Omega_3              , 0;
          0                                                                                              , 0 , 0];
A_2d=[                    sin_alpha_psi/(cos(alpha_phi)*(Omega.'*Omega)^0.5),                    (-cos_alpha_psi)/(cos(alpha_phi)*(Omega.'*Omega)^0.5),                    0 ;
          (cos(alpha_theta))^2*cos_alpha_psi/Omega_3                                                       , (cos(alpha_theta))^2*sin_alpha_psi/Omega_3                                          ,  -(cos(alpha_theta))^2*tan(alpha_theta)/Omega_3;
          0                                                                                              , 0 , 0                                                              ] - [tan(alpha_phi)/(Omega.'*Omega); 0; 0]*Omega.';
h_4 = -(q_e.'*R_2_alpha_q*T_alpha_2_inverse*A_2d*Partial_Omega_wrt_partial_n_1e).';

h_4 = h_4 / L2_to_L3;
Omega_d = Partial_Omega_wrt_partial_n_1e*(v_1_hat - n_1d_dot) + Partial_Omega_wrt_partial_v_1_hat  *(-D_1*v_1_hat - g*[0;0;1]+(((Omega.'*Omega)^0.5)/m) * R*[0;0;1] + h_2 + h_3+h_4) + Partial_Omega_wrt_partial_n_1d_dot  * n_1d_double_dot + Partial_Omega_wrt_partial_n_1d_double_dot * n_1d_triple_dot;

[Partial_L_wrt_q_times_vector , Partial_L_wrt_alpha_q_times_vector,L] = calc_L_MRP_norm_squared (q, alpha_q,G_1.'*v_1e/(1+ v_1e.'*v_1e)^0.5);
%Omega_d
alpha_w = R_2_q_inverse * ( -K_3 *q_e+ R_2_alpha_q*T_alpha_2_inverse*(A_2d*(Omega_d + Partial_Omega_wrt_partial_n_1e*2*gamma_1* G_1.'*v_1e/(1+v_1e.'*v_1e)^0.5 )+ (A_1d*Omega + [0;0;1])*alpha_psi_dot)-2*gamma_1*L2_to_L3*(1/m)*(Omega.'*Omega)^0.5 * L*G_1.'*v_1e/(1+v_1e.'*v_1e)^0.5);

[Partial_h_4_wrt_n_1e, Partial_h4_wrt_partial_n_1d_dot, Partial_h4_wrt_partial_n_1d_double_dot, Partial_h4_wrt_partial_v_1_hat, Partial_h_4_wrt_q, Partial_h_4_wrt_alpha_psi ] = h_4_Partial_derivatives(m, epsilon_2, epsilon_3, D_1, min_eigen_value_D_1, gamma_1, K_1, K_2, c, n_1e, v_1_hat, v_1e, n_1d_dot, n_1d_double_dot, q_e, Sum_M1_to_M6, Partial_M_1_to_6_wrt_partial_n_1e, Partial_M_1_to_6_wrt_partial_v_1_hat, Partial_M_1_to_6_wrt_partial_n_1d_dot, Partial_M_1_to_6_wrt_partial_n_1d_double_dot, alpha_1, Partial_Omega_wrt_partial_n_1e, Partial_Omega_wrt_partial_v_1_hat , Partial_Omega_wrt_partial_n_1d_dot, Partial_Omega_wrt_partial_n_1d_double_dot, A_1d, A_2d,
```

```
T_alpha_2_inverse,R_2_alpha_q ,alpha_psi, alpha_phi, alpha_theta, alpha_q, Omega,L2_to_L3 );
%[Omega_2d_dot_wrt_n_1e, Omega_2d_dot_wrt_v_1_hat, Omega_2d_dot_wrt_n_1d_dot, Omega_2d_dot_wrt_n_1d_double_dot, Omega_2d_dot_wrt_n_1d_triple_dot, Omega_2d_dot_wrt_q, Omega_2d_dot_wrt_alpha_psi ] = Omega_d_dot(m, epsilon_2, epsilon_3, D_1, min_eigen_value_D_1, gamma_1, K_1, K_2, c, n_1e, v_1_hat, v_1e, n_1d_dot, n_1d_double_dot, n_1d_triple_dot,q , Omega, Sum_M1_to_M6, Partial_M_1_to_6_wrt_partial_n_1e, Partial_M_1_to_6_wrt_partial_v_1_hat, Partial_M_1_to_6_wrt_partial_n_1d_dot, Partial_M_1_to_6_wrt_partial_n_1d_double_dot, h_4, R, alpha_1, Partial_Omega_wrt_partial_n_1e, Partial_Omega_wrt_partial_v_1_hat,Partial_Omega_wrt_partial_n_1d_dot,
Partial_Omega_wrt_partial_n_1d_double_dot, Partial_h_4_wrt_n_1e, Partial_h4_wrt_partial_n_1d_dot, Partial_h4_wrt_partial_n_1d_double_dot, Partial_h4_wrt_partial_v_1_hat, Partial_h_4_wrt_q, Partial_h_4_wrt_alpha_psi);
[Partial_alpha_w_wrt_partial_n_1e, Partial_alpha_w_wrt_partial_v_1_hat, Partial_alpha_w_wrt_partial_n_1d_dot, Partial_alpha_w_wrt_partial_n_1d_double_dot, Partial_alpha_w_wrt_partial_n_1d_triple_dot, Partial_alpha_w_wrt_partial_q, Partial_alpha_w_wrt_partial_alpha_psi, Partial_alpha_w_wrt_partial_alpha_psi_dot] = Partial_derivatives_of_alpha_w(m, epsilon_2, epsilon_3, epsilon_4, epsilon_5, D_1, min_eigen_value_D_1, gamma_1, K_1, K_2, K_3, c, n_1e, v_1_hat, v_1e, n_1d_dot, n_1d_double_dot, n_1d_triple_dot, n_2, alpha_1, alpha_2, q, alpha_q, Omega, Sum_M1_to_M6, Partial_M_1_to_6_wrt_partial_n_1e, Partial_M_1_to_6_wrt_partial_v_1_hat, Partial_M_1_to_6_wrt_partial_n_1d_dot, Partial_M_1_to_6_wrt_partial_n_1d_double_dot, h_4, R, Partial_Omega_wrt_partial_n_1e, Partial_Omega_wrt_partial_v_1_hat,Partial_Omega_wrt_partial_n_1d_dot,
Partial_Omega_wrt_partial_n_1d_double_dot, Omega_d, Partial_h_4_wrt_n_1e, Partial_h4_wrt_partial_n_1d_dot, Partial_h4_wrt_partial_n_1d_double_dot, Partial_h4_wrt_partial_v_1_hat, Partial_h_4_wrt_q, Partial_h_4_wrt_alpha_psi, A_1d, A_2d, alpha_psi_dot,L2_to_L3);
%Omega_2d_dot_wrt_v_1_hat
w = B_Frame_angulare_velocity;
w_e = w - alpha_w;

%alpha_w
h_5 = -w_e.'*w_e*(w_e.'*Partial_alpha_w_wrt_partial_n_1e).';
h_5 =  0.1* h_5/(constant+(w_e.'*w_e)^2)^0.5;


  if isreal(  Partial_alpha_w_wrt_partial_n_1e)
     real_numbers =1;
  end
  if real_numbers ==0
     pause(10000000);
  end
  %real_numbers=0;




%[n_1e, n_2e*180/pi, n_1_tilda, v_1_hat_tilda]

%Controller_count = 0;

v_1_hat_dot =-D_1*v_1_hat - g *[0;0;1] + (Omega.'*Omega)^0.5/m*R*[0;0;1]  + h_2 + h_3 + h_4 + h_5;
```

```
alphа_w_dot = Partial_alpha_w_wrt_partial_n_1e* n_1e_dot + Partial_alpha_w_wrt_partial_v_1_hat* v_1_hat_dot + Partial_alpha_w_wrt_partial_n_1d_dot*n_1d_double_dot + Partial_alpha_w_wrt_partial_n_1d_double_dot*n_1d_triple_dot + Partial_alpha_w_wrt_partial_n_1d_triple_dot*n_1d_quad_dot + Partial_alpha_w_wrt_partial_q * R_2_q * w + Partial_alpha_w_wrt_partial_alpha_psi * alpha_psi_dot + Partial_alpha_w_wrt_partial_alpha_psi_dot * alpha_psi_double_dot;
w_e = w - alpha_w;
if Loop_Count == 1
   tor_aero_hat_deterministic =[0;0;0];
end
I_XX = I_H(1,1);
I_YY = I_H(2,2);
I_ZZ = I_H(3,3);

Coriolis = [ 0, I_ZZ * B_Frame_angulare_velocity(3,1), -I_YY * B_Frame_angulare_velocity(2,1);
            -I_ZZ * B_Frame_angulare_velocity(3,1), 0, I_XX * B_Frame_angulare_velocity(1,1);
            I_YY * B_Frame_angulare_velocity(2,1), -I_XX * B_Frame_angulare_velocity(1,1), 0 ];
    I_XX_A = I_A(1,1);
    I_YY_A = I_A(2,2);
    I_ZZ_A = I_A(3,3);

      Coriolis_A = [ 0, I_ZZ_A * B_Frame_angulare_velocity(3,1), -I_YY_A * B_Frame_angulare_velocity(2,1);
            -I_ZZ_A * B_Frame_angulare_velocity(3,1), 0, I_XX_A * B_Frame_angulare_velocity(1,1);
            I_YY_A * B_Frame_angulare_velocity(2,1), -I_XX_A * B_Frame_angulare_velocity(1,1), 0 ];
      skew_w = [ 0, -B_Frame_angulare_velocity(3,1), B_Frame_angulare_velocity(2,1);
              B_Frame_angulare_velocity(3,1), 0, -B_Frame_angulare_velocity(1,1);
              -B_Frame_angulare_velocity(2,1), B_Frame_angulare_velocity(1,1), 0];
          Multiplicitiveand_aditive_noise = inv(I_H+I_A)*(D_2*eye(3)+0*0.05*eye(3)+ Coriolis_A +skew_w *I_A);

%sigma_Stochastic_1= Projection_algorithm(gamma_1^2*(9*epsilon_7)*(norm(Multiplicitiveand_aditive_noise ))^4*(norm(w_e))^4/(constant+(norm(w_e))^4)^1.0,sigma_hat_Stochastic_1,sigma_hat_Stochastic_dot_1 , xi_aero_Stochastic, tor_aero_Stochastic_MAX, gamma_projection_algorthim, Delta_time);
sigma_Stochastic_1 = [0;0];
sigma_hat_Stochastic_1 =1*sigma_Stochastic_1(1,1)
sigma_hat_Stochastic_dot_1 = sigma_Stochastic_1(2,1);
%alpha_w_dot = (alpha_w - alpha_w_old)/Delta_time;
%sigma_Stochastic = 0;
if sigma_hat_Stochastic_1 <0
   sigma_hat_Stochastic_1 =0;
%    pause(100)
end
%{
if sigma_hat_Stochastic >1
   sigma_hat_Stochastic =1;
```

```
%    pause(100)
end
%}
%pause(10000)


mew_projection_algorthim_2 =0.00001;%0.00812/norm(inv(I_H))^4;
tor_aero_Stochastic_MAX=0.1^4;
xi_aero_Stochastic=0.1*tor_aero_Stochastic_MAX;

sigma_Stochastic_2= Projection_algorithm(gamma_3^2*(9/4*epsilon_6)*(norm(inv(I_H+I_A)))^4*(norm(w_e))^4/(constant+(norm(w_e))^4)^0.75,sigma_hat_Stochastic_2,sigma_hat_Stochastic_dot_2 , xi_aero_Stochastic, tor_aero_Stochastic_MAX, mew_projection_algorthim_2, Delta_time);


sigma_hat_Stochastic_2       =1*sigma_Stochastic_2(1,1);
sigma_hat_Stochastic_dot_2  =  sigma_Stochastic_2(2,1);
%alpha_w_dot = (alpha_w - alpha_w_old)/Delta_time;
%sigma_Stochastic = 0;
if sigma_hat_Stochastic_2 <0
   sigma_hat_Stochastic_2 =0;
%    pause(100)
end


%sigma_hat_Stochastic_2 ;

 w_e_dot            = -K_4 * w_e;
 %additive noise only
%  tor = I_H*(  D_2  *alpha_w  +  w_e_dot  - 0.1*((9*epsilon_7/4)*(norm(inv(I_H)))^4*sigma_hat_Stochastic)*w_e/(1+(w_e .'*w_e )^2)^0.75 + alpha_w_dot - 10*((1/L2_to_L3)/(4*min_eigen_value_K_3)^2)*norm(R_2_alpha_q)^4/4 *w_e/(1+(w_e.'*w_e)^2)^3.25- 10*(1/L2_to_L3)*(q_e.'*R_2_alpha_q*w_e)*w_e/(1+(w_e.'*w_e)^2)^0.25 - ((2*gamma_1*v_1e.'*G_1 /(1+v_1e.'*v_1e)^0.5- (1/L2_to_L3)*q_e.'*R_2_alpha_q*T_alpha_2_inverse)*Partial_alpha_w_wrt_partial_n_1e).') - tor_aero_hat_deterministic + cross(B_Frame_angulare_velocity, (I_H*B_Frame_angulare_velocity));
%additive and multiplicitive noise

%sigma_hat_Stochastic_2 = 0.01;
%sigma_hat_Stochastic_1 = 1;
%sigma_hat_Stochastic_2 = 0.0001;
%sigma_hat_Stochastic_1 = 1;

sigma_hat_Stochastic_2 = 0;
sigma_hat_Stochastic_1 = 0;
tor_aero_hat_deterministic_dot = [0;0;0];
tor_aero_hat = tor_aero_hat_deterministic  ;
tor = (I_H+I_A)*(  D_2  *alpha_w  +  w_e_dot  - gamma_3*((9*epsilon_7)*(norm(Multiplicitiveand_aditive_noise))^4*sigma_hat_Stochastic_1)*w_e/(constant+1*(w_e .'*w_e )^2)^0.25  -  gamma_3*  (9/4*epsilon_6)*(norm(inv(I_H+I_A)
```

```
))^4*sigma_hat_Stochastic_2*w_e/(constant+(norm(w_e))^4)^0.25+ alpha_w_dot -
100*constant^2*((1/L2_to_L3)/(4*min_eigen_value_K_3)^2)*norm(R_2_alpha_q)^4/4
*w_e/(constant+(w_e.'*w_e)^2)^3.5-
10*(1/L2_to_L3)*(q_e.'*R_2_alpha_q*w_e)*w_e/(constant+(w_e.'*w_e)^2)^0.5 +
((2*gamma_1*v_1e.'*G_1 /(1+v_1e.'*v_1e)^0.5-
(1/L2_to_L3)*q_e.'*R_2_alpha_q*T_alpha_2_inverse)*Partial_alpha_w_wrt_partial_n_1e).') -
tor_aero_hat_deterministic + cross(B_Frame_angulare_velocity, ((I_H +
I_A)*B_Frame_angulare_velocity));


%-K_d *( -cross(B_Frame_angulare_velocity, ((I_H+I_A) *B_Frame_angulare_velocity))+tor
+tor_aero_hat);


   zeta_disturbance_observer= zeta_disturbance_observer- K_d*(I_H+I_A)*((inv(I_H+I_A)
*(tor-cross(B_Frame_angulare_velocity, ((I_H + I_A)*B_Frame_angulare_velocity)))-
alpha_w_dot -Partial_alpha_w_wrt_partial_v_1_hat*K_02*n_1_tilda
+tor_aero_hat+Partial_alpha_w_wrt_partial_n_1e*v_1_hat)* Delta_time -
Partial_alpha_w_wrt_partial_n_1e*(n_1-n_1_old));
if Loop_Count ==0
   zeta_disturbance_observer= -K_d*(I_H+I_A)*w_e;
end
   tor_aero_hat_deterministic  = zeta_disturbance_observer+ K_d*(I_H+I_A)*w_e;
```

All other functions ccan be found in the next Appen

Appendix K –Simulation Code, Stochastic One-step Ahead Backstepping Controller for Quadrotor Aircraft

```
close all
clc
clear
alpha_w_old = [0;0;0];
counting_Loop  = 0
End_Time                    =100%398.44;
Delta_Time                  = 1/1000;
controller_update_time=Delta_Time   ;
controller_time_steps = controller_update_time/Delta_Time;
Z_Desired_Possition         = 25;
X_START              = 0;
Y_START              = 0;
Z_START              = 0;
PHI_START            = 7;
THETA_START    =9;
PSI_START           = 0;

K_01 =20*  [1 0 0;
           0 1 0;
           0 0 1 ] ;
K_01 =1* eye(3);
K_01 =1* [ 1.5 0 0;
             0 1.5 0;
             0 0 1];
K_01 =1* eye(3);
gamma_4 =    1   ;
K_02 = gamma_4*eye(3);
K_1 = 0.5*eye(3);
K_1 = 0.85*eye(3);
K_2 = 2*eye(3);
K_3 = 5*eye(3);
K_4 =10*eye(3);
%K_4 = 5*eye(3);
  %      gamma =0.5
F_Collision_x = 0;
F_Collision_y = 0;
 F_Collision_z = 0;
Collision_time              = -1;
Wind_Start_Time               = 0;
Wind_frequancy               = [1; 1; 1];
Wind_Velocity               = [0 ;0;0];
Air_Flow_Type                 = 0;
Collision_Point             = 0;
Motor_spped                 = zeros(6, 1);
Motor_Voltage                = zeros(6, 1);
```

```
MOTOR_Spped_Voltage_Ratio   = 570;
PROP_Thrust_Coef          =  5.83368*10^-6;  %0.583368*10^-6;
PROP_Rotation_Coef      = PROP_Thrust_Coef /20 ; %0.27 * 10^-6;%0.00623535 * 10^-6;
Delta_OMEGA           = 0;
%% model paramters physical
Copter_Radius              = 0.30;
Copter_Radius_Along_X_Axis  = 0.26;
Copter_Radius_Along_Y_Axis  = 0.15;
g   = 9.81;
m = 2.23;
JTP =  2.5172e-006;
Motor_radius = 20/1000                        ;
copterradius = Copter_Radius;
Two_pi = 2*pi;
pitch =     4;
Propeller_radius =        (7.8/2)*.75;
alpha = atan(pitch/(2*pi*Propeller_radius ));
alpha = alpha * 180/pi;
Blade_pitch_angle = alpha ;
INCH_to_meter          = 0.0254    ;
BLade_diametre         = 6         ;
L                      = BLade_diametre * INCH_to_meter/2  ;
 p=1.1839; %     wikipedia at 25 degrees C
Blade_leangth          = L;
Effective_blade_leangth = 0.75*Blade_leangth     ;
total_Propeller_surface_area =    2.8873*10^-5;
air_density     =1.1839;
Hex_copter_height   =   1;

I_XX = 0.016507;
I_YY = 0.016507;
I_ZZ = 0.016284;

I_H     = [ I_XX, 0, 0;
            0, I_YY, 0;
            0, 0, I_ZZ];
I_A     = 0.5 * I_H;
%% mathamtical deffinitions
T_inverse_Phi_derivative    = zeros(3, 3);
T_inverse_Theta_derivative  = zeros(3, 3);
Plot_info   = zeros(6, 1);
T_inverse                  = zeros(3, 3);
T_inverse_inverse          = zeros(3, 3);
E_Frame_Linear_possition       = zeros(3, 1);
E_Frame_Linear_velocity        = zeros(3, 1);
angles                     = zeros(3, 1);
Control_Signal_in_F            = zeros(3, 1);

temp                       = zeros(2, 1);
T_Collision                  = zeros(3, 1);
Possition                   = zeros(6, 1);
Possition_OLD              =  zeros(6, 1);
Desired_Possition_OLD =  zeros(6, 1);
 Zeta                        =  zeros(6, 1);
 Motor_Speed                 =  zeros(6, 1);

%% dessired possitions
Time_steps                 = End_Time / Delta_Time;
Lyapunov_Signal = zeros(2, Time_steps);
 x_1_d_dot_plot = zeros(3, Time_steps);
```

```
MOTOR =  zeros(6, Time_steps);
%results                      = zeros(35, Time_steps);
results                   = zeros(100, Time_steps);
stochastic_disturbance_results  = zeros(9, Time_steps);
W_1 = [0;0;0];
W_2 = [0;0;0];
W_3 = [0;0;0];

Increment_Refference_Time = 0;
Refference_Signal_Time_Steps =20;
Refference_Signal_Time_Steps_Inverse = 20* End_Time;
Input                     = zeros(4, Time_steps);
Input_rate               = zeros(4, Time_steps);
Refference_signal                       = zeros(6, Refference_Signal_Time_Steps );
E_Frame_angular_velocity   = zeros(3,1);
B_Frame_angular_velocity   = zeros(3,1);

x_1_d_triple_dot_old =0;
 alpha_PSI_dot_old  = 0;
Loop_Count              = 1;
loop_count_2 = 10;
Controller_count = controller_time_steps;
x_1_d_dot_old              = [0;0;0];
x_1_d_double_dot_old    = [0;0;0];

n_1_hat_old = 0;
v_1_hat_old = 0;
ready_to_start =1;
if ready_to_start == 1

    E_Frame_Linear_possition = [0; 0;0];
    Possition = [E_Frame_Linear_possition;0;0;0];
end
simulation_start_time = clock;
v_1_hat_dot  = [0; 0; 0];
v_1_hat = [0; 0; 0];
n_1_tilda  = [0;0;0];
n_1_hat = E_Frame_Linear_possition ;
%dt = 0.02;
dt = Delta_Time;
%dt = 1/1000;
v_1_hat_TEMP  = [0;0;0];
alpha_w_old = [0;0;0];
L2_to_L3 = 1/100;
L2_to_L3 = 1/500;
L2_to_L3 = 1/500;
L2_to_L3 = 1/200;
gamma_2 = 1/L2_to_L3 ;
gamma_2 =200
L2_to_L3 = 1/gamma_2;
gamma_3 = 0.1;
last_coverience_dW                        = zeros(3,1);
coverience                                =0*0.1* [ 1 0 0;
                                               0 1 0;
                                               0 0 1];
coverience_angulare_system            =0*1.25*0.1^0.5* [ 1 0 0;
                                               0 1 0;
                                               0 0 1];
coverience_angulare_system            =2* [ 1 0 0;
                                               0 1 0;
```

```
                                             0 0 1];
coverience_3                              =0.1*eye(3);

tor_aero_hat_deterministic         = [0 ; 0; 0];
tor_aero_hat_deterministic_dot   = [0 ; 0; 0];
sigma_hat_Stochastic_1            = 0;
sigma_hat_Stochastic_dot_1      = 0;
sigma_hat_Stochastic_2            = 0;
sigma_hat_Stochastic_dot_2      = 0;
Inverse_I_H = inv(I_H);
Inverse_I_H_plus_I_A = inv(I_H + I_A);
   f_aero_Deterministic = 0*[0.5 ; 0.3;0.1];
   r=0.5
      Ca = 0.05;
MOTOR_FORCE_CALC = [ 1/4,        0, -1/(2*r), -1/(4*Ca);
                                  1/4, -1/(2*r),        0,  1/(4*Ca);
                                  1/4,        0,  1/(2*r), -1/(4*Ca);
                                  1/4,  1/(2*r),        0,  1/(4*Ca)];
Possition=0*[-15;1;0; -0*pi/4;-0*pi/4;0];
Possition_initial = Possition;
E_Frame_Linear_possition = [ Possition(1,1);
         Possition(2,1);
         Possition(3,1) ];

ROE =0.15*[50 0 0; 0 50 0; 0 0 50];%K_d;
ROE =5*0.115* ROE;
mew_1 = 10;
mew_2 = 57000000;
tor_aero_deterministic_MAX=[1;1;0.1];
tor_aero_Stochastic_MAX_1=0.1^4;
tor_aero_Stochastic_MAX_2= 1;
xi_aero_deterministic=0.005*[10;10;1];
xi_aero_Stochastic_1=0.1*tor_aero_Stochastic_MAX_1;
xi_aero_Stochastic_2=0.8*tor_aero_Stochastic_MAX_2;
gamma_3 = 0.1;
gamma_2 = 200;
sigma_hat_Stochastic_1 = 0;
sigma_hat_Stochastic_2 = 0;
sigma_hat_Stochastic_dot_1 =0;
sigma_hat_Stochastic_dot_2 = 0;
Possition = [0;0;0; 0;-pi/8;-pi/3]

      gamma_5 = 1;
for  t = 0: Delta_Time: End_Time
   %         [Desired_Possition,      Plot_info]       =       Get_Refference_Signal(Desired_Possition_2,
Desired_Possition_Time, t, Plot_info );
   Desired_Possition(3,1) = 0.1*t;
   n_1 =  [ Possition(1,1);
             Possition(2,1);
             Possition(3,1) ];
   if Desired_Possition(3,1) >= 0
      Desired_Possition(3,1) = 0.1*t;
      Desired_Possition(2,1)  = 10*sin((0.01*t));
      Desired_Possition(1,1)  = 10*(cos((0.01*t)))-20;
      Desired_Possition(6,1) = 0.01*t +0*pi/4;
      Desired_Possition(6,1) = 0;
      if Possition (3,1) > 0
         Wind_Velocity             = [0 ;0;0];
      end
      if Possition (3,1) <= 0
```

```
        Wind_Velocity            = [0 ;0;0];
      end
    end

    angles = [  Possition(4,1);
                Possition(5,1);
                Possition(6,1) ];
    phi     = Possition(4,1);
    theta   = Possition(5,1);
    R       = rotation(angles);
    q = Euler_to_Modified_rodrigues_paramater(angles);
    R_1_MPR = R_MRP(q);
    E_Frame_angulare_velocity                                          =
B_FRAME_ANGULARE_Velocity_TO_E_FRAME_Angulare_velocity(B_Frame_angulare_velocity,
angles);
    Velocity            = (Possition-Possition_OLD) / Delta_Time;
    Position_ERROR      = (Possition - Desired_Possition );
    Velocity_ERRPR      = (Desired_Possition-Desired_Possition_OLD)/Delta_Time-Velocity;
    Desired_Velocity     = [ -0.1*(sin((0.01*t)));
                             0.1*cos((0.01*t));
                             0.1; 0;0;0];
    x_1_d_dot            = [ Desired_Velocity(1,1);
                             Desired_Velocity(2,1);
                             Desired_Velocity(3,1) ];
    x_3_d_dot            = [ Desired_Velocity(4,1);
                             Desired_Velocity(5,1);
                             Desired_Velocity(6,1) ];
   x_1d                  = [ Desired_Possition(1,1);
                             Desired_Possition(2,1);
                             Desired_Possition(3,1) ];
%    x_1_d_double_dot    = (x_1_d_dot -x_1_d_dot_old) / Delta_Time;
    x_1_d_double_dot    = [ -0.001*(cos((0.01*t)));
                            -0.001*sin((0.01*t));
                             0];
     n_1d                = [ 10*(cos((0.1*t)))-20;
                             10*sin((0.1*t));
                             0.1*t-0.1];
    n_1d_dot           =  [ -1*sin(0.1*t);
                             1*cos((0.1*t));
                             0.1];
    n_1d_double_dot =  [ -0.1*(cos((0.1*t)));
                             -0.1*sin((0.1*t));
                             0];
    n_1d_triple_dot    = [ 0.01*(sin((0.1*t)));
                            -0.01*cos((0.1*t));
                            0];
    n_1d_quad_dot    = [ 0.001*(cos((0.1*t)));
                            0.001*sin((0.1*t));
                            0];
                %% Second stage set up converssions between frames of refference
    alpha_PSI              = 0*Desired_Possition(6,1);
    alpha_PSI_dot          = 0*Desired_Velocity(6,1);
    alpha_PSI_double_dot        = 0*( alpha_PSI_dot  - alpha_PSI_dot_old) / Delta_Time;
    alpha_psi = alpha_PSI;
    n_1e = n_1 - n_1d;
    epsilon_3=1;
    epsilon_4 =10;
    epsilon_5 =10;

gamma_1 =5.12;
```

```
    min_eigen_value_D_1=1/8;
    D_1= [ 0.25 0 0;
            0 0.25 0;
            0 0 0.125];
    D_2= 10^(-3)*[ 2.5 0 0;
            0 2.5 0;
            0 0 0.03];
%    c=0.054/gamma_1/2;
    c_2    =    ((norm(coverience/m))^2/2+0.025+0.025+f_aero_Deterministic    .'*f_aero_Deterministic
*min_eigen_value_D_1/3.9 + 3*min_eigen_value_D_1/6);
    c_1=c_2/gamma_1/2+gamma_1*min_eigen_value_D_1*0.5;
    c=c_1;
if Controller_count == controller_time_steps
    %% LINEAR POSSITION SYSTEM
counting_Loop  = counting_Loop  +1;
    %if gps_read_time >= gps_update_time
        delta_1_v_1_hat = 1 + 0.5 * (v_1_hat.')*v_1_hat;
        sigma_n_1e= [ n_1e(1,1)/(1+n_1e(1,1)^2)^0.5;
                      n_1e(2,1)/(1+n_1e(2,1)^2)^0.5;
                      n_1e(3,1)/(1+n_1e(3,1)^2)^0.5];
        sigma_dash_n_1e = [ 1/(1+n_1e(1,1)^2)^1.5, 0, 0;
                            0, 1/(1+n_1e(2,1)^2)^1.5, 0;
                            0, 0, 1/(1+n_1e(3,1)^2)^1.5];
        alpha_1              =  -K_1* sigma_n_1e/delta_1_v_1_hat + n_1d_dot;
        v_1e =  v_1_hat - alpha_1  ;
        [Omega,    Partial_Omega_wrt_partial_n_1e,    Partial_Omega_wrt_partial_v_1_hat    ,
Partial_Omega_wrt_partial_n_1d_dot        ,        Partial_Omega_wrt_partial_n_1d_double_dot,
Partial_M_1_to_6_wrt_partial_n_1e,                        Partial_M_1_to_6_wrt_partial_v_1_hat,
Partial_M_1_to_6_wrt_partial_n_1d_dot,            Partial_M_1_to_6_wrt_partial_n_1d_double_dot,
Sum_M1_to_M6]          =          Omega_Partial_stochastic_one_step_ahead_backstepping(m,
epsilon_2,gamma_1,  epsilon_3,min_eigen_value_D_1,K_1,  K_2,  D_1,c  ,  n_1e,  v_1_hat,  v_1e,
n_1d_dot, n_1d_double_dot, alpha_1);
        OMEGA = Omega;
        alpha_PHI                    =  asin((sin(alpha_PSI)*OMEGA   (1,1)  -  cos(alpha_PSI)*OMEGA
(2,1))/(OMEGA' * OMEGA)^0.5);
        alpha_THETA                  =  atan((cos(alpha_PSI)*OMEGA   (1,1)  +  sin(alpha_PSI)*OMEGA
(2,1))/OMEGA  (3,1));
        alpha_2              = [alpha_PHI ; alpha_THETA ; alpha_PSI ];
        h_2 =min_eigen_value_D_1*0.5* gamma_1*(sigma_n_1e.').';
        h_3 =2* gamma_1*(v_1e.'*K_1*sigma_dash_n_1e/delta_1_v_1_hat).'/(1+v_1e.'*v_1e)^0.5;
        alpha_psi_dot  =  alpha_PSI_dot       ;
        alpha_psi_double_dot  =   alpha_PSI_double_dot ;
        G_1 = eye(3)-(K_1*sigma_n_1e*(v_1_hat.')/delta_1_v_1_hat^2);
        cos_alpha_psi = cos(alpha_psi);
        alpha_phi = alpha_PHI;
        alpha_theta = alpha_THETA;
        Omega_3 = Omega(3,1);
        sin_alpha_psi = sin(alpha_psi);
        sigma_v_1e = [ v_1e(1,1)/(1+v_1e(1,1)^2)^0.5;
                       v_1e(2,1)/(1+v_1e(2,1)^2)^0.5;
                       v_1e(3,1)/(1+v_1e(3,1)^2)^0.5];
        clac_v_1_hat = 1;

    %end

%% ANGULARE POSSITION SYSTEM

    n_2e = angles -alpha_2;
```

```
n_1 =  [ Possition(1,1);
         Possition(2,1);
         Possition(3,1) ];
      epsilon_1  = 1;
       epsilon_6  = 1;

       Delta_time = Delta_Time;


      [ tor, alpha_w, alpha_w_dot, tor_aero_hat_deterministic, tor_aero_hat_deterministic_dot, sigma_hat_Stochastic_1, sigma_hat_Stochastic_dot_1,sigma_hat_Stochastic_2, sigma_hat_Stochastic_dot_2, h_4, h_5, q_e, w_e, Partial_alpha_w_wrt_partial_v_1_hat] = Attitude_Backstepping_Controller_Complete_System_Stochastic(m, I_H, I_A, g, L2_to_L3, epsilon_1,epsilon_2, epsilon_3, epsilon_4, epsilon_5, epsilon_6,epsilon_7, D_1, D_2,min_eigen_value_D_1, gamma_1, K_1, K_2, K_3, K_4, c, n_1e, v_1_hat, v_1e, n_1d, n_1d_dot, n_1d_double_dot, n_1d_triple_dot, n_1d_quad_dot, angles, alpha_1, alpha_2, alpha_psi, alpha_psi_dot, alpha_psi_double_dot, R, B_Frame_angulare_velocity, Delta_time, tor_aero_hat_deterministic, sigma_hat_Stochastic_1, sigma_hat_Stochastic_dot_1,sigma_hat_Stochastic_2, sigma_hat_Stochastic_dot_2, Loop_Count, Sum_M1_to_M6, Partial_M_1_to_6_wrt_partial_n_1e, Partial_M_1_to_6_wrt_partial_v_1_hat, Partial_M_1_to_6_wrt_partial_n_1d_dot, Partial_M_1_to_6_wrt_partial_n_1d_double_dot, Omega, Partial_Omega_wrt_partial_n_1e, Partial_Omega_wrt_partial_v_1_hat , Partial_Omega_wrt_partial_n_1d_dot , Partial_Omega_wrt_partial_n_1d_double_dot, alpha_w_old)
      alpha_w_old = alpha_w;
       Controller_count=0;
  tau_aero_hat  = tor_aero_hat_deterministic;


  Control_Input = [(Omega.'*Omega)^0.5;tor];



  if clac_v_1_hat ==1
     alpha_q = Euler_to_Modified_rodrigues_paramater(alpha_2)
     R_2_alpha_q = R_2_MRP_calc( alpha_q)
     T_alpha_2_inverse              = Angular_velocity_cordinant_transform(alpha_2);

     A_1d=[              cos_alpha_psi*1/(cos(alpha_phi)*(Omega.'*Omega)^0.5)              , sin_alpha_psi*1/(cos(alpha_phi)*(Omega.'*Omega)^0.5) , 0;
        -sin_alpha_psi*(cos(alpha_theta))^2/Omega_3                , cos_alpha_psi*(cos(alpha_theta))^2/Omega_3           , 0;
        0                                                                , 0 , 0];
     A_2d=[              sin_alpha_psi/(cos(alpha_phi)*(Omega.'*Omega)^0.5),              (-cos_alpha_psi)/(cos(alpha_phi)*(Omega.'*Omega)^0.5),              0 ;
        (cos(alpha_theta))^2*cos_alpha_psi/Omega_3              , (cos(alpha_theta))^2*sin_alpha_psi/Omega_3          , -(cos(alpha_theta))^2*tan(alpha_theta)/Omega_3;
        0                                                                , 0 , 0                    ] - [tan(alpha_phi)/(Omega.'*Omega); 0; 0]*Omega.';


q = Euler_to_Modified_rodrigues_paramater(angles);
R_1_MPR = R_MRP(q);


R = R_1_MPR;
```

```
        h_1 = K_02* (2*gamma_1*v_1e.'*G_1/(1+v_1e.'*v_1e)^0.5 -(1/L2_to_L3)* q_e.'*R_2_alpha_q *
T_alpha_2_inverse*A_2d*Partial_Omega_wrt_partial_v_1_hat                                    -
0.1*(w_e.'*w_e)*w_e.'*Partial_alpha_w_wrt_partial_v_1_hat/(gamma_5+(w_e.'*w_e)^2)^0.5).';
        v_1_hat_dot = -D_1*v_1_hat - g *[0;0;1] + (Omega.'*Omega)^0.5/m*R*[0;0;1] + K_02 * n_1_tilda
+ h_2 + h_3 + h_4 + h_5;
        v_1_hat = v_1_hat_dot*dt + v_1_hat   ;
        %[v_1_hat ,E_Frame_Linear_velocity];
        v_1_hat_tilda =  E_Frame_Linear_velocity -v_1_hat;
        v_1_tilda =  E_Frame_Linear_velocity -v_1_hat;

        n_1_hat_dot = K_01*(n_1 - n_1_hat) + h_1/gamma_4  +v_1_hat;
        n_1_hat = n_1_hat  +  n_1_hat_dot*dt;
        n_1_tilda = (n_1 - n_1_hat) ;
        clac_v_1_hat =0;
    end

end

    control_time = t ;

    loop_count_2 = -1;



    q= Euler_to_Modified_rodrigues_paramater(angles );
alpha_2 * 180/pi;
alpha_q = Euler_to_Modified_rodrigues_paramater(alpha_2);
q_e = q - alpha_q;


[Partial_L_wrt_q_times_vector             ,             Partial_L_wrt_alpha_q_times_vector,L_m]           =
calc_L_MRP_norm_squared (q, alpha_q,G_1.'*v_1e/(1+ v_1e.'*v_1e)^0.5);
temp_X                  =                   -2*gamma_1*L2_to_L3*(1/m)*(Omega.'*Omega)^0.5               *
L_m*G_1.'*v_1e/(1+v_1e.'*v_1e)^0.5-K_3*q_e;
%% motor spped to propeller thrust force and rotation toruqe
%v_1_hat =E_Frame_Linear_velocity;

%%disturbance dynamics
            Air_Velocity  = E_Frame_Linear_velocity -Wind_Velocity ;
            Aero_Disturbance         =  Aero_Disturbance_Calc(Air_Velocity,  Air_Flow_Type,  angles,  t,
Wind_Start_Time, Wind_frequancy, Hex_copter_height,copterradius);

            Collision_Disturbance                  =Collision_Disturbance_calc(t,                   Collision_time,
Collision_Point,F_Collision_x,       F_Collision_y,       F_Collision_z,       E_Frame_Linear_possition)
;
            Torque_Disturbance = 0*[0.1; 0.2; 0.1];
            %
            Collision_Disturbance = [ 0*Collision_Disturbance(1,1);
                                      0* Collision_Disturbance(2,1);
                                      0* Collision_Disturbance(3,1);
                                       Torque_Disturbance];

 Collision_Disturbance = [ f_aero_Deterministic ;0 ;0;0];
            %}
                               tau_error = Torque_Disturbance - tau_aero_hat  ;
                 %dW_last =  dW;
        dW                                   = sqrt(Delta_Time)*randn * [1;1;1];
        dW_2                                 = sqrt(Delta_Time)*randn * [1;1;1];
        dW_3                                 = sqrt(Delta_Time)*randn * [1;1;1];
```

```
        if Loop_Count == 1
            dW_initial    =  dW  ;
             %dW   =  dW   - dW_initial ;
        end
     I_XX_A = I_A(1,1);
     I_YY_A = I_A(2,2);
     I_ZZ_A = I_A(3,3);

        Coriolis      =      [      0,      I_ZZ_A      *      B_Frame_angulare_velocity(3,1),      -I_YY_A      *
B_Frame_angulare_velocity(2,1);
                 -I_ZZ_A        *        B_Frame_angulare_velocity(3,1),        0,        I_XX_A        *
B_Frame_angulare_velocity(1,1);
                 I_YY_A         *         B_Frame_angulare_velocity(2,1),          -I_XX_A          *
B_Frame_angulare_velocity(1,1), 0 ];
        skew_w = [ 0, -B_Frame_angulare_velocity(3,1),  B_Frame_angulare_velocity(2,1);
                   B_Frame_angulare_velocity(3,1), 0, -B_Frame_angulare_velocity(1,1);
                   -B_Frame_angulare_velocity(2,1), B_Frame_angulare_velocity(1,1), 0];
        last_coverience_dW                    = coverience*dW;%+last_coverience_dW;
        last_coverience_dW_2                                                              =
coverience_angulare_system*(D_2*eye(3)+0*0.05*eye(3)+Coriolis          -          skew_w          *
I_A)*dW_2;%+last_coverience_dW;
        last_coverience_dW_3                  = coverience_3*dW_3;
%%     // 6 dof dynamics
W_1 =W_1+ last_coverience_dW   ;
W_2 =W_2+ last_coverience_dW_2;
W_3 =W_3+ last_coverience_dW_3;

stochastic_disturbance_results(:,Loop_Count) =[W_1;W_2; W_3];
            Zeta_dot     =     SIX_DOF_Dynamics(R,     E_Frame_angulare_velocity,     angles,g,m,(I_H+
I_A),(I_XX+I_XX_A),(I_YY+I_YY_A),(I_ZZ+I_ZZ_A),      Aero_Disturbance,      Collision_Disturbance,
Control_Input, JTP,Zeta,Omega , D_1,  E_Frame_Linear_velocity,D_2 );

           % Advance system state.
        B_Frame_angulare_acceleration   = [Zeta_dot(4,1); Zeta_dot(5,1); Zeta_dot(6,1)];
        E_Frame_Linear_acceleration     = [Zeta_dot(1,1); Zeta_dot(2,1); Zeta_dot(3,1)];
 %             B_Frame_angulare_velocity   =   B_Frame_angulare_velocity   +   Delta_Time   *
B_Frame_angulare_acceleration;
  %                                                        E_Frame_angulare_velocity            =
B_FRAME_ANGULARE_Velocity_TO_E_FRAME_Angulare_velocity(B_Frame_angulare_velocity,
angles);
      %  angles = angles + Delta_Time * E_Frame_angulare_velocity;
        B_Frame_angulare_velocity     =     B_Frame_angulare_velocity      +          Delta_Time      *
B_Frame_angulare_acceleration        +        Inverse_I_H_plus_I_A*(last_coverience_dW_2        +
last_coverience_dW_3);
        E_Frame_angulare_velocity                                                         =
B_FRAME_ANGULARE_Velocity_TO_E_FRAME_Angulare_velocity(B_Frame_angulare_velocity,
angles);

        angles = angles +  Delta_Time * E_Frame_angulare_velocity;
 %                E_Frame_Linear_velocity    =   E_Frame_Linear_velocity    +    Delta_Time    *
E_Frame_Linear_acceleration;
 %               E_Frame_Linear_possition    =   E_Frame_Linear_possition    +    Delta_Time    *
E_Frame_Linear_velocity;
               E_Frame_Linear_velocity      =      E_Frame_Linear_velocity      +      Delta_Time      *
E_Frame_Linear_acceleration;
               E_Frame_Linear_velocity = E_Frame_Linear_velocity + (1/m)* last_coverience_dW  ;
        E_Frame_Linear_possition      =      E_Frame_Linear_possition      +      Delta_Time      *
E_Frame_Linear_velocity;
```

```
        if (E_Frame_Linear_possition(3,1) <= 0)
            E_Frame_Linear_possition(3,1) =0;
        end


        Zeta  = [ E_Frame_Linear_acceleration;
                  B_Frame_angulare_velocity           ];
              psi =  angles(3);

angles = Check_Angular_Position( angles);
  angles(3) = psi;
Possition_OLD   = Possition;
        Possition = [ E_Frame_Linear_possition(1);
                      E_Frame_Linear_possition(2);
                      E_Frame_Linear_possition(3);
                      angles(1);
                      angles(2);
                      angles(3) ];
        Desired_Possition_OLD =     Desired_Possition;
%% function to plot the refference signal
    Input(:,Loop_Count) = Control_Input;
    if Loop_Count > 1
        Input_rate(:,Loop_Count)  = (Input(:,Loop_Count) - Input(:,(Loop_Count-1))) / Delta_Time;
    end
    if Loop_Count == 1
        Input_rate(:,Loop_Count)  = 0;
    end
     x_1_d_dot_plot(:,Loop_Count) =  x_1_d_dot;

     %{
     n_2_error = [ Possition(4,1) - alpha_2(1,1);
                   Possition(5,1) - alpha_2(1,1);
                   Possition(6,1) - alpha_2(1,1)];
     %}
     n_2_error = n_2e;
                   w_error = B_Frame_angulare_velocity - alpha_w;


        c_2    =    ((norm(coverience/m))^2/2+0.025+0.025+f_aero_Deterministic    .'*f_aero_Deterministic
*min_eigen_value_D_1/3.9 + sigma_n_1e.'*sigma_n_1e*min_eigen_value_D_1/6);
      infinite_generator_of_V_4                                     =                              -
gamma_1*0.5*min_eigen_value_D_1*(1/delta_1_v_1_hat)*(sigma_n_1e.'*K_1*sigma_n_1e)              -
2*gamma_1*sigma_v_1e.'*K_2*v_1e/(1+v_1e.'*v_1e)^0.5
+2*gamma_1*v_1e.'*K_1*sigma_dash_n_1e/delta_1_v_1_hat*v_1e/(1+v_1e.'*v_1e)^0.5                -
2*gamma_1*c_1*v_1e.'*(v_1e+2*alpha_1)/(2*delta_1_v_1_hat)         -2*gamma_1*v_1e.'*(D_1       -
(1/8)*min_eigen_value_D_1*eye(3)*(7.6-sigma_n_1e.'*sigma_n_1e/7.6))*v_1e/(1+v_1e.'*v_1e)^0.5  -
n_1_tilda.'*K_01*n_1_tilda        -v_1_tilda.'*(D_1-eye(3)*min_eigen_value_D_1*3.9/4)*v_1_tilda       -
gamma_2*q_e.'*(K_3                -(5/5.1)*min(eig(K_3))               *               eye(3))*q_e             -
gamma_3*(norm(w_e.'*w_e))^2*w_e.'*(K_4+D_2)*w_e/(1+(norm(w_e))^4)^0.75                         -
(norm(w_e.'*w_e))^2*w_e.'*2*gamma_1*c_1*w_e/(1+(norm(w_e))^4)  + c_2/(1+(norm(w_e))^4)  ;
      V_4  =  (gamma_1*0.5*min_eigen_value_D_1  *  ((1+n_1e(1,1)^2)^0.5  +  (1+n_1e(2,1)^2)^0.5  +
(1+n_1e(3,1)^2)^0.5    -    3)    +    2*gamma_1*(1+v_1e.'*v_1e)^0.5       +    gamma_2*0.5*q_e.'*q_e
+gamma_3*(1+(norm(w_e))^4)^0.25+0.5        *        (n_1_tilda.'*n_1_tilda+v_1_tilda.'*v_1_tilda)       +
0.5*(norm(coverience_angulare_system)^4         -sigma_hat_Stochastic_1         )^2/1000000         +
0.5*(norm(coverience_3)^4 -sigma_hat_Stochastic_2 )^2/-2*gamma_1-gamma_3 );
%                                                    infinite_generator_of_V_4           =           -
gamma_1*0.5*min_eigen_value_D_1*(1/delta_1_v_1_hat)*(sigma_n_1e.'*K_1*sigma_n_1e)              -
2*gamma_1*sigma_v_1e.'*K_2*v_1e/(1+v_1e.'*v_1e)^0.5
+2*gamma_1*v_1e.'*K_1*sigma_dash_n_1e/delta_1_v_1_hat*v_1e/(1+v_1e.'*v_1e)^0.5                -
2*gamma_1*c_1*v_1e.'*(v_1e+2*alpha_1)/(2*delta_1_v_1_hat)         -2*gamma_1*v_1e.'*(D_1       -
```

```
(1/8)*min_eigen_value_D_1*eye(3)*(7.6-sigma_n_1e.'*sigma_n_1e/7.6))*v_1e/(1+v_1e.'*v_1e)^0.5 - n_1_tilda.'*K_01*n_1_tilda -v_1_tilda.'*(D_1-eye(3)*min_eigen_value_D_1*3.9/4)*v_1_tilda - gamma_2*q_e.'*(K_3 -(5/5.1)*min(eig(K_3)) * eye(3))*q_e - gamma_3*(norm(w_e.'*w_e))^2*w_e.'*(K_4+D_2)*w_e/(1+(norm(w_e))^4)^0.75 - (norm(w_e.'*w_e))^2*w_e.'*2*gamma_1*c_1*w_e/(1+(norm(w_e))^4) + c_2/(1+(norm(w_e))^4-0.5*(norm(coverience_angulare_system)^4 -sigma_hat_Stochastic )^2/1000000++ 0.5*(norm(coverience_angulare_system)^4 *2 )^2/1000000 ) ;
V_4_on_L_V_4 = (-gamma_1*0.5*min_eigen_value_D_1*(1/delta_1_v_1_hat)*(sigma_n_1e.'*K_1*sigma_n_1e) - 2*gamma_1*sigma_v_1e.'*K_2*v_1e/(1+v_1e.'*v_1e)^0.5 +2*gamma_1*v_1e.'*K_1*sigma_dash_n_1e/delta_1_v_1_hat*v_1e/(1+v_1e.'*v_1e)^0.5 - 2*gamma_1*c_1*v_1e.'*(v_1e+2*alpha_1)/(2*delta_1_v_1_hat) -2*gamma_1*v_1e.'*(D_1 - (1/8)*min_eigen_value_D_1*eye(3)*(7.6-sigma_n_1e.'*sigma_n_1e/7.6))*v_1e/(1+v_1e.'*v_1e)^0.5 - n_1_tilda.'*K_01*n_1_tilda -v_1_tilda.'*(D_1-eye(3)*min_eigen_value_D_1*3.9/4)*v_1_tilda - gamma_2*q_e.'*(K_3 -(5/5.1)*min(eig(K_3)) * eye(3))*q_e - gamma_3*(norm(w_e.'*w_e))^2*w_e.'*(K_4+D_2)*w_e/(1+(norm(w_e))^4)^0.75 - (norm(w_e.'*w_e))^2*w_e.'*2*gamma_1*c_1*w_e/(1+(norm(w_e))^4) + c_2/(1+(norm(w_e))^4-0.5*(norm(coverience_angulare_system)^4 -sigma_hat_Stochastic_1 )^2/1000000-0.5*(norm(coverience_3)^4 -sigma_hat_Stochastic_2 )^2+ 0.5*(norm(coverience_angulare_system)^4 *2 )^2/1000000 ) ) / V_4;



if Loop_Count ==1
   %V_sum_initial = ( ((1+n_1e(1,1)^2)^0.5 + (1+n_1e(2,1)^2)^0.5 + (1+n_1e(3,1)^2)^0.5 - 3) + (1+v_1e.'*v_1e)^0.5 + q_e.'*q_e +(1+(norm(w_e))^4)^0.25+ (n_1_tilda.'*n_1_tilda+v_1_tilda.'*v_1_tilda) + (norm(coverience_angulare_system)^4 - sigma_hat_Stochastic )^2 -3)
   V_sum_initial = V_4;
   b_7 = (1/3)*((1+n_1e(1,1)^2)^-0.5 + (1+n_1e(2,1)^2)^-0.5 + (1+n_1e(3,1)^2)^-0.5)/delta_1_v_1_hat * c_2/(1+(norm(w_e))^4) *(2+alpha_1.'*alpha_1)/2;
%      b_6 = min([(gamma_1*0.5*min_eigen_value_D_1*(1/delta_1_v_1_hat)*min(eig(K_1)) - c_1*(1/delta_1_v_1_hat)/(1+(norm(w_e))^4)) ,min(eig(D_1-eye(3)*min_eigen_value_D_1*3.9/4))])/max([gamma_1*0.5*min_eigen_value_D_1, 2 * gamma_1, gamma_2/2, 0.5, gamma_3]);
   b_6 = min([
       (gamma_1*0.5*min_eigen_value_D_1*(min(eig(K_1)) - c_1)*(1/delta_1_v_1_hat)*((1+n_1e(1,1)^2)^-0.5 + (1+n_1e(2,1)^2)^-0.5 + (1+n_1e(3,1)^2)^-0.5))/(gamma_1*0.5);
       (2*gamma_1*min(eig(D_1-eye(3)*min_eigen_value_D_1/8*(epsilon_1+3/epsilon_1)))/(2*gamma_1));
       (2*gamma_1*(min(eig(K_2))-min(eig(K_1)))/2*gamma_1);
       (min(eig(K_01))/0.5);
       (min(eig(D_1-eye(3)*min_eigen_value_D_1*3.9/4))/0.5);
       (gamma_2*min(eig(K_3))/(0.5*gamma_2));
       (gamma_3*min(eig(K_4))/(gamma_3))
       ((1+n_1e(1,1)^2)^-0.5 + (1+n_1e(2,1)^2)^-0.5 + (1+n_1e(3,1)^2)^-0.5)/delta_1_v_1_hat/0.5])
   b_7/b_6
end

V_sum =    V_sum_initial *exp(-b_6*t) + b_7/b_6;
   results(:,Loop_Count) = [ Possition;%1-6
                             t;%time ;%7
                             B_Frame_angulare_velocity ;%8-10
                             tau_aero_hat;%111 - 13
                             w_error;%14-16
                             n_2_error;%17-19
                             alpha_w_dot ;%20-22
                            Torque_Disturbance;%23-25
                             tau_error;%26-28
```

```
                                        alpha_2;%eta_2_d;%   ];%eta_2_noise];%29 - 31
                                        0;%exponentialMA ;%32
                                        Delta_time;%33
                                        v_1_hat;%q%34 - 36
                                        alpha_q;%37 - 39
                                        0;%count_3; %40
                                        0;%reread; %41
                                        0;%Time_to_read_seriel;%42
                                        sigma_hat_Stochastic_1;%43
                                        OMEGA;%44 - 46
                                        v_1_hat_tilda;% 47 - 49
                                        h_4; % 50 - 52
                                        h_5; % 53 - 55
                                        alpha_w;
                                        n_1_tilda;
                                        E_Frame_Linear_velocity ;
                                        v_1_hat;
                                        v_1e;
                                        temp_X;
                                         infinite_generator_of_V_4 ;
                                         V_4;
                                         V_sum;
                                         last_coverience_dW_2/dt;
                                         MOTOR_FORCE_CALC*Control_Input;
                                         sigma_hat_Stochastic_2;
                                         last_coverience_dW/dt;
                                         last_coverience_dW_3/dt;
                                         norm(w_e)
                                         h_1;
                                         h_2;
                                         h_3;];


   x_1_d_dot_old            =x_1_d_dot;
   x_1_d_double_dot_old  = x_1_d_double_dot;
   x_1_d_triple_dot_old    =x_1_d_triple_dot ;
   alpha_PSI_dot_old       = alpha_PSI_dot ;

   %{
for i = 1:3
   Refference_signal(i,Loop_Count) = Desired_Possition(i,1) ;
end
%}
Refference_signal(:,Loop_Count) = [ n_1d ;
                                            alpha_2];
%alpha_2 = [0;0;0];
Refference_signal(4,Loop_Count) = alpha_2(1,1);
Refference_signal(5,Loop_Count) = alpha_2(2,1);
Loop_Count             = Loop_Count + 1;
%% END OF SIMULATION LOOP%%
t;
%      [ v_1_hat_tilda.' t]
      [ n_1e.' v_1_hat_tilda.'  (1+v_1e.'*v_1e)^0.5/delta_1_v_1_hat t]

loop_count_2 = loop_count_2 + 1;
Controller_count = Controller_count+1;
end
simulation_end_time = clock;
```

```
simulation_elapsed_time = 60*60*(simulation_end_time(4) - simulation_start_time(4)) + 60*(simulation_end_time(5)-simulation_start_time(5)) + (simulation_end_time(6)-simulation_start_time(6));


figure
plot(results(7,:) ,results(11,:) - results(23,:), ...
        'LineWidth',2)
title('Torque Distrubance estimate error  deterministic Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,results(12,:) - results(24,:),'g', ...
        'LineWidth',2)
hold on
plot(results(7,:) ,results(13,:)  - results(25,:),'r', ...
        'LineWidth',2)
hold on
axis([0, End_Time, - 2, 2])
%axis([0, End_Time, - -45, 45])
legend({'\it{\tau_p} \rm{(Nm)}','\it{\tau_q} \rm{(Nm)}','\it{\tau_r} \rm{(Nm)}'}, 'FontSize',16)
xlabel('Time(s)', 'FontSize',14)
ylabel('Estimated torque disturbance (Nm)', 'FontSize',14)
%axis([0, End_Time,-0.4,0.4])
hold off

RAD_to_DEG = 180/pi;

figure
plot(results(7,:) ,RAD_to_DEG*results(17,:), ...
        'LineWidth',2)
title('Angulare possition Euler error Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,RAD_to_DEG*results(18,:),'g', ...
        'LineWidth',2)
hold on
plot(results(7,:) ,RAD_to_DEG*results(19,:),'r', ...
        'LineWidth',2)
  axis([0, End_Time, - 2, 18])
xlabel('Time(s)', 'FontSize',14)
ylabel('Attitude (degrees)', 'FontSize',14)
%set(legend({'\boldmath${\eta}_{1e,1}$','\boldmath${\eta}_{1e,2}$'       '\boldmath${\eta}_{1e,3}$'}, 'FontSize',20,'Location','northeast'),'interpreter','latex')
set(legend({'${\phi}_e$','${\theta}_e$'                    '${\psi}_e$'}, 'FontSize',20,'Location','northeast'),'interpreter','latex')
hold off

figure
subplot(2,2,1);
plot(results(7,:) ,results(94,:),'b' , ...
        'LineWidth',2)
      title('h_3 Vs Time')
hold on
plot(results(7,:) ,results(95,:),'g' , ...
        'LineWidth',2)
hold on
plot(results(7,:) ,results(96,:),'r' , ...
        'LineWidth',2)
hold off

subplot(2,2,2);
plot(results(7,:) ,results(98,:),'b' , ...
```

```
            'LineWidth',2)
         title('h_3 Vs Time')
hold on
plot(results(7,:) ,results(99,:),'g' , ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(100,:),'r' , ...
            'LineWidth',2)
hold off
subplot(2,2,3);

plot(results(7,:) ,results(50,:),'b' , ...
            'LineWidth',2)
         title('h_4 Vs Time')
hold on
plot(results(7,:) ,results(51,:),'g' , ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(52,:),'r' , ...
            'LineWidth',2)
hold off

subplot(2,2,4);
plot(results(7,:) ,results(53,:),'b' , ...
            'LineWidth',2)
         title('h_5 Vs Time')
hold on
plot(results(7,:) ,results(54,:),'g' , ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(55,:),'r' , ...
            'LineWidth',2)
hold off

figure
plot(results(7,:) ,results(47,:),'b' , ...
            'LineWidth',2)
         title('v 1 tilda Vs Time')
hold on
plot(results(7,:) ,results(48,:),'g' , ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(49,:),'r' , ...
            'LineWidth',2)
%hold on
%grid on
%axis([0, 0.010, - 1.6, 70])
hold off

figure
plot(results(7,:) ,results(1,:)-Refference_signal(1,:), ...
            'LineWidth',2)
title('Possition error Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,results(2,:)-Refference_signal(2,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(3,:)-Refference_signal(3,:),'r', ...
            'LineWidth',2)
xlabel('Time(s)', 'FontSize',14)
```

```
ylabel('Distance (m)', 'FontSize',14)
%set(legend({'\boldmath${\eta}_{1e,1}$','\boldmath${\eta}_{1e,2}$'            '\boldmath${\eta}_{1e,3}$'},
'FontSize',20,'Location','northeast'),'interpreter','latex')
set(legend({'$x_e$','$y_e$' '$z_e$'}, 'FontSize',20,'Location','northeast'),'interpreter','latex')
hold off

RAD_to_DEG = 180/pi;

figure
plot(results(7,:) ,RAD_to_DEG*results(4,:), ...
            'LineWidth',2)
title('Attitude Euler RepresentationVs Time', 'FontSize',14)
hold on
plot(results(7,:) ,RAD_to_DEG*results(5,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,RAD_to_DEG*results(6,:),'r', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,RAD_to_DEG*results(29,:),'--b' , ...
            'LineWidth',1)
hold on
plot(results(7,:) ,RAD_to_DEG*results(30,:),'--g' , ...
            'LineWidth',1)
 hold on
plot(results(7,:) ,RAD_to_DEG*results(31,:),'--r' , ...
            'LineWidth',1)
axis([0, End_Time, - RAD_to_DEG*2, RAD_to_DEG*2])
%axis([0, End_Time, - -45, 45])
legend({'\it{\phi} \rm{(Pitch Sensor reading)}','\it{\theta} \rm{(Roll Sensor reading)}','\it{\psi} \rm{(Yaw
Sensor reading)}','\it{\phi_d} \rm{(Pitch reference signal)}','\it{\theta_d} \rm{(Roll reference
signal)}','\it{\psi_d} \rm{(Yaw reference signal)}'}, 'FontSize',16)
xlabel('Time(s)', 'FontSize',14)
ylabel('Radians', 'FontSize',14)
%axis([0, End_Time,-0.4,0.4])
hold off

figure
subplot(2,1,1);
plot(results(7,:) , Input(1,:),'k', ...
            'LineWidth',2)
title('Control Thrust U_1 Vs Time', 'FontSize',14)
xlabel('Time(s)', 'FontSize',14)
ylabel('Force(N)', 'FontSize',14)
%ylabel('Estimated torque disturbance (Nm)', 'FontSize',14)
axis([0, End_Time,15,30])
hold off
subplot(2,1,2);
plot(results(7,:) ,Input(2,:) ,'b', ...
            'LineWidth',2)
title('Control TorqueVs Time', 'FontSize',14)
hold on
plot(results(7,:) ,Input(3,:)  ,'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,Input(4,:)  ,'r', ...
            'LineWidth',2)
hold on
legend({'U_2','U_3','U_4'}, 'FontSize',12,'Location','northeast', 'Orientation', 'Horizontal')
xlabel('Time(s)', 'FontSize',14)
```

```
ylabel('Torque(Nm)', 'FontSize',14)
hold off

plot(asdfasd,aafd)
figure
plot(results(7,:) ,results(84,:), ...
            'LineWidth',2)
title('sigma 2Vs Time')
hold off

figure
plot(results(7,:) ,results(91,:), ...
            'LineWidth',2)
title('norm w_eVs Time')
hold off

figure
plot(results(7,:) ,stochastic_disturbance_results(1,:), ...
            'LineWidth',2)
title('stochastic  linear disturbance Vs Time')
hold on
plot(results(7,:) ,stochastic_disturbance_results(2,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,stochastic_disturbance_results(3,:),'r', ...
            'LineWidth',2)
hold off

figure
plot(results(7,:) ,stochastic_disturbance_results(4,:), ...
            'LineWidth',2)
title('stochastic  angulre disturbance Vs Time')
hold on
plot(results(7,:) ,stochastic_disturbance_results(5,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,stochastic_disturbance_results(6,:),'r', ...
            'LineWidth',2)
hold off

figure
plot(results(7,:) ,stochastic_disturbance_results(7,:), ...
            'LineWidth',2)
title('W_3 Vs Time')
hold on
plot(results(7,:) ,stochastic_disturbance_results(8,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,stochastic_disturbance_results(9,:),'r', ...
            'LineWidth',2)
hold off

figure
plot(results(7,:) ,Refference_signal(1,:), ...
            'LineWidth',2)
title('Refference signal  Vs Time')
hold on
plot(results(7,:) ,Refference_signal(2,:),'g', ...
            'LineWidth',2)
hold on
```

```
hold off

figure
plot3(results(1,:),results(2,:), results(3,:), ...
        'LineWidth',2)
title('Linear Position 3d Vs Time')
hold on
plot3(Refference_signal(1,:),Refference_signal(2,:),Refference_signal(3,:),'g', ...
        'LineWidth',4)
grid on
axis([-10,10,-10,10])
hold off

figure
get(0,'Factory')
set(0,'defaultfigurecolor',[1 1 1])
plot3(results(1,:),results(2,:), results(3,:), ...
        'LineWidth',2)
title('Linear Position 3d Vs Time')
hold on
plot3(Refference_signal(1,:),Refference_signal(2,:),Refference_signal(3,:),'g', ...
        'LineWidth',4)
grid on
axis([-30,10,-20 20])
title('Linear Position 3d Vs Time', 'FontSize',14)
set(legend({'\boldmath${\eta}_1$',                    '\boldmath${\eta}_{1d}$'},
'FontSize',20,'Location','northeast'),'interpreter','latex')
xlabel('X(n)', 'FontSize',14)
ylabel('Y(m)', 'FontSize',14)
zlabel('Z(m)', 'FontSize',14)
hold off

figure
plot(results(7,:) ,results(1,:)-Refference_signal(1,:), ...
        'LineWidth',2)
title('Possition error Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,results(2,:)-Refference_signal(2,:),'g', ...
        'LineWidth',2)
hold on
plot(results(7,:) ,results(3,:)-Refference_signal(3,:),'r', ...
        'LineWidth',2)
xlabel('Time(s)', 'FontSize',14)
ylabel('Distance (m)', 'FontSize',14)
%set(legend({'\boldmath${\eta}_{1e,1}$','\boldmath${\eta}_{1e,2}$'        '\boldmath${\eta}_{1e,3}$'},
'FontSize',20,'Location','northeast'),'interpreter','latex')
set(legend({'$x_e$','$y_e$' '$z_e$'}, 'FontSize',20,'Location','northeast'),'interpreter','latex')
hold off

figure
plot(results(7,:) ,results(1,:), ...
        'LineWidth',2)
title('possition Vs Time')
hold on
plot(results(7,:) ,results(2,:),'g', ...
        'LineWidth',2)
hold on
plot(results(7,:) ,results(3,:),'r', ...
        'LineWidth',2)
hold off
```

```
figure
plot(results(7,:) ,results(32,:), ...
        'LineWidth',2)
title('Torque disturbance estimate Vs Time')
hold on
plot(results(7,:) ,results(33,:),'g', ...
        'LineWidth',2)
hold on
plot(results(7,:) ,results(34,:),'r', ...
        'LineWidth',2)
axis([0, End_Time, -2, 2])
hold off

figure
plot(results(7,:) ,results(35,:), ...
        'LineWidth',2)
title('Sigman hat stochastic estimate Vs Time')
axis([0, End_Time, 0, 0.0002])
hold off

figure
plot(results(7,:) ,results(68,:), ...
        'LineWidth',2)
title('v_1e  Vs Time')
hold on
plot(results(7,:) ,results(69,:),'g', ...
        'LineWidth',2)
hold on
plot(results(7,:) ,results(70,:),'r', ...
        'LineWidth',2)
  axis([0, End_Time, - .1, .5])
hold off

figure
subplot(2,1,1);
plot(results(7,:) ,results(62,:), ...
        'LineWidth',2)
title('Actual linear velcoity Vs Time')
hold on
plot(results(7,:) ,results(63,:),'g', ...
        'LineWidth',2)
hold on
plot(results(7,:) ,results(64,:),'r', ...
        'LineWidth',2)
  axis([0, End_Time, - .1, .5])
hold off
subplot(2,1,2);
plot(results(7,:) ,results(65,:), ...
        'LineWidth',2)
title('Linear velocity estimate Vs Time')
hold on
plot(results(7,:) ,results(66,:),'g', ...
        'LineWidth',2)
hold on
plot(results(7,:) ,results(67,:),'r', ...
        'LineWidth',2)
  axis([0, End_Time, - .1, .5])
hold off
%%
```

```
RAD_to_DEG = 180/pi;

figure
plot(results(7,:) ,RAD_to_DEG*results(4,:), ...
            'LineWidth',2)
title('Angulare possition Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,RAD_to_DEG*results(5,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,RAD_to_DEG*results(6,:),'r', ...
            'LineWidth',2)
 axis([0, End_Time, - 5, 5])
hold off
xlabel('Time(s)', 'FontSize',14)
ylabel('Attitude (degrees)', 'FontSize',14)
hold off

figure
plot(results(7,:) ,RAD_to_DEG*results(17,:), ...
            'LineWidth',2)
title('Angulare possition Euler error Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,RAD_to_DEG*results(18,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,RAD_to_DEG*results(19,:),'r', ...
            'LineWidth',2)
 axis([0, End_Time, - 2, 18])
xlabel('Time(s)', 'FontSize',14)
ylabel('Attitude (degrees)', 'FontSize',14)
set(legend({'${\phi}_e$','${\theta}_e$'                                    '${\psi}_e$'},
'FontSize',20,'Location','northeast'),'interpreter','latex')
hold off

 figure
plot(results(7,:) ,results(33,:), ...
            'LineWidth',2)
title('delta timer   Vs Time')
set(legend({'$\hat{\delta}_time$'}, 'FontSize',20),'interpreter','latex')
xlabel('Time(s)')
ylabel('delta time (s)')
hold off

figure
plot(results(7,:) ,results(33,:), ...
            'LineWidth',2)
title('delta timer   Vs Time')
set(legend({'$\hat{\delta}_time$'}, 'FontSize',20),'interpreter','latex')
xlabel('Time(s)')
ylabel('delta time (s)')
hold off

figure
plot(results(7,:) ,results(43,:), ...
            'LineWidth',2)
xlabel('Time(s)', 'FontSize',14)
set(ylabel({'$\hat{\delta}$'}, 'FontSize',20),'interpreter','latex')
set(title({'$\hat{\delta}$'}, 'FontSize',20),'interpreter','latex')
```

```
axis([0, End_Time, - 0, 1.5])
hold off

hold off

figure
plot(results(7,:) ,results(41,:), ...
            'LineWidth',2)
title('reread   Vs Time')
set(legend({'$\hat{\delta}_time$'}, 'FontSize',20),'interpreter','latex')
xlabel('Time(s)')
ylabel('delta time (s)')
hold off
figure
plot(results(7,:) ,results(42,:), ...
            'LineWidth',2)
title('Time to read seriel Vs Time')
set(legend({'$\hat{\delta}_time$'}, 'FontSize',20),'interpreter','latex')
xlabel('Time(s)')
ylabel('delta time (s)')
hold off

figure
plot(results(7,:) ,results(11,:), ...
            'LineWidth',2)
title('Torque Distrubance estimate deterministic Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,results(12,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(13,:),'r', ...
            'LineWidth',2)
hold on
axis([0, End_Time, - 2, 2])
legend({'\it{\tau_p} \rm{(Nm)}','\it{\tau_q} \rm{(Nm)}','\it{\tau_r} \rm{(Nm)}'}, 'FontSize',16)
xlabel('Time(s)', 'FontSize',14)
ylabel('Estimated torque disturbance (Nm)', 'FontSize',14)
hold off

figure
plot(results(7,:) ,results(11,:) - results(23,:), ...
            'LineWidth',2)
title('Torque Distrubance estimate error  deterministic Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,results(12,:) - results(24,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(13,:)  - results(25,:),'r', ...
            'LineWidth',2)
hold on
axis([0, End_Time, - 2, 2])
legend({'\it{\tau_p} \rm{(Nm)}','\it{\tau_q} \rm{(Nm)}','\it{\tau_r} \rm{(Nm)}'}, 'FontSize',16)
xlabel('Time(s)', 'FontSize',14)
ylabel('Estimated torque disturbance (Nm)', 'FontSize',14)
hold off

figure
plot(results(7,:) ,results(44,:), ...
            'LineWidth',2)
title('OMEGA Vs Time', 'FontSize',14)
```

```
hold on
plot(results(7,:) ,results(45,:) ,'g', ...
          'LineWidth',2)
hold on
plot(results(7,:) ,results(46,:)  ,'r', ...
          'LineWidth',2)
hold on
axis([0, End_Time, - 40, 40])
xlabel('Time(s)', 'FontSize',14)
hold off

figure
subplot(2,1,1);
plot(results(7,:) , Input(1,:),'k', ...
          'LineWidth',2)
title('Control Thrust U_1 Vs Time', 'FontSize',14)
xlabel('Time(s)', 'FontSize',14)
ylabel('Force(N)', 'FontSize',14)
axis([0, End_Time,15,30])
hold off
subplot(2,1,2);
plot(results(7,:) ,Input(2,:) ,'b', ...
          'LineWidth',2)
title('Control TorqueVs Time', 'FontSize',14)
hold on
plot(results(7,:) ,Input(3,:)  ,'g', ...
          'LineWidth',2)
hold on
plot(results(7,:) ,Input(4,:)  ,'r', ...
          'LineWidth',2)
hold on
axis([0, End_Time, - 1, 1])
legend({'U_2','U_3','U_4'}, 'FontSize',12,'Location','northeast', 'Orientation', 'Horizontal')
xlabel('Time(s)', 'FontSize',14)
ylabel('Torque(Nm)', 'FontSize',14)
hold off

RAD_to_DEG = 180/pi;
figure
plot(results(7,:) ,RAD_to_DEG*results(4,:), ...
          'LineWidth',2)
title('Attitude Euler RepresentationVs Time', 'FontSize',14)
hold on
plot(results(7,:) ,RAD_to_DEG*results(5,:),'g', ...
          'LineWidth',2)
hold on
plot(results(7,:) ,RAD_to_DEG*results(6,:),'r', ...
          'LineWidth',2)
hold on
plot(results(7,:) ,RAD_to_DEG*results(29,:),'--b' , ...
          'LineWidth',1)
hold on
plot(results(7,:) ,RAD_to_DEG*results(30,:),'--g' , ...
          'LineWidth',1)
 hold on
plot(results(7,:) ,RAD_to_DEG*results(31,:),'--r' , ...
          'LineWidth',1)
axis([0, End_Time, - RAD_to_DEG*2, RAD_to_DEG*2])
```

```
legend({'\it{\phi} \rm{(Pitch Sensor reading)}','\it{\theta} \rm{(Roll Sensor reading)}','\it{\psi} \rm{(Yaw Sensor reading)}','\it{\phi_d} \rm{(Pitch reference signal)}','\it{\theta_d} \rm{(Roll reference signal)}','\it{\psi_d} \rm{(Yaw reference signal)}'}, 'FontSize',16)
xlabel('Time(s)', 'FontSize',14)
ylabel('Radians', 'FontSize',14)
hold off

figure
plot(results(7,:) ,Input(2,:),'b', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,Input(3,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,Input(4,:),'r', ...
            'LineWidth',2)
hold on

title('Control input Vs Time')
legend({'\it{U_2} \rm{(Nm)}','\it{U_3} \rm{(Nm)}','\it{U_4} \rm{(Nm)}'}, 'FontSize',16)
xlabel('Time(s)')
ylabel('delta time (s)')
axis([0, End_Time,-10,10])
hold off

figure
plot(results(7,:) ,results(47,:),'b' , ...
            'LineWidth',2)
        title('v 1 tilda Vs Time')
hold on
plot(results(7,:) ,results(48,:),'g' , ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(49,:),'r' , ...
            'LineWidth',2)
hold off

figure
plot(results(7,:) ,results(59,:),'b' , ...
            'LineWidth',2)
        title('n 1 tilda Vs Time')
hold on
plot(results(7,:) ,results(60,:),'g' , ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(61,:),'r' , ...
            'LineWidth',2)
hold off

figure
plot(results(7,:) ,results(50,:),'b' , ...
            'LineWidth',2)
        title('h_4 Vs Time')
hold on
plot(results(7,:) ,results(51,:),'g' , ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(52,:),'r' , ...
            'LineWidth',2)
hold off
```

```
figure
plot(results(7,:) ,results(53,:),'b' , ...
            'LineWidth',2)
        title('h_5 Vs Time')
hold on
plot(results(7,:) ,results(54,:),'g' , ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(55,:),'r' , ...
            'LineWidth',2)
hold off

figure
plot(results(7,:) ,Input(2,:),'b', ...
            'LineWidth',2)
title('Control input Vs Time')
xlabel('Time(s)')
ylabel('delta time (s)')
%axis([0, End_Time,-10,10])
hold off

figure
plot(results(7,:) ,Input(3,:),'g', ...
            'LineWidth',2)
hold on
title('Control input Vs Time')
xlabel('Time(s)')
ylabel('delta time (s)')
axis([0, End_Time,-10,10])
hold off

 figure
plot(results(7,:) ,Input(4,:),'r', ...
            'LineWidth',2)
hold on
 title('Control input Vs Time')
xlabel('Time(s)')
ylabel('delta time (s)')
axis([0, End_Time,-10,10])
hold off

figure
plot(results(7,:) ,results(20,:), ...
            'LineWidth',2)
title('alpha w dot Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,results(21,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(22,:),'r', ...
            'LineWidth',2)
hold on
legend({'\it{\phi} \rm{(Pitch Sensor reading)}','\it{\theta} \rm{(Roll Sensor reading)}','\it{\psi} \rm{(Yaw Sensor reading)}'}, 'FontSize',16)
xlabel('Time(s)', 'FontSize',14)
ylabel('Radians', 'FontSize',14)
hold off

figure
```

```
plot(results(7,:) ,results(14,:), ...
            'LineWidth',2)
title('w_error Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,results(15,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(16,:),'r', ...
            'LineWidth',2)
hold on
legend({'\it{\phi} \rm{(Pitch Sensor reading)}','\it{\theta} \rm{(Roll Sensor reading)}','\it{\psi} \rm{(Yaw
Sensor reading)}'}, 'FontSize',16)
xlabel('Time(s)', 'FontSize',14)
ylabel('Radians', 'FontSize',14)
axis([0, End_Time, -45, 45])
hold off

figure
plot(results(7,:) ,results(56,:), ...
            'LineWidth',2)
title('alpha_w Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,results(57,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(58,:),'r', ...
            'LineWidth',2)
hold on
legend({'\it{\phi} \rm{(Pitch Sensor reading)}','\it{\theta} \rm{(Roll Sensor reading)}','\it{\psi} \rm{(Yaw
Sensor reading)}'}, 'FontSize',16)
xlabel('Time(s)', 'FontSize',14)
ylabel('Radians', 'FontSize',14)
hold off

figure
plot(results(7,:) ,results(71,:), ...
            'LineWidth',2)
title('temp_X Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,results(72,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(73,:),'r', ...
            'LineWidth',2)
hold on
legend({'\it{\phi} \rm{(Pitch Sensor reading)}','\it{\theta} \rm{(Roll Sensor reading)}','\it{\psi} \rm{(Yaw
Sensor reading)}'}, 'FontSize',16)
xlabel('Time(s)', 'FontSize',14)
ylabel('Radians', 'FontSize',14)
hold off

figure
plot(results(7,:) , results(23,:)/Delta_Time, ...
            'LineWidth',2)
title('Torque Distrubance estimate error  deterministic Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,results(24,:)/Delta_Time,'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(25,:)/Delta_Time,'r', ...
```

```
            'LineWidth',2)
hold on
axis([0, End_Time, - 2, 2])
legend({'\it{\tau_p} \rm{(Nm)}','\it{\tau_q} \rm{(Nm)}','\it{\tau_r} \rm{(Nm)}'}, 'FontSize',16)
xlabel('Time(s)', 'FontSize',14)
ylabel('Estimated torque disturbance (Nm)', 'FontSize',14)
hold off

figure
plot(results(7,:) ,results(74,:), ...
            'LineWidth',2)
title('LV_4 and V_4 V_4_on_L_V_4 ', 'FontSize',14)
hold on
plot(results(7,:) ,results(75,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(76,:),'r', ...
            'LineWidth',2)
hold on
legend({'\it{LV_4}','\rm{(V_4)}','\it{V_4_on_L_V_4}'}, 'FontSize',16)
xlabel('Time(s)', 'FontSize',14)
ylabel('LV_4', 'FontSize',14)
axis([0, End_Time,-10,100])
hold off

figure
plot(results(7,:) ,results(75,:),'g', ...
            'LineWidth',2)
title(' V_4 and V_{sum} Vs Time ', 'FontSize',14)
hold on
plot(results(7,:) ,results(76,:),'r', ...
            'LineWidth',2)
hold on
axis([0, End_Time,0,10])
legend({'\it{V_4}','\it{V_{sum}}'}, 'FontSize',16)
xlabel('Time(s)', 'FontSize',14)
ylabel('V_4, V_{sum}', 'FontSize',14)
hold off

figure
plot(results(7,:) ,results(77,:), ...
            'LineWidth',2)
title('Stochastic  multi plicitivedisturbance Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,results(78,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) , results(79,:),'r', ...
            'LineWidth',2)
hold on
axis([0, End_Time, - 2, 2])
%axis([0, End_Time, - -45, 45])
legend({'\it{\tau_p} \rm{(Nm)}','\it{\tau_q} \rm{(Nm)}','\it{\tau_r} \rm{(Nm)}'}, 'FontSize',16)
xlabel('Time(s)', 'FontSize',14)
ylabel('Estimated torque disturbance (Nm)', 'FontSize',14)
hold off

figure
plot(results(7,:) ,dt^0.5*results(85,:), ...
            'LineWidth',2)
```

```
title('Stochastic  linaer disturbance Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,dt^0.5*results(86,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) , dt^0.5*results(87,:),'r', ...
            'LineWidth',2)
hold on
legend({'\it{\tau_p} \rm{(Nm)}','\it{\tau_q} \rm{(Nm)}','\it{\tau_r} \rm{(Nm)}'}, 'FontSize',16)
xlabel('Time(s)', 'FontSize',14)
ylabel('Estimated torque disturbance (Nm)', 'FontSize',14)
hold off


figure
plot(results(7,:) ,dt^0.5*results(88,:), ...
            'LineWidth',2)
title('Stochastic  angulare disturbance Vs Time', 'FontSize',14)
hold on
plot(results(7,:) ,dt^0.5*results(89,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,dt^0.5* results(90,:),'r', ...
            'LineWidth',2)
hold on
axis([0, End_Time, - 2, 2])
legend({'\it{\tau_p} \rm{(Nm)}','\it{\tau_q} \rm{(Nm)}','\it{\tau_r} \rm{(Nm)}'}, 'FontSize',16)
xlabel('Time(s)', 'FontSize',14)
ylabel('Estimated torque disturbance (Nm)', 'FontSize',14)
hold off

figure
plot(results(7,:) , results(80,:),'b', ...
            'LineWidth',2)
title('Motor Force Vs Time', 'FontSize',14)
hold on
plot(results(7,:) , results(81,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) , results(82,:),'r', ...
            'LineWidth',2)
hold on
plot(results(7,:) , results(83,:),'k', ...
            'LineWidth',2)

xlabel('Time(s)', 'FontSize',14)
ylabel('Force(N)', 'FontSize',14)
axis([0, End_Time,0,30])
hold off

figure
plot(results(7,:) , results(34,:),'b', ...
            'LineWidth',2)
title('velocity estimate Vs Time', 'FontSize',14)
hold on
plot(results(7,:) , results(35,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) , results(36,:),'r', ...
            'LineWidth',2)
```

```
xlabel('Time(s)', 'FontSize',14)
ylabel('Velocity(ms^-1)', 'FontSize',14)
hold off

%% check initial roll refference as it is 90 degrees
```

```
function [Omega, Partial_Omega_wrt_partial_n_1e, Partial_Omega_wrt_partial_v_1_hat ,
Partial_Omega_wrt_partial_n_1d_dot , Partial_Omega_wrt_partial_n_1d_double_dot,
Partial_M_1_to_6_wrt_partial_n_1e, Partial_M_1_to_6_wrt_partial_v_1_hat,
Partial_M_1_to_6_wrt_partial_n_1d_dot, Partial_M_1_to_6_wrt_partial_n_1d_double_dot,
Sum_M1_to_M6] = Omega_Partial_stochastic_one_step_ahead_backstepping(m, epsilon_2,gamma,
epsilon_3,min_eigen_value_D_1,K_1, K_2, D_1,c , n_1e, v_1_hat, v_1e, n_1d_dot,
n_1d_double_dot, alpha_1)

sigma_n_1e= [ n_1e(1,1)/(1+n_1e(1,1)^2)^0.5;
              n_1e(2,1)/(1+n_1e(2,1)^2)^0.5;
              n_1e(3,1)/(1+n_1e(3,1)^2)^0.5];
sigma_v_1e = [ v_1e(1,1)/(1+v_1e(1,1)^2)^0.5;
              v_1e(2,1)/(1+v_1e(2,1)^2)^0.5;
              v_1e(3,1)/(1+v_1e(3,1)^2)^0.5];
sigma_dash_n_1e = [ 1/(1+n_1e(1,1)^2)^1.5, 0, 0;
                    0, 1/(1+n_1e(2,1)^2)^1.5, 0;
                    0, 0, 1/(1+n_1e(3,1)^2)^1.5];
sigma_double_dash_n_1e = -3*[ 1/(1+n_1e(1,1)^2)^2.5, 0, 0;
                              0, 1/(1+n_1e(2,1)^2)^2.5, 0;
                              0, 0, 1/(1+n_1e(3,1)^2)^2.5];
sigma_dash_v_1e = [ 1/(1+v_1e(1,1)^2)^1.5, 0, 0;
                    0, 1/(1+v_1e(2,1)^2)^1.5, 0;
                    0, 0, 1/(1+v_1e(3,1)^2)^1.5];
delta_1_v_1_hat = 1 + 0.5 * ((v_1_hat.'))*v_1_hat;
norm_sigma_n_1e = (sigma_n_1e.'*sigma_n_1e)^0.5;
F_1 = K_1*sigma_dash_n_1e*(v_1_hat - n_1d_dot)/(delta_1_v_1_hat);
g=9.81;
mge_3 = m*g*[0;0;1];
gamma_1 = gamma;
sigma_n_1e_squared =sigma_n_1e.'*sigma_n_1e;
alpha_1 = -K_1*sigma_n_1e/delta_1_v_1_hat + n_1d_dot;

G_1 = eye(3)-(K_1*sigma_n_1e*(v_1_hat.')/delta_1_v_1_hat^2);
norm_v_1e = ((v_1e.')*v_1e)^0.5;

N = eye(3)+(K_1*sigma_n_1e*(v_1_hat.')/delta_1_v_1_hat^2)/(1-
sigma_n_1e.'*K_1*v_1_hat/delta_1_v_1_hat^2);
N_1 = N;
M_1                                = D_1*alpha_1-K_2*sigma_v_1e;
M_2                                =  -sigma_n_1e*gamma_1 * 0.5 *
min_eigen_value_D_1*(2+alpha_1.'*alpha_1)/delta_1_v_1_hat/2;
M_3                                =
(gamma_1*min_eigen_value_D_1/2)*K_1*sigma_n_1e*v_1_hat.'*sigma_n_1e/delta_1_v_1_hat^2;
M_4                                 = -((v_1_hat.')*D_1*v_1_hat)/(delta_1_v_1_hat)^2 *
K_1*sigma_n_1e;
M_5                                 = c*(2*alpha_1+v_1e)/(delta_1_v_1_hat)*(1+v_1e.'*v_1e)^0.5;
M_6                                = K_1*sigma_dash_n_1e*(K_1*sigma_n_1e)/(delta_1_v_1_hat)^2+
n_1d_double_dot;
M_7                                = mge_3-
2*gamma_1*m*(K_1*sigma_dash_n_1e)/(delta_1_v_1_hat)*v_1e/(1+v_1e.'*v_1e)^0.5;

Partial_N_with_respect_to_n_1e_times_M8      =
(v_1_hat.')*(M_1+M_2+M_3+M_4+M_5+M_6)*N*((K_1*(sigma_dash_n_1e)/(delta_1_v_1_hat)^2))/(1-
sigma_n_1e.'*K_1*v_1_hat/(delta_1_v_1_hat)^2);
Partial_N_with_respect_to_v_1_hat_times_M8 =
+(K_1*sigma_n_1e*(M_1+M_2+M_3+M_4+M_5+M_6).'*N.')/((1-
sigma_n_1e.'*K_1*v_1_hat/(delta_1_v_1_hat)^2))*(eye(3)/(delta_1_v_1_hat)^2-
2*(v_1_hat*(v_1_hat.'))/(delta_1_v_1_hat)^3);
```

```
Partial_M1_by_partial_v_1_hat   = (D_1*K_1*(sigma_n_1e*(v_1_hat.'))/(delta_1_v_1_hat)^2-
K_2*sigma_dash_v_1e*G_1);
Partial_M1_by_partial_n_1e       = -
(D_1+K_2*sigma_dash_v_1e)*K_1*(sigma_dash_n_1e)/(delta_1_v_1_hat);
Partial_M1_by_partial_n_1d_dot = (D_1+K_2*sigma_dash_v_1e);

Partial_M2_by_partial_v_1_hat   = -M_3/delta_1_v_1_hat*v_1_hat.' - sigma_n_1e*gamma_1 * 0.5 *
min_eigen_value_D_1*(2*alpha_1.'*  (K_1*sigma_n_1e*(v_1_hat.')/delta_1_v_1_hat^2)
)/delta_1_v_1_hat/2;
Partial_M2_by_partial_n_1e       = -sigma_dash_n_1e*gamma_1 * 0.5 *
min_eigen_value_D_1*(2+alpha_1.'*alpha_1)/delta_1_v_1_hat/2 + sigma_n_1e*gamma_1 * 0.5 *
min_eigen_value_D_1*(2*alpha_1.'*K_1*sigma_dash_n_1e/delta_1_v_1_hat)/delta_1_v_1_hat/2;
Partial_M2_by_partial_n_1d_dot = -sigma_n_1e*gamma_1 * 0.5 *
min_eigen_value_D_1*(2*alpha_1.')/delta_1_v_1_hat/2;

epsilon_2 = 0;
Partial_M3_by_partial_v_1e = -(gamma_1^3*epsilon_2*min_eigen_value_D_1/4) *  (eye(3)-
2*(v_1e*v_1e.')/(1+norm_v_1e^2))/(1+norm_v_1e^2);
Partial_M3_by_partial_v_1_hat  = Partial_M3_by_partial_v_1e*G_1-
(gamma_1*K_1*min_eigen_value_D_1/2)*(sigma_n_1e*sigma_n_1e.')*(eye(3) -
2*(v_1_hat*v_1_hat.')/delta_1_v_1_hat)/delta_1_v_1_hat^2;
Partial_M3_by_partial_n_1e
=Partial_M3_by_partial_v_1e*K_1*sigma_dash_n_1e/delta_1_v_1_hat-
(gamma_1*min_eigen_value_D_1/2)*K_1*(sigma_dash_n_1e*(v_1_hat.'*sigma_n_1e)+sigma_n_1e*
v_1_hat.'*sigma_dash_n_1e)/delta_1_v_1_hat^2  ;
Partial_M3_by_partial_n_1d_dot = -Partial_M3_by_partial_v_1e;

Partial_M4_by_partial_v_1_hat    = (-
2*(K_1*sigma_n_1e*(v_1_hat.'))/(delta_1_v_1_hat)^2*D_1+2*((v_1_hat.')*D_1*v_1_hat)/(delta_1_v_1
_hat)^3*K_1*sigma_n_1e*(v_1_hat.'));
Partial_M4_by_partial_n_1e       = -
((v_1_hat.')*D_1*v_1_hat)/(delta_1_v_1_hat)^2*K_1*sigma_dash_n_1e;
Partial_M4_by_partial_n_1d_dot = [0 0 0; 0 0 0; 0 0 0];

% c = c -+ min_eigen_value_D_1 that is the c already has had
% min_eigen_value_D_1 added
M_5                              = c*(2*alpha_1+v_1e)/(delta_1_v_1_hat)*(1+v_1e.'*v_1e)^0.5;
temp = c*(2*alpha_1+v_1e)/(delta_1_v_1_hat)*(1+v_1e.'*v_1e)^-0.5*v_1e.';
Partial_M5_by_partial_v_1_hat   =
(c)*1/(delta_1_v_1_hat)*((eye(3)+K_1*(sigma_n_1e*(v_1_hat.'))/(delta_1_v_1_hat)^2)-
(2*alpha_1+v_1e)/(delta_1_v_1_hat)*(v_1_hat.'));
Partial_M5_by_partial_n_1e       =
(c)*1/(delta_1_v_1_hat)*(K_1*(sigma_dash_n_1e)/(delta_1_v_1_hat));
Partial_M5_by_partial_n_1d_dot = (c)*1/(delta_1_v_1_hat)*eye(3);

Partial_M5_by_partial_v_1_hat   = Partial_M5_by_partial_v_1_hat *(1+v_1e.'*v_1e)^0.5 +temp*G_1;
Partial_M5_by_partial_n_1e       = Partial_M5_by_partial_n_1e *(1+v_1e.'*v_1e)^0.5 +
temp*K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Partial_M5_by_partial_n_1d_dot = Partial_M5_by_partial_n_1d_dot *(1+v_1e.'*v_1e)^0.5-temp;

Partial_M6_by_partial_v_1_hat  = K_1*sigma_dash_n_1e*(K_1*sigma_n_1e)/(delta_1_v_1_hat)^3*(-
2*v_1_hat.');
Partial_M6_by_partial_n_1e   =
K_1*sigma_dash_n_1e*(K_1*sigma_dash_n_1e)/(delta_1_v_1_hat)^2 +
(K_1/(delta_1_v_1_hat)^2)*sigma_double_dash_n_1e*K_1*[ sigma_n_1e(1,1) 0 0;

0, sigma_n_1e(2,1), 0;

0, 0, sigma_n_1e(3,1)];
Partial_M6_by_partial_n_1d_dot   = 0*eye(3);
```

```
Partial_M6_by_partial_n_1d_double_dot        = eye(3);

Partial_M7_by_partial_v_1e                   = -
m*(K_1*sigma_dash_n_1e)/(delta_1_v_1_hat)*(eye(3) -
v_1e*(v_1e.')/(1+v_1e.'*v_1e))/(1+v_1e.'*v_1e)^0.5;
Partial_M7_by_partial_v_1_hat                =
m*(K_1*sigma_dash_n_1e)/(delta_1_v_1_hat)*((v_1e/(1+v_1e.'*v_1e)^0.5)*(v_1_hat.'))/delta_1_v_1_
hat +Partial_M7_by_partial_v_1e    *G_1;
Partial_M7_by_partial_n_1e                   = Partial_M7_by_partial_v_1e *
(K_1*sigma_dash_n_1e)/(delta_1_v_1_hat) -m*(
(K_1*sigma_double_dash_n_1e)/(delta_1_v_1_hat)*(1/(1+v_1e.'*v_1e)^0.5)*[v_1e(1,1),0,0;

0,v_1e(2,1),0;

0,0,v_1e(3,1)]);
Partial_M7_by_partial_n_1d_dot               = -Partial_M7_by_partial_v_1e    ;

Partial_M7_by_partial_v_1_hat    = 2*gamma_1*Partial_M7_by_partial_v_1_hat   ;
Partial_M7_by_partial_n_1e = 2*gamma_1*Partial_M7_by_partial_n_1e;
Partial_M7_by_partial_n_1d_dot  = 2*gamma_1*Partial_M7_by_partial_n_1d_dot ;

Partial_Omega_wrt_partial_n_1e               = m*Partial_N_with_respect_to_n_1e_times_M8 +
m*N_1*(Partial_M1_by_partial_n_1e + Partial_M2_by_partial_n_1e + Partial_M3_by_partial_n_1e +
Partial_M4_by_partial_n_1e + Partial_M5_by_partial_n_1e + Partial_M6_by_partial_n_1e)
+Partial_M7_by_partial_n_1e;
Partial_Omega_wrt_partial_v_1_hat            = m*Partial_N_with_respect_to_v_1_hat_times_M8 +
m*N_1*(Partial_M1_by_partial_v_1_hat + Partial_M2_by_partial_v_1_hat +
Partial_M3_by_partial_v_1_hat + Partial_M4_by_partial_v_1_hat + Partial_M5_by_partial_v_1_hat +
Partial_M6_by_partial_v_1_hat) + Partial_M7_by_partial_v_1_hat;
Partial_Omega_wrt_partial_n_1d_dot           = m*N_1*(Partial_M1_by_partial_n_1d_dot +
Partial_M2_by_partial_n_1d_dot + Partial_M3_by_partial_n_1d_dot +
Partial_M4_by_partial_n_1d_dot + Partial_M5_by_partial_n_1d_dot +
Partial_M6_by_partial_n_1d_dot ) + Partial_M7_by_partial_n_1d_dot ;
Partial_Omega_wrt_partial_n_1d_double_dot = m*N_1 * Partial_M6_by_partial_n_1d_double_dot;
Partial_M_1_to_6_wrt_partial_n_1d_double_dot = Partial_M6_by_partial_n_1d_double_dot;

Partial_M_1_to_6_wrt_partial_n_1e=Partial_M1_by_partial_n_1e+Partial_M2_by_partial_n_1e+Partia
l_M3_by_partial_n_1e+Partial_M4_by_partial_n_1e+Partial_M5_by_partial_n_1e+Partial_M6_by_par
tial_n_1e;
Partial_M_1_to_6_wrt_partial_v_1_hat=Partial_M1_by_partial_v_1_hat+Partial_M2_by_partial_v_1_h
at+Partial_M3_by_partial_v_1_hat+Partial_M4_by_partial_v_1_hat+Partial_M5_by_partial_v_1_hat+
Partial_M6_by_partial_v_1_hat;
Partial_M_1_to_6_wrt_partial_n_1d_dot=Partial_M1_by_partial_n_1d_dot+Partial_M2_by_partial_n_
1d_dot+Partial_M3_by_partial_n_1d_dot+Partial_M4_by_partial_n_1d_dot+Partial_M5_by_partial_n_
1d_dot+Partial_M6_by_partial_n_1d_dot;

Sum_M1_to_M6 = M_1+M_2+M_3+M_4+M_5+M_6;
Omega = m*N*(M_1+M_2+M_3+M_4+M_5+M_6) + M_7;

end
```

```
function  [Partial_L_wrt_q_times_vector , Partial_L_wrt_alpha_q_times_vector,L] =
calc_L_MRP_norm_squared (q, alpha_q,vector)
%function [L_norm_squared, Partial_L_norm_squared_wrt_q, Partial_L_norm_squared_wrt_alpha_q ]
= calc_L_MRP_norm_squared (q, alpha_q)
vector_1 = vector(1,1);
vector_2 = vector(2,1);
vector_3 = vector(3,1);
q_1 = q(1,1);
q_2 = q(2,1);
q_3 = q(3,1);
alpha_q_1 =alpha_q(1,1);
alpha_q_2 =alpha_q(2,1);
alpha_q_3 =alpha_q(3,1);


q_e = q-alpha_q;
q_e_squared = q_e.'*q_e;
X= 1 + q.'*q;
Y= 1+(alpha_q.')*alpha_q;
%L=(8/X^2);
L1=(8 /X^2) *[ q(3,1), 1, -q(1,1)-alpha_q(1,1);
               -1, -q(3,1), q(2,1) - alpha_q(2,1);
               alpha_q(1,1), alpha_q(2,1) 0];
L2=((4/X)*eye(3) -4/(X*Y)*(q+alpha_q)*(alpha_q.'))*[ 0 -1 0;
                                                      1 0 0;
                                                      0 0 0];
L3=-8*(q_e*(q_e.'*q_e+2+2*(alpha_q.')*alpha_q) +4*alpha_q*(1+q.'*q-q_e.'*alpha_q))/(X*X*Y*Y);
L4=[0 0 1]*(alpha_q*((alpha_q.'))-(alpha_q.')*alpha_q*eye(3))+(alpha_q.')*[0 -1 0; 1 0 0; 0 0 0];
L = L1+L2+L3*L4;

norm(L);

L_norm_squared = trace(L.'*L);
L_norm_squared = norm(L)^2;

Partial_L_wrt_alpha_q_times_vector = [ - vector(2,1)*((32*alpha_q(1,1)*(alpha_q(1,1)^2 -
alpha_q(1,1)*q(1,1) + alpha_q(2,1)^2 - alpha_q(2,1)*q(2,1) + alpha_q(3,1)^2 - alpha_q(3,1)*q(3,1) +
X) - 8*(alpha_q(1,1) - q(1,1))*(3*alpha_q(1,1)^2 - 2*alpha_q(1,1)*q(1,1) + 3*alpha_q(2,1)^2 -
2*alpha_q(2,1)*q(2,1) + 3*alpha_q(3,1)^2 - 2*alpha_q(3,1)*q(3,1) + q(1,1)^2 + q(2,1)^2 + q(3,1)^2 +
2))/((Y)^2*(X)^2) - (4*(alpha_q(1,1) + q(1,1)))/((Y)*(X)) - (4*alpha_q(1,1))/((Y)*(X)) + ((alpha_q(1,1) +
alpha_q(2,1)*alpha_q(3,1))*(24*alpha_q(1,1)^2 + 16*alpha_q(1,1)*q(1,1) + 8*alpha_q(2,1)^2 -
16*alpha_q(2,1)*q(2,1) + 8*alpha_q(3,1)^2 - 16*alpha_q(3,1)*q(3,1) + 8*q(1,1)^2 + 24*q(2,1)^2 +
24*q(3,1)^2 + 16))/((Y)^2*(X)^2) + (8*alpha_q(1,1)^2*(alpha_q(1,1) + q(1,1)))/((Y)^2*(X)) -
(4*alpha_q(1,1)*(32*alpha_q(1,1)*(alpha_q(1,1)^2 - alpha_q(1,1)*q(1,1) + alpha_q(2,1)^2 -
alpha_q(2,1)*q(2,1) + alpha_q(3,1)^2 - alpha_q(3,1)*q(3,1) + X) - 8*(alpha_q(1,1) -
q(1,1))*(3*alpha_q(1,1)^2 - 2*alpha_q(1,1)*q(1,1) + 3*alpha_q(2,1)^2 - 2*alpha_q(2,1)*q(2,1) +
3*alpha_q(3,1)^2 - 2*alpha_q(3,1)*q(3,1) + q(1,1)^2 + q(2,1)^2 + q(3,1)^2 + 2))*(alpha_q(1,1) +
alpha_q(2,1)*alpha_q(3,1)))/((Y)^3*(X)^2)) - vector(1,1)*((4*alpha_q(2,1))/((Y)*(X)) - ((alpha_q(2,1) -
alpha_q(1,1)*alpha_q(3,1))*(24*alpha_q(1,1)^2 + 16*alpha_q(1,1)*q(1,1) + 8*alpha_q(2,1)^2 -
16*alpha_q(2,1)*q(2,1) + 8*alpha_q(3,1)^2 - 16*alpha_q(3,1)*q(3,1) + 8*q(1,1)^2 + 24*q(2,1)^2 +
24*q(3,1)^2 + 16))/((Y)^2*(X)^2) + (alpha_q(3,1)*(32*alpha_q(1,1)*(alpha_q(1,1)^2 -
alpha_q(1,1)*q(1,1) + alpha_q(2,1)^2 - alpha_q(2,1)*q(2,1) + alpha_q(3,1)^2 - alpha_q(3,1)*q(3,1) +
X) - 8*(alpha_q(1,1) - q(1,1))*(3*alpha_q(1,1)^2 - 2*alpha_q(1,1)*q(1,1) + 3*alpha_q(2,1)^2 -
2*alpha_q(2,1)*q(2,1) + 3*alpha_q(3,1)^2 - 2*alpha_q(3,1)*q(3,1) + q(1,1)^2 + q(2,1)^2 + q(3,1)^2 +
2)))/((Y)^2*(X)^2) + (4*alpha_q(1,1)*(32*alpha_q(1,1)*(alpha_q(1,1)^2 - alpha_q(1,1)*q(1,1) +
alpha_q(2,1)^2 - alpha_q(2,1)*q(2,1) + alpha_q(3,1)^2 - alpha_q(3,1)*q(3,1) + X) - 8*(alpha_q(1,1) -
q(1,1))*(3*alpha_q(1,1)^2 - 2*alpha_q(1,1)*q(1,1) + 3*alpha_q(2,1)^2 - 2*alpha_q(2,1)*q(2,1) +
3*alpha_q(3,1)^2 - 2*alpha_q(3,1)*q(3,1) + q(1,1)^2 + q(2,1)^2 + q(3,1)^2 + 2))*(alpha_q(2,1) -
alpha_q(1,1)*alpha_q(3,1)))/((Y)^3*(X)^2) - (8*alpha_q(1,1)*alpha_q(2,1)*(alpha_q(1,1) +
```

q(1,1)))/((Y)^2*(X))) - vector(3,1)*(8/(X)^2 - ((alpha_q(1,1)^2 + alpha_q(2,1)^2)*(24*alpha_q(1,1)^2 + 16*alpha_q(1,1)*q(1,1) + 8*alpha_q(2,1)^2 - 16*alpha_q(2,1)*q(2,1) + 8*alpha_q(3,1)^2 - 16*alpha_q(3,1)*q(3,1) + 8*q(1,1)^2 + 24*q(2,1)^2 + 24*q(3,1)^2 + 16))/((Y)^2*(X)^2) - (2*alpha_q(1,1)*(32*alpha_q(1,1)*(alpha_q(1,1)^2 - alpha_q(1,1)*q(1,1) + alpha_q(2,1)^2 - alpha_q(2,1)*q(2,1) + alpha_q(3,1)^2 - alpha_q(3,1)*q(3,1) + X) - 8*(alpha_q(1,1) - q(1,1))*(3*alpha_q(1,1)^2 - 2*alpha_q(1,1)*q(1,1) + 3*alpha_q(2,1)^2 - 2*alpha_q(2,1)*q(2,1) + 3*alpha_q(3,1)^2 - 2*alpha_q(3,1)*q(3,1) + q(1,1)^2 + q(2,1)^2 + q(3,1)^2 + 2)))/((Y)^2*(X)^2) + (4*alpha_q(1,1)*(32*alpha_q(1,1)*(alpha_q(1,1)^2 - alpha_q(1,1)*q(1,1) + alpha_q(2,1)^2 - alpha_q(2,1)*q(2,1) + alpha_q(3,1)^2 - alpha_q(3,1)*q(3,1) + X) - 8*(alpha_q(1,1) - q(1,1))*(3*alpha_q(1,1)^2 - 2*alpha_q(1,1)*q(1,1) + 3*alpha_q(2,1)^2 - 2*alpha_q(2,1)*q(2,1) + 3*alpha_q(3,1)^2 - 2*alpha_q(3,1)*q(3,1) + q(1,1)^2 + q(2,1)^2 + q(3,1)^2 + 2))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^3*(X)^2)),
vector(1,1)*(((alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1))*(32*alpha_q(1,1)*(2*alpha_q(2,1) - q(2,1)) - 8*(alpha_q(1,1) - q(1,1))*(6*alpha_q(2,1) - 2*q(2,1))))/((Y)^2*(X)^2) - (4*(alpha_q(1,1) + q(1,1)))/((Y)*(X)) - (8*(alpha_q(1,1) - q(1,1))*(2*Y + q_e_squared) - 32*alpha_q(1,1)*(-alpha_q.'*q_e + X))/((Y)^2*(X)^2) + (8*alpha_q(2,1)^2*(alpha_q(1,1) + q(1,1)))/((Y)^2*(X)) + (4*alpha_q(2,1)*(8*(alpha_q(1,1) - q(1,1))*(2*Y + q_e_squared) - 32*alpha_q(1,1)*(-alpha_q.'*q_e + X))*(alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1)))/((Y)^3*(X)^2)) + vector(3,1)*(((alpha_q(1,1)^2 + alpha_q(2,1)^2)*(32*alpha_q(1,1)*(2*alpha_q(2,1) - q(2,1)) - 8*(alpha_q(1,1) - q(1,1))*(6*alpha_q(2,1) - 2*q(2,1))))/((Y)^2*(X)^2) - (2*alpha_q(2,1)*(8*(alpha_q(1,1) - q(1,1))*(2*Y + q_e_squared) - 32*alpha_q(1,1)*(-alpha_q.'*q_e + X)))/((Y)^2*(X)^2) + (4*alpha_q(2,1)*(8*(alpha_q(1,1) - q(1,1))*(2*Y + q_e_squared) - 32*alpha_q(1,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^3*(X)^2)) - vector(2,1)*(((alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1))*(32*alpha_q(1,1)*(2*alpha_q(2,1) - q(2,1)) - 8*(alpha_q(1,1) - q(1,1))*(6*alpha_q(2,1) - 2*q(2,1))))/((Y)^2*(X)^2) - (alpha_q(3,1)*(8*(alpha_q(1,1) - q(1,1))*(2*Y + q_e_squared) - 32*alpha_q(1,1)*(-alpha_q.'*q_e + X)))/((Y)^2*(X)^2) + (4*alpha_q(2,1)*(8*(alpha_q(1,1) - q(1,1))*(2*Y + q_e_squared) - 32*alpha_q(1,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1)))/((Y)^3*(X)^2) + (8*alpha_q(1,1)*alpha_q(2,1)*(alpha_q(1,1) + q(1,1)))/((Y)^2*(X))),
vector(1,1)*((alpha_q(1,1)*(8*(alpha_q(1,1) - q(1,1))*(2*Y + q_e_squared) - 32*alpha_q(1,1)*(-alpha_q.'*q_e + X)))/((Y)^2*(X)^2) + ((alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1))*(32*alpha_q(1,1)*(2*alpha_q(3,1) - q(3,1)) - 8*(alpha_q(1,1) - q(1,1))*(6*alpha_q(3,1) - 2*q(3,1))))/((Y)^2*(X)^2) + (4*alpha_q(3,1)*(8*(alpha_q(1,1) - q(1,1))*(2*Y + q_e_squared) - 32*alpha_q(1,1)*(-alpha_q.'*q_e + X))*(alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1)))/((Y)^3*(X)^2) + (8*alpha_q(2,1)*alpha_q(3,1)*(alpha_q(1,1) + q(1,1)))/((Y)^2*(X))) - vector(2,1)*(((alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1))*(32*alpha_q(1,1)*(2*alpha_q(3,1) - q(3,1)) - 8*(alpha_q(1,1) - q(1,1))*(6*alpha_q(3,1) - 2*q(3,1))))/((Y)^2*(X)^2) - (alpha_q(2,1)*(8*(alpha_q(1,1) - q(1,1))*(2*Y + q_e_squared) - 32*alpha_q(1,1)*(-alpha_q.'*q_e + X)))/((Y)^2*(X)^2) + (4*alpha_q(3,1)*(8*(alpha_q(1,1) - q(1,1))*(2*Y + q_e_squared) - 32*alpha_q(1,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1)))/((Y)^3*(X)^2) + (8*alpha_q(1,1)*alpha_q(3,1)*(alpha_q(1,1) + q(1,1)))/((Y)^2*(X))) + vector(3,1)*(((alpha_q(1,1)^2 + alpha_q(2,1)^2)*(32*alpha_q(1,1)*(2*alpha_q(3,1) - q(3,1)) - 8*(alpha_q(1,1) - q(1,1))*(6*alpha_q(3,1) - 2*q(3,1))))/((Y)^2*(X)^2) + (4*alpha_q(3,1)*(8*(alpha_q(1,1) - q(1,1))*(2*Y + q_e_squared) - 32*alpha_q(1,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^3*(X)^2));

vector(3,1)*(((alpha_q(1,1)^2 + alpha_q(2,1)^2)*(32*alpha_q(2,1)*(2*alpha_q(1,1) - q(1,1)) - 8*(alpha_q(2,1) - q(2,1))*(6*alpha_q(1,1) - 2*q(1,1))))/((Y)^2*(X)^2) - (2*alpha_q(1,1)*(8*(alpha_q(2,1) - q(2,1))*(2*Y + q_e_squared) - 32*alpha_q(2,1)*(-alpha_q.'*q_e + X)))/((Y)^2*(X)^2) + (4*alpha_q(1,1)*(8*(alpha_q(2,1) - q(2,1))*(2*Y + q_e_squared) - 32*alpha_q(2,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^3*(X)^2)) - vector(2,1)*(((alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1))*(32*alpha_q(2,1)*(2*alpha_q(1,1) - q(1,1)) - 8*(alpha_q(2,1) - q(2,1))*(6*alpha_q(1,1) - 2*q(1,1))))/((Y)^2*(X)^2) - (4*(alpha_q(2,1) + q(2,1)))/((Y)*(X)) - (8*(alpha_q(2,1) - q(2,1))*(2*Y + q_e_squared) - 32*alpha_q(2,1)*(-alpha_q.'*q_e + X))/((Y)^2*(X)^2) + (8*alpha_q(1,1)^2*(alpha_q(2,1) + q(2,1)))/((Y)^2*(X)) + (4*alpha_q(1,1)*(8*(alpha_q(2,1) - q(2,1))*(2*Y + q_e_squared) - 32*alpha_q(2,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1)))/((Y)^3*(X)^2)) + vector(1,1)*((alpha_q(3,1)*(8*(alpha_q(2,1) - q(2,1))*(2*Y + q_e_squared) - 32*alpha_q(2,1)*(-alpha_q.'*q_e + X)))/((Y)^2*(X)^2) + ((alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1))*(32*alpha_q(2,1)*(2*alpha_q(1,1) - q(1,1)) - 8*(alpha_q(2,1) - q(2,1))*(6*alpha_q(1,1) - 2*q(1,1))))/((Y)^2*(X)^2) + (4*alpha_q(1,1)*(8*(alpha_q(2,1) - q(2,1))*(2*Y +

q_e_squared) - 32*alpha_q(2,1)*(-alpha_q.'*q_e + X))*(alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1)))/((Y)^3*(X)^2) + (8*alpha_q(1,1)*alpha_q(2,1)*(alpha_q(2,1) + q(2,1)))/((Y)^2*(X))), vector(1,1)*((32*alpha_q(2,1)*(alpha_q(1,1)^2 - alpha_q(1,1)*q(1,1) + alpha_q(2,1)^2 - alpha_q(2,1)*q(2,1) + alpha_q(3,1)^2 - alpha_q(3,1)*q(3,1) + X) - 8*(alpha_q(2,1) - q(2,1))*(3*alpha_q(1,1)^2 - 2*alpha_q(1,1)*q(1,1) + 3*alpha_q(2,1)^2 - 2*alpha_q(2,1)*q(2,1) + 3*alpha_q(3,1)^2 - 2*alpha_q(3,1)*q(3,1) + q(1,1)^2 + q(2,1)^2 + q(3,1)^2 + 2))/((Y)^2*(X)^2) - (4*(alpha_q(2,1) + q(2,1)))/((Y)*(X)) - (4*alpha_q(2,1))/((Y)*(X)) + ((alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1))*(8*alpha_q(1,1)^2 - 16*alpha_q(1,1)*q(1,1) + 24*alpha_q(2,1)^2 + 16*alpha_q(2,1)*q(2,1) + 8*alpha_q(3,1)^2 - 16*alpha_q(3,1)*q(3,1) + 24*q(1,1)^2 + 8*q(2,1)^2 + 24*q(3,1)^2 + 16))/((Y)^2*(X)^2) + (8*alpha_q(2,1)^2*(alpha_q(2,1) + q(2,1)))/((Y)^2*(X)) - (4*alpha_q(2,1)*(32*alpha_q(2,1)*(alpha_q(1,1)^2 - alpha_q(1,1)*q(1,1) + alpha_q(2,1)^2 - alpha_q(2,1)*q(2,1) + alpha_q(3,1)^2 - alpha_q(3,1)*q(3,1) + X) - 8*(alpha_q(2,1) - q(2,1))*(3*alpha_q(1,1)^2 - 2*alpha_q(1,1)*q(1,1) + 3*alpha_q(2,1)^2 - 2*alpha_q(2,1)*q(2,1) + 3*alpha_q(3,1)^2 - 2*alpha_q(3,1)*q(3,1) + q(1,1)^2 + q(2,1)^2 + q(3,1)^2 + 2))*(alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1)))/((Y)^3*(X)^2)) - vector(2,1)*(((alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1))*(8*alpha_q(1,1)^2 - 16*alpha_q(1,1)*q(1,1) + 24*alpha_q(2,1)^2 + 16*alpha_q(2,1)*q(2,1) + 8*alpha_q(3,1)^2 - 16*alpha_q(3,1)*q(3,1) + 24*q(1,1)^2 + 8*q(2,1)^2 + 24*q(3,1)^2 + 16))/((Y)^2*(X)^2) - (4*alpha_q(1,1))/((Y)*(X)) + (alpha_q(3,1)*(32*alpha_q(2,1)*(alpha_q(1,1)^2 - alpha_q(1,1)*q(1,1) + alpha_q(2,1)^2 - alpha_q(2,1)*q(2,1) + alpha_q(3,1)^2 - alpha_q(3,1)*q(3,1) + X) - 8*(alpha_q(2,1) - q(2,1))*(3*alpha_q(1,1)^2 - 2*alpha_q(1,1)*q(1,1) + 3*alpha_q(2,1)^2 - 2*alpha_q(2,1)*q(2,1) + 3*alpha_q(3,1)^2 - 2*alpha_q(3,1)*q(3,1) + q(1,1)^2 + q(2,1)^2 + q(3,1)^2 + 2)))/((Y)^2*(X)^2) - (4*alpha_q(2,1)*(32*alpha_q(2,1)*(alpha_q(1,1)^2 - alpha_q(1,1)*q(1,1) + alpha_q(2,1)^2 - alpha_q(2,1)*q(2,1) + alpha_q(3,1)^2 - alpha_q(3,1)*q(3,1) + X) - 8*(alpha_q(2,1) - q(2,1))*(3*alpha_q(1,1)^2 - 2*alpha_q(1,1)*q(1,1) + 3*alpha_q(2,1)^2 - 2*alpha_q(2,1)*q(2,1) + 3*alpha_q(3,1)^2 - 2*alpha_q(3,1)*q(3,1) + q(1,1)^2 + q(2,1)^2 + q(3,1)^2 + 2))*(alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1)))/((Y)^3*(X)^2) + (8*alpha_q(1,1)*alpha_q(2,1)*(alpha_q(2,1) + q(2,1)))/((Y)^2*(X))) - vector(3,1)*(8/(X)^2 - ((alpha_q(1,1)^2 + alpha_q(2,1)^2)*(8*alpha_q(1,1)^2 - 16*alpha_q(1,1)*q(1,1) + 24*alpha_q(2,1)^2 + 16*alpha_q(2,1)*q(2,1) + 8*alpha_q(3,1)^2 - 16*alpha_q(3,1)*q(3,1) + 24*q(1,1)^2 + 8*q(2,1)^2 + 24*q(3,1)^2 + 16))/((Y)^2*(X)^2) - (2*alpha_q(2,1)*(32*alpha_q(2,1)*(alpha_q(1,1)^2 - alpha_q(1,1)*q(1,1) + alpha_q(2,1)^2 - alpha_q(2,1)*q(2,1) + alpha_q(3,1)^2 - alpha_q(3,1)*q(3,1) + X) - 8*(alpha_q(2,1) - q(2,1))*(3*alpha_q(1,1)^2 - 2*alpha_q(1,1)*q(1,1) + 3*alpha_q(2,1)^2 - 2*alpha_q(2,1)*q(2,1) + 3*alpha_q(3,1)^2 - 2*alpha_q(3,1)*q(3,1) + q(1,1)^2 + q(2,1)^2 + q(3,1)^2 + 2)))/((Y)^2*(X)^2) + (4*alpha_q(2,1)*(32*alpha_q(2,1)*(alpha_q(1,1)^2 - alpha_q(1,1)*q(1,1) + alpha_q(2,1)^2 - alpha_q(2,1)*q(2,1) + alpha_q(3,1)^2 - alpha_q(3,1)*q(3,1) + X) - 8*(alpha_q(2,1) - q(2,1))*(3*alpha_q(1,1)^2 - 2*alpha_q(1,1)*q(1,1) + 3*alpha_q(2,1)^2 - 2*alpha_q(2,1)*q(2,1) + 3*alpha_q(3,1)^2 - 2*alpha_q(3,1)*q(3,1) + q(1,1)^2 + q(2,1)^2 + q(3,1)^2 + 2))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^3*(X)^2)),
vector(1,1)*((alpha_q(1,1)*(8*(alpha_q(2,1) - q(2,1))*(2*Y + q_e_squared) - 32*alpha_q(2,1)*(-alpha_q.'*q_e + X)))/((Y)^2*(X)^2) + ((alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1))*(32*alpha_q(2,1)*(2*alpha_q(3,1) - q(3,1)) - 8*(alpha_q(2,1) - q(2,1))*(6*alpha_q(3,1) - 2*q(3,1))))/((Y)^2*(X)^2) + (4*alpha_q(3,1)*(8*(alpha_q(2,1) - q(2,1))*(2*Y + q_e_squared) - 32*alpha_q(2,1)*(-alpha_q.'*q_e + X))*(alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1)))/((Y)^3*(X)^2) + (8*alpha_q(2,1)*alpha_q(3,1)*(alpha_q(2,1) + q(2,1)))/((Y)^2*(X))) - vector(2,1)*(((alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1))*(32*alpha_q(2,1)*(2*alpha_q(3,1) - q(3,1)) - 8*(alpha_q(2,1) - q(2,1))*(6*alpha_q(3,1) - 2*q(3,1))))/((Y)^2*(X)^2) - (alpha_q(2,1)*(8*(alpha_q(2,1) - q(2,1))*(2*Y + q_e_squared) - 32*alpha_q(2,1)*(-alpha_q.'*q_e + X)))/((Y)^2*(X)^2) + (4*alpha_q(3,1)*(8*(alpha_q(2,1) - q(2,1))*(2*Y + q_e_squared) - 32*alpha_q(2,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1)))/((Y)^3*(X)^2) + (8*alpha_q(1,1)*alpha_q(3,1)*(alpha_q(2,1) + q(2,1)))/((Y)^2*(X))) + vector(3,1)*(((alpha_q(1,1)^2 + alpha_q(2,1)^2)*(32*alpha_q(2,1)*(2*alpha_q(3,1) - q(3,1)) - 8*(alpha_q(2,1) - q(2,1))*(6*alpha_q(3,1) - 2*q(3,1))))/((Y)^2*(X)^2) + (4*alpha_q(3,1)*(8*(alpha_q(2,1) - q(2,1))*(2*Y + q_e_squared) - 32*alpha_q(2,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^3*(X)^2));

vector(1,1)*(8/(X)^2 + (alpha_q(3,1)*(8*(alpha_q(3,1) - q(3,1))*(2*Y + q_e_squared) - 32*alpha_q(3,1)*(-alpha_q.'*q_e + X)))/((Y)^2*(X)^2) + ((alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1))*(32*alpha_q(3,1)*(2*alpha_q(1,1) - q(1,1)) - 8*(alpha_q(3,1) - q(3,1))*(6*alpha_q(1,1) - 2*q(1,1))))/((Y)^2*(X)^2) + (4*alpha_q(1,1)*(8*(alpha_q(3,1) - q(3,1))*(2*Y +

q_e_squared) - 32*alpha_q(3,1)*(-alpha_q.'*q_e + X))*(alpha_q(2,1) -
alpha_q(1,1)*alpha_q(3,1)))/((Y)^3*(X)^2) + (8*alpha_q(1,1)*alpha_q(2,1)*(alpha_q(3,1) +
q(3,1)))/((Y)^2*(X))) - vector(2,1)*(((alpha_q(1,1) +
alpha_q(2,1)*alpha_q(3,1))*(32*alpha_q(3,1)*(2*alpha_q(1,1) - q(1,1)) - 8*(alpha_q(3,1) -
q(3,1))*(6*alpha_q(1,1) - 2*q(1,1))))/((Y)^2*(X)^2) - (4*(alpha_q(3,1) + q(3,1)))/((Y)*(X)) -
(8*(alpha_q(3,1) - q(3,1))*(2*Y + q_e_squared) - 32*alpha_q(3,1)*(-alpha_q.'*q_e + X))/((Y)^2*(X)^2)
+ (8*alpha_q(1,1)^2*(alpha_q(3,1) + q(3,1)))/((Y)^2*(X)) + (4*alpha_q(1,1)*(8*(alpha_q(3,1) -
q(3,1))*(2*Y + q_e_squared) - 32*alpha_q(3,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1) +
alpha_q(2,1)*alpha_q(3,1)))/((Y)^3*(X)^2)) - (2*alpha_q(1,1)*vector(3,1)*(8*(alpha_q(3,1) -
q(3,1))*(2*Y + q_e_squared) - 32*alpha_q(3,1)*(-alpha_q.'*q_e + X)))/((Y)^2*(X)^2) +
(vector(3,1)*(alpha_q(1,1)^2 + alpha_q(2,1)^2)*(32*alpha_q(3,1)*(2*alpha_q(1,1) - q(1,1)) -
8*(alpha_q(3,1) - q(3,1))*(6*alpha_q(1,1) - 2*q(1,1))))/((Y)^2*(X)^2) +
(4*alpha_q(1,1)*vector(3,1)*(8*(alpha_q(3,1) - q(3,1))*(2*Y + q_e_squared) - 32*alpha_q(3,1)*(-
alpha_q.'*q_e + X))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^3*(X)^2),
vector(1,1)*(((alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1))*(32*alpha_q(3,1)*(2*alpha_q(2,1) - q(2,1)) -
8*(alpha_q(3,1) - q(3,1))*(6*alpha_q(2,1) - 2*q(2,1))))/((Y)^2*(X)^2) - (4*(alpha_q(3,1) +
q(3,1)))/((Y)*(X)) - (8*(alpha_q(3,1) - q(3,1))*(2*Y + q_e_squared) - 32*alpha_q(3,1)*(-alpha_q.'*q_e +
X))/((Y)^2*(X)^2) + (8*alpha_q(2,1)^2*(alpha_q(3,1) + q(3,1)))/((Y)^2*(X)) +
(4*alpha_q(2,1)*(8*(alpha_q(3,1) - q(3,1))*(2*Y + q_e_squared) - 32*alpha_q(3,1)*(-alpha_q.'*q_e +
X))*(alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1)))/((Y)^3*(X)^2)) - vector(2,1)*(((alpha_q(1,1) +
alpha_q(2,1)*alpha_q(3,1))*(32*alpha_q(3,1)*(2*alpha_q(2,1) - q(2,1)) - 8*(alpha_q(3,1) -
q(3,1))*(6*alpha_q(2,1) - 2*q(2,1))))/((Y)^2*(X)^2) - (alpha_q(3,1)*(8*(alpha_q(3,1) - q(3,1))*(2*Y +
q_e_squared) - 32*alpha_q(3,1)*(-alpha_q.'*q_e + X)))/((Y)^2*(X)^2) - 8/(X)^2 +
(4*alpha_q(2,1)*(8*(alpha_q(3,1) - q(3,1))*(2*Y + q_e_squared) - 32*alpha_q(3,1)*(-alpha_q.'*q_e +
X))*(alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1)))/((Y)^3*(X)^2) +
(8*alpha_q(1,1)*alpha_q(2,1)*(alpha_q(3,1) + q(3,1)))/((Y)^2*(X))) -
(2*alpha_q(2,1)*vector(3,1)*(8*(alpha_q(3,1) - q(3,1))*(2*Y + q_e_squared) - 32*alpha_q(3,1)*(-
alpha_q.'*q_e + X)))/((Y)^2*(X)^2) + (vector(3,1)*(alpha_q(1,1)^2 +
alpha_q(2,1)^2)*(32*alpha_q(3,1)*(2*alpha_q(2,1) - q(2,1)) - 8*(alpha_q(3,1) - q(3,1))*(6*alpha_q(2,1)
- 2*q(2,1))))/((Y)^2*(X)^2) + (4*alpha_q(2,1)*vector(3,1)*(8*(alpha_q(3,1) - q(3,1))*(2*Y +
q_e_squared) - 32*alpha_q(3,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1)^2 +
alpha_q(2,1)^2))/((Y)^3*(X)^2), (vector(3,1)*(alpha_q(1,1)^2 + alpha_q(2,1)^2)*(8*alpha_q(1,1)^2 -
16*alpha_q(1,1)*q(1,1) + 8*alpha_q(2,1)^2 - 16*alpha_q(2,1)*q(2,1) + 24*alpha_q(3,1)^2 +
16*alpha_q(3,1)*q(3,1) + 24*q(1,1)^2 + 24*q(2,1)^2 + 8*q(3,1)^2 + 16))/((Y)^2*(X)^2) -
vector(1,1)*((4*alpha_q(2,1))/((Y)*(X)) - ((alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1))*(8*alpha_q(1,1)^2
- 16*alpha_q(1,1)*q(1,1) + 8*alpha_q(2,1)^2 - 16*alpha_q(2,1)*q(2,1) + 24*alpha_q(3,1)^2 +
16*alpha_q(3,1)*q(3,1) + 24*q(1,1)^2 + 24*q(2,1)^2 + 8*q(3,1)^2 + 16))/((Y)^2*(X)^2) +
(alpha_q(1,1)*(32*alpha_q(3,1)*(alpha_q(1,1)^2 - alpha_q(1,1)*q(1,1) + alpha_q(2,1)^2 -
alpha_q(2,1)*q(2,1) + alpha_q(3,1)^2 - alpha_q(3,1)*q(3,1) + X) - 8*(alpha_q(3,1) -
q(3,1))*(3*alpha_q(1,1)^2 - 2*alpha_q(1,1)*q(1,1) + 3*alpha_q(2,1)^2 - 2*alpha_q(2,1)*q(2,1) +
3*alpha_q(3,1)^2 - 2*alpha_q(3,1)*q(3,1) + q(1,1)^2 + q(2,1)^2 + q(3,1)^2 + 2)))/((Y)^2*(X)^2) +
(4*alpha_q(3,1)*(32*alpha_q(3,1)*(alpha_q(1,1)^2 - alpha_q(1,1)*q(1,1) + alpha_q(2,1)^2 -
alpha_q(2,1)*q(2,1) + alpha_q(3,1)^2 - alpha_q(3,1)*q(3,1) + X) - 8*(alpha_q(3,1) -
q(3,1))*(3*alpha_q(1,1)^2 - 2*alpha_q(1,1)*q(1,1) + 3*alpha_q(2,1)^2 - 2*alpha_q(2,1)*q(2,1) +
3*alpha_q(3,1)^2 - 2*alpha_q(3,1)*q(3,1) + q(1,1)^2 + q(2,1)^2 + q(3,1)^2 + 2))*(alpha_q(2,1) -
alpha_q(1,1)*alpha_q(3,1)))/((Y)^3*(X)^2) - (8*alpha_q(2,1)*alpha_q(3,1)*(alpha_q(3,1) +
q(3,1)))/((Y)^2*(X))) - vector(2,1)*(((alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1))*(8*alpha_q(1,1)^2 -
16*alpha_q(1,1)*q(1,1) + 8*alpha_q(2,1)^2 - 16*alpha_q(2,1)*q(2,1) + 24*alpha_q(3,1)^2 +
16*alpha_q(3,1)*q(3,1) + 24*q(1,1)^2 + 24*q(2,1)^2 + 8*q(3,1)^2 + 16))/((Y)^2*(X)^2) -
(4*alpha_q(1,1))/((Y)*(X)) + (alpha_q(2,1)*(32*alpha_q(3,1)*(alpha_q(1,1)^2 - alpha_q(1,1)*q(1,1) +
alpha_q(2,1)^2 - alpha_q(2,1)*q(2,1) + alpha_q(3,1)^2 - alpha_q(3,1)*q(3,1) + X) - 8*(alpha_q(3,1) -
q(3,1))*(3*alpha_q(1,1)^2 - 2*alpha_q(1,1)*q(1,1) + 3*alpha_q(2,1)^2 - 2*alpha_q(2,1)*q(2,1) +
3*alpha_q(3,1)^2 - 2*alpha_q(3,1)*q(3,1) + q(1,1)^2 + q(2,1)^2 + q(3,1)^2 + 2)))/((Y)^2*(X)^2) -
(4*alpha_q(3,1)*(32*alpha_q(3,1)*(alpha_q(1,1)^2 - alpha_q(1,1)*q(1,1) + alpha_q(2,1)^2 -
alpha_q(2,1)*q(2,1) + alpha_q(3,1)^2 - alpha_q(3,1)*q(3,1) + X) - 8*(alpha_q(3,1) -
q(3,1))*(3*alpha_q(1,1)^2 - 2*alpha_q(1,1)*q(1,1) + 3*alpha_q(2,1)^2 - 2*alpha_q(2,1)*q(2,1) +
3*alpha_q(3,1)^2 - 2*alpha_q(3,1)*q(3,1) + q(1,1)^2 + q(2,1)^2 + q(3,1)^2 + 2))*(alpha_q(1,1) +
alpha_q(2,1)*alpha_q(3,1)))/((Y)^3*(X)^2) + (8*alpha_q(1,1)*alpha_q(3,1)*(alpha_q(3,1) +
q(3,1)))/((Y)^2*(X))) - (4*alpha_q(3,1)*vector(3,1)*(32*alpha_q(3,1)*(alpha_q(1,1)^2 -
alpha_q(1,1)*q(1,1) + alpha_q(2,1)^2 - alpha_q(2,1)*q(2,1) + alpha_q(3,1)^2 - alpha_q(3,1)*q(3,1) +

X) - 8*(alpha_q(3,1) - q(3,1))*(3*alpha_q(1,1)^2 - 2*alpha_q(1,1)*q(1,1) + 3*alpha_q(2,1)^2 - 2*alpha_q(2,1)*q(2,1) + 3*alpha_q(3,1)^2 - 2*alpha_q(3,1)*q(3,1) + q(1,1)^2 + q(2,1)^2 + q(3,1)^2 + 2))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^3*(X)^2)];

Partial_L_wrt_q_times_vector =[vector(3,1)*((32*q(1,1)*(alpha_q(1,1) + q(1,1)))/(X)^3 - 8/(X)^2 + ((alpha_q(1,1)^2 + alpha_q(2,1)^2)*(16*alpha_q(1,1)^2 - 32*alpha_q(1,1)*(alpha_q(1,1) - 2*q(1,1)) + 16*alpha_q(2,1)^2 + 16*alpha_q(3,1)^2 + 8*(alpha_q(1,1) - q(1,1))*(2*alpha_q(1,1) - 2*q(1,1)) + 8*q_e_squared + 16))/((Y)^2*(X)^2) + (4*q(1,1)*(8*(alpha_q(1,1) - q(1,1))*(2*Y + q_e_squared) - 32*alpha_q(1,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^2*(X)^3)) - vector(2,1)*((32*q(1,1))/(X)^3 - (8*q(1,1))/(X)^2 - (4*alpha_q(1,1))/((Y)*(X)) + ((alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1))*(16*alpha_q(1,1)^2 - 32*alpha_q(1,1)*(alpha_q(1,1) - 2*q(1,1)) + 16*alpha_q(2,1)^2 + 16*alpha_q(3,1)^2 + 8*(alpha_q(1,1) - q(1,1))*(2*alpha_q(1,1) - 2*q(1,1)) + 8*q_e_squared + 16))/((Y)^2*(X)^2) + (4*q(1,1)*(8*(alpha_q(1,1) - q(1,1))*(2*Y + q_e_squared) - 32*alpha_q(1,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1)))/((Y)^2*(X)^3) + (8*alpha_q(1,1)*q(1,1)*(alpha_q(1,1) + q(1,1)))/((Y)*(X)^2)) + vector(1,1)*(((alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1))*(16*alpha_q(1,1)^2 - 32*alpha_q(1,1)*(alpha_q(1,1) - 2*q(1,1)) + 16*alpha_q(2,1)^2 + 16*alpha_q(3,1)^2 + 8*(alpha_q(1,1) - q(1,1))*(2*alpha_q(1,1) - 2*q(1,1)) + 8*q_e_squared + 16))/((Y)^2*(X)^2) - (4*alpha_q(2,1))/((Y)*(X)) - (32*q(1,1)*q(3,1))/(X)^3 + (4*q(1,1)*(8*(alpha_q(1,1) - q(1,1))*(2*Y + q_e_squared) - 32*alpha_q(1,1)*(-alpha_q.'*q_e + X))*(alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1)))/((Y)^2*(X)^3) + (8*alpha_q(2,1)*q(1,1)*(alpha_q(1,1) + q(1,1)))/((Y)*(X)^2)),
vector(3,1)*((32*q(2,1)*(alpha_q(1,1) + q(1,1)))/(X)^3 - ((32*alpha_q(1,1)*(alpha_q(2,1) - 2*q(2,1)) - 8*(alpha_q(1,1) - q(1,1))*(2*alpha_q(2,1) - 2*q(2,1)))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^2*(X)^2) + (4*q(2,1)*(8*(alpha_q(1,1) - q(1,1))*(2*Y + q_e_squared) - 32*alpha_q(1,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^2*(X)^3)) - vector(1,1)*((32*q(2,1)*q(3,1))/(X)^3 + ((32*alpha_q(1,1)*(alpha_q(2,1) - 2*q(2,1)) - 8*(alpha_q(1,1) - q(1,1))*(2*alpha_q(2,1) - 2*q(2,1)))*(alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1)))/((Y)^2*(X)^2) - (4*q(2,1)*(8*(alpha_q(1,1) - q(1,1))*(2*Y + q_e_squared) - 32*alpha_q(1,1)*(-alpha_q.'*q_e + X))*(alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1)))/((Y)^2*(X)^3) - (8*alpha_q(2,1)*q(2,1)*(alpha_q(1,1) + q(1,1)))/((Y)*(X)^2)) - vector(2,1)*((32*q(2,1))/(X)^3 - (8*q(2,1))/(X)^2 - ((32*alpha_q(1,1)*(alpha_q(2,1) - 2*q(2,1)) - 8*(alpha_q(1,1) - q(1,1))*(2*alpha_q(2,1) - 2*q(2,1)))*(alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1)))/((Y)^2*(X)^2) + (4*q(2,1)*(8*(alpha_q(1,1) - q(1,1))*(2*Y + q_e_squared) - 32*alpha_q(1,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1)))/((Y)^2*(X)^3) + (8*alpha_q(1,1)*q(2,1)*(alpha_q(1,1) + q(1,1)))/((Y)*(X)^2)),
vector(1,1)*(8/(X)^2 - (32*q(3,1)^2)/(X)^3 - ((32*alpha_q(1,1)*(alpha_q(3,1) - 2*q(3,1)) - 8*(alpha_q(1,1) - q(1,1))*(2*alpha_q(3,1) - 2*q(3,1)))*(alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1)))/((Y)^2*(X)^2) + (4*q(3,1)*(8*(alpha_q(1,1) - q(1,1))*(2*Y + q_e_squared) - 32*alpha_q(1,1)*(-alpha_q.'*q_e + X))*(alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1)))/((Y)^2*(X)^3) + (8*alpha_q(2,1)*q(3,1)*(alpha_q(1,1) + q(1,1)))/((Y)*(X)^2)) - vector(2,1)*((32*q(3,1))/(X)^3 - (8*q(3,1))/(X)^2 - ((32*alpha_q(1,1)*(alpha_q(3,1) - 2*q(3,1)) - 8*(alpha_q(1,1) - q(1,1))*(2*alpha_q(3,1) - 2*q(3,1)))*(alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1)))/((Y)^2*(X)^2) + (4*q(3,1)*(8*(alpha_q(1,1) - q(1,1))*(2*Y + q_e_squared) - 32*alpha_q(1,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1)))/((Y)^2*(X)^3) + (8*alpha_q(1,1)*q(3,1)*(alpha_q(1,1) + q(1,1)))/((Y)*(X)^2)) + vector(3,1)*((32*q(3,1)*(alpha_q(1,1) + q(1,1)))/(X)^3 - ((32*alpha_q(1,1)*(alpha_q(3,1) - 2*q(3,1)) - 8*(alpha_q(1,1) - q(1,1))*(2*alpha_q(3,1) - 2*q(3,1)))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^2*(X)^2) + (4*q(3,1)*(8*(alpha_q(1,1) - q(1,1))*(2*Y + q_e_squared) - 32*alpha_q(1,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^2*(X)^3));

vector(1,1)*((32*q(1,1))/(X)^3 - (8*q(1,1))/(X)^2 - ((32*alpha_q(2,1)*(alpha_q(1,1) - 2*q(1,1)) - 8*(alpha_q(2,1) - q(2,1))*(2*alpha_q(1,1) - 2*q(1,1)))*(alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1)))/((Y)^2*(X)^2) + (4*q(1,1)*(8*(alpha_q(2,1) - q(2,1))*(2*Y + q_e_squared) - 32*alpha_q(2,1)*(-alpha_q.'*q_e + X))*(alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1)))/((Y)^2*(X)^3) + (8*alpha_q(2,1)*q(1,1)*(alpha_q(2,1) + q(2,1)))/((Y)*(X)^2)) - vector(2,1)*((32*q(1,1)*q(3,1))/(X)^3 - ((32*alpha_q(2,1)*(alpha_q(1,1) - 2*q(1,1)) - 8*(alpha_q(2,1) - q(2,1))*(2*alpha_q(1,1) - 2*q(1,1)))*(alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1)))/((Y)^2*(X)^2) + (4*q(1,1)*(8*(alpha_q(2,1) - q(2,1))*(2*Y + q_e_squared) - 32*alpha_q(2,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1)))/((Y)^2*(X)^3) + (8*alpha_q(1,1)*q(1,1)*(alpha_q(2,1) +

```
q(2,1)))/((Y)*(X)^2)) + vector(3,1)*((32*q(1,1)*(alpha_q(2,1) + q(2,1)))/(X)^3 -
((32*alpha_q(2,1)*(alpha_q(1,1) - 2*q(1,1)) - 8*(alpha_q(2,1) - q(2,1))*(2*alpha_q(1,1) -
2*q(1,1)))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^2*(X)^2) + (4*q(1,1)*(8*(alpha_q(2,1) - q(2,1))*(2*Y
+ q_e_squared) - 32*alpha_q(2,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1)^2 +
alpha_q(2,1)^2))/((Y)^2*(X)^3)), vector(1,1)*((32*q(2,1))/(X)^3 - (8*q(2,1))/(X)^2 -
(4*alpha_q(2,1))/((Y)*(X)) + ((alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1))*(16*alpha_q(1,1)^2 -
32*alpha_q(2,1)*(alpha_q(2,1) - 2*q(2,1)) + 16*alpha_q(2,1)^2 + 16*alpha_q(3,1)^2 + 8*(alpha_q(2,1)
- q(2,1))*(2*alpha_q(2,1) - 2*q(2,1)) + 8*q_e_squared + 16))/((Y)^2*(X)^2) +
(4*q(2,1)*(8*(alpha_q(2,1) - q(2,1))*(2*Y + q_e_squared) - 32*alpha_q(2,1)*(-alpha_q.'*q_e +
X))*(alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1)))/((Y)^2*(X)^3) + (8*alpha_q(2,1)*q(2,1)*(alpha_q(2,1) +
q(2,1)))/((Y)*(X)^2)) + vector(3,1)*((32*q(2,1)*(alpha_q(2,1) + q(2,1)))/(X)^3 - 8/(X)^2 +
((alpha_q(1,1)^2 + alpha_q(2,1)^2)*(16*alpha_q(1,1)^2 - 32*alpha_q(2,1)*(alpha_q(2,1) - 2*q(2,1)) +
16*alpha_q(2,1)^2 + 16*alpha_q(3,1)^2 + 8*(alpha_q(2,1) - q(2,1))*(2*alpha_q(2,1) - 2*q(2,1)) +
8*q_e_squared + 16))/((Y)^2*(X)^2) + (4*q(2,1)*(8*(alpha_q(2,1) - q(2,1))*(2*Y + q_e_squared) -
32*alpha_q(2,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^2*(X)^3)) -
vector(2,1)*((32*q(2,1)*q(3,1))/(X)^3 - (4*alpha_q(1,1))/((Y)*(X)) + ((alpha_q(1,1) +
alpha_q(2,1)*alpha_q(3,1))*(16*alpha_q(1,1)^2 - 32*alpha_q(2,1)*(alpha_q(2,1) - 2*q(2,1)) +
16*alpha_q(2,1)^2 + 16*alpha_q(3,1)^2 + 8*(alpha_q(2,1) - q(2,1))*(2*alpha_q(2,1) - 2*q(2,1)) +
8*q_e_squared + 16))/((Y)^2*(X)^2) + (4*q(2,1)*(8*(alpha_q(2,1) - q(2,1))*(2*Y + q_e_squared) -
32*alpha_q(2,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1)))/((Y)^2*(X)^3) +
(8*alpha_q(1,1)*q(2,1)*(alpha_q(2,1) + q(2,1)))/((Y)*(X)^2)),
vector(1,1)*((32*q(3,1))/(X)^3 - (8*q(3,1))/(X)^2 - ((32*alpha_q(2,1)*(alpha_q(3,1) - 2*q(3,1)) -
8*(alpha_q(2,1) - q(2,1))*(2*alpha_q(3,1) - 2*q(3,1)))*(alpha_q(2,1) -
alpha_q(1,1)*alpha_q(3,1)))/((Y)^2*(X)^2) + (4*q(3,1)*(8*(alpha_q(2,1) - q(2,1))*(2*Y + q_e_squared) -
32*alpha_q(2,1)*(-alpha_q.'*q_e + X))*(alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1)))/((Y)^2*(X)^3) +
(8*alpha_q(2,1)*q(3,1)*(alpha_q(2,1) + q(2,1)))/((Y)*(X)^2)) - vector(2,1)*((32*q(3,1)^2)/(X)^3 - 8/(X)^2
- ((32*alpha_q(2,1)*(alpha_q(3,1) - 2*q(3,1)) - 8*(alpha_q(2,1) - q(2,1))*(2*alpha_q(3,1) -
2*q(3,1)))*(alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1)))/((Y)^2*(X)^2) + (4*q(3,1)*(8*(alpha_q(2,1) -
q(2,1))*(2*Y + q_e_squared) - 32*alpha_q(2,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1) +
alpha_q(2,1)*alpha_q(3,1)))/((Y)^2*(X)^3) + (8*alpha_q(1,1)*q(3,1)*(alpha_q(2,1) +
q(2,1)))/((Y)*(X)^2)) + vector(3,1)*((32*q(3,1)*(alpha_q(2,1) + q(2,1)))/(X)^3 -
((32*alpha_q(2,1)*(alpha_q(3,1) - 2*q(3,1)) - 8*(alpha_q(2,1) - q(2,1))*(2*alpha_q(3,1) -
2*q(3,1)))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^2*(X)^2) + (4*q(3,1)*(8*(alpha_q(2,1) - q(2,1))*(2*Y
+ q_e_squared) - 32*alpha_q(2,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1)^2 +
alpha_q(2,1)^2))/((Y)^2*(X)^3));

(4*q(1,1)*vector(3,1)*(8*(alpha_q(3,1) - q(3,1))*(2*Y + q_e_squared) - 32*alpha_q(3,1)*(-
alpha_q.'*q_e + X))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^2*(X)^3) -
vector(2,1)*((32*alpha_q(2,1)*q(1,1))/(X)^3 - ((32*alpha_q(3,1)*(alpha_q(1,1) - 2*q(1,1)) -
8*(alpha_q(3,1) - q(3,1))*(2*alpha_q(1,1) - 2*q(1,1)))*(alpha_q(1,1) +
alpha_q(2,1)*alpha_q(3,1)))/((Y)^2*(X)^2) + (4*q(1,1)*(8*(alpha_q(3,1) - q(3,1))*(2*Y + q_e_squared) -
32*alpha_q(3,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1)))/((Y)^2*(X)^3) +
(8*alpha_q(1,1)*q(1,1)*(alpha_q(3,1) + q(3,1)))/((Y)*(X)^2)) -
(vector(3,1)*(32*alpha_q(3,1)*(alpha_q(1,1) - 2*q(1,1)) - 8*(alpha_q(3,1) - q(3,1))*(2*alpha_q(1,1) -
2*q(1,1)))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^2*(X)^2) -
vector(1,1)*((32*alpha_q(1,1)*q(1,1))/(X)^3 + ((32*alpha_q(3,1)*(alpha_q(1,1) - 2*q(1,1)) -
8*(alpha_q(3,1) - q(3,1))*(2*alpha_q(1,1) - 2*q(1,1)))*(alpha_q(2,1) -
alpha_q(1,1)*alpha_q(3,1)))/((Y)^2*(X)^2) - (4*q(1,1)*(8*(alpha_q(3,1) - q(3,1))*(2*Y + q_e_squared) -
32*alpha_q(3,1)*(-alpha_q.'*q_e + X))*(alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1)))/((Y)^2*(X)^3) -
(8*alpha_q(2,1)*q(1,1)*(alpha_q(3,1) + q(3,1)))/((Y)*(X)^2)),
(4*q(2,1)*vector(3,1)*(8*(alpha_q(3,1) - q(3,1))*(2*Y + q_e_squared) - 32*alpha_q(3,1)*(-
alpha_q.'*q_e + X))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^2*(X)^3) -
vector(2,1)*((32*alpha_q(2,1)*q(2,1))/(X)^3 - ((32*alpha_q(3,1)*(alpha_q(2,1) - 2*q(2,1)) -
8*(alpha_q(3,1) - q(3,1))*(2*alpha_q(2,1) - 2*q(2,1)))*(alpha_q(1,1) +
alpha_q(2,1)*alpha_q(3,1)))/((Y)^2*(X)^2) + (4*q(2,1)*(8*(alpha_q(3,1) - q(3,1))*(2*Y + q_e_squared) -
32*alpha_q(3,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1)))/((Y)^2*(X)^3) +
(8*alpha_q(1,1)*q(2,1)*(alpha_q(3,1) + q(3,1)))/((Y)*(X)^2)) -
(vector(3,1)*(32*alpha_q(3,1)*(alpha_q(2,1) - 2*q(2,1)) - 8*(alpha_q(3,1) - q(3,1))*(2*alpha_q(2,1) -
2*q(2,1)))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^2*(X)^2) -
vector(1,1)*((32*alpha_q(1,1)*q(2,1))/(X)^3 + ((32*alpha_q(3,1)*(alpha_q(2,1) - 2*q(2,1)) -
```

```
8*(alpha_q(3,1) - q(3,1))*(2*alpha_q(2,1) - 2*q(2,1)))*(alpha_q(2,1) -
alpha_q(1,1)*alpha_q(3,1)))/((Y)^2*(X)^2) - (4*q(2,1)*(8*(alpha_q(3,1) - q(3,1))*(2*Y + q_e_squared) -
32*alpha_q(3,1)*(-alpha_q.'*q_e + X))*(alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1)))/((Y)^2*(X)^3) -
(8*alpha_q(2,1)*q(2,1)*(alpha_q(3,1) + q(3,1)))/((Y)*(X)^2)), vector(1,1)*(((alpha_q(2,1) -
alpha_q(1,1)*alpha_q(3,1))*(16*alpha_q(1,1)^2 - 32*alpha_q(3,1)*(alpha_q(3,1) - 2*q(3,1)) +
16*alpha_q(2,1)^2 + 16*alpha_q(3,1)^2 + 8*(alpha_q(3,1) - q(3,1))*(2*alpha_q(3,1) - 2*q(3,1)) +
8*q_e_squared + 16))/((Y)^2*(X)^2) - (4*alpha_q(2,1))/((Y)*(X)) - (32*alpha_q(1,1)*q(3,1))/(X)^3 +
(4*q(3,1)*(8*(alpha_q(3,1) - q(3,1))*(2*Y + q_e_squared) - 32*alpha_q(3,1)*(-alpha_q.'*q_e +
X))*(alpha_q(2,1) - alpha_q(1,1)*alpha_q(3,1)))/((Y)^2*(X)^3) + (8*alpha_q(2,1)*q(3,1)*(alpha_q(3,1) +
q(3,1)))/((Y)*(X)^2)) - vector(2,1)*((32*alpha_q(2,1)*q(3,1))/(X)^3 - (4*alpha_q(1,1))/((Y)*(X)) +
((alpha_q(1,1) + alpha_q(2,1)*alpha_q(3,1))*(16*alpha_q(1,1)^2 - 32*alpha_q(3,1)*(alpha_q(3,1) -
2*q(3,1)) + 16*alpha_q(2,1)^2 + 16*alpha_q(3,1)^2 + 8*(alpha_q(3,1) - q(3,1))*(2*alpha_q(3,1) -
2*q(3,1)) + 8*q_e_squared + 16))/((Y)^2*(X)^2) + (4*q(3,1)*(8*(alpha_q(3,1) - q(3,1))*(2*Y +
q_e_squared) - 32*alpha_q(3,1)*(-alpha_q.'*q_e + X))*(alpha_q(1,1) +
alpha_q(2,1)*alpha_q(3,1)))/((Y)^2*(X)^3) + (8*alpha_q(1,1)*q(3,1)*(alpha_q(3,1) +
q(3,1)))/((Y)*(X)^2)) + (vector(3,1)*(alpha_q(1,1)^2 + alpha_q(2,1)^2)*(16*alpha_q(1,1)^2 -
32*alpha_q(3,1)*(alpha_q(3,1) - 2*q(3,1)) + 16*alpha_q(2,1)^2 + 16*alpha_q(3,1)^2 + 8*(alpha_q(3,1)
- q(3,1))*(2*alpha_q(3,1) - 2*q(3,1)) + 8*q_e_squared + 16))/((Y)^2*(X)^2) +
(4*q(3,1)*vector(3,1)*(8*(alpha_q(3,1) - q(3,1))*(2*Y + q_e_squared) - 32*alpha_q(3,1)*(-
alpha_q.'*q_e + X))*(alpha_q(1,1)^2 + alpha_q(2,1)^2))/((Y)^2*(X)^3)];

end
```

```
function [ tor, alpha_w, alpha_w_dot, tor_aero_hat_deterministic,  tor_aero_hat_deterministic_dot,
sigma_hat_Stochastic_1, sigma_hat_Stochastic_dot_1,sigma_hat_Stochastic_2,
sigma_hat_Stochastic_dot_2, h_4, h_5, q_e, w_e, Partial_alpha_w_wrt_partial_v_1_hat] =
Attitude_Backstepping_Controller_Complete_System_Stochastic(m, I_H, I_A, g, L2_to_L3,
epsilon_1,epsilon_2, epsilon_3, epsilon_4, epsilon_5, epsilon_6,epsilon_7,  D_1,
D_2,min_eigen_value_D_1, gamma_1, K_1, K_2, K_3, K_4, c, n_1e, v_1_hat, v_1e, n_1d, n_1d_dot,
n_1d_double_dot, n_1d_triple_dot, n_1d_quad_dot, angles, alpha_1, alpha_2, alpha_psi,
alpha_psi_dot, alpha_psi_double_dot, R, B_Frame_angulare_velocity, Delta_time,
tor_aero_hat_deterministic, sigma_hat_Stochastic_1,
sigma_hat_Stochastic_dot_1,sigma_hat_Stochastic_2, sigma_hat_Stochastic_dot_2, Loop_Count,
Sum_M1_to_M6, Partial_M_1_to_6_wrt_partial_n_1e, Partial_M_1_to_6_wrt_partial_v_1_hat,
Partial_M_1_to_6_wrt_partial_n_1d_dot, Partial_M_1_to_6_wrt_partial_n_1d_double_dot,  Omega,
Partial_Omega_wrt_partial_n_1e, Partial_Omega_wrt_partial_v_1_hat ,
Partial_Omega_wrt_partial_n_1d_dot , Partial_Omega_wrt_partial_n_1d_double_dot, alpha_w_old)
constant = 1;
min_eigen_value_K_3 = (5/5.1)*min(eig(K_3));
gamma_3 = 0.1;
sigma_n_1e= [ n_1e(1,1)/(1+n_1e(1,1)^2)^0.5;
                    n_1e(2,1)/(1+n_1e(2,1)^2)^0.5;
                    n_1e(3,1)/(1+n_1e(3,1)^2)^0.5];
sigma_dash_n_1e = [ 1/(1+n_1e(1,1)^2)^1.5, 0, 0;
                            0, 1/(1+n_1e(2,1)^2)^1.5, 0;
                            0, 0, 1/(1+n_1e(3,1)^2)^1.5];
delta_1_v_1_hat = 1 + 0.5 * ((v_1_hat.'))*v_1_hat;
n_1e_dot = v_1_hat - n_1d_dot;
G_1 = eye(3)-(K_1*sigma_n_1e*(v_1_hat.')/delta_1_v_1_hat^2);
n_2 = angles;
(n_2-alpha_2)*180/pi;

q= Euler_to_Modified_rodrigues_paramater(n_2);
alpha_2 * 180/pi;
alpha_q = Euler_to_Modified_rodrigues_paramater(alpha_2);
q_e = q - alpha_q
R_2_alpha_q = R_2_MRP_calc( alpha_q);
T_alpha_2_inverse                       = Angular_velocity_cordinant_transform(alpha_2);
h_2 =min_eigen_value_D_1*0.5* gamma_1*(sigma_n_1e.').';
h_3 = 2*gamma_1*(v_1e.'*K_1*sigma_dash_n_1e/delta_1_v_1_hat).'/(1+v_1e.'*v_1e)^0.5;

alpha_phi = alpha_2(1,1);%alpha_PHI;
alpha_theta = alpha_2(2,1);%alpha_THETA;
alpha_psi = alpha_2(3,1);%alpha_PSI;

Omega_3 = Omega(3,1);
cos_alpha_psi =cos(alpha_psi);
sin_alpha_psi = sin(alpha_psi);

R_alpha_q = R_MRP(alpha_q);
R_q = R_MRP(q);
R_2_q = R_2_MRP_calc(q);
R_2_q_inverse  = (16/(1+q.'*q)^2)*R_2_q.';
R_2_alpha_q=R_2_MRP_calc(alpha_q);
A_1d=[ cos_alpha_psi*1/(cos(alpha_phi)*(Omega.'*Omega)^0.5) ,
sin_alpha_psi*1/(cos(alpha_phi)*(Omega.'*Omega)^0.5) , 0;
          -sin_alpha_psi*(cos(alpha_theta))^2/Omega_3                    ,
cos_alpha_psi*(cos(alpha_theta))^2/Omega_3              , 0;
          0                                                                                      , 0
, 0];
A_2d=[ sin_alpha_psi/(cos(alpha_phi)*(Omega.'*Omega)^0.5), (-
cos_alpha_psi)/(cos(alpha_phi)*(Omega.'*Omega)^0.5), 0
;
```

```
        (cos(alpha_theta))^2*cos_alpha_psi/Omega_3            ,
(cos(alpha_theta))^2*sin_alpha_psi/Omega_3                  , -
(cos(alpha_theta))^2*tan(alpha_theta)/Omega_3;
        0                                                          , 0
, 0                                                 ] - [tan(alpha_phi)/(Omega.'*Omega); 0;
0]*Omega.';
h_4 = -(q_e.'*R_2_alpha_q*T_alpha_2_inverse*A_2d*Partial_Omega_wrt_partial_n_1e).';

h_4 = h_4 / L2_to_L3;
Omega_d = Partial_Omega_wrt_partial_n_1e*(v_1_hat - n_1d_dot) +
Partial_Omega_wrt_partial_v_1_hat  *(-D_1*v_1_hat - g*[0;0;1]+(((Omega.'*Omega)^0.5)/m) *
R*[0;0;1] + h_2 + h_3+h_4) + Partial_Omega_wrt_partial_n_1d_dot  * n_1d_double_dot +
Partial_Omega_wrt_partial_n_1d_double_dot * n_1d_triple_dot;

[Partial_L_wrt_q_times_vector , Partial_L_wrt_alpha_q_times_vector,L] =
calc_L_MRP_norm_squared (q, alpha_q,G_1.'*v_1e/(1+ v_1e.'*v_1e)^0.5);
%Omega_d
alpha_w = R_2_q_inverse * ( -K_3 *q_e+ R_2_alpha_q*T_alpha_2_inverse*(A_2d*(Omega_d +
Partial_Omega_wrt_partial_n_1e*2*gamma_1* G_1.'*v_1e/(1+v_1e.'*v_1e)^0.5 )+ (A_1d*Omega +
[0;0;1])*alpha_psi_dot)-2*gamma_1*L2_to_L3*(1/m)*(Omega.'*Omega)^0.5 *
L*G_1.'*v_1e/(1+v_1e.'*v_1e)^0.5);

[Partial_h_4_wrt_n_1e, Partial_h4_wrt_partial_n_1d_dot, Partial_h4_wrt_partial_n_1d_double_dot,
Partial_h4_wrt_partial_v_1_hat, Partial_h_4_wrt_q,  Partial_h_4_wrt_alpha_psi ] =
h_4_Partial_derivatives(m, epsilon_2, epsilon_3, D_1, min_eigen_value_D_1, gamma_1, K_1, K_2,
c, n_1e, v_1_hat, v_1e, n_1d_dot, n_1d_double_dot, q_e, Sum_M1_to_M6,
Partial_M_1_to_6_wrt_partial_n_1e, Partial_M_1_to_6_wrt_partial_v_1_hat,
Partial_M_1_to_6_wrt_partial_n_1d_dot, Partial_M_1_to_6_wrt_partial_n_1d_double_dot, alpha_1,
Partial_Omega_wrt_partial_n_1e, Partial_Omega_wrt_partial_v_1_hat ,
Partial_Omega_wrt_partial_n_1d_dot, Partial_Omega_wrt_partial_n_1d_double_dot, A_1d, A_2d,
T_alpha_2_inverse,R_2_alpha_q ,alpha_psi, alpha_phi, alpha_theta, alpha_q, Omega,L2_to_L3 );
%[Omega_2d_dot_wrt_n_1e, Omega_2d_dot_wrt_v_1_hat, Omega_2d_dot_wrt_n_1d_dot,
Omega_2d_dot_wrt_n_1d_double_dot, Omega_2d_dot_wrt_n_1d_triple_dot, Omega_2d_dot_wrt_q,
Omega_2d_dot_wrt_alpha_psi ] = Omega_d_dot(m, epsilon_2, epsilon_3, D_1,
min_eigen_value_D_1, gamma_1, K_1, K_2, c, n_1e, v_1_hat, v_1e, n_1d_dot, n_1d_double_dot,
n_1d_triple_dot,q , Omega, Sum_M1_to_M6, Partial_M_1_to_6_wrt_partial_n_1e,
Partial_M_1_to_6_wrt_partial_v_1_hat, Partial_M_1_to_6_wrt_partial_n_1d_dot,
Partial_M_1_to_6_wrt_partial_n_1d_double_dot, h_4, R, alpha_1, Partial_Omega_wrt_partial_n_1e,
Partial_Omega_wrt_partial_v_1_hat,Partial_Omega_wrt_partial_n_1d_dot,
Partial_Omega_wrt_partial_n_1d_double_dot, Partial_h_4_wrt_n_1e,
Partial_h4_wrt_partial_n_1d_dot, Partial_h4_wrt_partial_n_1d_double_dot,
Partial_h4_wrt_partial_v_1_hat, Partial_h_4_wrt_q,  Partial_h_4_wrt_alpha_psi);
[Partial_alpha_w_wrt_partial_n_1e, Partial_alpha_w_wrt_partial_v_1_hat,
Partial_alpha_w_wrt_partial_n_1d_dot, Partial_alpha_w_wrt_partial_n_1d_double_dot,
Partial_alpha_w_wrt_partial_n_1d_triple_dot, Partial_alpha_w_wrt_partial_q,
Partial_alpha_w_wrt_partial_alpha_psi, Partial_alpha_w_wrt_partial_alpha_psi_dot] =
Partial_derivatives_of_alpha_w(m, epsilon_2, epsilon_3, epsilon_4, epsilon_5, D_1,
min_eigen_value_D_1, gamma_1, K_1, K_2, K_3,  c, n_1e, v_1_hat, v_1e, n_1d_dot,
n_1d_double_dot, n_1d_triple_dot, n_2, alpha_1, alpha_2, q, alpha_q, Omega, Sum_M1_to_M6,
Partial_M_1_to_6_wrt_partial_n_1e, Partial_M_1_to_6_wrt_partial_v_1_hat,
Partial_M_1_to_6_wrt_partial_n_1d_dot, Partial_M_1_to_6_wrt_partial_n_1d_double_dot, h_4, R,
Partial_Omega_wrt_partial_n_1e,
Partial_Omega_wrt_partial_v_1_hat,Partial_Omega_wrt_partial_n_1d_dot,
Partial_Omega_wrt_partial_n_1d_double_dot, Omega_d, Partial_h_4_wrt_n_1e,
Partial_h4_wrt_partial_n_1d_dot, Partial_h4_wrt_partial_n_1d_double_dot,
Partial_h4_wrt_partial_v_1_hat, Partial_h_4_wrt_q,  Partial_h_4_wrt_alpha_psi, A_1d, A_2d,
alpha_psi_dot,L2_to_L3);
%Omega_2d_dot_wrt_v_1_hat
w = B_Frame_angulare_velocity;
w_e = w - alpha_w;
```

```
%alpha_w
h_5 = -w_e.'*w_e*(w_e.'*Partial_alpha_w_wrt_partial_n_1e).';
h_5 =  0.1* h_5/(constant+(w_e.'*w_e)^2)^0.5;




v_1_hat_dot =-D_1*v_1_hat - g *[0;0;1] + (Omega.'*Omega)^0.5/m*R*[0;0;1]  + h_2 + h_3 + h_4 +
h_5;

alpha_w_dot  = Partial_alpha_w_wrt_partial_n_1e* n_1e_dot +
Partial_alpha_w_wrt_partial_v_1_hat* v_1_hat_dot +
Partial_alpha_w_wrt_partial_n_1d_dot*n_1d_double_dot +
Partial_alpha_w_wrt_partial_n_1d_double_dot*n_1d_triple_dot +
Partial_alpha_w_wrt_partial_n_1d_triple_dot*n_1d_quad_dot + Partial_alpha_w_wrt_partial_q *
R_2_q * w + Partial_alpha_w_wrt_partial_alpha_psi * alpha_psi_dot +
Partial_alpha_w_wrt_partial_alpha_psi_dot * alpha_psi_double_dot;
w_e = w - alpha_w;


xi_aero_deterministic=1;
ROE =100*0.15*[1 0 0; 0 1 0; 0 0 1];%K_d;
ROE = 1*eye(3);
tor_aero_deterministic_MAX=1*0.2;


tor_aero_hat_deterministic_dot = [0;0;0];

INV_I_H_plus_I_A = inv(I_H+I_A);
tor_aero_deterministic_MAX=[0.1; 0.2; 0.1];
xi_aero_deterministic=0.005*[0.1; 0.2; 0.1];
tor_aero_deterministic_p          = Projection_algorithm(gamma_3*((norm(w_e))^2*([1 0 0
]*INV_I_H_plus_I_A)*w_e)/(1+(w_e.'*w_e)^2)^0.5,[1 0 0]*tor_aero_hat_deterministic,[1 0
0]*tor_aero_hat_deterministic_dot, [1 0 0]*xi_aero_deterministic, [1 0 0]*tor_aero_deterministic_MAX,
ROE(1,1), Delta_time);
tor_aero_deterministic_q          = Projection_algorithm(gamma_3*((norm(w_e))^2*([0 1 0
]*INV_I_H_plus_I_A)*w_e)/(1+(w_e.'*w_e)^2)^0.5,[0 1 0]*tor_aero_hat_deterministic,[0 1
0]*tor_aero_hat_deterministic_dot, [0 1 0]*xi_aero_deterministic, [0 1
0]*tor_aero_deterministic_MAX,ROE(2,2), Delta_time);
tor_aero_deterministic_r          = Projection_algorithm(gamma_3*((norm(w_e))^2*([0 0 1
]*INV_I_H_plus_I_A)*w_e)/(1+(w_e.'*w_e)^2)^0.5,[0 0 1]*tor_aero_hat_deterministic,[0 0
1]*tor_aero_hat_deterministic_dot, [0 0 1]*xi_aero_deterministic, [0 0 1]*tor_aero_deterministic_MAX,
ROE(3,3), Delta_time);
tor_aero_deterministic            = [ tor_aero_deterministic_p;
                                      tor_aero_deterministic_q;
                                      tor_aero_deterministic_r ];

tor_aero_hat_deterministic        =[ tor_aero_deterministic(1,1);
                                      tor_aero_deterministic(3,1);
                                      tor_aero_deterministic(5,1)];
tor_aero_hat_deterministic_dot = [ tor_aero_deterministic(2,1);
                                      tor_aero_deterministic(4,1);
                                      tor_aero_deterministic(6,1)];


if Loop_Count   == 1
   tor_aero_hat_deterministic  =[0;0;0];
end
```

```
gamma_projection_algorthim =800;
tor_aero_Stochastic_MAX= 1;
xi_aero_Stochastic=0.1*tor_aero_Stochastic_MAX;


%sigma_hat_Stochastic
%sigma_Stochastic= Projection_algorithm(
0.01*(9*epsilon_7/4)*(norm(inv(I_H)))^4*((norm(w_e))^4)/(1+(norm(w_e))^4)^1.5,sigma_hat_Stochasti
c,sigma_hat_Stochastic_dot  , xi_aero_Stochastic, tor_aero_Stochastic_MAX,
gamma_projection_algorthim, dt);
I_XX = I_H(1,1);
I_YY = I_H(2,2);
I_ZZ = I_H(3,3);

Coriolis = [ 0, I_ZZ * B_Frame_angulare_velocity(3,1), -I_YY * B_Frame_angulare_velocity(2,1);
                    -I_ZZ * B_Frame_angulare_velocity(3,1), 0, I_XX * B_Frame_angulare_velocity(1,1);
                    I_YY * B_Frame_angulare_velocity(2,1), -I_XX * B_Frame_angulare_velocity(1,1), 0
];
    I_XX_A = I_A(1,1);
    I_YY_A = I_A(2,2);
    I_ZZ_A = I_A(3,3);

      Coriolis_A = [ 0, I_ZZ_A * B_Frame_angulare_velocity(3,1), -I_YY_A *
B_Frame_angulare_velocity(2,1);
                    -I_ZZ_A * B_Frame_angulare_velocity(3,1), 0, I_XX_A *
B_Frame_angulare_velocity(1,1);
                    I_YY_A * B_Frame_angulare_velocity(2,1), -I_XX_A *
B_Frame_angulare_velocity(1,1), 0 ];
      skew_w = [ 0, -B_Frame_angulare_velocity(3,1),  B_Frame_angulare_velocity(2,1);
                     B_Frame_angulare_velocity(3,1), 0, -B_Frame_angulare_velocity(1,1);
                     -B_Frame_angulare_velocity(2,1), B_Frame_angulare_velocity(1,1), 0];
                 Multiplicitiveand_aditive_noise = inv(I_H+I_A)*(D_2*eye(3)+0*0.05*eye(3)+ Coriolis_A
+skew_w *I_A);

sigma_Stochastic_1= Projection_algorithm(gamma_1^2*
(9*epsilon_7)*(norm(Multiplicitiveand_aditive_noise
))^4*(norm(w_e))^4/(constant+(norm(w_e))^4)^1.0,sigma_hat_Stochastic_1,sigma_hat_Stochastic_do
t_1  , xi_aero_Stochastic, tor_aero_Stochastic_MAX, gamma_projection_algorthim, Delta_time);

sigma_hat_Stochastic_1        =1*sigma_Stochastic_1(1,1)
sigma_hat_Stochastic_dot_1  =  sigma_Stochastic_1(2,1);

if sigma_hat_Stochastic_1 <0
   sigma_hat_Stochastic_1 =0;
%    pause(100)
end

mew_projection_algorthim_2 =0.00001;%0.00812/norm(inv(I_H))^4;
tor_aero_Stochastic_MAX=0.1^4;
xi_aero_Stochastic=0.1*tor_aero_Stochastic_MAX;

sigma_Stochastic_2= Projection_algorithm(gamma_3^2* (9/4*epsilon_6)*(norm(inv(I_H+I_A)
))^4*(norm(w_e))^4/(constant+(norm(w_e))^4)^0.75,sigma_hat_Stochastic_2,sigma_hat_Stochastic_d
ot_2  , xi_aero_Stochastic, tor_aero_Stochastic_MAX, mew_projection_algorthim_2, Delta_time);


sigma_hat_Stochastic_2        =1*sigma_Stochastic_2(1,1);
sigma_hat_Stochastic_dot_2  =  sigma_Stochastic_2(2,1);
%alpha_w_dot = (alpha_w - alpha_w_old)/Delta_time;
```

```
%sigma_Stochastic = 0;
if sigma_hat_Stochastic_2 <0
   sigma_hat_Stochastic_2 =0;
%    pause(100)
end


 w_e_dot             = -K_4 * w_e;

tor                          = (I_H+I_A)*( D_2 *alpha_w + w_e_dot -
gamma_3*((9*epsilon_7)*(norm(Multiplicitiveand_aditive_noise))^4*sigma_hat_Stochastic_1)*w_e/(co
nstant+1*(w_e .'*w_e )^2)^0.25  - gamma_3* (9/4*epsilon_6)*(norm(inv(I_H+I_A)
))^4*sigma_hat_Stochastic_2*w_e/(constant+(norm(w_e))^4)^0.25+ alpha_w_dot -
100*constant^2*((1/L2_to_L3)/(4*min_eigen_value_K_3)^2)*norm(R_2_alpha_q)^4/4
*w_e/(constant+(w_e.'*w_e)^2)^3.5-
10*(1/L2_to_L3)*(q_e.'*R_2_alpha_q*w_e)*w_e/(constant+(w_e.'*w_e)^2)^0.5 +
((2*gamma_1*v_1e.'*G_1 /(1+v_1e.'*v_1e)^0.5-
(1/L2_to_L3)*q_e.'*R_2_alpha_q*T_alpha_2_inverse)*Partial_alpha_w_wrt_partial_n_1e).') -
tor_aero_hat_deterministic + cross(B_Frame_angulare_velocity, ((I_H +
I_A)*B_Frame_angulare_velocity));
```

```
function [Partial_alpha_w_wrt_partial_n_1e, Partial_alpha_w_wrt_partial_v_1_hat,
Partial_alpha_w_wrt_partial_n_1d_dot, Partial_alpha_w_wrt_partial_n_1d_double_dot,
Partial_alpha_w_wrt_partial_n_1d_triple_dot, Partial_alpha_w_wrt_partial_q,
Partial_alpha_w_wrt_partial_alpha_psi, Partial_alpha_w_wrt_partial_alpha_psi_dot] =
Partial_derivatives_of_alpha_w(m, epsilon_2, epsilon_3, epsilon_4, epsilon_5, D_1,
min_eigen_value_D_1, gamma_1, K_1, K_2, K_3,  c, n_1e, v_1_hat, v_1e, n_1d_dot,
n_1d_double_dot, n_1d_triple_dot, n_2, alpha_1, alpha_2, q, alpha_q, Omega, Sum_M1_to_M6,
Partial_M_1_to_6_wrt_partial_n_1e, Partial_M_1_to_6_wrt_partial_v_1_hat,
Partial_M_1_to_6_wrt_partial_n_1d_dot, Partial_M_1_to_6_wrt_partial_n_1d_double_dot, h_4, R,
Partial_Omega_wrt_partial_n_1e,
Partial_Omega_wrt_partial_v_1_hat,Partial_Omega_wrt_partial_n_1d_dot,
Partial_Omega_wrt_partial_n_1d_double_dot, Omega_d, Partial_h_4_wrt_n_1e,
Partial_h4_wrt_partial_n_1d_dot, Partial_h4_wrt_partial_n_1d_double_dot,
Partial_h4_wrt_partial_v_1_hat, Partial_h_4_wrt_q,  Partial_h_4_wrt_alpha_psi, A_1d, A_2d,
alpha_psi_dot,L2_to_L3)

q_error = q-alpha_q;
T_inv                    = Angular_velocity_cordinant_transform(alpha_2);
T_inverse_alpha_2 = T_inv;
T_alpha_2_inverse = T_inverse_alpha_2;
R_2_alpha_q = R_2_MRP_calc(alpha_q);
sigma_n_1e= [ n_1e(1,1)/(1+n_1e(1,1)^2)^0.5;
                    n_1e(2,1)/(1+n_1e(2,1)^2)^0.5;
                    n_1e(3,1)/(1+n_1e(3,1)^2)^0.5];
sigma_dash_n_1e = [ 1/(1+n_1e(1,1)^2)^1.5, 0, 0;
                          0, 1/(1+n_1e(2,1)^2)^1.5, 0;
                          0, 0, 1/(1+n_1e(3,1)^2)^1.5];

delta_1_v_1_hat = 1 + 0.5 * ((v_1_hat.'))*v_1_hat;

G_1 = eye(3)-(K_1*sigma_n_1e*(v_1_hat.')/delta_1_v_1_hat^2);
%norm_v_1e = ((v_1e.')*v_1e)^0.5;
q_e = q - alpha_q;

 [Partial_L_wrt_q_times_vector , Partial_L_wrt_alpha_q_times_vector,L] =
calc_L_MRP_norm_squared (q, alpha_q,G_1.'*v_1e/(1+ v_1e.'*v_1e)^0.5);

vector_0 =( -K_3 *q_e+ R_2_alpha_q*T_alpha_2_inverse*(A_2d*(Omega_d +
Partial_Omega_wrt_partial_n_1e* G_1.'*v_1e )+ (A_1d*Omega + [0;0;1])*alpha_psi_dot)-
(Omega.'*Omega)^0.5 * L*G_1.'*v_1e/(1+v_1e.'*v_1e)^0.5);;
q_e_times_inverse_tanh_qe =[  0;  0;  0];
vector_0 =( -K_3 *q_e - q_e_times_inverse_tanh_qe +
R_2_alpha_q*T_alpha_2_inverse*(A_2d*(Omega_d + 2*gamma_1*Partial_Omega_wrt_partial_n_1e*
G_1.'*v_1e /(1+v_1e.'*v_1e)^0.5)+ (A_1d*Omega + [0;0;1])*alpha_psi_dot)-
(L2_to_L3*gamma_1*2)*(1/m)*(Omega.'*Omega)^0.5 * L*G_1.'*v_1e/(1+v_1e.'*v_1e)^0.5);


Derivative_q_e_times_nverse_tanh_qe = 0 *eye(3);

vector_3 = Omega_d+Partial_Omega_wrt_partial_n_1e * G_1.'*
v_1e/(1+v_1e.'*v_1e)^0.5*2*gamma_1;
vector_2 = (A_2d*(Omega_d+2*gamma_1*Partial_Omega_wrt_partial_n_1e * G_1.'*
v_1e/(1+v_1e.'*v_1e)^0.5 )+(A_1d*Omega+[0;0;1])*alpha_psi_dot);
vector_1 =T_inverse_alpha_2 * vector_2;

Partial_R_2_alpha_q_wrt_partial_alpha_q_times_vector_1 = R_2_derivitive_times_vector (alpha_q,
R_2_alpha_q, vector_1);
Partial_T_inverse_alpha_2_wrt_alpha_2_times_vector_2 =
Partial_T_alpha_2_inverse_wrt_n_2_times_vector (alpha_2, vector_2);
```

```
R_2_q = R_2_MRP_calc(q);
R_2_inverse_dot_times_x = R_2_inverse_derivative_times_vector(R_2_q,  q, vector_0);
R_2_q_inverse  = R_2_q.'*(16/(1+q.'*q)^2);
P_1 = R_2_inverse_dot_times_x ;


P_2 =R_2_q_inverse * (-(K_3 +
Derivative_q_e_times_nverse_tanh_qe+(L2_to_L3*gamma_1*2)*((Omega.'*Omega)^0.5/m)*Partial_L
_wrt_q_times_vector));
P_3 =(K_3) + Derivative_q_e_times_nverse_tanh_qe;
P_4 =-
(L2_to_L3*gamma_1*2)*(1/(m*(Omega.'*Omega)^0.5))*L*(G_1.'*v_1e/(1+v_1e.'*v_1e)^0.5)*Omega.';
P_5 =-(L2_to_L3*gamma_1*2)*((Omega.'*Omega)^0.5/m)*Partial_L_wrt_alpha_q_times_vector;


P_6 =Partial_R_2_alpha_q_wrt_partial_alpha_q_times_vector_1;
P_7 =R_2_alpha_q*Partial_T_inverse_alpha_2_wrt_alpha_2_times_vector_2;

[Partial_A_2d_wrt_alpha_phi_and_alpha_theta_times_vector_3,
Partial_A_2d_wrt_alpha_psi_times_vector_3, Partial_A_2d_wrt_Partial_Omega_times_vector_3] =
Partial_derivative_A_2d_times_vector_3(alpha_2, vector_3, Omega);
P_8   =
R_2_alpha_q*T_inverse_alpha_2*Partial_A_2d_wrt_alpha_phi_and_alpha_theta_times_vector_3;
P_9   = R_2_alpha_q*T_inverse_alpha_2*Partial_A_2d_wrt_Partial_Omega_times_vector_3;
P_10 = R_2_alpha_q*T_inverse_alpha_2*Partial_A_2d_wrt_alpha_psi_times_vector_3;

[Partial_A_1d_wrt_alpha_phi_and_alpha_theta_times_Omega,
Partial_A_1d_wrt_alpha_psi_times_Omega, Partial_A_1d_wrt_Partial_Omega_times_Omega] =
Partial_derivative_A_1d_times_Omega(alpha_2, A_1d, Omega);
%yes alpha_psi_dot should be here
P_12 = R_2_alpha_q*T_inverse_alpha_2 *
alpha_psi_dot*Partial_A_1d_wrt_Partial_Omega_times_Omega;
P_13 = R_2_alpha_q*T_inverse_alpha_2 *
Partial_A_1d_wrt_alpha_phi_and_alpha_theta_times_Omega;
P_14 = R_2_alpha_q*T_inverse_alpha_2 *
alpha_psi_dot*Partial_A_1d_wrt_alpha_psi_times_Omega;
P_15 = R_2_alpha_q*T_inverse_alpha_2*(A_1d*Omega+[0;0;1]);

Partial_alpha_w_wrt_alpha_q = R_2_q_inverse *(P_3 + P_5 + P_6) ;
Partial_alpha_w_wrt_alpha_phi_and_alpha_theta = R_2_q_inverse *(P_7 + P_8 + P_13) +
Partial_alpha_w_wrt_alpha_q*R_2_alpha_q * T_inverse_alpha_2;


P_200 =-(L2_to_L3*gamma_1*2)*(1/m)*R_2_q_inverse * (Omega.'*Omega)^0.5 * L*(G_1.'*(eye(3) -
v_1e*v_1e.'/(1+v_1e.'*v_1e))/(1+v_1e.'*v_1e)^0.5);
P_20  =-(L2_to_L3*gamma_1*2)*(1/m)*R_2_q_inverse * (Omega.'*Omega)^0.5 * L*(-
sigma_n_1e.'*K_1*v_1e/(1+v_1e.'*v_1e)^0.5)*(eye(3)-
2*v_1_hat*v_1_hat.'/delta_1_v_1_hat)/delta_1_v_1_hat^2 + P_200*G_1;
P_21  =-(L2_to_L3*gamma_1*2)*(1/m)*R_2_q_inverse * (Omega.'*Omega)^0.5 * L*((-
v_1_hat/delta_1_v_1_hat^2)*v_1e.'/(1+v_1e.'*v_1e)^0.5)*K_1*sigma_dash_n_1e +
P_200*K_1*sigma_dash_n_1e/delta_1_v_1_hat;
P_22  =-P_200;


Partial_alpha_w_wrt_Omega = Partial_alpha_w_wrt_alpha_phi_and_alpha_theta * A_2d +
R_2_q_inverse *(P_4 +P_9 +P_12);
 [Omega_2d_dot_wrt_n_1e, Omega_2d_dot_wrt_v_1_hat, Omega_2d_dot_wrt_n_1d_dot,
Omega_2d_dot_wrt_n_1d_double_dot, Omega_2d_dot_wrt_n_1d_triple_dot, Omega_2d_dot_wrt_q,
Omega_2d_dot_wrt_alpha_psi ] = Omega_d_dot(m, epsilon_2, epsilon_3, D_1,
min_eigen_value_D_1, gamma_1, K_1, K_2, c, n_1e, v_1_hat, v_1e, n_1d_dot, n_1d_double_dot,
```

```
n_1d_triple_dot,q , Omega, Sum_M1_to_M6, Partial_M_1_to_6_wrt_partial_n_1e,
Partial_M_1_to_6_wrt_partial_v_1_hat, Partial_M_1_to_6_wrt_partial_n_1d_dot,
Partial_M_1_to_6_wrt_partial_n_1d_double_dot, h_4, R, alpha_1, Partial_Omega_wrt_partial_n_1e,
Partial_Omega_wrt_partial_v_1_hat,Partial_Omega_wrt_partial_n_1d_dot,
Partial_Omega_wrt_partial_n_1d_double_dot, Partial_h_4_wrt_n_1e,
Partial_h4_wrt_partial_n_1d_dot, Partial_h4_wrt_partial_n_1d_double_dot,
Partial_h4_wrt_partial_v_1_hat, Partial_h_4_wrt_q,  Partial_h_4_wrt_alpha_psi);


Partial_alpha_w_wrt_partial_n_1e                 = Partial_alpha_w_wrt_Omega *
Partial_Omega_wrt_partial_n_1e + R_2_q_inverse *
R_2_alpha_q*T_inverse_alpha_2*A_2d*Omega_2d_dot_wrt_n_1e;
Partial_alpha_w_wrt_partial_v_1_hat              = Partial_alpha_w_wrt_Omega *
Partial_Omega_wrt_partial_v_1_hat + R_2_q_inverse *
R_2_alpha_q*T_inverse_alpha_2*A_2d*Omega_2d_dot_wrt_v_1_hat;
Partial_alpha_w_wrt_partial_n_1d_dot             = Partial_alpha_w_wrt_Omega *
Partial_Omega_wrt_partial_n_1d_dot + R_2_q_inverse *
R_2_alpha_q*T_inverse_alpha_2*A_2d*Omega_2d_dot_wrt_n_1d_dot;
Partial_alpha_w_wrt_partial_n_1d_double_dot = Partial_alpha_w_wrt_Omega *
Partial_Omega_wrt_partial_n_1d_double_dot + R_2_q_inverse *
R_2_alpha_q*T_inverse_alpha_2*A_2d*Omega_2d_dot_wrt_n_1d_double_dot;
Partial_alpha_w_wrt_partial_n_1d_triple_dot    = R_2_q_inverse *
R_2_alpha_q*T_inverse_alpha_2*A_2d*Omega_2d_dot_wrt_n_1d_triple_dot;
Partial_alpha_w_wrt_partial_q                    =P_1+ P_2 + R_2_q_inverse *
R_2_alpha_q*T_inverse_alpha_2*A_2d*Omega_2d_dot_wrt_q;

Partial_alpha_w_wrt_partial_v_1_hat  = Partial_alpha_w_wrt_partial_v_1_hat +P_20;
Partial_alpha_w_wrt_partial_n_1e = Partial_alpha_w_wrt_partial_n_1e + P_21;
Partial_alpha_w_wrt_partial_n_1d_dot = Partial_alpha_w_wrt_partial_n_1d_dot + P_22;
Partial_alpha_w_wrt_partial_alpha_psi            = Partial_alpha_w_wrt_alpha_q *
R_2_alpha_q*T_inverse_alpha_2* (A_1d*Omega+[0;0;1]) + R_2_q_inverse * (P_10 + P_14) +
Partial_alpha_w_wrt_alpha_phi_and_alpha_theta * (A_1d*Omega+[0;0;1]) + R_2_q_inverse *
R_2_alpha_q*T_inverse_alpha_2*A_2d*Omega_2d_dot_wrt_alpha_psi ;
Partial_alpha_w_wrt_partial_alpha_psi_dot      = P_15;
%pause(100);

end




function R_2_MRP = R_2_MRP_calc(q)
skew_q = [ 0, -q(3,1), q(2,1);
               q(3,1), 0, -q(1,1);
               -q(2,1), q(1,1), 0];
R_2_MRP = 0.5*(eye(3) + q*q.' + skew_q - ((1+(q.'*q))/2)*eye(3) );
end
```

```
function Modified_rodrigues_paramater = Euler_to_Modified_rodrigues_paramater(Euler_angle)

Phi    = Euler_angle(1,1);
Theta = Euler_angle(2,1);
Psi    = Euler_angle(3,1);
 q_0 = (1 + cos(Phi/2)*cos(Theta/2)*cos(Psi/2) + sin(Phi/2)*sin(Theta/2)*sin(Psi/2) );
q_1 = (sin(Phi/2)*cos(Theta/2)*cos(Psi/2) - cos(Phi/2)*sin(Theta/2)*sin(Psi/2))/q_0;
q_2 = (cos(Phi/2)*sin(Theta/2)*cos(Psi/2) + sin(Phi/2)*cos(Theta/2)*sin(Psi/2))/q_0;
q_3 = (cos(Phi/2)*cos(Theta/2)*sin(Psi/2) - sin(Phi/2)*sin(Theta/2)*cos(Psi/2))/q_0;

Modified_rodrigues_paramater = [ q_1;
                                 q_2;
                                 q_3];

end


function  Partial_R_2_alpha_q_wrt_partial_q_times_vector  = R_2_derivitive_times_vector (q, R_2,
vector)
Partial_R_2_alpha_q_wrt_partial_q_times_vector = 1/2*((q.'*vector*eye(3)+q*(vector).'-vector*q.')-
([0,0,0;0,0,1;0,-1,0]*(vector*[1,0,0])+[0,0,-1;0,0,0;1,0,0]*(vector*[0,1,0])+[0,1,0;-
1,0,0;0,0,0]*(vector*[0,0,1])));
End


function R_2_inverse_dot_times_vector = R_2_inverse_derivative_times_vector(R_2,  q, vector)

R_2_inverse_dot_times_vector=8/(1+q.'*q)^2 * ((q.'*vector*eye(3)+q*vector.'-
vector*q.')+([0,0,0;0,0,1;0,-1,0]*(vector*[1,0,0])+[0,0,-1;0,0,0;1,0,0]*(vector*[0,1,0])+[0,1,0;-
1,0,0;0,0,0]*(vector*[0,0,1])))-64/(1+q.'*q)^3  * R_2.'*vector*q.';

End


function  Partial_T_eta_2_d_inverse_wrt_eta_2_times_vector_x =
Partial_T_eta_2_d_inverse_wrt_eta_2_times_vector (alpha_2, vector)
   alpha_phi    = alpha_2(1,1);
   alpha_theta = alpha_2(2,1);
   Partial_T_eta_2_d_inverse_wrt_eta_2_times_vector_x = ([0,0,0;0,-
sin(alpha_phi),cos(alpha_theta)*cos(alpha_phi);0,-cos(alpha_phi),-
cos(alpha_theta)*sin(alpha_phi)]*vector*[1;0;0].'+[0,0,-cos(alpha_theta);0,0,-
sin(alpha_theta)*sin(alpha_phi);0,0,-sin(alpha_theta)*cos(alpha_phi)]*vector*[0;1;0].');
end
```

```
function  Projection = Projection_algorithm(omega_Macron,omega_hat, omega_hat_dot, xi,
omega_MAX, gamma, Delta_time)
   XI =( (norm(omega_hat))^2 - omega_MAX^2) / (xi^2 + 2*xi*omega_MAX);
   Derivative_XI = 2*omega_hat'/ (xi^2 + 2*xi*omega_MAX);
   if(XI < 0)
      omega_hat_dot = gamma * omega_Macron;
      Derivative_XI = 2*omega_hat'/ (xi^2 + 2*xi*omega_MAX);
   end
   if((XI >=  0) && ((Derivative_XI * omega_Macron)<=0))
      omega_hat_dot = gamma * omega_Macron;
   end
   if((XI >=  0) && ((Derivative_XI * omega_Macron)>0))
      omega_hat_dot = gamma * (1-XI)*omega_Macron;
   end
    omega_hat = omega_hat + Delta_time * omega_hat_dot ;
   Projection  = [omega_hat; omega_hat_dot ];
end
```

```
function  Zeta_dot =SIX_DOF_Dynamics(R, E_Frame_angulare_velocity, angles,g,m,I_H,
I_A,IXX,IYY,IZZ, Aero_Disturbance,      Collision_Disturbance, Control_Input, JTP,Zeta, D_1, D_2 )

   B_Frame_angulare_velocity =
E_FRAME_ANGULARE_Velocity_TO_B_FRAME_Angulare_velocity(E_Frame_angulare_velocity,
angles);
   p                                    = B_Frame_angulare_velocity(1,1);
   q                                    = B_Frame_angulare_velocity(2,1);
   r                                   = B_Frame_angulare_velocity(3,1);
   EH = [ R, zeros(3,3);
           zeros(3,3), eye(3)];
   MH = [ m * eye(3), zeros(3,3);
               zeros(3,3), I_H    + I_A      ];
   GH = [0;0; m*g;0;0;0];
   I_XX_A = I_A(1,1);
   I_YY_A = I_A(2,2);
   I_ZZ_A = I_A(3,3);
   Coriolis_A = [ 0, I_ZZ_A * r, -I_YY_A * q;
                    -I_ZZ_A * r, 0, I_XX_A * p;
                    I_YY_A * q, -I_XX_A * p, 0 ];
   coriolis = [  0, IZZ*r, -IYY*q;
                  -IZZ*r, 0, IXX*p;
                  IYY*q, -IXX*p, 0];
   CH = [ zeros(3,3), zeros(3,3);
               zeros(3,3), (coriolis +Coriolis_A)     ];
   OH = [ 0, 0, 0, 0 0, 0;
               0, 0, 0, 0 0, 0;
               0, 0, 0, 0 0, 0;
               -p, p, -p, p, -p, p;
               -q, q, -q, q, -q, q;
               0, 0, 0, 0 0, 0 ] ;
   Control_H =  [ 0;
                    0;
                    Control_Input ];
   Motor_Speed = zeros(6,1);
   Zeta_dot = (inv(MH))*(-CH*Zeta - GH +JTP*OH * Motor_Speed  +EH*Control_H
+Aero_Disturbance + Collision_Disturbance) - [D_1 0*eye(3); 0*eye(3) D_2] * Zeta;
end
```

```
function [Omega_2d_dot_wrt_n_1e, Omega_2d_dot_wrt_v_1_hat, Omega_2d_dot_wrt_n_1d_dot, Omega_2d_dot_wrt_n_1d_double_dot, Omega_2d_dot_wrt_n_1d_triple_dot, Omega_2d_dot_wrt_q, Omega_2d_dot_wrt_alpha_psi ] = Omega_d_dot(m, epsilon_2, epsilon_3, D_1, min_eigen_value_D_1, gamma_1, K_1, K_2, c, n_1e, v_1_hat, v_1e, n_1d_dot, n_1d_double_dot, n_1d_tripple_dot,q , Omega, Sum_M1_to_M6, Partial_M_1_to_6_wrt_partial_n_1e, Partial_M_1_to_6_wrt_partial_v_1_hat, Partial_M_1_to_6_wrt_partial_n_1d_dot, Partial_M_1_to_6_wrt_partial_n_1d_double_dot, h_4, R, alpha_1, Partial_Omega_wrt_partial_n_1e, Partial_Omega_wrt_partial_v_1_hat,Partial_Omega_wrt_partial_n_1d_dot,
Partial_Omega_wrt_partial_n_1d_double_dot, Partial_h_4_wrt_n_1e, Partial_h4_wrt_partial_n_1d_dot, Partial_h4_wrt_partial_n_1d_double_dot, Partial_h4_wrt_partial_v_1_hat, Partial_h_4_wrt_q, Partial_h_4_wrt_alpha_psi);

R_1_n_2 =R;
constant =8*4;
constant =1;
Partial_M6_by_partial_n_1d_double_dot = Partial_M_1_to_6_wrt_partial_n_1d_double_dot;
sigma_n_1e= [ n_1e(1,1)/(1+n_1e(1,1)^2)^0.5;
              n_1e(2,1)/(1+n_1e(2,1)^2)^0.5;
              n_1e(3,1)/(1+n_1e(3,1)^2)^0.5];
sigma_dash_n_1e = [ 1/(1+n_1e(1,1)^2)^1.5, 0, 0;
                    0, 1/(1+n_1e(2,1)^2)^1.5, 0;
                    0, 0, 1/(1+n_1e(3,1)^2)^1.5];
sigma_n_1e_dash =       sigma_dash_n_1e;
sigma_double_dash_n_1e = -3*[ 1/(1+n_1e(1,1)^2)^2.5, 0, 0;
                              0, 1/(1+n_1e(2,1)^2)^2.5, 0;
                              0, 0, 1/(1+n_1e(3,1)^2)^2.5];
sigma_n_1e_double_dash =  sigma_double_dash_n_1e ;
sigma_n_1e_triple_dash = 15*[ 1/(1+n_1e(1,1)^2)^2.5, 0, 0;
                              0, 1/(1+n_1e(2,1)^2)^2.5, 0;
                              0, 0, 1/(1+n_1e(3,1)^2)^3.5];
sigma_dash_v_1e = [ 1/(1+v_1e(1,1)^2)^1.5, 0, 0;
                    0, 1/(1+v_1e(2,1)^2)^1.5, 0;
                    0, 0, 1/(1+v_1e(3,1)^2)^1.5];
sigma_v_1e_dash= sigma_dash_v_1e;
sigma_double_dash_v_1e = -3*[ 1/(1+v_1e(1,1)^2)^2.5, 0, 0;
                              0, 1/(1+v_1e(2,1)^2)^2.5, 0;
                              0, 0, 1/(1+v_1e(3,1)^2)^2.5];
delta_1_v_1_hat = 1 + 0.5 * ((v_1_hat.'))*v_1_hat;
g=9.81;
G_1_transpose = eye(3) - v_1_hat*(K_1*sigma_n_1e).'/delta_1_v_1_hat^2;
G_1 = eye(3) - (K_1*sigma_n_1e*(v_1_hat.'))/delta_1_v_1_hat^2;
%norm_v_1e_squared = sigma_n_1e.'*sigma_n_1e;
norm_v_1e_squared = v_1e.'*v_1e;
sigma_n_1e_squared=sigma_n_1e.'*sigma_n_1e;
n_1e_dot = v_1_hat -n_1d_dot + 2*gamma_1*G_1_transpose  * v_1e/(1+v_1e.'*v_1e)^0.5;
n_1e_dot_wrt_v_1_hat   = eye(3) ;
n_1e_dot_wrt_n_1e      = 0*eye(3);
n_1e_dot_wrt_n_1d_dot = -eye(3)  ;
h_4_from_L2       = 2*gamma_1*G_1_transpose  *  v_1e/(1+v_1e.'*v_1e)^0.5;
v_1_hat_dot         =         -D_1*v_1_hat         -g*[0;0;1]          +(Omega.'*Omega)^0.5*R*[0;0;1]/m +min_eigen_value_D_1*0.5*gamma_1*sigma_n_1e + 2*gamma_1*(K_1*sigma_dash_n_1e/delta_1_v_1_hat)*v_1e/(1+v_1e.'*v_1e)^0.5 +h_4;
N = eye(3)+(K_1*sigma_n_1e*(v_1_hat.')/delta_1_v_1_hat^2)/(1-sigma_n_1e.'*K_1*v_1_hat/delta_1_v_1_hat^2);
N_1 = N;
%N1
%Partial_M_1_to_6_wrt_partial_v_1_hat
Y_1_1_1=(-(K_1*sigma_n_1e*v_1_hat_dot.')/(delta_1_v_1_hat^2-
sigma_n_1e.'*K_1*v_1_hat)^2*(eye(3)-
2*(v_1_hat*(v_1_hat.'))/(delta_1_v_1_hat))*N_1*(Sum_M1_to_M6)*(2*delta_1_v_1_hat*(v_1_hat.')-
```

```
sigma_n_1e.'*K_1)+(K_1*sigma_n_1e*(Sum_M1_to_M6).')/(delta_1_v_1_hat^2-
sigma_n_1e.'*K_1*v_1_hat)*N_1.'*(-
2/(delta_1_v_1_hat)*((v_1_hat.')*v_1_hat_dot*eye(3)++v_1_hat*v_1_hat_dot.')+2*(v_1_hat*(v_1_hat.'
))/(delta_1_v_1_hat)^2*v_1_hat_dot*(v_1_hat.'))+(K_1*sigma_n_1e*v_1_hat_dot.')/(delta_1_v_1_hat^
2-sigma_n_1e.'*K_1*v_1_hat)*(eye(3)-
2*(v_1_hat*(v_1_hat.'))/(delta_1_v_1_hat))*(K_1*sigma_n_1e*(Sum_M1_to_M6).')/(delta_1_v_1_hat^
2-sigma_n_1e.'*K_1*v_1_hat)*N_1.'*(eye(3)-
2*(v_1_hat*(v_1_hat.'))/(delta_1_v_1_hat))+(K_1*sigma_n_1e*v_1_hat_dot.')/(delta_1_v_1_hat^2-
sigma_n_1e.'*K_1*v_1_hat)*(eye(3)-
2*(v_1_hat*(v_1_hat.'))/(delta_1_v_1_hat))*N_1*(Partial_M_1_to_6_wrt_partial_v_1_hat));
Y_1_1_2=((K_1*sigma_n_1e*v_1_hat_dot.')/(delta_1_v_1_hat^2-
sigma_n_1e.'*K_1*v_1_hat)^2*(eye(3)-
2*(v_1_hat*(v_1_hat.'))/(delta_1_v_1_hat))*N_1*(Sum_M1_to_M6)*(v_1_hat.')*K_1*sigma_n_1e_das
h+(v_1_hat_dot.')/(delta_1_v_1_hat^2-sigma_n_1e.'*K_1*v_1_hat)*(eye(3)-
2*(v_1_hat*(v_1_hat.'))/(delta_1_v_1_hat))*N_1*(Sum_M1_to_M6)*K_1*sigma_n_1e_dash+(K_1*sig
ma_n_1e*v_1_hat_dot.')/(delta_1_v_1_hat^2-sigma_n_1e.'*K_1*v_1_hat)*(eye(3)-
2*(v_1_hat*(v_1_hat.'))/(delta_1_v_1_hat))*((v_1_hat.')*(Sum_M1_to_M6))/(delta_1_v_1_hat^2-
sigma_n_1e.'*K_1*v_1_hat)*N_1*K_1*sigma_n_1e_dash+(K_1*sigma_n_1e*v_1_hat_dot.')/(delta_1_
v_1_hat^2-sigma_n_1e.'*K_1*v_1_hat)*(eye(3)-
2*(v_1_hat*(v_1_hat.'))/(delta_1_v_1_hat))*N_1*(Partial_M_1_to_6_wrt_partial_n_1e));
Y_1_1_3=(K_1*sigma_n_1e*v_1_hat_dot.')/(delta_1_v_1_hat^2-sigma_n_1e.'*K_1*v_1_hat)*(eye(3)-
2*(v_1_hat*(v_1_hat.'))/(delta_1_v_1_hat))*N_1*(Partial_M_1_to_6_wrt_partial_n_1d_dot);
%may not need thisa one
%Y_1_1_4=(K_1*sigma_n_1e*v_1_hat_dot.')/(delta_1_v_1_hat^2-
sigma_n_1e.'*K_1*v_1_hat)*(eye(3)-
2*(v_1_hat*(v_1_hat.'))/(delta_1_v_1_hat))*N_1*Partial_M6_by_partial_n_1d_double_dot;
Y_1_2_1=(-((v_1_hat.')*(Sum_M1_to_M6))/(delta_1_v_1_hat^2-
sigma_n_1e.'*K_1*v_1_hat)^2*N_1*K_1*sigma_n_1e_dash*n_1e_dot*(2*delta_1_v_1_hat*(v_1_hat.')
-sigma_n_1e.'*K_1)+(N_1*K_1*sigma_n_1e_dash*n_1e_dot)/(delta_1_v_1_hat^2-
sigma_n_1e.'*K_1*v_1_hat)*(Sum_M1_to_M6).'+(N_1*K_1*sigma_n_1e_dash*n_1e_dot*(v_1_hat.'))/
(delta_1_v_1_hat^2-
sigma_n_1e.'*K_1*v_1_hat)*(Partial_M_1_to_6_wrt_partial_v_1_hat)+((v_1_hat.')*(Sum_M1_to_M6))
/(delta_1_v_1_hat^2-
sigma_n_1e.'*K_1*v_1_hat)*(K_1*sigma_n_1e*(K_1*sigma_n_1e_dash*n_1e_dot).')/(delta_1_v_1_h
at^2-sigma_n_1e.'*K_1*v_1_hat)*N_1.'*(eye(3)-2*(v_1_hat*(v_1_hat.'))/(delta_1_v_1_hat)));
Y_1_2_2=((v_1_hat.')*(Sum_M1_to_M6))/(delta_1_v_1_hat^2-
sigma_n_1e.'*K_1*v_1_hat)^2*N_1*K_1*sigma_n_1e_dash*n_1e_dot*(v_1_hat.')*K_1*sigma_n_1e_d
ash+((v_1_hat.')*(Sum_M1_to_M6))/(delta_1_v_1_hat^2-
sigma_n_1e.'*K_1*v_1_hat)*N_1*K_1*sigma_n_1e_double_dash*[n_1e_dot(1,1),0,0;0,n_1e_dot(2,1),
0;0,0,n_1e_dot(3,1)]+(N_1*K_1*sigma_n_1e_dash*n_1e_dot*(v_1_hat.'))/(delta_1_v_1_hat^2-
sigma_n_1e.'*K_1*v_1_hat)*(Partial_M_1_to_6_wrt_partial_n_1e)+((v_1_hat.')*(Sum_M1_to_M6))/(d
elta_1_v_1_hat^2-
sigma_n_1e.'*K_1*v_1_hat)^2*((v_1_hat.')*K_1*sigma_n_1e_dash*n_1e_dot)*N_1*K_1*sigma_n_1e
_dash;
Y_1_2_3=(N_1*K_1*sigma_n_1e_dash*n_1e_dot*(v_1_hat.'))/(delta_1_v_1_hat^2-
sigma_n_1e.'*K_1*v_1_hat)*(Partial_M_1_to_6_wrt_partial_n_1d_dot);
%may not need thisa one
%Y_1_2_4=(N_1*K_1*sigma_n_1e_dash*n_1e_dot*(v_1_hat.'))/(delta_1_v_1_hat^2-
sigma_n_1e.'*K_1*v_1_hat)*Partial_M6_by_partial_n_1d_double_dot;

%M1
%D_1*K_1*sigma_n_1e*((v_1_hat_dot.')/delta_1_v_1_hat^2-
2*((v_1_hat.')*v_1_hat_dot)/delta_1_v_1_hat^3*(v_1_hat.'))-
K_2*sigma_double_dash_v_1e*[(([1,0,0]*(G_1*v_1_hat_dot)),0,0;0,([0,1,0]*(G_1*v_1_hat_dot)),0;0,0,(
[0,0,1]*(G_1*v_1_hat_dot))]                    *                    G_1-K_2*sigma_v_1e_dash*(-
K_1*sigma_n_1e*v_1_hat_dot.'*(eye(3)/delta_1_v_1_hat^2-
2*(v_1_hat*(v_1_hat.'))/delta_1_v_1_hat^3));

%Y_2_1_1=D_1*K_1*sigma_n_1e*((v_1_hat_dot.')/delta_1_v_1_hat^2-
2*((v_1_hat.')*v_1_hat_dot)/delta_1_v_1_hat^3*(v_1_hat.'))-
```

```
K_2*sigma_double_dash_v_1e*[(([1,0,0]*(G_1*v_1_hat_dot)),1,0,0;0,([0,1,0]*(G_1*v_1_hat_dot)),1,0;
0,0,([0,0,1]*(G_1*v_1_hat_dot)),1]*G_1-K_2*sigma_v_1e_dash*(-
K_1*sigma_n_1e*v_1_hat_dot.'*(eye(3)/delta_1_v_1_hat^2-
2*(v_1_hat*(v_1_hat.'))/delta_1_v_1_hat^3));
Y_2_1_1=D_1*K_1*sigma_n_1e*((v_1_hat_dot.')/delta_1_v_1_hat^2-
2*((v_1_hat.')*v_1_hat_dot)/delta_1_v_1_hat^3*(v_1_hat.'))-
K_2*sigma_double_dash_v_1e*[(([1,0,0]*(G_1*v_1_hat_dot)),0,0;0,([0,1,0]*(G_1*v_1_hat_dot)),0;0,0,(
[0,0,1]*(G_1*v_1_hat_dot))]           *           G_1-K_2*sigma_v_1e_dash*(-
K_1*sigma_n_1e*v_1_hat_dot.'*(eye(3)/delta_1_v_1_hat^2-
2*(v_1_hat*(v_1_hat.'))/delta_1_v_1_hat^3));
Y_2_1_2=D_1*K_1*((v_1_hat.')*v_1_hat_dot)/delta_1_v_1_hat^2*sigma_n_1e_dash+K_2*sigma_v_1
e_dash*(K_1*((v_1_hat.')*v_1_hat_dot)/delta_1_v_1_hat^2*sigma_n_1e_dash)-
K_2*sigma_double_dash_v_1e*[(([1,0,0]*(G_1*v_1_hat_dot)),0,0;0,([0,1,0]*(G_1*v_1_hat_dot)),0;0,0,(
[0,0,1]*(G_1*v_1_hat_dot))]*((K_1*sigma_n_1e_dash)/(delta_1_v_1_hat));
Y_2_1_3=K_2*sigma_double_dash_v_1e*[(([1,0,0]*(G_1*v_1_hat_dot)),0,0;0,([0,1,0]*(G_1*v_1_hat_d
ot)),0;0,0,([0,0,1]*(G_1*v_1_hat_dot))];
Y_2_2_1=-
(K_2*sigma_double_dash_v_1e)*(K_1*sigma_n_1e_dash)/(delta_1_v_1_hat)*[n_1e_dot(1,1),0,0;0,n_
1e_dot(2,1),0;0,0,n_1e_dot(3,1)]*G_1+(D_1+K_2*sigma_v_1e_dash)*K_1*(sigma_n_1e_dash)/(delta
_1_v_1_hat)^2*n_1e_dot*(v_1_hat.');
Y_2_2_2=-
K_2*sigma_double_dash_v_1e*(K_1*sigma_n_1e_dash)/(delta_1_v_1_hat)*(K_1*sigma_n_1e_dash)
/(delta_1_v_1_hat)*[n_1e_dot(1,1),0,0;0,n_1e_dot(2,1),0;0,0,n_1e_dot(3,1)]-
(D_1+K_2*sigma_v_1e_dash)*K_1*(sigma_n_1e_double_dash)/(delta_1_v_1_hat)*[n_1e_dot(1,1),0,
0;0,n_1e_dot(2,1),0;0,0,n_1e_dot(3,1)];
Y_2_2_3=(K_2*sigma_double_dash_v_1e)*(K_1*sigma_n_1e_dash)/(delta_1_v_1_hat)*[n_1e_dot(1,
1),0,0;0,n_1e_dot(2,1),0;0,0,n_1e_dot(3,1)];
Y_2_3_1=K_2*sigma_double_dash_v_1e*[n_1d_double_dot(1,1),0,0;0,n_1d_double_dot(2,1),0;0,0,n
_1d_double_dot(3,1)]*G_1;
Y_2_3_2=K_2*sigma_double_dash_v_1e*[n_1d_double_dot(1,1),0,0;0,n_1d_double_dot(2,1),0;0,0,n
_1d_double_dot(3,1)]*(K_1*sigma_n_1e_dash)/(delta_1_v_1_hat);
Y_2_3_3=-
K_2*sigma_double_dash_v_1e*[n_1d_double_dot(1,1),0,0;0,n_1d_double_dot(2,1),0;0,0,n_1d_doubl
e_dot(3,1)];
%may not need thisa one
%Y_2_3_4=(D_1+K_2*sigma_v_1e_dash);

%M2
n_1e_dot_1 = n_1e_dot(1,1);
n_1e_dot_2 = n_1e_dot(2,1);
n_1e_dot_3 = n_1e_dot(3,1);
n_1d_dot_1 = n_1d_dot(1,1);
n_1d_dot_2 = n_1d_dot(2,1);
n_1d_dot_3 = n_1d_dot(3,1);
 v_1_hat_dot_1 = v_1_hat_dot(1,1);
 v_1_hat_dot_2 = v_1_hat_dot(2,1);
 v_1_hat_dot_3 = v_1_hat_dot(3,1);
n_1d_double_dot_1 = n_1d_double_dot(1,1);
n_1d_double_dot_2 = n_1d_double_dot(2,1);
n_1d_double_dot_3 = n_1d_double_dot(3,1);
 k_1= K_1(1,1);
 k_2 = K_1(2,2);
 k_3 = K_1(3,3);
n_1e_1  = n_1e(1,1);
n_1e_2  = n_1e(2,1);
n_1e_3  = n_1e(3,1);
v_1e_1  = v_1e(1,1);
v_1e_2  = v_1e(2,1);
v_1e_3  = v_1e(3,1);
v_1_hat_1 = v_1_hat(1,1);
```

```
v_1_hat_2 = v_1_hat(2,1);
v_1_hat_3 = v_1_hat(3,1);

%% Copy symbolic calulated values from the scritp symbolic omegad and h_4 derivative cal
Y_3_1_0 =
Y_3_1_1 =
Y_3_1_2 =
Y_3_2_0 =
Y_3_2_1 =
Y_3_2_2 =
Y_3_3_0 =
Y_3_3_1 = 0*eye(3);
 Y_3_3_2 =


Y_3_1_1 = Y_3_1_1 + Y_3_1_0*G_1;
Y_3_1_2 = Y_3_1_2 + Y_3_1_0*K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Y_3_1_3  = -Y_3_1_0;
Y_3_2_1 = Y_3_2_1 + Y_3_2_0*G_1;
Y_3_2_2 = Y_3_2_2 + Y_3_2_0*K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Y_3_2_3  = -Y_3_2_0;
Y_3_3_1 =Y_3_3_0*G_1;
Y_3_3_2 = Y_3_3_2 + Y_3_3_0*K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Y_3_3_3  = -Y_3_3_0;

%M3
%% Copy symbolic calulated values from the scritp symbolic omegad and h_4 derivative cal
Y_4_1_1 =
Y_4_1_2 =
Y_4_1_3 =
Y_4_2_1 =
Y_4_2_2 =
Y_4_2_3 =
Y_4_3_1 =
Y_4_3_2 =
Y_4_3_3 =


%M4
Y_5_1_1=K_1*sigma_n_1e*((-
2*(v_1_hat_dot.')/delta_1_v_1_hat^2*D_1.'+4*((v_1_hat.')*D_1*v_1_hat_dot)/delta_1_v_1_hat^3*(v_1
_hat.')+2*((v_1_hat.')*D_1*v_1_hat)/delta_1_v_1_hat^3*v_1_hat_dot.'+2*((v_1_hat.')*D_1)/delta_1_v
_1_hat^3*(v_1_hat_dot.'*v_1_hat)+2*((v_1_hat.')*D_1.')/delta_1_v_1_hat^3*(v_1_hat_dot.'*v_1_hat)-
6*((v_1_hat.')*D_1*v_1_hat)/delta_1_v_1_hat^4*(v_1_hat.')*v_1_hat_dot*(v_1_hat.')+2*v_1_hat_dot.'*
K_1*sigma_n_1e_dash*(eye(3)/delta_1_v_1_hat^3-3*(v_1_hat*(v_1_hat.'))/delta_1_v_1_hat^4)-
6*((v_1_hat.')*v_1_hat_dot*(v_1_hat.'))/delta_1_v_1_hat^4*(K_1*sigma_n_1e_dash).'-
3*((v_1_hat.')*K_1*sigma_n_1e_dash*v_1_hat)/delta_1_v_1_hat^4*v_1_hat_dot.'+12*((v_1_hat.')*K_
1*sigma_n_1e_dash)/delta_1_v_1_hat^5*v_1_hat*(v_1_hat.')*v_1_hat_dot*(v_1_hat.')));
Y_5_1_2=(((-
2*((v_1_hat.')*D_1)/delta_1_v_1_hat^2+2*((v_1_hat.')*D_1*v_1_hat)/delta_1_v_1_hat^3*(v_1_hat.')+
2*((v_1_hat.')*K_1*sigma_n_1e_dash)/delta_1_v_1_hat^3-
3*((v_1_hat.')*K_1*sigma_n_1e_dash*v_1_hat)/delta_1_v_1_hat^4*(v_1_hat.'))*v_1_hat_dot)*K_1*si
gma_n_1e_dash+K_1*sigma_n_1e*(2*(v_1_hat_dot.')/delta_1_v_1_hat^3-
3*(v_1_hat.')*v_1_hat_dot*((v_1_hat.'))/delta_1_v_1_hat^4)*K_1*sigma_n_1e_double_dash*[v_1_hat
(1,1),0,0;0,v_1_hat(2,1),0;0,0,v_1_hat(3,1)]);
Y_5_2_1=(-K_1*sigma_n_1e_dash*n_1e_dot*(v_1_hat.')*D_1*(2*eye(3)/delta_1_v_1_hat^2-
2*(v_1_hat*(v_1_hat.'))/delta_1_v_1_hat^3)+(K_1*sigma_n_1e_dash*n_1e_dot*(v_1_hat.')*K_1*sigm
a_n_1e_dash+K_1*sigma_n_1e*(v_1_hat.')*K_1*sigma_n_1e_double_dash*[n_1e_dot(1,1),0,0;0,n_1
e_dot(2,1),0;0,0,n_1e_dot(3,1)])*(2*eye(3)/delta_1_v_1_hat^3-
3*(v_1_hat*(v_1_hat.'))/delta_1_v_1_hat^4));
```

```
Y_5_2_2=((-
((v_1_hat.')*D_1*v_1_hat)/delta_1_v_1_hat^2+((v_1_hat.')*K_1*sigma_n_1e_dash*v_1_hat)/delta_1_
v_1_hat^3)*K_1*sigma_n_1e_double_dash*[n_1e_dot(1,1),0,0;0,n_1e_dot(2,1),0;0,0,n_1e_dot(3,1)]+
(K_1*sigma_n_1e_dash*n_1e_dot)/delta_1_v_1_hat^3*(v_1_hat.')*K_1*sigma_n_1e_double_dash*[v
_1_hat(1,1),0,0;0,v_1_hat(2,1),0;0,0,v_1_hat(3,1)]+K_1*(sigma_n_1e*(v_1_hat.'))/delta_1_v_1_hat^3
*K_1*sigma_n_1e_triple_dash*[v_1_hat(1,1),0,0;0,v_1_hat(2,1),0;0,0,v_1_hat(3,1)]*[n_1e_dot(1,1),0,
0;0,n_1e_dot(2,1),0;0,0,n_1e_dot(3,1)]+(v_1_hat.')*K_1*sigma_n_1e_double_dash*[v_1_hat(1,1),0,0;
0,v_1_hat(2,1),0;0,0,v_1_hat(3,1)]*n_1e_dot*K_1*(sigma_n_1e_dash)/delta_1_v_1_hat^3);

%% M5
%% Copy symbolic calulated values from the scritp symbolic omegad and h_4 derivative cal
Y_6_1_0 =
Y_6_1_1 =
Y_6_1_2 =
 Y_6_2_0 =
Y_6_2_1 =
Y_6_2_2 =
Y_6_3_0 =
Y_6_3_1 =
Y_6_3_2 =
 Y_6_1_1  = Y_6_1_1  + Y_6_1_0 * G_1;
Y_6_1_2  = Y_6_1_2  + Y_6_1_0 * K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Y_6_1_3  = -Y_6_1_0 ;

Y_6_2_1  = Y_6_2_1  + Y_6_2_0 * G_1;
Y_6_2_2  = Y_6_2_2  + Y_6_2_0 * K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Y_6_2_3  = -Y_6_2_0 ;
Y_6_3_1  = Y_6_3_1  + Y_6_3_0 * G_1;
Y_6_3_2  = Y_6_3_2  + Y_6_3_0 * K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Y_6_3_3  = -Y_6_3_0 ;


%% M6
%% Copy symbolic calulated values from the scritp symbolic omegad and h_4 derivative cal
Y_7_1_1=0*eye(3);
Y_7_1_2=0*eye(3);
Y_7_1_3=0*eye(3);
Y_7_2_1=0*eye(3);
Y_7_2_2=0*eye(3);
Y_7_2_3=0*eye(3);
Y_7_3_1=0*eye(3);
Y_7_3_2=0*eye(3);
Y_7_1_0 = 0*eye(3);
 Y_7_1_1 =
 Y_7_1_2 =
 Y_7_2_0 = 0*eye(3);
 Y_7_2_1 =
 Y_7_2_2 =
Y_7_3_0 = 0*eye(3);
Y_7_3_1 = 0*eye(3);
Y_7_3_2 = 0*eye(3);

%% M7
%% Copy symbolic calulated values from the scritp symbolic omegad and h_4 derivative cal
Y_8_1_0 =
Y_8_1_1 =
Y_8_1_2 =
Y_8_2_0 =
Y_8_2_1 =
Y_8_2_2 =
```

```
Y_8_3_0 =
Y_8_3_1 =
Y_8_3_2 =

Y_8_1_1  = Y_8_1_1  + Y_8_1_0 * G_1;
Y_8_1_2  = Y_8_1_2  + Y_8_1_0 * K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Y_8_1_3  = -Y_8_1_0 ;

Y_8_2_1  = Y_8_2_1  + Y_8_2_0 * G_1;
Y_8_2_2  = Y_8_2_2  + Y_8_2_0 * K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Y_8_2_3  = -Y_8_2_0 ;

Y_8_3_1  = Y_8_3_1  + Y_8_3_0 * G_1;
Y_8_3_2  = Y_8_3_2  + Y_8_3_0 * K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Y_8_3_3  = -Y_8_3_0 ;

%% M8
Y_9_1_1=(K_1*sigma_n_1e)/(1-
sigma_n_1e.'*K_1*v_1_hat/delta_1_v_1_hat^2)*(N*((Partial_M_1_to_6_wrt_partial_v_1_hat)*v_1_hat
_dot+(Partial_M_1_to_6_wrt_partial_n_1e)*n_1e_dot+(Partial_M_1_to_6_wrt_partial_n_1d_dot)*n_1
d_double_dot+Partial_M6_by_partial_n_1d_double_dot*n_1d_tripple_dot)).'*(eye(3)/delta_1_v_1_hat
^2-2*(v_1_hat*(v_1_hat.'))/delta_1_v_1_hat^3);
Y_9_1_2=(v_1_hat.')*((Partial_M_1_to_6_wrt_partial_v_1_hat)*v_1_hat_dot+(Partial_M_1_to_6_wrt_
partial_n_1e)*n_1e_dot+(Partial_M_1_to_6_wrt_partial_n_1d_dot)*n_1d_double_dot+Partial_M6_by_
partial_n_1d_double_dot*n_1d_tripple_dot)*(N_1*K_1*(sigma_n_1e_dash)/delta_1_v_1_hat^2)/(1-
sigma_n_1e.'*K_1*v_1_hat/delta_1_v_1_hat^2);
%clc

e_3 = [0;.0;1];
%%%%%%%%%%%%%%%%CHANGE THIS
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Partial_R_1_q__times_e3_ert_q =  10^20 * eye(3);
Partial_R_1_q__times_e3_ert_q = Partial_R_1_q__wrt_partial_q_times_vecor(q,e_3);
%%%%%%%%%%%%%%%%CHANGE THIS
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
h_3 = 2*gamma_1*(K_1*sigma_dash_n_1e/delta_1_v_1_hat) * v_1e/(1+v_1e.'*v_1e)^0.5;
Partial_h_3_wrt_v_1e = 2*constant*gamma_1*(K_1*sigma_dash_n_1e/delta_1_v_1_hat)
*(eye(3)-v_1e*v_1e.'/(1+v_1e.'*v_1e))/(1+v_1e.'*v_1e)^0.5;
Partial_h_3_wrt_n_1e =
2*constant*gamma_1*(K_1*sigma_n_1e_double_dash)/(delta_1_v_1_hat*(1+v_1e.'*v_1e)^0.5
)*[v_1e(1,1),0,0;0,v_1e(2,1),0;0,0,v_1e(3,1) ] + Partial_h_3_wrt_v_1e
*K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Partial_h_3_wrt_v_1_hat = -2*constant*gamma_1*(K_1* sigma_dash_n_1e/delta_1_v_1_hat^2)*(
v_1e/(1+v_1e.'*v_1e)^0.5)*(v_1_hat.') + Partial_h_3_wrt_v_1e * G_1;
Partial_h_3_wrt_n_1d_dot = -Partial_h_3_wrt_v_1e ;

%/(1+v_1e.'*v_1e)^0.5

Partial_v_1_hat_dot_wrt_n_1e = (min_eigen_value_D_1*0.5*gamma_1*sigma_dash_n_1e +
Partial_h_3_wrt_n_1e+(R_1_n_2*e_3*Omega.')/(m*(Omega.'*Omega)^0.5)*Partial_Omega_wrt_parti
al_n_1e+ Partial_h_3_wrt_n_1e +Partial_h_4_wrt_n_1e);
Partial_v_1_hat_dot_wrt_v_1_hat =-(D_1 + Partial_h_3_wrt_v_1_hat + (R_1_n_2 * e_3 *
Omega.')/(m*(Omega.'* Omega)^0.5) *
Partial_Omega_wrt_partial_v_1_hat+Partial_h4_wrt_partial_v_1_hat);
Partial_v_1_hat_dot_wrt_n_1d_dot = (R_1_n_2* e_3*Omega.')/(m*(Omega.'*Omega)^0.5) *
Partial_Omega_wrt_partial_n_1d_dot +Partial_h4_wrt_partial_n_1d_dot + Partial_h_3_wrt_n_1d_dot
;
Partial_v_1_hat_dot_wrt_n_1d_double_dot = (R_1_n_2* e_3*Omega.')/(m*(Omega.'*Omega)^0.5) *
Partial_Omega_wrt_partial_n_1d_double_dot +Partial_h4_wrt_partial_n_1d_double_dot;
```

```
Partial_v_1_hat_dot_wrt_q = (Omega.'*Omega)^0.5/m * Partial_R_1_q__times_e3_ert_q +Partial_h_4_wrt_q;
Partial_v_1_hat_dot_wrt_alpha_psi = Partial_h_4_wrt_alpha_psi;

Partial_h_4_from_L2_wrt_v_1e = 2*gamma_1*G_1_transpose * (eye(3) - v_1e*v_1e.'/(1+v_1e.'*v_1e))/(1+v_1e.'*v_1e)^0.5;
Partial_h_4_from_L2_wrt_n_1e = Partial_h_4_from_L2_wrt_v_1e* K_1* sigma_n_1e_dash/delta_1_v_1_hat - 2*gamma_1*(v_1_hat*(v_1e.'/(1+v_1e.'*v_1e)^0.5)*K_1*sigma_dash_n_1e/delta_1_v_1_hat^2);
Partial_h_4_from_L2_wrt_v_1_hat = Partial_h_4_from_L2_wrt_v_1e*G_1 -2* gamma_1*(((v_1e.'/(1+v_1e.'*v_1e)^0.5)*K_1*sigma_n_1e)*(eye(3)/delta_1_v_1_hat^2- v_1_hat*(v_1_hat.')/delta_1_v_1_hat^3));
Partial_h_4_from_L2_wrt_n_1d_dot = -Partial_h_4_from_L2_wrt_v_1e;


Omega_2d_dot_wrt_n_1e =m*(Y_1_1_2+Y_1_2_2+N_1*(Y_2_1_2 + Y_2_2_2 + Y_2_3_2 + Y_3_1_2 + Y_3_2_2 + Y_3_3_2 + Y_4_1_2 + Y_4_2_2 + Y_4_3_2 + Y_5_1_2 + Y_5_2_2 + Y_6_1_2 + Y_6_2_2 + Y_6_3_2 + Y_7_1_2 + Y_7_2_2 + Y_7_3_2) )+ Y_8_1_2 + Y_8_2_2 + Y_8_3_2 + Y_9_1_2 + Partial_Omega_wrt_partial_v_1_hat * Partial_v_1_hat_dot_wrt_n_1e + Partial_Omega_wrt_partial_n_1e*Partial_h_4_from_L2_wrt_n_1e;
Omega_2d_dot_wrt_v_1_hat =m*(Y_1_1_1 + Y_1_2_1 + N_1 * (Y_2_1_1 + Y_2_2_1 + Y_2_3_1 + Y_3_1_1 + Y_3_2_1 + Y_3_3_1 + Y_4_1_1 + Y_4_2_1 + Y_4_3_1 + Y_5_1_1+ Y_5_2_1 + Y_6_1_1 + Y_6_2_1 + Y_6_3_1 + Y_7_1_1+ Y_7_2_1 + Y_7_3_1)) + Y_8_1_1 + Y_8_2_1 + Y_8_3_1 + Y_9_1_1 + Partial_Omega_wrt_partial_n_1e *n_1e_dot_wrt_v_1_hat+ Partial_Omega_wrt_partial_v_1_hat * Partial_v_1_hat_dot_wrt_v_1_hat + Partial_Omega_wrt_partial_n_1e*Partial_h_4_from_L2_wrt_v_1_hat ;
 Omega_2d_dot_wrt_n_1d_dot = m*(Y_1_1_3 + Y_1_2_3 + N_1 * (Y_2_1_3 + Y_2_2_3 + Y_2_3_3 + Y_3_1_3 + Y_3_2_3 + Y_3_3_3 + Y_4_1_3 + Y_4_2_3 + Y_4_3_3 + Y_6_1_3 +Y_6_2_3 + Y_6_3_3 + Y_7_1_3 + Y_7_2_3 ))+ Y_8_1_3 + Y_8_2_3 + Partial_Omega_wrt_partial_n_1e *n_1e_dot_wrt_n_1d_dot + Partial_Omega_wrt_partial_v_1_hat * Partial_v_1_hat_dot_wrt_n_1d_dot + Partial_Omega_wrt_partial_n_1e*Partial_h_4_from_L2_wrt_n_1d_dot;


Omega_2d_dot_wrt_n_1d_double_dot = Partial_Omega_wrt_partial_n_1d_dot + Partial_Omega_wrt_partial_v_1_hat * Partial_v_1_hat_dot_wrt_n_1d_double_dot ;
Omega_2d_dot_wrt_q = Partial_Omega_wrt_partial_v_1_hat * Partial_v_1_hat_dot_wrt_q ;
Omega_2d_dot_wrt_alpha_psi = Partial_Omega_wrt_partial_v_1_hat * Partial_v_1_hat_dot_wrt_alpha_psi ;
Omega_2d_dot_wrt_n_1d_triple_dot = Partial_Omega_wrt_partial_n_1d_double_dot;

end
```

```
function [Partial_h_4_wrt_n_1e, Partial_h4_wrt_partial_n_1d_dot, Partial_h4_wrt_partial_n_1d_double_dot, Partial_h4_wrt_partial_v_1_hat, Partial_h_4_wrt_q, Partial_h_4_wrt_alpha_psi ] = h_4_Partial_derivatives(m, epsilon_2, epsilon_3, D_1, min_eigen_value_D_1, gamma_1, K_1, K_2, c, n_1e, v_1_hat, v_1e, n_1d_dot, n_1d_double_dot, q_e, Sum_M1_to_M6, Partial_M_1_to_6_wrt_partial_n_1e, Partial_M_1_to_6_wrt_partial_v_1_hat, Partial_M_1_to_6_wrt_partial_n_1d_dot, Partial_M_1_to_6_wrt_partial_n_1d_double_dot, alpha_1, Partial_Omega_wrt_partial_n_1e, Partial_Omega_wrt_partial_v_1_hat , Partial_Omega_wrt_partial_n_1d_dot, Partial_Omega_wrt_partial_n_1d_double_dot, A_1d, A_2d, T_alpha_2_inverse,R_2_alpha_q ,alpha_psi, alpha_phi, alpha_theta, alpha_q, Omega,L2_to_L3 )

norm_v_1e = (v_1e.'*v_1e)^0.5;
norm_v_1e_squared = (v_1e.'*v_1e);
angles = [alpha_phi; alpha_theta; alpha_psi];
min_eigen_value_D1 = min_eigen_value_D_1;
R_2_alpha_q =  R_2_MRP_calc(alpha_q);
R_2_alpha_q_transpose = R_2_alpha_q.';
norm_q_e_norm_square = q_e.'*q_e;
norm_q_e_norm_square  = 1;
T_inv                    = Angular_velocity_cordinant_transform(angles);
T_alpha_2_inverse = T_inv;
T_alpha_2_inverse_transpose = T_alpha_2_inverse.';
norm_v_1e_square = (v_1e.')*v_1e;
delta_1_v_1_hat = 1 + 0.5*(v_1_hat.')*v_1_hat;
q_e_norm_square = q_e.'*q_e;
q_e_norm_square = 1;
sigma_n_1e= [ n_1e(1,1)/(1+n_1e(1,1)^2)^0.5;
                  n_1e(2,1)/(1+n_1e(2,1)^2)^0.5;
                  n_1e(3,1)/(1+n_1e(3,1)^2)^0.5];
sigma_dash_n_1e = [ 1/(1+n_1e(1,1)^2)^1.5, 0, 0;
                          0, 1/(1+n_1e(2,1)^2)^1.5, 0;
                          0, 0, 1/(1+n_1e(3,1)^2)^1.5];
sigma_double_dash_n_1e = -3*[ 1/(1+n_1e(1,1)^2)^2.5, 0, 0;
                                    0, 1/(1+n_1e(2,1)^2)^2.5, 0;
                                    0, 0, 1/(1+n_1e(3,1)^2)^2.5];
sigma_n_1e_triple_dash = 15*[ 1/(1+n_1e(1,1)^2)^3.5, 0, 0;
                                    0, 1/(1+n_1e(2,1)^3)^2.5, 0;
                                    0, 0, 1/(1+n_1e(3,1)^2)^3.5];
sigma_tripple_dash_n_1e = sigma_n_1e_triple_dash ;
sigma_dash_v_1e = [ 1/(1+v_1e(1,1)^2)^1.5, 0, 0;
                          0, 1/(1+v_1e(2,1)^2)^1.5, 0;
                          0, 0, 1/(1+v_1e(3,1)^2)^1.5];
sigma_v_1e_dash= sigma_dash_v_1e;
sigma_double_dash_v_1e = -3*[ 1/(1+v_1e(1,1)^2)^2.5, 0, 0;
                                    0, 1/(1+v_1e(2,1)^2)^2.5, 0;
                                    0, 0, 1/(1+v_1e(3,1)^2)^2.5];
N_1= eye(3)+(K_1*sigma_n_1e*((v_1_hat.'))/delta_1_v_1_hat^2)/(1-sigma_n_1e.'*K_1*v_1_hat/delta_1_v_1_hat^2);
G_1= eye(3)-(K_1*sigma_n_1e*((v_1_hat.'))/delta_1_v_1_hat^2);
sigma_n_1e_squared = sigma_n_1e.'*sigma_n_1e;
vector = -N_1.'*A_2d.'*T_alpha_2_inverse.'*R_2_alpha_q.'*q_e;
vector_1 =vector(1,1);
vector_2 =vector(2,1);
vector_3 =vector(3,1);
 n_1d_dot_1  = n_1d_dot(1,1);
 n_1d_dot_2  = n_1d_dot(2,1);
 n_1d_dot_3  = n_1d_dot(3,1);

 k_1= K_1(1,1);
 k_2 = K_1(2,2);
 k_3 = K_1(3,3);
```

```
n_1e_1  = n_1e(1,1);
n_1e_2  = n_1e(2,1);
n_1e_3  = n_1e(3,1);
v_1e_1  = v_1e(1,1);
v_1e_2  = v_1e(2,1);
v_1e_3  = v_1e(3,1);
v_1_hat_1 = v_1_hat(1,1);
v_1_hat_2 = v_1_hat(2,1);
v_1_hat_3 = v_1_hat(3,1);

%N_1
Z_1_1=(((K_1*sigma_dash_n_1e*N_1.'*q_e_norm_square*A_2d.'*T_alpha_2_inverse.'*R_2_alpha_q.
'*q_e)/(1-sigma_n_1e.'*K_1*v_1_hat/delta_1_v_1_hat^2)*Sum_M1_to_M6
.'+((v_1_hat.')*Sum_M1_to_M6                                                      )/(1-
sigma_n_1e.'*K_1*v_1_hat/delta_1_v_1_hat^2)^2*(K_1*sigma_dash_n_1e)/delta_1_v_1_hat^2*(N_1.
'*q_e_norm_square*A_2d.'*T_alpha_2_inverse.'*R_2_alpha_q.'*q_e*sigma_n_1e.'*K_1+(q_e_norm_s
quare*A_2d.'*T_alpha_2_inverse.'*R_2_alpha_q.'*q_e).'*K_1*sigma_n_1e*N_1.'))*(eye(3)/delta_1_v_
1_hat^2-
2*(v_1_hat*(v_1_hat.'))/delta_1_v_1_hat^3)+(K_1*sigma_dash_n_1e)/delta_1_v_1_hat^2*N_1.'*(q_e
_norm_square*A_2d.'*T_alpha_2_inverse.'*R_2_alpha_q.'*q_e*(v_1_hat.'))/(1-
sigma_n_1e.'*K_1*v_1_hat/delta_1_v_1_hat^2)*(Partial_M_1_to_6_wrt_partial_v_1_hat));
Z_1_2=(((v_1_hat.')*Sum_M1_to_M6                                                  )/(1-
sigma_n_1e.'*K_1*v_1_hat/delta_1_v_1_hat^2)*(K_1*sigma_double_dash_n_1e)/delta_1_v_1_hat^2*
([1,0,0;0,0,0;0,0,0]*([1;0;0].'*N_1.'*q_e_norm_square*A_2d.'*T_alpha_2_inverse.'*R_2_alpha_q.'*q_e)
+[0,0,0;0,1,0;0,0,0]*([0;1;0].'*N_1.'*q_e_norm_square*A_2d.'*T_alpha_2_inverse.'*R_2_alpha_q.'*q_e
)+[0,0,0;0,0,0;0,0,1]*([0;0;1].'*N_1.'*q_e_norm_square*A_2d.'*T_alpha_2_inverse.'*R_2_alpha_q.'*q_
e))+((v_1_hat.')*Sum_M1_to_M6                                                     )/(1-
sigma_n_1e.'*K_1*v_1_hat/delta_1_v_1_hat^2)^2*(K_1*sigma_dash_n_1e)/delta_1_v_1_hat^2*N_1.'
*q_e_norm_square*A_2d.'*T_alpha_2_inverse.'*R_2_alpha_q.'*q_e*(v_1_hat/delta_1_v_1_hat^2).'*K
_1*sigma_dash_n_1e+(K_1*sigma_dash_n_1e)/delta_1_v_1_hat^2*(N_1.'*q_e_norm_square*A_2d.'*
T_alpha_2_inverse.'*R_2_alpha_q.'*q_e*(v_1_hat.'))/(1-
sigma_n_1e.'*K_1*v_1_hat/delta_1_v_1_hat^2)*(Partial_M_1_to_6_wrt_partial_n_1e)+((v_1_hat.')*Su
m_M1_to_M6                                                                        )/(1-
sigma_n_1e.'*K_1*v_1_hat/delta_1_v_1_hat^2)*(K_1*sigma_dash_n_1e)/delta_1_v_1_hat^2*((v_1_h
at/delta_1_v_1_hat^2*(q_e_norm_square*A_2d.'*T_alpha_2_inverse.'*R_2_alpha_q.'*q_e).')/(1-
sigma_n_1e.'*K_1*v_1_hat/delta_1_v_1_hat^2)*N_1)*K_1*sigma_dash_n_1e);

%% M1
Z_2_1=(-
(K_2*sigma_double_dash_v_1e)*(K_1*sigma_dash_n_1e)/(delta_1_v_1_hat)*([1,0,0;0,0,0;0,0,0]*([1;
0;0].'*N_1.'*q_e_norm_square*A_2d.'*T_alpha_2_inverse.'*R_2_alpha_q.'*q_e)+[0,0,0;0,1,0;0,0,0]*([0
;1;0].'*N_1.'*q_e_norm_square*A_2d.'*T_alpha_2_inverse.'*R_2_alpha_q.'*q_e)+[0,0,0;0,0,0;0,0,1]*([
0;0;1].'*N_1.'*q_e_norm_square*A_2d.'*T_alpha_2_inverse.'*R_2_alpha_q.'*q_e))*G_1+(D_1+K_2*si
gma_v_1e_dash)*K_1*(sigma_dash_n_1e)/(delta_1_v_1_hat)^2*(N_1.'*q_e_norm_square*A_2d.'*T_
alpha_2_inverse.'*R_2_alpha_q.'*q_e)*(v_1_hat.'));
Z_2_2=-
(K_2*sigma_double_dash_v_1e*(K_1*sigma_dash_n_1e)/(delta_1_v_1_hat)*(K_1*sigma_dash_n_1e
)/(delta_1_v_1_hat)+(D_1+K_2*sigma_v_1e_dash)*K_1*(sigma_double_dash_n_1e)/(delta_1_v_1_h
at))*([1,0,0;0,0,0;0,0,0]*([1;0;0].'*N_1.'*q_e_norm_square*A_2d.'*T_alpha_2_inverse.'*R_2_alpha_q.'*
q_e)+[0,0,0;0,1,0;0,0,0]*([0;1;0].'*N_1.'*q_e_norm_square*A_2d.'*T_alpha_2_inverse.'*R_2_alpha_q.'
*q_e)+[0,0,0;0,0,0;0,0,1]*([0;0;1].'*N_1.'*q_e_norm_square*A_2d.'*T_alpha_2_inverse.'*R_2_alpha_q.
'*q_e));
Z_2_3=K_2*sigma_double_dash_v_1e*(K_1*sigma_dash_n_1e)/(delta_1_v_1_hat)*([1,0,0;0,0,0;0,0,
0]*([1;0;0].'*N_1.'*q_e_norm_square*A_2d.'*T_alpha_2_inverse.'*R_2_alpha_q.'*q_e)+[0,0,0;0,1,0;0,0
,0]*([0;1;0].'*N_1.'*q_e_norm_square*A_2d.'*T_alpha_2_inverse.'*R_2_alpha_q.'*q_e)+[0,0,0;0,0,0;0,
0,1]*([0;0;1].'*N_1.'*q_e_norm_square*A_2d.'*T_alpha_2_inverse.'*R_2_alpha_q.'*q_e));

%% M2
%% Copy symbolic calulated values from the scritp symbolic omegad and h_4 derivative cal
```

```
Z_3_0 =
Z_3_1 =
Z_3_2 =
Z_3_1 = Z_3_1 +Z_3_0 * G_1;
Z_3_2 = Z_3_2 +Z_3_0 * K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Z_3_3 = -Z_3_0 ;

%% M3
%% Copy symbolic calulated values from the scritp symbolic omegad and h_4 derivative cal

Z_4_1=0*eye(3);
Z_4_2=0*eye(3);
Z_4_3 = 0*eye(3);
Z_4_1 =
Z_4_2 =
Z_4_3 =

%% M4
Z_5_1= -K_1*sigma_dash_n_1e*vector*2*(((v_1_hat.')*D_1)/(delta_1_v_1_hat)^2 -((v_1_hat.')*D_1*v_1_hat)/(delta_1_v_1_hat)^3*(v_1_hat.'));
Z_5_2 = -((v_1_hat.')*D_1*v_1_hat)/(delta_1_v_1_hat)^2*K_1*sigma_double_dash_n_1e*[vector(1,1) 0 0;
0 vector(2,1) 0;
0 0 vector(3,1)];
%% M5
%% Copy symbolic calulated values from the scritp symbolic omegad and h_4 derivative cal

Z_6_0 =
Z_6_1 =
 Z_6_2 =
Z_6_3 =
Z_6_1 = Z_6_1 + Z_6_0*G_1;
Z_6_2 = Z_6_2 + Z_6_0* K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Z_6_3 = -Z_6_0;

%% M6
%% Copy symbolic calulated values from the scritp symbolic omegad and h_4 derivative cal

Z_7_1 =
Z_7_2 = 0, 0, -(12*k_3^2*vector_3*(- 5*n_1e_3^2 + 2*n_1e_3 + 1))/((n_1e_3^2 + 1)^4*(2*delta_1_v_1_hat)^2)];
 Z_7_3 =

%% M7
%% Copy symbolic calulated values from the scritp symbolic omegad and h_4 derivative cal

vector = -A_2d.'*T_alpha_2_inverse.'*R_2_alpha_q.'*q_e;
vector_1 =vector(1,1);
vector_2 =vector(2,1);
vector_3 =vector(3,1);
Z_8_0 =
Z_8_1 =
Z_8_2 =

Z_8_1 =( Z_8_1 +Z_8_0 * G_1);
Z_8_2 =(Z_8_2 +Z_8_0 * K_1*sigma_dash_n_1e/delta_1_v_1_hat);
Z_8_3 =( -Z_8_0 );
```

```
%% M8
Z_9_1=m*(Partial_M_1_to_6_wrt_partial_n_1e).'*((q_e_norm_square*A_2d.'*T_alpha_2_inverse.'*R_
2_alpha_q.'*q_e).'*K_1*sigma_n_1e*N_1.')/(1-
sigma_n_1e.'*K_1*v_1_hat/delta_1_v_1_hat^2)*(eye(3)/delta_1_v_1_hat^2-
2*(v_1_hat*(v_1_hat.'))/delta_1_v_1_hat^3);
Z_9_2=m*(Partial_M_1_to_6_wrt_partial_n_1e).'*(v_1_hat*q_e_norm_square*q_e.'*R_2_alpha_q*T_
alpha_2_inverse*A_2d*N_1)/(1-
sigma_n_1e.'*K_1*v_1_hat/delta_1_v_1_hat^2)*(K_1*sigma_dash_n_1e)/delta_1_v_1_hat^2;

Z_1_3 = 0 * eye(3);
Z_5_3 = 0 * eye(3);
Z_6_3 = 0 * eye(3);
Z_9_3 = 0 * eye(3);




Partial_Partial_Omega_wrt_partial_n_1e_transpose_wrt_n_1e        = m*(Z_1_2 + Z_2_2 + Z_3_2 +
Z_4_2 + Z_5_2 + Z_6_2 + Z_7_2) + Z_8_2 + Z_9_2;
Partial_Partial_Omega_wrt_partial_n_1e_transpose_wrt_v_1_hat    = m*(Z_1_1 + Z_2_1 + Z_3_1 +
Z_4_1 + Z_5_1 + Z_6_1 + Z_7_1) + Z_8_1 + Z_9_1;
Partial_Partial_Omega_wrt_partial_n_1e_transpose_wrt_n_1d_dot = m*(Z_1_3 + Z_2_3 + Z_3_3 +
Z_4_3 + Z_5_3 + Z_6_3 + Z_7_3) + Z_8_3 + Z_9_3;

Q_1=([-
sin(alpha_psi)/cos(alpha_phi),0,0;cos(alpha_psi)/cos(alpha_phi),0,0;0,0,0]*(T_alpha_2_inverse_trans
pose*R_2_alpha_q_transpose*q_e*Omega.')/(Omega.'*Omega)^1.5+[0,-
(cos(alpha_theta))^2*cos(alpha_psi)/(Omega(3,1)^2),0;0,-
(cos(alpha_theta))^2*sin(alpha_psi)/(Omega(3,1)^2),0;0,(cos(alpha_theta))^2*tan(alpha_theta)/(Ome
ga(3,1)^2),0]*(T_alpha_2_inverse_transpose*R_2_alpha_q_transpose*q_e)/(Omega.'*Omega)^1.5*[0
;0;1].'-
eye(3)*([tan(alpha_phi)/(Omega.'*Omega);0;0].'*T_alpha_2_inverse_transpose*R_2_alpha_q_transpo
se*q_e)+2*Omega*([tan(alpha_phi)/(Omega.'*Omega)^2;0;0].')*T_alpha_2_inverse_transpose*R_2_a
lpha_q_transpose*q_e*Omega.');
Q_2=([sin(alpha_psi)*sin(alpha_phi)/((cos(alpha_phi))^2*(Omega.'*Omega)^0.5),0,0;-
cos(alpha_psi)*sin(alpha_phi)/((cos(alpha_phi))^2*(Omega.'*Omega)^0.5),0,0;0,0,0]*T_alpha_2_inver
se_transpose*R_2_alpha_q_transpose*q_e*[1;0;0].'+[0,-
cos(alpha_psi)*sin(2*alpha_theta)/Omega(3,1),0;0,-
sin(alpha_psi)*sin(2*alpha_theta)/Omega(3,1),0;0,-
cos(2*alpha_theta)/Omega(3,1),0]*T_alpha_2_inverse_transpose*R_2_alpha_q_transpose*q_e*[0;1;
0].'-
Omega*[1/((cos(alpha_phi))^2*Omega.'*Omega);0;0].'*T_alpha_2_inverse_transpose*R_2_alpha_q_t
ranspose*q_e*[1;0;0].');
Q_3=[cos(alpha_psi)/(cos(alpha_phi)*(Omega.'*Omega)^0.5),-
(cos(alpha_theta))^2*sin(alpha_psi)/Omega(3,1),0;sin(alpha_psi)/(cos(alpha_phi)*(Omega.'*Omega)^
0.5),(cos(alpha_theta))^2*cos(alpha_psi)/Omega(3,1),0;0,0,0]*T_alpha_2_inverse_transpose*R_2_al
pha_q_transpose*q_e;
Q_1= Partial_Omega_wrt_partial_n_1e.'*Q_1;
Q_2= Partial_Omega_wrt_partial_n_1e.'*Q_2;
Q_3= Partial_Omega_wrt_partial_n_1e.'*Q_3;
Q_4=norm_q_e_norm_square*Partial_Omega_wrt_partial_n_1e.'*A_2d.'*([0,0,0;0,-
sin(alpha_phi),cos(alpha_theta)*cos(alpha_phi);0,-cos(alpha_phi),-
cos(alpha_theta)*sin(alpha_phi)].'*R_2_alpha_q_transpose*q_e*[1;0;0].'+[0,0,-cos(alpha_theta);0,0,-
sin(alpha_theta)*sin(alpha_phi);0,0,-
sin(alpha_theta)*cos(alpha_phi)]*R_2_alpha_q_transpose*q_e*[0;1;0].');
Q_5=norm_q_e_norm_square*Partial_Omega_wrt_partial_n_1e.'*A_2d.'*T_alpha_2_inverse_transpo
se*(1/2*(alpha_q.'*q_e*eye(3)+alpha_q*q_e.'-([0,0,0;0,0,1;0,-1,0]*([1,0,0]*q_e)+[0,0,-
1;0,0,0;1,0,0]*([0,1,0]*q_e)+[0,1,0;-1,0,0;0,0,0]*([0,0,1]*q_e))-
q_e*alpha_q.'))*R_2_alpha_q*T_alpha_2_inverse;
```

```
Q_6=2*Partial_Omega_wrt_partial_n_1e.'*A_2d.'*T_alpha_2_inverse_transpose*R_2_alpha_q_transpose*q_e*q_e.';
Q_7=-2*Partial_Omega_wrt_partial_n_1e.'*A_2d.'*T_alpha_2_inverse_transpose*R_2_alpha_q_transpose*q_e*q_e.'*R_2_alpha_q*T_alpha_2_inverse;
Q_6= 0 * eye(3);
Q_7= 0 * eye(3);
Q_8=norm_q_e_norm_square*Partial_Omega_wrt_partial_n_1e.'*A_2d.'*T_alpha_2_inverse_transpose*R_2_alpha_q_transpose;
Q_9=-norm_q_e_norm_square*Partial_Omega_wrt_partial_n_1e.'*A_2d.'*T_alpha_2_inverse_transpose*R_2_alpha_q_transpose*R_2_alpha_q*T_alpha_2_inverse;

Partial_Q_sum_1_to_9_wrt_n_1e = (Q_1+(Q_2+Q_4+Q_5+Q_7+Q_9)*A_2d)* Partial_Omega_wrt_partial_n_1e;
Partial_Q_sum_1_to_9_wrt_v_1_hat = (Q_1+(Q_2+Q_4+Q_5+Q_7+Q_9)*A_2d) * Partial_Omega_wrt_partial_v_1_hat;
Partial_Q_sum_1_to_9_wrt_n_1d_dot = (Q_1+(Q_2+Q_4+Q_5+Q_7+Q_9)*A_2d )* Partial_Omega_wrt_partial_n_1d_dot;
Partial_Q_sum_1_to_9_wrt_n_1d_double_dot = (Q_1+(Q_2+Q_4+Q_5+Q_7+Q_9)*A_2d) * Partial_Omega_wrt_partial_n_1d_double_dot;
Partial_Q_sum_1_to_9_wrt_alpha_psi = Q_3+(Q_2+Q_4+Q_5+Q_7+Q_9)*([0;0;1]+A_1d*Omega);
Partial_Q_sum_1_to_9_wrt_q = Q_6+Q_8;

Partial_h_4_wrt_n_1e = Partial_Partial_Omega_wrt_partial_n_1e_transpose_wrt_n_1e + Partial_Q_sum_1_to_9_wrt_n_1e ;
Partial_h4_wrt_partial_n_1d_dot = Partial_Partial_Omega_wrt_partial_n_1e_transpose_wrt_v_1_hat + Partial_Q_sum_1_to_9_wrt_v_1_hat;
Partial_h4_wrt_partial_v_1_hat = Partial_Partial_Omega_wrt_partial_n_1e_transpose_wrt_n_1d_dot + Partial_Q_sum_1_to_9_wrt_n_1d_dot;
Partial_h4_wrt_partial_n_1d_double_dot = Partial_Q_sum_1_to_9_wrt_n_1d_double_dot ;
Partial_h_4_wrt_q = Partial_Q_sum_1_to_9_wrt_q ;
Partial_h_4_wrt_alpha_psi = Partial_Q_sum_1_to_9_wrt_alpha_psi;

constant = 1+0*2*gamma_1;
Partial_h_4_wrt_n_1e = (-1*Partial_h_4_wrt_n_1e /L2_to_L3)*constant;
Partial_h4_wrt_partial_n_1d_dot = (-1*Partial_h4_wrt_partial_n_1d_dot/L2_to_L3) *constant;
Partial_h4_wrt_partial_v_1_hat = (-1*Partial_h4_wrt_partial_v_1_hat/L2_to_L3)*constant;
Partial_h4_wrt_partial_n_1d_double_dot = (-1*Partial_h4_wrt_partial_n_1d_double_dot /L2_to_L3)*constant;
Partial_h_4_wrt_q = (-1*Partial_h_4_wrt_q/L2_to_L3)*constant;
Partial_h_4_wrt_alpha_psi = (-1*Partial_h_4_wrt_alpha_psi/L2_to_L3)*constant;
end
```

```
%% CALC partial for h_4 and second derivtives of M terms
clear
clc
syms  v_1e_1  v_1e_2  v_1e_3  n_1e_dot_1  n_1e_dot_2  v_1e_3  v_1_hat_dot_1
v_1_hat_dot_2    v_1_hat_dot_3       k_1    k_2    k_3    gamma_1    epsilon_2
n_1d_double_dot_1 n_1d_double_dot_2 n_1d_double_dot_3 n_1e_dot_3  v_1_hat_1
v_1_hat_2  v_1_hat_3  n_1e_1  n_1e_2  n_1e_3  min_eigen_value_D_1  vector_1
vector_3 vector_2 m n_1d_dot_1 n_1d_dot_2 n_1d_dot_3
vector = [vector_1;vector_2;vector_3];

K_1 = [k_1 0 0; 0 k_2 0; 0 0 k_3];
v_1_hat_dot =[ v_1_hat_dot_1;
                  v_1_hat_dot_2;
                  v_1_hat_dot_3];
n_1e_dot = [n_1e_dot_1; n_1e_dot_2; n_1e_dot_3];
v_1_hat =[ v_1_hat_1;
                  v_1_hat_2;
                  v_1_hat_3];
n_1e = [n_1e_1; n_1e_2; n_1e_3];
n_1d_dot = [n_1d_dot_1;n_1d_dot_2;n_1d_dot_3];

n_1d_double_dot = [n_1d_double_dot_1;n_1d_double_dot_2;n_1d_double_dot_3];
v_1e = [v_1e_1; v_1e_2; v_1e_3];

norm_v_1e = (v_1e.'*v_1e)^0.5;
sigma_n_1e= [ n_1e(1,1)/(1+n_1e(1,1)^2)^0.5;
                         n_1e(2,1)/(1+n_1e(2,1)^2)^0.5;
                         n_1e(3,1)/(1+n_1e(3,1)^2)^0.5];
sigma_dash_n_1e = [ 1/(1+n_1e(1,1)^2)^1.5, 0, 0;
                                   0, 1/(1+n_1e(2,1)^2)^1.5, 0;
                                   0, 0, 1/(1+n_1e(3,1)^2)^1.5];
sigma_double_dash_n_1e = -3*[ 1/(1+n_1e(1,1)^2)^2.5, 0, 0;
                                                  0,
1/(1+n_1e(2,1)^2)^2.5, 0;
                                                  0,                      0,
1/(1+n_1e(3,1)^2)^2.5];
sigma_n_1e_squared =sigma_n_1e.'*sigma_n_1e;
delta_1_v_1_hat = 1 + 0.5 * ((v_1_hat.'))*v_1_hat;
norm_sigma_n_1e = (sigma_n_1e.'*sigma_n_1e)^0.5;
G_1 = eye(3)-(K_1*sigma_n_1e*(v_1_hat.')/delta_1_v_1_hat^2);


alpha_1 = -K_1*sigma_n_1e/delta_1_v_1_hat + n_1d_dot;

M_3                                            =  -sigma_n_1e*gamma_1 * 0.5
* min_eigen_value_D_1*(2+alpha_1.'*alpha_1)/delta_1_v_1_hat/2;
Partial_M3_by_partial_v_1_hat         =   -M_3/delta_1_v_1_hat*v_1_hat.'   -
sigma_n_1e*gamma_1     *     0.5     *     min_eigen_value_D_1*(2*alpha_1.'*
(K_1*sigma_n_1e*(v_1_hat.')/delta_1_v_1_hat^2) )/delta_1_v_1_hat/2;
Partial_M3_by_partial_n_1e             = -sigma_dash_n_1e*gamma_1  *  0.5  *
min_eigen_value_D_1*(2+alpha_1.'*alpha_1)/delta_1_v_1_hat/2                +
sigma_n_1e*gamma_1                *                  0.5                   *
min_eigen_value_D_1*(2*alpha_1.'*K_1*sigma_dash_n_1e/delta_1_v_1_hat)/delta
_1_v_1_hat/2;
Partial_M3_by_partial_n_1d_dot     =    -sigma_n_1e*gamma_1    *    0.5    *
min_eigen_value_D_1*(2*alpha_1.')/delta_1_v_1_hat/2;


Partial_M3_by_partial_v_1_hat_dot       =  Partial_M3_by_partial_v_1_hat   *
v_1_hat_dot;
```

```
Partial_M3_by_partial_n_1e_dot              = Partial_M3_by_partial_n_1e  *
n_1d_dot ;
Partial_M3_by_partial_n_1d_dot_dot  =  Partial_M3_by_partial_n_1d_dot     *
n_1d_double_dot ;

Z = Partial_M3_by_partial_n_1e.'*vector
Z= simplify(Z);

started = 1
%

Y_4_1_1 = jacobian(Partial_M3_by_partial_v_1_hat_dot , v_1_hat); %+ Y_4_1_0
*G_1;
Y_4_1_1 = simplify(Y_4_1_1 )
Y_4_1_2  = jacobian(Partial_M3_by_partial_v_1_hat_dot , n_1e) ;%+ Y_4_1_0
*K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Y_4_1_2 = simplify(Y_4_1_2 )
Y_4_1_3   =  jacobian(Partial_M3_by_partial_v_1_hat_dot   ,   n_1d_dot)   ;%+
Y_4_1_0 *K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Y_4_1_3 = simplify(Y_4_1_3)

Y_4_1_3 = 0*eye(3);
Y_4_2_0 = jacobian(Partial_M3_by_partial_n_1e_dot , v_1e);
Y_4_2_0 = simplify(Y_4_2_0 )
Y_4_2_1 = jacobian(Partial_M3_by_partial_n_1e_dot , v_1_hat);
Y_4_2_1 = simplify(Y_4_2_1 )
Y_4_2_2 = jacobian(Partial_M3_by_partial_n_1e_dot , n_1e);
Y_4_2_2 = simplify(Y_4_2_2 )
Y_4_2_3  = jacobian(Partial_M3_by_partial_n_1e_dot , n_1d_dot) ;%+ Y_4_1_0
*K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Y_4_2_3 = simplify(Y_4_2_3 )

Y_4_3_1 = jacobian(Partial_M3_by_partial_n_1d_dot_dot , v_1_hat);
Y_4_3_1 = simplify(Y_4_3_1 )
Y_4_3_2 = jacobian(Partial_M3_by_partial_n_1d_dot_dot , n_1e);
Y_4_3_2 = simplify(Y_4_3_2 )
Y_4_3_3 =0*eye(3);
Y_4_3_3   =  jacobian(Partial_M3_by_partial_n_1d_dot_dot   ,   n_1d_dot)   ;%+
Y_4_1_0 *K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Y_4_3_3 = simplify(Y_4_3_3 )
%}

Started_Z_1=1
Z_4_1 =jacobian(Z , v_1_hat);
Z_4_1 =simplify(Z_4_1 )
Z_4_2 =jacobian(Z , n_1e);
Z_4_2 =simplify(Z_4_2 )
Z_4_3 =jacobian(Z , n_1d_dot);
Z_4_3 =simplify(Z_4_2 )




%}
DONE = 1


syms  v_1e_1  v_1e_2  v_1e_3  n_1e_dot_1  n_1e_dot_2  v_1e_3  v_1_hat_dot_1
v_1_hat_dot_2    v_1_hat_dot_3        k_1    k_2    k_3    gamma_1    epsilon_2
n_1d_double_dot_1 n_1d_double_dot_2 n_1d_double_dot_3 n_1e_dot_3  v_1_hat_1
```

```
v_1_hat_2  v_1_hat_3  n_1e_1  n_1e_2  n_1e_3  min_eigen_value_D_1  vector_1
vector_3 vector_2 m c n_1d_dot_1 n_1d_dot_2 n_1d_dot_3
vector = [vector_1;vector_2;vector_3];

K_1 = [k_1 0 0; 0 k_2 0; 0 0 k_3];
v_1_hat_dot =[ v_1_hat_dot_1;
                  v_1_hat_dot_2;
                  v_1_hat_dot_3];
n_1e_dot = [n_1e_dot_1; n_1e_dot_2; n_1e_dot_3];
v_1_hat =[ v_1_hat_1;
                  v_1_hat_2;
                  v_1_hat_3];
n_1e = [n_1e_1; n_1e_2; n_1e_3];
n_1d_dot = [n_1d_dot_1;n_1d_dot_2;n_1d_dot_3];
n_1d_double_dot = [n_1d_double_dot_1;n_1d_double_dot_2;n_1d_double_dot_3];
v_1e = [v_1e_1; v_1e_2; v_1e_3];

norm_v_1e = (v_1e.'*v_1e)^0.5;
sigma_n_1e= [ n_1e(1,1)/(1+n_1e(1,1)^2)^0.5;
                         n_1e(2,1)/(1+n_1e(2,1)^2)^0.5;
                         n_1e(3,1)/(1+n_1e(3,1)^2)^0.5];
sigma_dash_n_1e = [ 1/(1+n_1e(1,1)^2)^1.5, 0, 0;
                                   0, 1/(1+n_1e(2,1)^2)^1.5, 0;
                                   0, 0, 1/(1+n_1e(3,1)^2)^1.5];
sigma_double_dash_n_1e = -3*[ 1/(1+n_1e(1,1)^2)^2.5, 0, 0;
                                                   0,
1/(1+n_1e(2,1)^2)^2.5, 0;
                                                   0,                      0,
1/(1+n_1e(3,1)^2)^2.5];
sigma_n_1e_squared =sigma_n_1e.'*sigma_n_1e;
delta_1_v_1_hat = 1 + 0.5 * ((v_1_hat.'))*v_1_hat;
norm_sigma_n_1e = (sigma_n_1e.'*sigma_n_1e)^0.5;
G_1 = eye(3)-(K_1*sigma_n_1e*(v_1_hat.')/delta_1_v_1_hat^2);

alpha_1 = -K_1*sigma_n_1e/delta_1_v_1_hat +n_1d_dot
%{
temp = c*(2*alpha_1+v_1e)/(delta_1_v_1_hat)*(1+v_1e.'*v_1e)^-0.5*v_1e.';
Partial_M5_by_partial_v_1_hat                                              =
(c)*1/(delta_1_v_1_hat)*((eye(3)+K_1*(sigma_n_1e*(v_1_hat.'))/(delta_1_v_1_
hat)^2)-(2*alpha_1+v_1e)/(delta_1_v_1_hat)*(v_1_hat.'));
Partial_M5_by_partial_n_1e                                                 =
(c)*1/(delta_1_v_1_hat)*(K_1*(sigma_dash_n_1e)/(delta_1_v_1_hat));
Partial_M5_by_partial_n_1d_dot = (c)*1/(delta_1_v_1_hat)*eye(3);

Partial_M5_by_partial_v_1_hat            =    Partial_M5_by_partial_v_1_hat
*(1+v_1e.'*v_1e)^0.5 +temp*G_1;
Partial_M5_by_partial_n_1e                    =   Partial_M5_by_partial_n_1e
*(1+v_1e.'*v_1e)^0.5 + temp*K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Partial_M5_by_partial_n_1d_dot       =       Partial_M5_by_partial_n_1d_dot
*(1+v_1e.'*v_1e)^0.5-temp;




Partial_M5_by_partial_v_1_hat_dot      =      Partial_M5_by_partial_v_1_hat
*v_1_hat_dot;
Partial_M5_by_partial_n_1e_dot               =   Partial_M5_by_partial_n_1e
*n_1e_dot;
Partial_M5_by_partial_n_1d_dot_dot     =     Partial_M5_by_partial_n_1d_dot
*n_1d_double_dot;
```

```
%}
%{
M_5                                                                          =
c*(2*alpha_1+v_1e)/(delta_1_v_1_hat)*(1+v_1e.'*v_1e)^0.5;
temp = c*(2*alpha_1+v_1e)/(delta_1_v_1_hat)*(1+v_1e.'*v_1e)^-0.5*v_1e.';
Partial_M5_by_partial_v_1_hat                                                =
(c)*1/(delta_1_v_1_hat)*((eye(3)+K_1*(sigma_n_1e*(v_1_hat.'))/(delta_1_v_1_
hat)^2)-(2*alpha_1+v_1e)/(delta_1_v_1_hat)*(v_1_hat.'));
Partial_M5_by_partial_n_1e                                                   =
(c)*1/(delta_1_v_1_hat)*(K_1*(sigma_dash_n_1e)/(delta_1_v_1_hat));
Partial_M5_by_partial_n_1d_dot = (c)*1/(delta_1_v_1_hat)*eye(3);

Partial_M5_by_partial_v_1_hat            =    Partial_M5_by_partial_v_1_hat
*(1+v_1e.'*v_1e)^0.5 +temp*G_1;
Partial_M5_by_partial_n_1e                         =   Partial_M5_by_partial_n_1e
*(1+v_1e.'*v_1e)^0.5 + temp*K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Partial_M5_by_partial_n_1d_dot       =       Partial_M5_by_partial_n_1d_dot
*(1+v_1e.'*v_1e)^0.5-temp;

Partial_M5_by_partial_v_1_hat_dot      =      Partial_M5_by_partial_v_1_hat
*v_1_hat_dot;
Partial_M5_by_partial_n_1e_dot                    =   Partial_M5_by_partial_n_1e
*n_1e_dot;
Partial_M5_by_partial_n_1d_dot_dot      =     Partial_M5_by_partial_n_1d_dot
*n_1d_double_dot;
%}


M_5                                                                          =
c*(2*alpha_1+v_1e)/(delta_1_v_1_hat)*(1+v_1e.'*v_1e)^0.5;
temp = c*(2*alpha_1+v_1e)/(delta_1_v_1_hat)*(1+v_1e.'*v_1e)^-0.5*v_1e.';
Partial_M5_by_partial_v_1_hat                                                =
(c)*1/(delta_1_v_1_hat)*((eye(3)+K_1*(sigma_n_1e*(v_1_hat.'))/(delta_1_v_1_
hat)^2)-(2*alpha_1+v_1e)/(delta_1_v_1_hat)*(v_1_hat.'));
Partial_M5_by_partial_n_1e                                                   =
(c)*1/(delta_1_v_1_hat)*(K_1*(sigma_dash_n_1e)/(delta_1_v_1_hat));
Partial_M5_by_partial_n_1d_dot = (c)*1/(delta_1_v_1_hat)*eye(3);

Partial_M5_by_partial_v_1_hat            =    Partial_M5_by_partial_v_1_hat
*(1+v_1e.'*v_1e)^0.5 +temp*G_1;
Partial_M5_by_partial_n_1e                         =   Partial_M5_by_partial_n_1e
*(1+v_1e.'*v_1e)^0.5 + temp*K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Partial_M5_by_partial_n_1d_dot       =       Partial_M5_by_partial_n_1d_dot
*(1+v_1e.'*v_1e)^0.5-temp;

Partial_M5_by_partial_v_1_hat_dot      =      Partial_M5_by_partial_v_1_hat
*v_1_hat_dot;
Partial_M5_by_partial_n_1e_dot                    =   Partial_M5_by_partial_n_1e
*n_1e_dot;
Partial_M5_by_partial_n_1d_dot_dot      =     Partial_M5_by_partial_n_1d_dot
*n_1d_double_dot;



Z = Partial_M5_by_partial_n_1e*vector
Z= simplify(Z);
started = 1
Y_6_1_0 = jacobian(Partial_M5_by_partial_v_1_hat_dot , v_1e);
Y_6_1_0 = simplify(Y_6_1_0 )
```

```
Y_6_1_1 = jacobian(Partial_M5_by_partial_v_1_hat_dot , v_1_hat); %+ Y_6_1_0
*G_1;
Y_6_1_1 = simplify(Y_6_1_1 )
Y_6_1_2  =  jacobian(Partial_M5_by_partial_v_1_hat_dot  ,  n_1e)  ;%+  Y_6_1_0
*K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Y_6_1_2 = simplify(Y_6_1_2 )
Y_6_1_3 = 0*eye(3);
Y_6_2_0 = jacobian(Partial_M5_by_partial_n_1e_dot , v_1e);
Y_6_2_0 = simplify(Y_6_2_0 )
Y_6_2_1 = jacobian(Partial_M5_by_partial_n_1e_dot , v_1_hat);
Y_6_2_1 = simplify(Y_6_2_1 )
Y_6_2_2 = jacobian(Partial_M5_by_partial_n_1e_dot , n_1e);
Y_6_2_2 = simplify(Y_6_2_2 )
Y_6_2_3 = 0*eye(3);
Y_6_3_0 = jacobian(Partial_M5_by_partial_n_1d_dot_dot , v_1e);
Y_6_3_0 = simplify(Y_6_3_0 )
Y_6_3_1 = jacobian(Partial_M5_by_partial_n_1d_dot_dot , v_1_hat);
Y_6_3_1 = simplify(Y_6_3_1 )
Y_6_3_2 = jacobian(Partial_M5_by_partial_n_1d_dot_dot , n_1e);
Y_6_3_2 = simplify(Y_6_3_2 )
Y_6_3_3 =0*eye(3);

Started_Z_1 = 1
Z_6_0 =jacobian(Z , v_1e);
Z_6_0 =simplify(Z_6_0 )
Z_6_1 =jacobian(Z , v_1_hat);
Z_6_1 =simplify(Z_6_1 )
Z_6_2 =jacobian(Z , n_1e);
Z_6_2 =simplify(Z_6_2 )
Z_6_3 =jacobian(Z , n_1d_dot);
Z_6_3 =simplify(Z_6_3 )
DONE = 1


syms  v_1e_1  v_1e_2  v_1e_3  n_1e_dot_1  n_1e_dot_2  v_1e_3  v_1_hat_dot_1
v_1_hat_dot_2    v_1_hat_dot_3       k_1    k_2    k_3    gamma_1    epsilon_2
n_1d_double_dot_1 n_1d_double_dot_2 n_1d_double_dot_3 n_1e_dot_3  v_1_hat_1
v_1_hat_2  v_1_hat_3  n_1e_1  n_1e_2  n_1e_3  min_eigen_value_D_1  vector_1
vector_3 vector_2 m

vector = [vector_1;vector_2;vector_3];

K_1 = [k_1 0 0; 0 k_2 0; 0 0 k_3];
v_1_hat_dot =[ v_1_hat_dot_1;
                  v_1_hat_dot_2;
                  v_1_hat_dot_3];
n_1e_dot = [n_1e_dot_1; n_1e_dot_2; n_1e_dot_3];
v_1_hat =[ v_1_hat_1;
                  v_1_hat_2;
                  v_1_hat_3];
n_1e = [n_1e_1; n_1e_2; n_1e_3];
n_1d_double_dot = [n_1d_double_dot_1;n_1d_double_dot_2;n_1d_double_dot_3];
v_1e = [v_1e_1; v_1e_2; v_1e_3];

norm_v_1e = (v_1e.'*v_1e)^0.5;
sigma_n_1e= [ n_1e(1,1)/(1+n_1e(1,1)^2)^0.5;
                         n_1e(2,1)/(1+n_1e(2,1)^2)^0.5;
                         n_1e(3,1)/(1+n_1e(3,1)^2)^0.5];
sigma_dash_n_1e = [ 1/(1+n_1e(1,1)^2)^1.5, 0, 0;
                                   0, 1/(1+n_1e(2,1)^2)^1.5, 0;
                                   0, 0, 1/(1+n_1e(3,1)^2)^1.5];
```

```
sigma_double_dash_n_1e = -3*[ 1/(1+n_1e(1,1)^2)^2.5, 0, 0;
                                                    0,
1/(1+n_1e(2,1)^2)^2.5, 0;
                                                    0,                      0,
1/(1+n_1e(3,1)^2)^2.5];
sigma_n_1e_squared =sigma_n_1e.'*sigma_n_1e;
delta_1_v_1_hat = 1 + 0.5 * ((v_1_hat.'))*v_1_hat;
norm_sigma_n_1e = (sigma_n_1e.'*sigma_n_1e)^0.5;
G_1 = eye(3)-(K_1*sigma_n_1e*(v_1_hat.')/delta_1_v_1_hat^2);

%{

Partial_M7_by_partial_v_1e                                             = -
m*(K_1*sigma_dash_n_1e)/(delta_1_v_1_hat)*(eye(3)                        -
v_1e*(v_1e.')/(1+v_1e.'*v_1e))/(1+v_1e.'*v_1e)^0.5;
Partial_M7_by_partial_v_1_hat                                            =
m*(K_1*sigma_dash_n_1e)/(delta_1_v_1_hat)*((v_1e/(1+v_1e.'*v_1e)^0.5)*(v_1_
hat.'))/delta_1_v_1_hat +Partial_M7_by_partial_v_1e    *G_1;
Partial_M7_by_partial_n_1e                                               =
Partial_M7_by_partial_v_1e  *  (K_1*sigma_dash_n_1e)/(delta_1_v_1_hat)  -m*(
(K_1*sigma_double_dash_n_1e)/(delta_1_v_1_hat)*(1/(1+v_1e.'*v_1e)^0.5)*[v_1
e(1,1),0,0;

0,v_1e(2,1),0;

0,0,v_1e(3,1)]);
Partial_M7_by_partial_n_1d_dot                                          =  -
Partial_M7_by_partial_v_1e     ;

Partial_M7_by_partial_v_1_hat       =  gamma_1*Partial_M7_by_partial_v_1_hat
;
Partial_M7_by_partial_n_1e = gamma_1*Partial_M7_by_partial_n_1e;
Partial_M7_by_partial_n_1d_dot  = gamma_1*Partial_M7_by_partial_n_1d_dot ;

Partial_M7_by_partial_v_1_hat_dot      =      Partial_M7_by_partial_v_1_hat
*v_1_hat_dot;
Partial_M7_by_partial_n_1e_dot               =   Partial_M7_by_partial_n_1e
*n_1e_dot;
Partial_M7_by_partial_n_1d_dot_dot     =     Partial_M7_by_partial_n_1d_dot
*n_1d_double_dot;
%}
%{
Partial_M7_by_partial_v_1e                                             = -
m*(K_1*sigma_dash_n_1e)/(delta_1_v_1_hat)*(eye(3)                        -
v_1e*(v_1e.')/(1+v_1e.'*v_1e))/(1+v_1e.'*v_1e)^0.5;
Partial_M7_by_partial_v_1_hat                                            =
m*(K_1*sigma_dash_n_1e)/(delta_1_v_1_hat)*((v_1e/(1+v_1e.'*v_1e)^0.5)*(v_1_
hat.'))/delta_1_v_1_hat +Partial_M7_by_partial_v_1e    *G_1;
Partial_M7_by_partial_n_1e                                               =
Partial_M7_by_partial_v_1e  *  (K_1*sigma_dash_n_1e)/(delta_1_v_1_hat)  -m*(
(K_1*sigma_double_dash_n_1e)/(delta_1_v_1_hat)*(1/(1+v_1e.'*v_1e)^0.5)*[v_1
e(1,1),0,0;

0,v_1e(2,1),0;

0,0,v_1e(3,1)]);
Partial_M7_by_partial_n_1d_dot                                          =  -
Partial_M7_by_partial_v_1e     ;

Partial_M7_by_partial_v_1_hat     = 2*gamma_1*Partial_M7_by_partial_v_1_hat
;
```

```
Partial_M7_by_partial_n_1e = 2*gamma_1*Partial_M7_by_partial_n_1e;
Partial_M7_by_partial_n_1d_dot   = 2*gamma_1*Partial_M7_by_partial_n_1d_dot
;

%}


Partial_M7_by_partial_v_1e                                                 = -
m*(K_1*sigma_dash_n_1e)/(delta_1_v_1_hat)*(eye(3)                            -
v_1e*(v_1e.')/(1+v_1e.'*v_1e))/(1+v_1e.'*v_1e)^0.5;
Partial_M7_by_partial_v_1_hat                                                =
m*(K_1*sigma_dash_n_1e)/(delta_1_v_1_hat)*((v_1e/(1+v_1e.'*v_1e)^0.5)*(v_1_
hat.'))/delta_1_v_1_hat +Partial_M7_by_partial_v_1e    *G_1;
Partial_M7_by_partial_n_1e                                                   =
Partial_M7_by_partial_v_1e  *  (K_1*sigma_dash_n_1e)/(delta_1_v_1_hat)  -m*(
(K_1*sigma_double_dash_n_1e)/(delta_1_v_1_hat)*(1/(1+v_1e.'*v_1e)^0.5)*[v_1
e(1,1),0,0;

0,v_1e(2,1),0;

0,0,v_1e(3,1)]);
Partial_M7_by_partial_n_1d_dot                                             =  -
Partial_M7_by_partial_v_1e     ;

Partial_M7_by_partial_v_1_hat     = 2*gamma_1*Partial_M7_by_partial_v_1_hat
;
Partial_M7_by_partial_n_1e = 2*gamma_1*Partial_M7_by_partial_n_1e;
Partial_M7_by_partial_n_1d_dot   = 2*gamma_1*Partial_M7_by_partial_n_1d_dot
;




Partial_M7_by_partial_v_1_hat_dot      =      Partial_M7_by_partial_v_1_hat
*v_1_hat_dot;
Partial_M7_by_partial_n_1e_dot               =   Partial_M7_by_partial_n_1e
*n_1e_dot;
Partial_M7_by_partial_n_1d_dot_dot     =     Partial_M7_by_partial_n_1d_dot
*n_1d_double_dot;



Z = Partial_M7_by_partial_n_1e.'*vector
Z= simplify(Z);

started = 1
%
Y_8_1_0 = jacobian(Partial_M7_by_partial_v_1_hat_dot , v_1e);
Y_8_1_0 = simplify(Y_8_1_0 )
Y_8_1_1 = jacobian(Partial_M7_by_partial_v_1_hat_dot , v_1_hat); %+ Y_8_1_0
*G_1;
Y_8_1_1 = simplify(Y_8_1_1 )
Y_8_1_2  =  jacobian(Partial_M7_by_partial_v_1_hat_dot  ,  n_1e)  ;%+  Y_8_1_0
*K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Y_8_1_2 = simplify(Y_8_1_2 )
Y_8_1_3 = 0*eye(3);
Y_8_2_0 = jacobian(Partial_M7_by_partial_n_1e_dot , v_1e);
Y_8_2_0 = simplify(Y_8_2_0 )
Y_8_2_1 = jacobian(Partial_M7_by_partial_n_1e_dot , v_1_hat);
```

```
Y_8_2_1 = simplify(Y_8_2_1 )
Y_8_2_2 = jacobian(Partial_M7_by_partial_n_1e_dot , n_1e);
Y_8_2_2 = simplify(Y_8_2_2 )
Y_8_2_3 = 0*eye(3);
Y_8_3_0 = jacobian(Partial_M7_by_partial_n_1d_dot_dot , v_1e);
Y_8_3_0 = simplify(Y_8_3_0 )
Y_8_3_1 = jacobian(Partial_M7_by_partial_n_1d_dot_dot , v_1_hat);
Y_8_3_1 = simplify(Y_8_3_1 )
Y_8_3_2 = jacobian(Partial_M7_by_partial_n_1d_dot_dot , n_1e);
Y_8_3_2 = simplify(Y_8_3_2 )
Y_8_3_3 =0*eye(3);
%}

Started_Z_1=1
Z_8_0 =jacobian(Z , v_1e);
Z_8_0 =simplify(Z_8_0 )
Z_8_1 =jacobian(Z , v_1_hat);
Z_8_1 =simplify(Z_8_1 )
Z_8_2 =jacobian(Z , n_1e);
Z_8_2 =simplify(Z_8_2 )
%}
DONE = 1


syms  v_1e_1  v_1e_2  v_1e_3  n_1e_dot_1  n_1e_dot_2  v_1e_3  v_1_hat_dot_1
v_1_hat_dot_2    v_1_hat_dot_3       k_1    k_2    k_3    gamma_1    epsilon_2
n_1d_double_dot_1 n_1d_double_dot_2 n_1d_double_dot_3 n_1e_dot_3  v_1_hat_1
v_1_hat_2  v_1_hat_3  n_1e_1  n_1e_2  n_1e_3  min_eigen_value_D_1  vector_1
vector_3 vector_2 m

vector = [vector_1;vector_2;vector_3];


K_1 = [k_1 0 0; 0 k_2 0; 0 0 k_3];
v_1_hat_dot =[ v_1_hat_dot_1;
                  v_1_hat_dot_2;
                  v_1_hat_dot_3];
n_1e_dot = [n_1e_dot_1; n_1e_dot_2; n_1e_dot_3];
v_1_hat =[ v_1_hat_1;
                  v_1_hat_2;
                  v_1_hat_3];
n_1e = [n_1e_1; n_1e_2; n_1e_3];
n_1d_double_dot = [n_1d_double_dot_1;n_1d_double_dot_2;n_1d_double_dot_3];
v_1e = [v_1e_1; v_1e_2; v_1e_3];

norm_v_1e = (v_1e.'*v_1e)^0.5;
sigma_n_1e= [ n_1e(1,1)/(1+n_1e(1,1)^2)^0.5;
                         n_1e(2,1)/(1+n_1e(2,1)^2)^0.5;
                         n_1e(3,1)/(1+n_1e(3,1)^2)^0.5];
sigma_dash_n_1e = [ 1/(1+n_1e(1,1)^2)^1.5, 0, 0;
                                   0, 1/(1+n_1e(2,1)^2)^1.5, 0;
                                   0, 0, 1/(1+n_1e(3,1)^2)^1.5];
sigma_n_1e_squared =sigma_n_1e.'*sigma_n_1e;
delta_1_v_1_hat = 1 + 0.5 * ((v_1_hat.'))*v_1_hat;
norm_sigma_n_1e = (sigma_n_1e.'*sigma_n_1e)^0.5;
G_1 = eye(3)-(K_1*sigma_n_1e*(v_1_hat.')/delta_1_v_1_hat^2);
%{
Partial_M2_by_partial_v_1e    =    -(gamma_1^2*epsilon_2/min_eigen_value_D_1)
*sigma_n_1e_squared       *               ((1+1/(1+v_1e.'*v_1e)^0.5)*(eye(3)-
(v_1e*v_1e.')/(1+norm_v_1e^2))/(1+norm_v_1e^2)^0.5                          -
v_1e*v_1e.'/(1+v_1e.'*v_1e)^1.5 );
```

```
Partial_M2_by_partial_v_1_hat          =     Partial_M2_by_partial_v_1e*G_1-
gamma_1*K_1*sigma_n_1e*sigma_n_1e.'*(eye(3)                                  -
2*(v_1_hat*v_1_hat.')/delta_1_v_1_hat)/delta_1_v_1_hat^2;
Partial_M2_by_partial_n_1e
=Partial_M2_by_partial_v_1e*K_1*sigma_dash_n_1e/delta_1_v_1_hat            -
gamma_1*K_1*(sigma_dash_n_1e*(v_1_hat.'*sigma_n_1e)+sigma_n_1e*v_1_hat.'*si
gma_dash_n_1e)/delta_1_v_1_hat^2                                             -
(gamma_1^2*epsilon_2/min_eigen_value_D_1)                                    *
(1+1/(1+v_1e.'*v_1e)^0.5)*v_1e/(1+v_1e.'*v_1e)^0.5*(2*sigma_n_1e.'*sigma_da
sh_n_1e);
Partial_M2_by_partial_n_1d_dot = -Partial_M2_by_partial_v_1e;


Partial_M2_by_partial_v_1_hat_dot      =      Partial_M2_by_partial_v_1_hat
*v_1_hat_dot;
Partial_M2_by_partial_n_1e_dot               =   Partial_M2_by_partial_n_1e
*n_1e_dot;
Partial_M2_by_partial_n_1d_dot_dot     =     Partial_M2_by_partial_n_1d_dot
*n_1d_double_dot;


Partial_M2_by_partial_v_1_hat          =     Partial_M2_by_partial_v_1e*G_1-
gamma_1*K_1*(sigma_n_1e*sigma_n_1e.')*(eye(3)                                -
2*(v_1_hat*v_1_hat.')/delta_1_v_1_hat)/delta_1_v_1_hat^2;
Partial_M2_by_partial_n_1e
=Partial_M2_by_partial_v_1e*K_1*sigma_dash_n_1e/delta_1_v_1_hat            -
gamma_1*K_1*(sigma_dash_n_1e*(v_1_hat.'*sigma_n_1e)+sigma_n_1e*v_1_hat.'*si
gma_dash_n_1e)/delta_1_v_1_hat^2                                             -
(gamma_1^2*epsilon_2*min_eigen_value_D_1/4)                                  *
(v_1e/(1+v_1e.'*v_1e))*(2*sigma_n_1e.'*sigma_dash_n_1e);
Partial_M2_by_partial_n_1d_dot = -Partial_M2_by_partial_v_1e;
%}
%{
Partial_M2_by_partial_v_1e  =  -(gamma_1^2*epsilon_2*min_eigen_value_D_1/4)
*sigma_n_1e_squared                *                                 (eye(3)-
2*(v_1e*v_1e.')/(1+norm_v_1e^2))/(1+norm_v_1e^2);
Partial_M2_by_partial_v_1_hat          =     Partial_M2_by_partial_v_1e*G_1-
(gamma_1*K_1*min_eigen_value_D_1/4)*(sigma_n_1e*sigma_n_1e.')*(eye(3)      -
2*(v_1_hat*v_1_hat.')/delta_1_v_1_hat)/delta_1_v_1_hat^2;
Partial_M2_by_partial_n_1e
=Partial_M2_by_partial_v_1e*K_1*sigma_dash_n_1e/delta_1_v_1_hat            -
(gamma_1*min_eigen_value_D_1/4)*K_1*(sigma_dash_n_1e*(v_1_hat.'*sigma_n_1e)
+sigma_n_1e*v_1_hat.'*sigma_dash_n_1e)/delta_1_v_1_hat^2                     -
(gamma_1^2*epsilon_2*min_eigen_value_D_1/4)                                  *
(v_1e/(1+v_1e.'*v_1e))*(2*sigma_n_1e.'*sigma_dash_n_1e);
Partial_M2_by_partial_n_1d_dot = -Partial_M2_by_partial_v_1e;




Partial_M2_by_partial_v_1e  =  -(gamma_1^2*epsilon_2*min_eigen_value_D_1/4)
*sigma_n_1e_squared                *                                 (eye(3)-
2*(v_1e*v_1e.')/(1+norm_v_1e^2))/(1+norm_v_1e^2);
Partial_M2_by_partial_v_1e      =     Partial_M2_by_partial_v_1e             -
(gamma_1^2*epsilon_2*min_eigen_value_D_1/4)                        *(eye(3)-
v_1e*v_1e.'/(1+v_1e.'*v_1e))/(1+v_1e.'*v_1e) ^0.5;
Partial_M2_by_partial_v_1_hat          =     Partial_M2_by_partial_v_1e*G_1-
(gamma_1*K_1*min_eigen_value_D_1/4)*(sigma_n_1e*sigma_n_1e.')*(eye(3)      -
2*(v_1_hat*v_1_hat.')/delta_1_v_1_hat)/delta_1_v_1_hat^2;
```

```
Partial_M2_by_partial_n_1e
=Partial_M2_by_partial_v_1e*K_1*sigma_dash_n_1e/delta_1_v_1_hat          -
(gamma_1*min_eigen_value_D_1/4)*K_1*(sigma_dash_n_1e*(v_1_hat.'*sigma_n_1e)
+sigma_n_1e*v_1_hat.'*sigma_dash_n_1e)/delta_1_v_1_hat^2                 -
(gamma_1^2*epsilon_2*min_eigen_value_D_1/4)                              *
(v_1e/(1+v_1e.'*v_1e))*(2*sigma_n_1e.'*sigma_dash_n_1e);
Partial_M2_by_partial_n_1d_dot = -Partial_M2_by_partial_v_1e;



Partial_M2_by_partial_v_1_hat_dot      =      Partial_M2_by_partial_v_1_hat
*v_1_hat_dot;
Partial_M2_by_partial_n_1e_dot               =   Partial_M2_by_partial_n_1e
*n_1e_dot;
Partial_M2_by_partial_n_1d_dot_dot     =     Partial_M2_by_partial_n_1d_dot
*n_1d_double_dot;
%}

%{
M_2 = -(gamma_1^3*epsilon_2*min_eigen_value_D_1/4) * v_1e/(1+v_1e.'*v_1e) +
(gamma_1*min_eigen_value_D_1/2)*K_1*sigma_n_1e*v_1_hat.'*sigma_n_1e/delta_1
_v_1_hat^2;

Partial_M2_by_partial_v_1e = -(gamma_1^3*epsilon_2*min_eigen_value_D_1/4) *
(eye(3)-2*(v_1e*v_1e.')/(1+norm_v_1e^2))/(1+norm_v_1e^2);
Partial_M2_by_partial_v_1_hat          =     Partial_M2_by_partial_v_1e*G_1-
(gamma_1*K_1*min_eigen_value_D_1/2)*(sigma_n_1e*sigma_n_1e.')*(eye(3)     -
2*(v_1_hat*v_1_hat.')/delta_1_v_1_hat)/delta_1_v_1_hat^2;
Partial_M2_by_partial_n_1e
=Partial_M2_by_partial_v_1e*K_1*sigma_dash_n_1e/delta_1_v_1_hat-
(gamma_1*min_eigen_value_D_1/2)*K_1*(sigma_dash_n_1e*(v_1_hat.'*sigma_n_1e)
+sigma_n_1e*v_1_hat.'*sigma_dash_n_1e)/delta_1_v_1_hat^2  ;
Partial_M2_by_partial_n_1d_dot = -Partial_M2_by_partial_v_1e;
%}


Partial_M2_by_partial_v_1e = -(gamma_1^3*epsilon_2*min_eigen_value_D_1/4) *
(eye(3)-2*(v_1e*v_1e.')/(1+norm_v_1e^2))/(1+norm_v_1e^2);
Partial_M2_by_partial_v_1_hat          =     Partial_M2_by_partial_v_1e*G_1-
(gamma_1*K_1*min_eigen_value_D_1/2)*(sigma_n_1e*sigma_n_1e.')*(eye(3)     -
2*(v_1_hat*v_1_hat.')/delta_1_v_1_hat)/delta_1_v_1_hat^2;
Partial_M2_by_partial_n_1e
=Partial_M2_by_partial_v_1e*K_1*sigma_dash_n_1e/delta_1_v_1_hat-
(gamma_1*min_eigen_value_D_1/2)*K_1*(sigma_dash_n_1e*(v_1_hat.'*sigma_n_1e)
+sigma_n_1e*v_1_hat.'*sigma_dash_n_1e)/delta_1_v_1_hat^2  ;
Partial_M2_by_partial_n_1d_dot = -Partial_M2_by_partial_v_1e;




Partial_M2_by_partial_v_1_hat_dot      =      Partial_M2_by_partial_v_1_hat
*v_1_hat_dot;
Partial_M2_by_partial_n_1e_dot               =   Partial_M2_by_partial_n_1e
*n_1e_dot;
Partial_M2_by_partial_n_1d_dot_dot     =     Partial_M2_by_partial_n_1d_dot
*n_1d_double_dot;
```

```
Z = Partial_M2_by_partial_n_1e.'*vector;
Z= simplify(Z);


started = 1
Y_3_1_0 = jacobian(Partial_M2_by_partial_v_1_hat_dot , v_1e);
Y_3_1_0 = simplify(Y_3_1_0 )
Y_3_1_1 = jacobian(Partial_M2_by_partial_v_1_hat_dot , v_1_hat); %+ Y_3_1_0
*G_1;
Y_3_1_1 = simplify(Y_3_1_1 )
Y_3_1_2 = jacobian(Partial_M2_by_partial_v_1_hat_dot , n_1e) ;%+ Y_3_1_0
*K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Y_3_1_2 = simplify(Y_3_1_2 )
Y_3_1_3 = 0*eye(3);
Y_3_2_0 = jacobian(Partial_M2_by_partial_n_1e_dot , v_1e);
Y_3_2_0 = simplify(Y_3_2_0 )
Y_3_2_1 = jacobian(Partial_M2_by_partial_n_1e_dot , v_1_hat);
Y_3_2_1 = simplify(Y_3_2_1 )
Y_3_2_2 = jacobian(Partial_M2_by_partial_n_1e_dot , n_1e);
Y_3_2_2 = simplify(Y_3_2_2 )
Y_3_2_3 = 0*eye(3);
Y_3_3_0 = jacobian(Partial_M2_by_partial_n_1d_dot_dot , v_1e);
Y_3_3_0 = simplify(Y_3_3_0 )
Y_3_3_1 = jacobian(Partial_M2_by_partial_n_1d_dot_dot , v_1_hat);
Y_3_3_1 = simplify(Y_3_3_1 )
Y_3_3_2 = jacobian(Partial_M2_by_partial_n_1d_dot_dot , n_1e);
Y_3_3_2 = simplify(Y_3_3_2 )
Y_3_3_3 =0*eye(3);

Started_Z_1 =1

Z_3_0 =jacobian(Z , v_1e);
Z_3_0 =simplify(Z_3_0)
Z_3_1 =jacobian(Z , v_1_hat);
Z_3_1 =simplify(Z_3_1 )
Z_3_2 =jacobian(Z , n_1e);
Z_3_2 =simplify(Z_3_2 )
Z_3_3 =jacobian(Z , n_1d_dot);
Z_3_3 =simplify(Z_3_3 )
DONE = 1



syms v_1e_1 v_1e_2 v_1e_3 n_1e_dot_1 n_1e_dot_2 v_1e_3 v_1_hat_dot_1
v_1_hat_dot_2 v_1_hat_dot_3 k_1 k_2 k_3 gamma_1 epsilon_2
n_1d_double_dot_1 n_1d_double_dot_2 n_1d_double_dot_3 n_1e_dot_3 v_1_hat_1
v_1_hat_2 v_1_hat_3 n_1e_1 n_1e_2 n_1e_3 min_eigen_value_D_1 vector_1
vector_3 vector_2 m

vector = [vector_1;vector_2;vector_3];


K_1 = [k_1 0 0; 0 k_2 0; 0 0 k_3];
v_1_hat_dot =[ v_1_hat_dot_1;
                  v_1_hat_dot_2;
                  v_1_hat_dot_3];
n_1e_dot = [n_1e_dot_1; n_1e_dot_2; n_1e_dot_3];
v_1_hat =[ v_1_hat_1;
                  v_1_hat_2;
```

```
                  v_1_hat_3];
n_1e = [n_1e_1; n_1e_2; n_1e_3];
n_1d_double_dot = [n_1d_double_dot_1;n_1d_double_dot_2;n_1d_double_dot_3];
v_1e = [v_1e_1; v_1e_2; v_1e_3];

norm_v_1e = (v_1e.'*v_1e)^0.5;
sigma_n_1e= [ n_1e(1,1)/(1+n_1e(1,1)^2)^0.5;
                         n_1e(2,1)/(1+n_1e(2,1)^2)^0.5;
                         n_1e(3,1)/(1+n_1e(3,1)^2)^0.5];
sigma_dash_n_1e = [ 1/(1+n_1e(1,1)^2)^1.5, 0, 0;
                                   0, 1/(1+n_1e(2,1)^2)^1.5, 0;
                                   0, 0, 1/(1+n_1e(3,1)^2)^1.5];
sigma_double_dash_n_1e = -3*[ 1/(1+n_1e(1,1)^2)^2.5, 0, 0;
                                                   0,
1/(1+n_1e(2,1)^2)^2.5, 0;
                                                   0,                      0,
1/(1+n_1e(3,1)^2)^2.5];
sigma_n_1e_squared =sigma_n_1e.'*sigma_n_1e;
delta_1_v_1_hat = 1 + 0.5 * ((v_1_hat.'))*v_1_hat;
norm_sigma_n_1e = (sigma_n_1e.'*sigma_n_1e)^0.5;
G_1 = eye(3)-(K_1*sigma_n_1e*(v_1_hat.')/delta_1_v_1_hat^2);

Partial_M6_by_partial_v_1_hat                                                  =
K_1*sigma_dash_n_1e*(K_1*sigma_n_1e)/(delta_1_v_1_hat)^3*(-2*v_1_hat.');
Partial_M6_by_partial_n_1e                                                     =
K_1*sigma_dash_n_1e*(K_1*sigma_dash_n_1e)/(delta_1_v_1_hat)^2             +
(K_1/(delta_1_v_1_hat)^2)*sigma_double_dash_n_1e*K_1*[ sigma_n_1e(1,1) 0 0;

0, sigma_n_1e(2,1), 0;

0, 0, sigma_n_1e(3,1)];
Partial_M6_by_partial_n_1d_dot   = 0*eye(3);

Partial_M6_by_partial_v_1_hat_dot      =      Partial_M6_by_partial_v_1_hat
*v_1_hat_dot;
Partial_M6_by_partial_n_1e_dot               =   Partial_M6_by_partial_n_1e
*n_1e_dot;
Partial_M6_by_partial_n_1d_dot_dot     =     Partial_M6_by_partial_n_1d_dot
*n_1d_double_dot;

Z = Partial_M6_by_partial_n_1e.'*vector;
Z= simplify(Z);


started = 1
Y_7_1_0 = jacobian(Partial_M6_by_partial_v_1_hat_dot , v_1e);
Y_7_1_0 = simplify(Y_7_1_0 )
Y_7_1_1 = jacobian(Partial_M6_by_partial_v_1_hat_dot , v_1_hat); %+ Y_7_1_0
*G_1;
Y_7_1_1 = simplify(Y_7_1_1 )
Y_7_1_2  =  jacobian(Partial_M6_by_partial_v_1_hat_dot  ,  n_1e)  ;%+  Y_7_1_0
*K_1*sigma_dash_n_1e/delta_1_v_1_hat;
Y_7_1_2 = simplify(Y_7_1_2 )
Y_7_1_3 = 0*eye(3);
Y_7_2_0 = jacobian(Partial_M6_by_partial_n_1e_dot , v_1e);
Y_7_2_0 = simplify(Y_7_2_0 )
Y_7_2_1 = jacobian(Partial_M6_by_partial_n_1e_dot , v_1_hat);
Y_7_2_1 = simplify(Y_7_2_1 )
Y_7_2_2 = jacobian(Partial_M6_by_partial_n_1e_dot , n_1e);
Y_7_2_2 = simplify(Y_7_2_2 )
Y_7_2_3 = 0*eye(3);
```

```
Y_7_3_0 = jacobian(Partial_M6_by_partial_n_1d_dot_dot , v_1e);
Y_7_3_0 = simplify(Y_7_3_0 )
Y_7_3_1 = jacobian(Partial_M6_by_partial_n_1d_dot_dot , v_1_hat);
Y_7_3_1 = simplify(Y_7_3_1 )
Y_7_3_2 = jacobian(Partial_M6_by_partial_n_1d_dot_dot , n_1e);
Y_7_3_2 = simplify(Y_7_3_2 )
Y_7_3_3 =0*eye(3);

Started_Z_1 =1

Z_7_0 =jacobian(Z , v_1e);
Z_7_0 =simplify(Z_7_0)
Z_7_1 =jacobian(Z , v_1_hat);
Z_7_1 =simplify(Z_7_1 )
Z_7_2 =jacobian(Z , n_1e);
Z_7_2 =simplify(Z_7_2 )
Z_7_3 =jacobian(Z , n_1d_dot);
Z_7_3 =simplify(Z_7_3 )
DONE = 1
```

Appendix L –SparkFun 9DOF IMU Code.

The following code is a modified berssion of SparkFun 9DOF code which can be found on GitHub[42].

Razor_AHRS.ino

```
/***************************************************************************************
*************************
* Razor AHRS Firmware v1.4.2
* 9 Degree of Measurement Attitude and Heading Reference System
* for SparkFun "9DOF Razor IMU" (SEN-10125 and SEN-10736)
* and "9DOF Sensor Stick" (SEN-10183, 10321 and SEN-10724)
*
* Released under GNU GPL (General Public License) v3.0
* Copyright (C) 2013 Peter Bartz [http://ptrbrtz.net]
* Copyright (C) 2011-2012 Quality & Usability Lab, Deutsche Telekom Laboratories, TU
Berlin
*
* Infos, updates, bug reports, contributions and feedback:
*     https://github.com/ptrbrtz/razor-9dof-ahrs
*
*
* History:
*   * Original code (http://code.google.com/p/sf9domahrs/) by Doug Weibel and Jose
Julio,
*     based on ArduIMU v1.5 by Jordi Munoz and William Premerlani, Jose Julio and Doug
Weibel. Thank you!
*
*   * Updated code (http://groups.google.com/group/sf_9dof_ahrs_update) by David Malik
(david.zsolt.malik@gmail.com)
*     for new SparkFun 9DOF Razor hardware (SEN-10125).
*
*   * Updated and extended by Peter Bartz (peter-bartz@gmx.de):
*     * v1.3.0
*       * Cleaned up, streamlined and restructured most of the code to make it more
comprehensible.
*       * Added sensor calibration (improves precision and responsiveness a lot!).
*       * Added binary yaw/pitch/roll output.
*       * Added basic serial command interface to set output modes/calibrate
sensors/synch stream/etc.
*       * Added support to synch automatically when using Rovering Networks Bluetooth
modules (and compatible).
*       * Wrote new easier to use test program (using Processing).
*       * Added support for new version of "9DOF Razor IMU": SEN-10736.
*       --> The output of this code is not compatible with the older versions!
*       --> A Processing sketch to test the tracker is available.
*     * v1.3.1
*       * Initializing rotation matrix based on start-up sensor readings ->
orientation OK right away.
*       * Adjusted gyro low-pass filter and output rate settings.
*     * v1.3.2
*       * Adapted code to work with new Arduino 1.0 (and older versions still).
*     * v1.3.3
*       * Improved synching.
*     * v1.4.0
```

```
*       * Added support for SparkFun "9DOF Sensor Stick" (versions SEN-10183, SEN-
10321 and SEN-10724).
*     * v1.4.1
*       * Added output modes to read raw and/or calibrated sensor data in text or
binary format.
*       * Added static magnetometer soft iron distortion compensation
*     * v1.4.2
*       * (No core firmware changes)
*
* TODOs:
*   * Allow optional use of EEPROM for storing and reading calibration values.
*   * Use self-test and temperature-compensation features of the sensors.
*************************************************************************************
*************************/

/*
  "9DOF Razor IMU" hardware versions: SEN-10125 and SEN-10736

  ATMega328@3.3V, 8MHz

  ADXL345  : Accelerometer
  HMC5843  : Magnetometer on SEN-10125
  HMC5883L : Magnetometer on SEN-10736
  ITG-3200 : Gyro

  Arduino IDE : Select board "Arduino Pro or Pro Mini (3.3v, 8Mhz) w/ATmega328"
*/

/*
  "9DOF Sensor Stick" hardware versions: SEN-10183, SEN-10321 and SEN-10724

  ADXL345  : Accelerometer
  HMC5843  : Magnetometer on SEN-10183 and SEN-10321
  HMC5883L : Magnetometer on SEN-10724
  ITG-3200 : Gyro
*/

/*
  Axis definition (differs from definition printed on the board!):
    X axis pointing forward (towards the short edge with the connector holes)
    Y axis pointing to the right
    and Z axis pointing down.

  Positive yaw   : clockwise
  Positive roll  : right wing down
  Positive pitch : nose up

  Transformation order: first yaw then pitch then roll.
*/

/*
  Serial commands that the firmware understands:

  "#o<params>" - Set OUTPUT mode and parameters. The available options are:

      // Streaming output
      "#o0" - DISABLE continuous streaming output. Also see #f below.
      "#o1" - ENABLE continuous streaming output.

      // Angles output
      "#ob" - Output angles in BINARY format (yaw/pitch/roll as binary float, so one
output frame
```

```
              is 3x4 = 12 bytes long).
      "#ot" - Output angles in TEXT format (Output frames have form like "#YPR=-
142.28,-5.38,33.52",
              followed by carriage return and line feed [\r\n]).

      // Sensor calibration
      "#oc" - Go to CALIBRATION output mode.
      "#on" - When in calibration mode, go on to calibrate NEXT sensor.

      // Sensor data output
      "#osct" - Output CALIBRATED SENSOR data of all 9 axes in TEXT format.
                One frame consist of three lines - one for each sensor: acc, mag, gyr.
      "#osrt" - Output RAW SENSOR data of all 9 axes in TEXT format.
                One frame consist of three lines - one for each sensor: acc, mag, gyr.
      "#osbt" - Output BOTH raw and calibrated SENSOR data of all 9 axes in TEXT
format.
                One frame consist of six lines - like #osrt and #osct combined (first
RAW, then CALIBRATED).
                NOTE: This is a lot of number-to-text conversion work for the little
8MHz chip on the Razor boards.
                In fact it's too much and an output frame rate of 50Hz can not be
maintained. #osbb.
      "#oscb" - Output CALIBRATED SENSOR data of all 9 axes in BINARY format.
                One frame consist of three 3x3 float values = 36 bytes. Order is: acc
x/y/z, mag x/y/z, gyr x/y/z.
      "#osrb" - Output RAW SENSOR data of all 9 axes in BINARY format.
                One frame consist of three 3x3 float values = 36 bytes. Order is: acc
x/y/z, mag x/y/z, gyr x/y/z.
      "#osbb" - Output BOTH raw and calibrated SENSOR data of all 9 axes in BINARY
format.
                One frame consist of 2x36 = 72 bytes - like #osrb and #oscb combined
(first RAW, then CALIBRATED).

      // Error message output
      "#oe0" - Disable ERROR message output.
      "#oe1" - Enable ERROR message output.


  "#f" - Request one output frame - useful when continuous output is disabled and
updates are
         required in larger intervals only. Though #f only requests one reply, replies
are still
         bound to the internal 20ms (50Hz) time raster. So worst case delay that #f
can add is 19.99ms.


  "#s<xy>" - Request synch token - useful to find out where the frame boundaries are
in a continuous
         binary stream or to see if tracker is present and answering. The tracker will
send
         "#SYNCH<xy>\r\n" in response (so it's possible to read using a readLine()
function).
         x and y are two mandatory but arbitrary bytes that can be used to find out
which request
         the answer belongs to.


  ("#C" and "#D" - Reserved for communication with optional Bluetooth module.)

  Newline characters are not required. So you could send "#ob#o1#s", which
  would set binary output mode, enable continuous streaming output and request
  a synch token all at once.
```

```
  The status LED will be on if streaming output is enabled and off otherwise.

  Byte order of binary output is little-endian: least significant byte comes first.
*/



/*****************************************************************/
/*********** USER SETUP AREA! Set your options here! *************/
/*****************************************************************/

// HARDWARE OPTIONS
/*****************************************************************/
// Select your hardware here by uncommenting one line!
//#define HW__VERSION_CODE 10125 // SparkFun "9DOF Razor IMU" version "SEN-10125"
(HMC5843 magnetometer)
#define HW__VERSION_CODE 10736 // SparkFun "9DOF Razor IMU" version "SEN-10736"
(HMC5883L magnetometer)
//#define HW__VERSION_CODE 10183 // SparkFun "9DOF Sensor Stick" version "SEN-10183"
(HMC5843 magnetometer)
//#define HW__VERSION_CODE 10321 // SparkFun "9DOF Sensor Stick" version "SEN-10321"
(HMC5843 magnetometer)
//#define HW__VERSION_CODE 10724 // SparkFun "9DOF Sensor Stick" version "SEN-10724"
(HMC5883L magnetometer)


// OUTPUT OPTIONS
/*****************************************************************/
// Set your serial port baud rate used to send out data here!
#define OUTPUT__BAUD_RATE 115200//57600

// Sensor data output interval in milliseconds
// This may not work, if faster than 20ms (=50Hz)
// Code is tuned for 20ms, so better leave it like that
#define OUTPUT__DATA_INTERVAL 0//10//20  // in milliseconds

// Output mode definitions (do not change)
#define OUTPUT__MODE_CALIBRATE_SENSORS 0 // Outputs sensor min/max values as text for
manual calibration
#define OUTPUT__MODE_ANGLES 1 // Outputs yaw/pitch/roll in degrees
#define OUTPUT__MODE_SENSORS_CALIB 2 // Outputs calibrated sensor values for all 9
axes
#define OUTPUT__MODE_SENSORS_RAW 3 // Outputs raw (uncalibrated) sensor values for all
9 axes
#define OUTPUT__MODE_SENSORS_BOTH 4 // Outputs calibrated AND raw sensor values for
all 9 axes
// Output format definitions (do not change)
#define OUTPUT__FORMAT_TEXT 0 // Outputs data as text
#define OUTPUT__FORMAT_BINARY 1 // Outputs data as binary float

// Select your startup output mode and format here!
int output_mode = OUTPUT__MODE_ANGLES;
int output_format = OUTPUT__FORMAT_TEXT;

// Select if serial continuous streaming output is enabled per default on startup.
#define OUTPUT__STARTUP_STREAM_ON true  // true or false

// If set true, an error message will be output if we fail to read sensor data.
// Message format: "!ERR: reading <sensor>", followed by "\r\n".
boolean output_errors = false;  // true or false
```

```
// Bluetooth
// You can set this to true, if you have a Rovering Networks Bluetooth Module
attached.
// The connect/disconnect message prefix of the module has to be set to "#".
// (Refer to manual, it can be set like this: SO,#)
// When using this, streaming output will only be enabled as long as we're connected.
That way
// receiver and sender are synchronzed easily just by connecting/disconnecting.
// It is not necessary to set this! It just makes life easier when writing code for
// the receiving side. The Processing test sketch also works without setting this.
// NOTE: When using this, OUTPUT__STARTUP_STREAM_ON has no effect!
#define OUTPUT__HAS_RN_BLUETOOTH false  // true or false


// SENSOR CALIBRATION
/*****************************************************************/
// How to calibrate? Read the tutorial at http://dev.qu.tu-berlin.de/projects/sf-
razor-9dof-ahrs
// Put MIN/MAX and OFFSET readings for your board here!
// Accelerometer
// "accel x,y,z (min/max) = X_MIN/X_MAX  Y_MIN/Y_MAX  Z_MIN/Z_MAX"
#define ACCEL_X_MIN ((float) -250)
#define ACCEL_X_MAX ((float) 250)
#define ACCEL_Y_MIN ((float) -250)
#define ACCEL_Y_MAX ((float) 250)
#define ACCEL_Z_MIN ((float) -250)
#define ACCEL_Z_MAX ((float) 250)

// Magnetometer (standard calibration mode)
// "magn x,y,z (min/max) = X_MIN/X_MAX  Y_MIN/Y_MAX  Z_MIN/Z_MAX"
#define MAGN_X_MIN ((float) -600)
#define MAGN_X_MAX ((float) 600)
#define MAGN_Y_MIN ((float) -600)
#define MAGN_Y_MAX ((float) 600)
#define MAGN_Z_MIN ((float) -600)
#define MAGN_Z_MAX ((float) 600)

// Magnetometer (extended calibration mode)
// Uncommend to use extended magnetometer calibration (compensates hard & soft iron
errors)
//#define CALIBRATION__MAGN_USE_EXTENDED true
//const float magn_ellipsoid_center[3] = {0, 0, 0};
//const float magn_ellipsoid_transform[3][3] = {{0, 0, 0}, {0, 0, 0}, {0, 0, 0}};

// Gyroscope
// "gyro x,y,z (current/average) = .../OFFSET_X  .../OFFSET_Y  .../OFFSET_Z
#define GYRO_AVERAGE_OFFSET_X ((float) 0.0)
#define GYRO_AVERAGE_OFFSET_Y ((float) 0.0)
#define GYRO_AVERAGE_OFFSET_Z ((float) 0.0)


// Calibration example:

// "accel x,y,z (min/max) = -277.00/264.00  -256.00/278.00  -299.00/235.00"
//#define ACCEL_X_MIN ((float) -277)
//#define ACCEL_X_MAX ((float) 264)
//#define ACCEL_Y_MIN ((float) -256)
//#define ACCEL_Y_MAX ((float) 278)
//#define ACCEL_Z_MIN ((float) -299)
//#define ACCEL_Z_MAX ((float) 235)

// "magn x,y,z (min/max) = -511.00/581.00  -516.00/568.00  -489.00/486.00"
```

```
//#define MAGN_X_MIN ((float) -511)
//#define MAGN_X_MAX ((float) 581)
//#define MAGN_Y_MIN ((float) -516)
//#define MAGN_Y_MAX ((float) 568)
//#define MAGN_Z_MIN ((float) -489)
//#define MAGN_Z_MAX ((float) 486)
/*
// Extended magn
#define CALIBRATION__MAGN_USE_EXTENDED true
const float magn_ellipsoid_center[3] = {91.5, -13.5, -48.1};
const float magn_ellipsoid_transform[3][3] = {{0.902, -0.00354, 0.000636}, {-0.00354,
0.9, -0.00599}, {0.000636, -0.00599, 1}};
*/
// Extended magn (with Sennheiser HD 485 headphones)
//#define CALIBRATION__MAGN_USE_EXTENDED true
//const float magn_ellipsoid_center[3] = {72.3360, 23.0954, 53.6261};
//const float magn_ellipsoid_transform[3][3] = {{0.879685, 0.000540833, -0.0106054},
{0.000540833, 0.891086, -0.0130338}, {-0.0106054, -0.0130338, 0.997494}};

//"gyro x,y,z (current/average) = -40.00/-42.05  98.00/96.20  -18.00/-18.36"
/*
#define GYRO_AVERAGE_OFFSET_X ((float) -42.05)
#define GYRO_AVERAGE_OFFSET_Y ((float) 96.20)
#define GYRO_AVERAGE_OFFSET_Z ((float) -18.36)
*/
#define GYRO_AVERAGE_OFFSET_X ((float) -42.05)
#define GYRO_AVERAGE_OFFSET_Y ((float) 99.20)
#define GYRO_AVERAGE_OFFSET_Z ((float) -32.66)
/*
#define GYRO_AVERAGE_OFFSET_X ((float) 0*30.0)
#define GYRO_AVERAGE_OFFSET_Y ((float) 0*79.0)
#define GYRO_AVERAGE_OFFSET_Z ((float) -0*3.0)
*/
//gyro x,y,z (current/average) = 30.00/29.76  78.00/78.90  -3.00/-2.50
/*
#define GYRO_AVERAGE_OFFSET_X ((float) 29.76)
#define GYRO_AVERAGE_OFFSET_Y ((float) 78.90)
#define GYRO_AVERAGE_OFFSET_Z ((float) -2.50)
*/
/*
#define CALIBRATION__MAGN_USE_EXTENDED true
const float magn_ellipsoid_center[3] = {120.987, -64.1249, -23.9885};
const float magn_ellipsoid_transform[3][3] = {{0.877829, -0.000779676, -0.00180051},
{-0.000779676, 0.882212, 0.0361621}, {-0.00180051, 0.0361621, 0.988864}};

*/
#define CALIBRATION__MAGN_USE_EXTENDED true
const float magn_ellipsoid_center[3] = {103.876, -77.6812, -54.3508};
const float magn_ellipsoid_transform[3][3] = {{0.904266, 0.00112385, 0.00256729},
{0.00112385, 0.900164, 0.0224625}, {0.00256729, 0.0224625, 0.994863}};

// DEBUG OPTIONS
/*****************************************************************/
// When set to true, gyro drift correction will not be applied
#define DEBUG__NO_DRIFT_CORRECTION false
// Print elapsed time after each I/O loop
#define DEBUG__PRINT_LOOP_TIME false


/*****************************************************************/
/****************** END OF USER SETUP AREA!  *********************/
/*****************************************************************/
```

```
// Check if hardware version code is defined
#ifndef HW__VERSION_CODE
  // Generate compile error
  #error YOU HAVE TO SELECT THE HARDWARE YOU ARE USING! See "HARDWARE OPTIONS" in
"USER SETUP AREA" at top of Razor_AHRS.ino!
#endif

#include <Wire.h>

// Sensor calibration scale and offset values
#define ACCEL_X_OFFSET ((ACCEL_X_MIN + ACCEL_X_MAX) / 2.0f)
#define ACCEL_Y_OFFSET ((ACCEL_Y_MIN + ACCEL_Y_MAX) / 2.0f)
#define ACCEL_Z_OFFSET ((ACCEL_Z_MIN + ACCEL_Z_MAX) / 2.0f)
#define ACCEL_X_SCALE (GRAVITY / (ACCEL_X_MAX - ACCEL_X_OFFSET))
#define ACCEL_Y_SCALE (GRAVITY / (ACCEL_Y_MAX - ACCEL_Y_OFFSET))
#define ACCEL_Z_SCALE (GRAVITY / (ACCEL_Z_MAX - ACCEL_Z_OFFSET))

#define MAGN_X_OFFSET ((MAGN_X_MIN + MAGN_X_MAX) / 2.0f)
#define MAGN_Y_OFFSET ((MAGN_Y_MIN + MAGN_Y_MAX) / 2.0f)
#define MAGN_Z_OFFSET ((MAGN_Z_MIN + MAGN_Z_MAX) / 2.0f)
#define MAGN_X_SCALE (100.0f / (MAGN_X_MAX - MAGN_X_OFFSET))
#define MAGN_Y_SCALE (100.0f / (MAGN_Y_MAX - MAGN_Y_OFFSET))
#define MAGN_Z_SCALE (100.0f / (MAGN_Z_MAX - MAGN_Z_OFFSET))


// Gain for gyroscope (ITG-3200)
#define GYRO_GAIN 0.06957 // Same gain on all axes
#define GYRO_SCALED_RAD(x) (x * TO_RAD(GYRO_GAIN)) // Calculate the scaled gyro
readings in radians per second

// DCM parameters

#define Kp_ROLLPITCH 0.002f
#define Ki_ROLLPITCH 0.00002f
#define Kp_YAW 1.2f
#define Ki_YAW 0.000002f

/*
#define Kp_ROLLPITCH 0.02f
#define Ki_ROLLPITCH 0.00002f
#define Kp_YAW 1.2f
#define Ki_YAW 0.00002f
*/
// Stuff
#define STATUS_LED_PIN 13  // Pin number of status LED
#define GRAVITY 256.0f // "1G reference" used for DCM filter and accelerometer
calibration
#define TO_RAD(x) (x * 0.01745329252)  // *pi/180
#define TO_DEG(x) (x * 57.2957795131)  // *180/pi

// Sensor variables
```

```
float accel[3];  // Actually stores the NEGATED acceleration (equals gravity, if board
not moving).
float accel_min[3];
float accel_max[3];

float magnetom[3];
float magnetom_min[3];
float magnetom_max[3];
float magnetom_tmp[3];

float gyro[3];
float gyro_average[3];
int gyro_num_samples = 0;

// DCM variables
float MAG_Heading;
float Accel_Vector[3]= {0, 0, 0}; // Store the acceleration in a vector
float Accel_Vector_old[3]= {0, 0, 0}; // Store the acceleration in a vector
float Gyro_Vector[3]= {0, 0, 0}; // Store the gyros turn rate in a vector
float Gyro_Vector_old[3]= {0, 0, 0};
float Omega_Vector[3]= {0, 0, 0}; // Corrected Gyro_Vector data
float Omega_P[3]= {0, 0, 0}; // Omega Proportional correction
float Omega_I[3]= {0, 0, 0}; // Omega Integrator
float Omega[3]= {0, 0, 0};
float errorRollPitch[3] = {0, 0, 0};
float errorYaw[3] = {0, 0, 0};
float DCM_Matrix[3][3] = {{1, 0, 0}, {0, 1, 0}, {0, 0, 1}};
float Update_Matrix[3][3] = {{0, 1, 2}, {3, 4, 5}, {6, 7, 8}};
float Temporary_Matrix[3][3] = {{0, 0, 0}, {0, 0, 0}, {0, 0, 0}};

// Euler angles
float yaw;
float pitch;
float roll;

// DCM timing in the main loop
unsigned long timestamp;
unsigned long timestamp_old;
float G_Dt; // Integration time for DCM algorithm

// More output-state variables
boolean output_stream_on;
boolean output_single_on;
int curr_calibration_sensor = 0;
boolean reset_calibration_session_flag = true;
int num_accel_errors = 0;
int num_magn_errors = 0;
int num_gyro_errors = 0;

void read_sensors() {
  Read_Gyro(); // Read gyroscope
  Read_Accel(); // Read accelerometer
  Read_Magn(); // Read magnetometer
}

// Read every sensor and record a time stamp
// Init DCM with unfiltered orientation
// TODO re-init global vars?
void reset_sensor_fusion() {
  float temp1[3];
  float temp2[3];
  float xAxis[] = {1.0f, 0.0f, 0.0f};
```

```
  read_sensors();
  timestamp = millis();

  // GET PITCH
  // Using y-z-plane-component/x-component of gravity vector
  pitch = -atan2(accel[0], sqrt(accel[1] * accel[1] + accel[2] * accel[2]));

  // GET ROLL
  // Compensate pitch of gravity vector
  Vector_Cross_Product(temp1, accel, xAxis);
  Vector_Cross_Product(temp2, xAxis, temp1);
  // Normally using x-z-plane-component/y-component of compensated gravity vector
  // roll = atan2(temp2[1], sqrt(temp2[0] * temp2[0] + temp2[2] * temp2[2]));
  // Since we compensated for pitch, x-z-plane-component equals z-component:
  roll = atan2(temp2[1], temp2[2]);

  // GET YAW
  Compass_Heading();
  yaw = MAG_Heading;

  // Init rotation matrix
  init_rotation_matrix(DCM_Matrix, yaw, pitch, roll);
}

// Apply calibration to raw sensor readings
void compensate_sensor_errors() {
    // Compensate accelerometer error
    accel[0] = (accel[0] - ACCEL_X_OFFSET) * ACCEL_X_SCALE;
    accel[1] = (accel[1] - ACCEL_Y_OFFSET) * ACCEL_Y_SCALE;
    accel[2] = (accel[2] - ACCEL_Z_OFFSET) * ACCEL_Z_SCALE;

    // Compensate magnetometer error
#if CALIBRATION__MAGN_USE_EXTENDED == true
    for (int i = 0; i < 3; i++)
      magnetom_tmp[i] = magnetom[i] - magn_ellipsoid_center[i];
    Matrix_Vector_Multiply(magn_ellipsoid_transform, magnetom_tmp, magnetom);
#else
    magnetom[0] = (magnetom[0] - MAGN_X_OFFSET) * MAGN_X_SCALE;
    magnetom[1] = (magnetom[1] - MAGN_Y_OFFSET) * MAGN_Y_SCALE;
    magnetom[2] = (magnetom[2] - MAGN_Z_OFFSET) * MAGN_Z_SCALE;
#endif

    // Compensate gyroscope error
    gyro[0] -= GYRO_AVERAGE_OFFSET_X;
    gyro[1] -= GYRO_AVERAGE_OFFSET_Y;
    gyro[2] -= GYRO_AVERAGE_OFFSET_Z;
}

// Reset calibration session if reset_calibration_session_flag is set
void check_reset_calibration_session()
{
  // Raw sensor values have to be read already, but no error compensation applied

  // Reset this calibration session?
  if (!reset_calibration_session_flag) return;

  // Reset acc and mag calibration variables
  for (int i = 0; i < 3; i++) {
    accel_min[i] = accel_max[i] = accel[i];
    magnetom_min[i] = magnetom_max[i] = magnetom[i];
  }
```

```
  // Reset gyro calibration variables
  gyro_num_samples = 0;  // Reset gyro calibration averaging
  gyro_average[0] = gyro_average[1] = gyro_average[2] = 0.0f;

  reset_calibration_session_flag = false;
}

void turn_output_stream_on()
{
  output_stream_on = true;
  digitalWrite(STATUS_LED_PIN, HIGH);
}

void turn_output_stream_off()
{
  output_stream_on = false;
  digitalWrite(STATUS_LED_PIN, LOW);
}

// Blocks until another byte is available on serial port
char readChar()
{
  while (Serial.available() < 1) { } // Block
  return Serial.read();
}

void setup()
{
  // Init serial output
  Serial.begin(OUTPUT__BAUD_RATE);

  // Init status LED
  pinMode (STATUS_LED_PIN, OUTPUT);
  digitalWrite(STATUS_LED_PIN, LOW);

  // Init sensors
  delay(50);  // Give sensors enough time to start
  I2C_Init();
  Accel_Init();
  Magn_Init();
  Gyro_Init();

  // Read sensors, init DCM algorithm
  delay(20);  // Give sensors enough time to collect data
  reset_sensor_fusion();

  // Init output
#if (OUTPUT__HAS_RN_BLUETOOTH == true) || (OUTPUT__STARTUP_STREAM_ON == false)
  turn_output_stream_off();
#else
  turn_output_stream_on();
#endif
}

// Main loop
void loop()
{
  // Read incoming control messages
  if (Serial.available() >= 2)
  {
    if (Serial.read() == '#') // Start of new control message
```

```
    {
      int command = Serial.read(); // Commands
      if (command == 'f') // request one output _f_rame
        output_single_on = true;
      else if (command == 's') // _s_ynch request
      {
        // Read ID
        byte id[2];
        id[0] = readChar();
        id[1] = readChar();

        // Reply with synch message
        Serial.print("#SYNCH");
        Serial.write(id, 2);
        Serial.println();
      }
      else if (command == 'o') // Set _o_utput mode
      {
        char output_param = readChar();
        if (output_param == 'n')  // Calibrate _n_ext sensor
        {
          curr_calibration_sensor = (curr_calibration_sensor + 1) % 3;
          reset_calibration_session_flag = true;
        }
        else if (output_param == 't') // Output angles as _t_ext
        {
          output_mode = OUTPUT__MODE_ANGLES;
          output_format = OUTPUT__FORMAT_TEXT;
        }
        else if (output_param == 'b') // Output angles in _b_inary format
        {
          output_mode = OUTPUT__MODE_ANGLES;
          output_format = OUTPUT__FORMAT_BINARY;
        }
        else if (output_param == 'c') // Go to _c_alibration mode
        {
          output_mode = OUTPUT__MODE_CALIBRATE_SENSORS;
          reset_calibration_session_flag = true;
        }
        else if (output_param == 's') // Output _s_ensor values
        {
          char values_param = readChar();
          char format_param = readChar();
          if (values_param == 'r')  // Output _r_aw sensor values
            output_mode = OUTPUT__MODE_SENSORS_RAW;
          else if (values_param == 'c')  // Output _c_alibrated sensor values
            output_mode = OUTPUT__MODE_SENSORS_CALIB;
          else if (values_param == 'b')  // Output _b_oth sensor values (raw and
calibrated)
            output_mode = OUTPUT__MODE_SENSORS_BOTH;

          if (format_param == 't') // Output values as _t_text
            output_format = OUTPUT__FORMAT_TEXT;
          else if (format_param == 'b') // Output values in _b_inary format
            output_format = OUTPUT__FORMAT_BINARY;
        }
        else if (output_param == '0') // Disable continuous streaming output
        {
          turn_output_stream_off();
          reset_calibration_session_flag = true;
        }
        else if (output_param == '1') // Enable continuous streaming output
```

```
        {
          reset_calibration_session_flag = true;
          turn_output_stream_on();
        }
        else if (output_param == 'e') // _e_rror output settings
        {
          char error_param = readChar();
          if (error_param == '0') output_errors = false;
          else if (error_param == '1') output_errors = true;
          else if (error_param == 'c') // get error count
          {
            Serial.print("#AMG-ERR:");
            Serial.print(num_accel_errors); Serial.print(",");
            Serial.print(num_magn_errors); Serial.print(",");
            Serial.println(num_gyro_errors);
          }
        }
      }
#if OUTPUT__HAS_RN_BLUETOOTH == true
      // Read messages from bluetooth module
      // For this to work, the connect/disconnect message prefix of the module has to
be set to "#".
      else if (command == 'C') // Bluetooth "#CONNECT" message (does the same as
"#o1")
        turn_output_stream_on();
      else if (command == 'D') // Bluetooth "#DISCONNECT" message (does the same as
"#o0")
        turn_output_stream_off();
#endif // OUTPUT__HAS_RN_BLUETOOTH == true
    }
    else
    { } // Skip character
  }

  // Time to read the sensors again?
  if((millis() - timestamp) >= OUTPUT__DATA_INTERVAL)
  {
    timestamp_old = timestamp;
    timestamp = millis();
    if (timestamp > timestamp_old)
      G_Dt = (float) (timestamp - timestamp_old) / 1000.0f; // Real time of loop run.
We use this on the DCM algorithm (gyro integration time)
    else G_Dt = 0;

    // Update sensor readings
    read_sensors();

    if (output_mode == OUTPUT__MODE_CALIBRATE_SENSORS)  // We're in calibration mode
    {
      check_reset_calibration_session();  // Check if this session needs a reset
      if (output_stream_on || output_single_on)
output_calibration(curr_calibration_sensor);
    }
    else if (output_mode == OUTPUT__MODE_ANGLES)  // Output angles
    {
      // Apply sensor calibration
      compensate_sensor_errors();

      // Run DCM algorithm
      Compass_Heading(); // Calculate magnetic heading
      Matrix_update();
      Normalize();
```

```
      Drift_correction();
      Euler_angles();

      if (output_stream_on || output_single_on) output_angles();
    }
    else  // Output sensor values
    {
      if (output_stream_on || output_single_on) output_sensors();
    }

    output_single_on = false;

#if DEBUG__PRINT_LOOP_TIME == true
    Serial.print("loop time (ms) = ");
    Serial.println(millis() - timestamp);
#endif
  }
#if DEBUG__PRINT_LOOP_TIME == true
  else
  {
    Serial.println("waiting...");
  }
#endif
}
```

Compass.ino

```
/* This file is part of the Razor AHRS Firmware */

void Compass_Heading()
{
  float mag_x;
  float mag_y;
  float cos_roll;
  float sin_roll;
  float cos_pitch;
  float sin_pitch;

  cos_roll = cos(roll);
  sin_roll = sin(roll);
  cos_pitch = cos(pitch);
  sin_pitch = sin(pitch);

  // Tilt compensated magnetic field X
  mag_x = magnetom[0] * cos_pitch + magnetom[1] * sin_roll * sin_pitch + magnetom[2] *
cos_roll * sin_pitch;
  // Tilt compensated magnetic field Y
  mag_y = magnetom[1] * cos_roll - magnetom[2] * sin_roll;
  // Magnetic Heading
  MAG_Heading = atan2(-mag_y, mag_x);
}
```

DCM.ino

```
/* This file is part of the Razor AHRS Firmware */

// DCM algorithm

/**************************************************/
void Normalize(void)
{
  float error=0;
  float temporary[3][3];
  float renorm=0;

  error= -Vector_Dot_Product(&DCM_Matrix[0][0],&DCM_Matrix[1][0])*.5; //eq.19

  Vector_Scale(&temporary[0][0], &DCM_Matrix[1][0], error); //eq.19
  Vector_Scale(&temporary[1][0], &DCM_Matrix[0][0], error); //eq.19

  Vector_Add(&temporary[0][0], &temporary[0][0], &DCM_Matrix[0][0]);//eq.19
  Vector_Add(&temporary[1][0], &temporary[1][0], &DCM_Matrix[1][0]);//eq.19

  Vector_Cross_Product(&temporary[2][0],&temporary[0][0],&temporary[1][0]); // c= a x
b //eq.20

  renorm= .5 *(3 - Vector_Dot_Product(&temporary[0][0],&temporary[0][0])); //eq.21
  Vector_Scale(&DCM_Matrix[0][0], &temporary[0][0], renorm);

  renorm= .5 *(3 - Vector_Dot_Product(&temporary[1][0],&temporary[1][0])); //eq.21
  Vector_Scale(&DCM_Matrix[1][0], &temporary[1][0], renorm);

  renorm= .5 *(3 - Vector_Dot_Product(&temporary[2][0],&temporary[2][0])); //eq.21
  Vector_Scale(&DCM_Matrix[2][0], &temporary[2][0], renorm);
}

/**************************************************/
void Drift_correction(void)
{
  float mag_heading_x;
  float mag_heading_y;
  float errorCourse;
  //Compensation the Roll, Pitch and Yaw drift.
  static float Scaled_Omega_P[3];
  static float Scaled_Omega_I[3];
  float Accel_magnitude;
  float Accel_weight;


  //*****Roll and Pitch***************

  // Calculate the magnitude of the accelerometer vector
  Accel_magnitude = sqrt(Accel_Vector[0]*Accel_Vector[0] +
Accel_Vector[1]*Accel_Vector[1] + Accel_Vector[2]*Accel_Vector[2]);
  Accel_magnitude = Accel_magnitude / GRAVITY; // Scale to gravity.
  // Dynamic weighting of accelerometer info (reliability filter)
  // Weight for accelerometer info (<0.5G = 0.0, 1G = 1.0 , >1.5G = 0.0)
  Accel_weight = constrain(1 - 2*abs(1 - Accel_magnitude),0,1);  //

  Vector_Cross_Product(&errorRollPitch[0],&Accel_Vector[0],&DCM_Matrix[2][0]);
//adjust the ground of reference
  Vector_Scale(&Omega_P[0],&errorRollPitch[0],Kp_ROLLPITCH*Accel_weight);

  Vector_Scale(&Scaled_Omega_I[0],&errorRollPitch[0],Ki_ROLLPITCH*Accel_weight);
```

```
  Vector_Add(Omega_I,Omega_I,Scaled_Omega_I);

  //*****YAW***************
  // We make the gyro YAW drift correction based on compass magnetic heading

  mag_heading_x = cos(MAG_Heading);
  mag_heading_y = sin(MAG_Heading);
  errorCourse=(DCM_Matrix[0][0]*mag_heading_y) - (DCM_Matrix[1][0]*mag_heading_x);
//Calculating YAW error
  Vector_Scale(errorYaw,&DCM_Matrix[2][0],errorCourse); //Applys the yaw correction to
the XYZ rotation of the aircraft, depeding the position.

  Vector_Scale(&Scaled_Omega_P[0],&errorYaw[0],Kp_YAW);//.01proportional of YAW.
  Vector_Add(Omega_P,Omega_P,Scaled_Omega_P);//Adding  Proportional.

  Vector_Scale(&Scaled_Omega_I[0],&errorYaw[0],Ki_YAW);//.00001Integrator
  Vector_Add(Omega_I,Omega_I,Scaled_Omega_I);//adding integrator to the Omega_I
}

void Matrix_update(void)
{

  Gyro_Vector[0]=GYRO_SCALED_RAD(gyro[0])-0.085; //gyro x roll
  Gyro_Vector[1]=GYRO_SCALED_RAD(gyro[1])+0.025; //gyro y pitch
  Gyro_Vector[2]=GYRO_SCALED_RAD(gyro[2])-0.035; //gyro z yaw

  /*
  Gyro_Vector[0]= (1-0.8)*Gyro_Vector_old[0] + 0.8 * Gyro_Vector[0];
  Gyro_Vector[1]= (1-0.8)*Gyro_Vector_old[1] + 0.8 * Gyro_Vector[1];
  Gyro_Vector[2]= (1-0.8)*Gyro_Vector_old[2] + 0.8 * Gyro_Vector[2];


  Gyro_Vector_old[0] = Gyro_Vector[0];
  Gyro_Vector_old[1] = Gyro_Vector[1];
  Gyro_Vector_old[2] = Gyro_Vector[2];
  */
  /*
  Gyro_Vector[0]= (1-0.8)*Gyro_Vector[0] + 0.8 * (GYRO_SCALED_RAD(gyro[0])-0.09);
//gyro x roll
  Gyro_Vector[1]= (1-0.8)*Gyro_Vector[1] + 0.8 * (GYRO_SCALED_RAD(gyro[1])+0.015);
//gyro y pitch
  Gyro_Vector[2]= (1-0.8)*Gyro_Vector[2] + 0.8 * (GYRO_SCALED_RAD(gyro[2])-0.01);
//gyro z yaw
  */
  Accel_Vector[0]=accel[0];
  Accel_Vector[1]=accel[1];
  Accel_Vector[2]=accel[2];
  /*
  Accel_Vector[0]= (1-0.5)*Accel_Vector_old[0] + 0.5 * Accel_Vector[0];
  Accel_Vector[1]= (1-0.5)*Accel_Vector_old[1] + 0.5* Accel_Vector[1];
  Accel_Vector[2]= (1-0.5)*Accel_Vector_old[2] + 0.5 * Accel_Vector[2];
  */
  /*
  Serial.print(Accel_Vector_old[0]);
  Serial.print(", ");
  Serial.print(Accel_Vector_old[1]);
  Serial.print(", ");
  Serial.println(Accel_Vector_old[2]);
  */
  /*
  Accel_Vector_old[0] = Accel_Vector[0];
  Accel_Vector_old[1] = Accel_Vector[1];
```

```
  Accel_Vector_old[2] = Accel_Vector[2];
    */
  Vector_Add(&Omega[0], &Gyro_Vector[0], &Omega_I[0]);  //adding proportional term
  Vector_Add(&Omega_Vector[0], &Omega[0], &Omega_P[0]); //adding Integrator term

#if DEBUG__NO_DRIFT_CORRECTION == true // Do not use drift correction
  Update_Matrix[0][0]=0;
  Update_Matrix[0][1]=-G_Dt*Gyro_Vector[2];//-z
  Update_Matrix[0][2]=G_Dt*Gyro_Vector[1];//y
  Update_Matrix[1][0]=G_Dt*Gyro_Vector[2];//z
  Update_Matrix[1][1]=0;
  Update_Matrix[1][2]=-G_Dt*Gyro_Vector[0];
  Update_Matrix[2][0]=-G_Dt*Gyro_Vector[1];
  Update_Matrix[2][1]=G_Dt*Gyro_Vector[0];
  Update_Matrix[2][2]=0;
#else // Use drift correction
  Update_Matrix[0][0]=0;
  Update_Matrix[0][1]=-G_Dt*Omega_Vector[2];//-z
  Update_Matrix[0][2]=G_Dt*Omega_Vector[1];//y
  Update_Matrix[1][0]=G_Dt*Omega_Vector[2];//z
  Update_Matrix[1][1]=0;
  Update_Matrix[1][2]=-G_Dt*Omega_Vector[0];//-x
  Update_Matrix[2][0]=-G_Dt*Omega_Vector[1];//-y
  Update_Matrix[2][1]=G_Dt*Omega_Vector[0];//x
  Update_Matrix[2][2]=0;
#endif

  Matrix_Multiply(DCM_Matrix,Update_Matrix,Temporary_Matrix); //a*b=c

  for(int x=0; x<3; x++) //Matrix Addition (update)
  {
    for(int y=0; y<3; y++)
    {
      DCM_Matrix[x][y]+=Temporary_Matrix[x][y];
    }
  }
}

void Euler_angles(void)
{
  pitch = -asin(DCM_Matrix[2][0]);
  roll = atan2(DCM_Matrix[2][1],DCM_Matrix[2][2]);
  yaw = atan2(DCM_Matrix[1][0],DCM_Matrix[0][0]);
}
```

Math.ino

```
/* This file is part of the Razor AHRS Firmware */

// Computes the dot product of two vectors
float Vector_Dot_Product(const float v1[3], const float v2[3])
{
  float result = 0;

  for(int c = 0; c < 3; c++)
  {
    result += v1[c] * v2[c];
  }

  return result;
}

// Computes the cross product of two vectors
// out has to different from v1 and v2 (no in-place)!
void Vector_Cross_Product(float out[3], const float v1[3], const float v2[3])
{
  out[0] = (v1[1] * v2[2]) - (v1[2] * v2[1]);
  out[1] = (v1[2] * v2[0]) - (v1[0] * v2[2]);
  out[2] = (v1[0] * v2[1]) - (v1[1] * v2[0]);
}

// Multiply the vector by a scalar
void Vector_Scale(float out[3], const float v[3], float scale)
{
  for(int c = 0; c < 3; c++)
  {
    out[c] = v[c] * scale;
  }
}

// Adds two vectors
void Vector_Add(float out[3], const float v1[3], const float v2[3])
{
  for(int c = 0; c < 3; c++)
  {
    out[c] = v1[c] + v2[c];
  }
}

// Multiply two 3x3 matrices: out = a * b
// out has to different from a and b (no in-place)!
void Matrix_Multiply(const float a[3][3], const float b[3][3], float out[3][3])
{
  for(int x = 0; x < 3; x++)  // rows
  {
    for(int y = 0; y < 3; y++)  // columns
    {
      out[x][y] = a[x][0] * b[0][y] + a[x][1] * b[1][y] + a[x][2] * b[2][y];
    }
  }
}

// Multiply 3x3 matrix with vector: out = a * b
// out has to different from b (no in-place)!
void Matrix_Vector_Multiply(const float a[3][3], const float b[3], float out[3])
{
  for(int x = 0; x < 3; x++)
  {
```

```
    out[x] = a[x][0] * b[0] + a[x][1] * b[1] + a[x][2] * b[2];
  }
}

// Init rotation matrix using euler angles
void init_rotation_matrix(float m[3][3], float yaw, float pitch, float roll)
{
  float c1 = cos(roll);
  float s1 = sin(roll);
  float c2 = cos(pitch);
  float s2 = sin(pitch);
  float c3 = cos(yaw);
  float s3 = sin(yaw);

  // Euler angles, right-handed, intrinsic, XYZ convention
  // (which means: rotate around body axes Z, Y', X'')
  m[0][0] = c2 * c3;
  m[0][1] = c3 * s1 * s2 - c1 * s3;
  m[0][2] = s1 * s3 + c1 * c3 * s2;

  m[1][0] = c2 * s3;
  m[1][1] = c1 * c3 + s1 * s2 * s3;
  m[1][2] = c1 * s2 * s3 - c3 * s1;

  m[2][0] = -s2;
  m[2][1] = c2 * s1;
  m[2][2] = c1 * c2;
}
```

Output.ino

```
/* This file is part of the Razor AHRS Firmware */

// Output angles: yaw, pitch, roll
void output_angles()
{

  if (output_format == OUTPUT__FORMAT_BINARY)
  {
    float ypr[3];
    ypr[0] = TO_DEG(yaw);
    ypr[1] = TO_DEG(pitch);
    ypr[2] = TO_DEG(roll);
    Serial.write((byte*) ypr, 12);  // No new-line
  }
  else if (output_format == OUTPUT__FORMAT_TEXT)
  {

    Serial.print("#YPR=");
    //Serial.print("#");
    Serial.print(TO_DEG(yaw)); Serial.print(",");
    Serial.print(TO_DEG(pitch)); Serial.print(",");
    Serial.print(TO_DEG(roll)); Serial.print(",");
    /*
    Serial.print(gyro[0]); Serial.print(",");
    Serial.print(gyro[1]); Serial.print(",");
    Serial.print(gyro[2]);
    */
      Serial.print(Gyro_Vector[0]); Serial.print(",");
  Serial.print(Gyro_Vector[1]); Serial.print(",");
  Serial.print(Gyro_Vector[2]);
  Serial.print("!");
    Serial.println();
  }
  //Serial.flush();
}

void output_calibration(int calibration_sensor)
{
  if (calibration_sensor == 0)  // Accelerometer
  {
    // Output MIN/MAX values
    Serial.print("accel x,y,z (min/max) = ");
    for (int i = 0; i < 3; i++) {
      if (accel[i] < accel_min[i]) accel_min[i] = accel[i];
      if (accel[i] > accel_max[i]) accel_max[i] = accel[i];
      Serial.print(accel_min[i]);
      Serial.print("/");
      Serial.print(accel_max[i]);
      if (i < 2) Serial.print("  ");
      else Serial.println();
    }
  }
  else if (calibration_sensor == 1)  // Magnetometer
  {
    // Output MIN/MAX values
    Serial.print("magn x,y,z (min/max) = ");
    for (int i = 0; i < 3; i++) {
      if (magnetom[i] < magnetom_min[i]) magnetom_min[i] = magnetom[i];
      if (magnetom[i] > magnetom_max[i]) magnetom_max[i] = magnetom[i];
      Serial.print(magnetom_min[i]);
```

```
      Serial.print("/");
      Serial.print(magnetom_max[i]);
      if (i < 2) Serial.print("  ");
      else Serial.println();
    }
  }
  else if (calibration_sensor == 2)  // Gyroscope
  {
    // Average gyro values
    for (int i = 0; i < 3; i++)
      gyro_average[i] += gyro[i];
    gyro_num_samples++;

    // Output current and averaged gyroscope values
    Serial.print("gyro x,y,z (current/average) = ");
    for (int i = 0; i < 3; i++) {
      Serial.print(gyro[i]);
      Serial.print("/");
      Serial.print(gyro_average[i] / (float) gyro_num_samples);
      if (i < 2) Serial.print("  ");
      else Serial.println();
    }
  }
}

void output_sensors_text(char raw_or_calibrated)
{
  Serial.print("#A-"); Serial.print(raw_or_calibrated); Serial.print('=');
  Serial.print(accel[0]); Serial.print(",");
  Serial.print(accel[1]); Serial.print(",");
  Serial.print(accel[2]); Serial.println();

  Serial.print("#M-"); Serial.print(raw_or_calibrated); Serial.print('=');
  Serial.print(magnetom[0]); Serial.print(",");
  Serial.print(magnetom[1]); Serial.print(",");
  Serial.print(magnetom[2]); Serial.println();

  Serial.print("#G-"); Serial.print(raw_or_calibrated); Serial.print('=');
  Serial.print(gyro[0]); Serial.print(",");
  Serial.print(gyro[1]); Serial.print(",");
  Serial.print(gyro[2]); Serial.println();
}

void output_sensors_binary()
{
  Serial.write((byte*) accel, 12);
  Serial.write((byte*) magnetom, 12);
  Serial.write((byte*) gyro, 12);
}

void output_sensors()
{
  if (output_mode == OUTPUT__MODE_SENSORS_RAW)
  {
    if (output_format == OUTPUT__FORMAT_BINARY)
      output_sensors_binary();
    else if (output_format == OUTPUT__FORMAT_TEXT)
      output_sensors_text('R');
  }
  else if (output_mode == OUTPUT__MODE_SENSORS_CALIB)
  {
    // Apply sensor calibration
```

```
    compensate_sensor_errors();

    if (output_format == OUTPUT__FORMAT_BINARY)
      output_sensors_binary();
    else if (output_format == OUTPUT__FORMAT_TEXT)
      output_sensors_text('C');
  }
  else if (output_mode == OUTPUT__MODE_SENSORS_BOTH)
  {
    if (output_format == OUTPUT__FORMAT_BINARY)
    {
      output_sensors_binary();
      compensate_sensor_errors();
      output_sensors_binary();
    }
    else if (output_format == OUTPUT__FORMAT_TEXT)
    {
      output_sensors_text('R');
      compensate_sensor_errors();
      output_sensors_text('C');
    }
  }
}
```

Sensors.ino

```
/* This file is part of the Razor AHRS Firmware */

// I2C code to read the sensors

// Sensor I2C addresses
#define ACCEL_ADDRESS ((int) 0x53) // 0x53 = 0xA6 / 2
#define MAGN_ADDRESS  ((int) 0x1E) // 0x1E = 0x3C / 2
#define GYRO_ADDRESS  ((int) 0x68) // 0x68 = 0xD0 / 2

// Arduino backward compatibility macros
#if ARDUINO >= 100
  #define WIRE_SEND(b) Wire.write((byte) b)
  #define WIRE_RECEIVE() Wire.read()
#else
  #define WIRE_SEND(b) Wire.send(b)
  #define WIRE_RECEIVE() Wire.receive()
#endif


void I2C_Init()
{
  Wire.begin();
}

void Accel_Init()
{
  Wire.beginTransmission(ACCEL_ADDRESS);
  WIRE_SEND(0x2D);  // Power register
  WIRE_SEND(0x08);  // Measurement mode
  Wire.endTransmission();
  delay(5);
  Wire.beginTransmission(ACCEL_ADDRESS);
  WIRE_SEND(0x31);  // Data format register
  WIRE_SEND(0x08);  // Set to full resolution
  Wire.endTransmission();
  delay(5);

  // Because our main loop runs at 50Hz we adjust the output data rate to 50Hz (25Hz
bandwidth)
  Wire.beginTransmission(ACCEL_ADDRESS);
  WIRE_SEND(0x2C);  // Rate
  //WIRE_SEND(0x09);  // Set to 50Hz, normal operation
  WIRE_SEND(0x0A);  // Set to 50Hz, normal operation
  Wire.endTransmission();
  delay(5);
}

// Reads x, y and z accelerometer registers
void Read_Accel()
{
  int i = 0;
  byte buff[6];

  Wire.beginTransmission(ACCEL_ADDRESS);
  WIRE_SEND(0x32);  // Send address to read from
  Wire.endTransmission();

  Wire.beginTransmission(ACCEL_ADDRESS);
  Wire.requestFrom(ACCEL_ADDRESS, 6);  // Request 6 bytes
  while(Wire.available())  // ((Wire.available())&&(i<6))
```

```
  {
    buff[i] = WIRE_RECEIVE();  // Read one byte
    i++;
  }
  Wire.endTransmission();

  if (i == 6)  // All bytes received?
  {
    // No multiply by -1 for coordinate system transformation here, because of double
negation:
    // We want the gravity vector, which is negated acceleration vector.
    accel[0] = (((int) buff[3]) << 8) | buff[2];  // X axis (internal sensor y axis)
    accel[1] = (((int) buff[1]) << 8) | buff[0];  // Y axis (internal sensor x axis)
    accel[2] = (((int) buff[5]) << 8) | buff[4];  // Z axis (internal sensor z axis)
  }
  else
  {
    num_accel_errors++;
    if (output_errors) Serial.println("!ERR: reading accelerometer");
  }
}

void Magn_Init()
{
  Wire.beginTransmission(MAGN_ADDRESS);
  WIRE_SEND(0x02);
  WIRE_SEND(0x00);  // Set continuous mode (default 10Hz)
  Wire.endTransmission();
  delay(5);

  Wire.beginTransmission(MAGN_ADDRESS);
  WIRE_SEND(0x00);
  WIRE_SEND(0b00011000);  // Set 50Hz
  Wire.endTransmission();
  delay(5);
}

void Read_Magn()
{
  int i = 0;
  byte buff[6];

  Wire.beginTransmission(MAGN_ADDRESS);
  WIRE_SEND(0x03);  // Send address to read from
  Wire.endTransmission();

  Wire.beginTransmission(MAGN_ADDRESS);
  Wire.requestFrom(MAGN_ADDRESS, 6);  // Request 6 bytes
  while(Wire.available())  // ((Wire.available())&&(i<6))
  {
    buff[i] = WIRE_RECEIVE();  // Read one byte
    i++;
  }
  Wire.endTransmission();

  if (i == 6)  // All bytes received?
  {
// 9DOF Razor IMU SEN-10125 using HMC5843 magnetometer
#if HW__VERSION_CODE == 10125
    // MSB byte first, then LSB; X, Y, Z
    magnetom[0] = -1 * ((((int) buff[2]) << 8) | buff[3]);  // X axis (internal sensor
-y axis)
```

```
    magnetom[1] = -1 * ((((int) buff[0]) << 8) | buff[1]);  // Y axis (internal sensor
-x axis)
    magnetom[2] = -1 * ((((int) buff[4]) << 8) | buff[5]);  // Z axis (internal sensor
-z axis)
// 9DOF Razor IMU SEN-10736 using HMC5883L magnetometer
#elif HW__VERSION_CODE == 10736
    // MSB byte first, then LSB; Y and Z reversed: X, Z, Y
    magnetom[0] = -1 * ((((int) buff[4]) << 8) | buff[5]);  // X axis (internal sensor
-y axis)
    magnetom[1] = -1 * ((((int) buff[0]) << 8) | buff[1]);  // Y axis (internal sensor
-x axis)
    magnetom[2] = -1 * ((((int) buff[2]) << 8) | buff[3]);  // Z axis (internal sensor
-z axis)
// 9DOF Sensor Stick SEN-10183 and SEN-10321 using HMC5843 magnetometer
#elif (HW__VERSION_CODE == 10183) || (HW__VERSION_CODE == 10321)
    // MSB byte first, then LSB; X, Y, Z
    magnetom[0] = (((int) buff[0]) << 8) | buff[1];         // X axis (internal sensor
x axis)
    magnetom[1] = -1 * ((((int) buff[2]) << 8) | buff[3]);  // Y axis (internal sensor
-y axis)
    magnetom[2] = -1 * ((((int) buff[4]) << 8) | buff[5]);  // Z axis (internal sensor
-z axis)
// 9DOF Sensor Stick SEN-10724 using HMC5883L magnetometer
#elif HW__VERSION_CODE == 10724
    // MSB byte first, then LSB; Y and Z reversed: X, Z, Y
    magnetom[0] = (((int) buff[0]) << 8) | buff[1];         // X axis (internal sensor
x axis)
    magnetom[1] = -1 * ((((int) buff[4]) << 8) | buff[5]);  // Y axis (internal sensor
-y axis)
    magnetom[2] = -1 * ((((int) buff[2]) << 8) | buff[3]);  // Z axis (internal sensor
-z axis)
#endif
  }
  else
  {
    num_magn_errors++;
    if (output_errors) Serial.println("!ERR: reading magnetometer");
  }
}

void Gyro_Init()
{
  // Power up reset defaults
  Wire.beginTransmission(GYRO_ADDRESS);
  WIRE_SEND(0x3E);
  WIRE_SEND(0x80);
  Wire.endTransmission();
  delay(5);

  // Select full-scale range of the gyro sensors
  // Set LP filter bandwidth to 42Hz
  Wire.beginTransmission(GYRO_ADDRESS);
  WIRE_SEND(0x16);
  //WIRE_SEND(0x1B);  // DLPF_CFG = 3, FS_SEL = 3
  WIRE_SEND(0x1B);  // DLPF_CFG = 3, FS_SEL = 3
  Wire.endTransmission();
  delay(5);

  // Set sample rato to 50Hz
  Wire.beginTransmission(GYRO_ADDRESS);
  WIRE_SEND(0x15);
  //WIRE_SEND(0x0A);  //  SMPLRT_DIV = 10 (50Hz)
```

```
  WIRE_SEND(0x0A);  //  SMPLRT_DIV = 80 (50Hz)
  Wire.endTransmission();
  delay(5);

  // Set clock to PLL with z gyro reference
  Wire.beginTransmission(GYRO_ADDRESS);
  WIRE_SEND(0x3E);
  WIRE_SEND(0x00);
  Wire.endTransmission();
  delay(5);
}

// Reads x, y and z gyroscope registers
void Read_Gyro()
{
  int i = 0;
  byte buff[6];

  Wire.beginTransmission(GYRO_ADDRESS);
  WIRE_SEND(0x1D);  // Sends address to read from
  Wire.endTransmission();

  Wire.beginTransmission(GYRO_ADDRESS);
  Wire.requestFrom(GYRO_ADDRESS, 6);  // Request 6 bytes
  while(Wire.available())  // ((Wire.available())&&(i<6))
  {
    buff[i] = WIRE_RECEIVE();  // Read one byte
    i++;
  }
  Wire.endTransmission();

  if (i == 6)  // All bytes received?
  {
    gyro[0] = -1 * ((((int) buff[2]) << 8) | buff[3]);    // X axis (internal sensor -
y axis)
    gyro[1] = -1 * ((((int) buff[0]) << 8) | buff[1]);    // Y axis (internal sensor -
x axis)
    gyro[2] = -1 * ((((int) buff[4]) << 8) | buff[5]);    // Z axis (internal sensor -
z axis)
  }
  else
  {
    num_gyro_errors++;
    if (output_errors) Serial.println("!ERR: reading gyroscope");
  }
}
```

Appendix M –Arduino PWM Generator Code

```
#include <math.h>
int loop_count = 0;
char a_3= '0';
int m =0;
double Motor_1_spped=0;
double Motor_2_spped=0;
double Motor_3_spped=0;
double Motor_4_spped=0;
double Motor_5_spped=0;
double Motor_6_spped=0;
int Motor_Voltage[6];
int MOTOR_Spped_Voltage_Ratio=570;
int ESC_Setup = 1500;//19661;//1500;
int PWM_RESOLUTION = 5000;//65535;//5000;
double PROP_Thrust_Coef =0.583368;//0.675654;//4.05117;//46.8397;//
4.05117;//9.62241;//1450;//9.62241;// 9.62241 * 10^-6 the 10^-6  added in
code//0.2;//0.01;
double PROP_Rotation_Coef = 0.00623535;//1;//450;//0.2975;// lose 6 decimal
placesas this data type only can save 6 -7 numbers in a number so add 10^-6
further down in code//2.975/10000000; //0.001;
int l = 0;
char Motor_PWM[21] ;
char U_1[4];
char U_2[4];
char U_3[4];
char U_4[4];
char u_1[4];
char u_2[4];
char u_3[4];
int u_4[4];
int i;
int k;
char a;
char character='0';
int set;
int check = 0;
int j;
int ROLL_SIGN = 0;
void setup()
{
	pinMode(13, OUTPUT);
	digitalWrite(13, HIGH);
	Serial.begin(115200);//57600);
	Serial.flush();
	Serial1.begin(115200);
	Serial2 .begin(57600);  // Baud rate fixed....
	Serial.flush();
	Serial1.flush();
	Serial2.flush();

	pinMode(2,OUTPUT);
	pinMode(3,OUTPUT);
```

```
	pinMode(4,OUTPUT);
	TCCR3A= 0b10101010;
	TCCR3B= 0b00011010;// changed to 8 prescaller//0b00011011;//64 BIT
PRESCALLER  fast
	OCR3AH= ESC_Setup>>8;//MIN
	OCR3AL= ESC_Setup;
	OCR3BH= ESC_Setup>>8;//MIN
	OCR3BL= ESC_Setup;

	ICR3H= PWM_RESOLUTION>>8; //MAX VALUE IF THIS VALUE IS REACHE THE SERVO
WILL MOVE 90 DEGREES CLOCKWISE PULSE WIDTH IS 2 MS
	ICR3L= PWM_RESOLUTION;
	ICR1=64;

	pinMode(8,OUTPUT);
	pinMode(6,OUTPUT);
	pinMode(7,OUTPUT);
	TCCR4A= 0b10101010;
	TCCR4B= 0b00011010;//changed to 8 prescaller //0b00011011;//64 prescaller

	pinMode(9,OUTPUT);
	pinMode(10,OUTPUT);

	pinMode(12,OUTPUT);
	pinMode(11,OUTPUT);

	pinMode(44,OUTPUT);
	pinMode(45,OUTPUT);
	pinMode(46,OUTPUT);
	TCCR5A= 0b10101010;
	TCCR5B= 0b00011010;
	OCR5CH= ESC_Setup>>8;//MIN
	OCR5CL= ESC_Setup;
	OCR5BH= ESC_Setup>>8;//MIN
	OCR5BL= ESC_Setup;

	ICR5H= PWM_RESOLUTION>>8; //MAX VALUE IF THIS VALUE IS REACHE THE SERVO
WILL MOVE 90 DEGREES CLOCKWISE PULSE WIDTH IS 2 MS
	ICR5L= PWM_RESOLUTION;
	DDRB=0xFF;
	DDRC=0xFF;
	DDRE=0xFF;

	analogReference(DEFAULT);

	Serial2.flush();


	Serial.println("SETUP DONE !");
	Serial.flush();

	Serial.println("Start Flight");
	delay(10);
	Motor_Voltage[0] = 2300;
	Motor_Voltage[1] = 2300;
	Motor_Voltage[2] = 2300;
	Motor_Voltage[3] = 2300;

	OCR3AH= (Motor_Voltage[2] ) >>8;//MIN
```

```
	OCR3AL= (Motor_Voltage[2] ) ;
	// PIN 44 MOTOR 1 used to bePIN 11 MOTOR 1 as PIN 5 is stuffed
	OCR5CH= (Motor_Voltage[0] ) >>8;//MIN
	OCR5CL= (Motor_Voltage[0] ) ;
	// PIN 3 motor 2 used to be 2
	OCR3BH= Motor_Voltage[1] >>8;//MIN
	OCR3BL= Motor_Voltage[1] ;
	OCR5BH= Motor_Voltage[3]>>8;//MIN
	OCR5BL= Motor_Voltage[3];
	delay(50);
}

void loop() {
	j=0;
	k=0;
	check = 0;
	int i = 0;
	char character='0';
	char Motor_PWM[21] ;
	if(Serial1.available()>21){
		char b = Serial1.read();
	}
	if(Serial1.available()==21)
	{
		character = Serial1.read();
		while(character != ']')
		{
			if (Serial1.available()>0)
				{
					Motor_PWM[i] = Serial1.read();
					character = Motor_PWM[i];
					i++;
				}
		}

		for(i =0; i<20; i++)
			{
				if(Motor_PWM[i] == '[')
				{
					i++;
					j++;
				}
				if (check ==0)
				{
					k=i-j;
					if (i <5)
					{
						U_1[k] =Motor_PWM[i];
					}
					if (Motor_PWM[i] == ';')
					{
						//U_1[k] = '\0';
						i++;
						check=1;
						m = i;
						Motor_Voltage[0] = atoi(U_1);
					}
				}
				if (check ==1)
```

```
                    {
                            k=i-m;
                            U_2[k] =Motor_PWM[i];
                            if (Motor_PWM[i] == ';')
                            {
                                    i++;
                                    check=2;
                                    m = i;
                                    Motor_Voltage[1] = atoi(U_2);
                            }
                    }
                    if (check ==2)
                    {
                            k=i-m;
                            U_3[k] =Motor_PWM[i];
                            if (Motor_PWM[i] == ';')
                            {
                                    i++;
                                    check=3;
                                    m = i;
                            }
                    }
                    if (check ==3)
                    {
                            k=i-m;
                            U_4[k] =Motor_PWM[i];
                            if (Motor_PWM[i] == ']')
                            {
                                    Motor_PWM[i] ='\n';
                                    i++;
                                    check=4;
                                    m = i;
                                    Motor_Voltage[3] = atoi(U_4);
                            }
                    }
            }

        Serial1.flush();
        u_1[0] = U_1[0];
        u_1[1] = U_1[1];
        u_1[2] = U_1[2];
        u_1[3] = U_1[3];
        //Motor_Voltage[0] = atoi(u_1);
        u_2[0] = U_2[0];
        u_2[1] = U_2[1];
        u_2[2] = U_2[2];
        u_2[3] = U_2[3];

        u_3[0] = U_3[0];
        u_3[1] = U_3[1];
        u_3[2] = U_3[2];
        u_3[3] = U_3[3];

        Motor_Voltage[2] = atoi(u_3);

        loop_count = loop_count + 1;
        a_3 = Motor_PWM[18];
        Serial.println(loop_count);
    } //if avaliable
```

```
        // PIN 5 motor 5 used to be 1
        OCR3AH= (Motor_Voltage[2] ) >>8;//MIN
        OCR3AL= (Motor_Voltage[2] ) ;
        // PIN 44 MOTOR 1 used to bePIN 11 MOTOR 1 as PIN 5 is stuffed
        OCR5CH= (Motor_Voltage[0] ) >>8;//MIN
        OCR5CL= (Motor_Voltage[0] ) ;
        // PIN 3 motor 2 used to be 2
        OCR3BH= Motor_Voltage[1] >>8;//MIN
        OCR3BL= Motor_Voltage[1] ;
        OCR5BH= Motor_Voltage[3]>>8;//MIN
        OCR5BL= Motor_Voltage[3];
}
```

Appendix N –Physical Implementation Code, Deterministic Backstepping Controller for Quadrotor Attitude Control

```
clear
clc
End_Time = 10
results_1= zeros(11,78000);
Time_steps = 78000
results                  = zeros(32, Time_steps);
results_initial          = zeros(7, Time_steps);

delete(instrfindall);
Yaw_count = 0;
Pitch_count = 0;
Roll_count = 0;
time_count =0;
coma_count =0;
Pitch = ['0' '0' '0' '0' '0' '0'];
Roll =['0' '0' '0' '0' '0' '0'];
Yaw = ['0' '0' '0' '0' '0' '0'];
Delta_time = ['9' '9' '9' '9' '9' '9'];
P = ['0' '0' '0' '0' '0' '0'];
Q = ['0' '0' '0' '0' '0' '0'];
R = ['0' '0' '0' '0' '0' '0'];
info =['0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0'
'0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0'];
time = 0;
TIME = 0;
 old_time =0;
%connect to serial port and set baud rate
%s1 = serial('COM4','BaudRate', 115200);
s1 = serial('COM5','BaudRate',115200)%57600);
fopen(s1);
fprintf(s1, '*IDN?');

s2 = serial('COM6','BaudRate',115200)%57600);
fopen(s2);
fprintf(s2, '*IDN?');

% open connection and send identification (to initalize it or something???)

Delta_time = 10*10^-3;
%number of points to be plotted
numberOfPoints = 50000;
```

```
%initialize with zeros, to hold yvalues
yVal= zeros(1,numberOfPoints);


flushinput(s1);
flushinput(s2);
%delay or pause so serial data doesnt stutter
%pause(1);
   a=datevec(now);
flushoutput(s1);
flushoutput(s2);
fprintf(s1, '%s','1');
fprintf(s2, '%s','1');
%pause (10)
TIME_START =(60* a(5)+a(6))  ;% cputime;
%pause (2)
initial_loop_start=9;
num = 0;
count_2 = 0;
count_3 = 0;
%%
PROP_Thrust_Coef          =  2.5569*10^-5;%2*  5.83368*10^-6;  %0.583368*10^-6;
PROP_Rotation_Coef     =5.7768*10^-6;% PROP_Thrust_Coef /10 ; %0.27 * 10^-6;%0.00623535 *
10^-6;
%PROP_Thrust_Coef          =  9.885*10^-6;
%PROP_Power_Coef           =
%PROP_Rotation_Coef     = 1.77*10^-6;%PROP_Power_Coef   / (2 * pi);
%% model paramters physical
Copter_Radius              = 0.30;
Copter_Radius_Along_X_Axis  = 0.26;
Copter_Radius_Along_Y_Axis  = 0.15;
Copter_Radius              = 0.225;
Copter_Radius_Along_X_Axis  =  0.225/2^0.5;
Copter_Radius_Along_Y_Axis  = 0.225/2^0.5;
g   = 9.81;
IXX =     0.0823;
IYY =     0.0539;
IZZ =     0.2169;
IXX = 0.022;
IYY = 0.022;
IZZ = 0.08
IXX = 0.016507;
IYY = 0.016507;
IZZ = 0.016284
m = 2.23;
%m = 1;
JTP =  2.5172e-006;
copterradius = Copter_Radius;
pitch =     4;
Propeller_radius =        (7.8/2)*.75;
alpha = atan(pitch/(2*pi*Propeller_radius ));
alpha = alpha * 180/pi;
Blade_pitch_angle = alpha ;
INCH_to_meter          = 0.0254    ;
BLade_diametre         = 6         ;
L                      = BLade_diametre * INCH_to_meter/2  ;
 p=1.1839; %     wikipedia at 25 degrees C
Blade_leangth          = L;
Effective_blade_leangth = 0.75*Blade_leangth     ;
```

```
air_density     =1.1839;
Hex_copter_height   =   1;
I_H          = [ IXX, 0, 0;
               0, IYY, 0;
               0, 0, IZZ];
%% mathamtical deffinitions
T_inverse_Phi_derivative    = zeros(3, 3);
T_inverse_Theta_derivative  = zeros(3, 3);
Plot_info   = zeros(6, 1);
T_inverse                   = zeros(3, 3);
T_inverse_inverse           = zeros(3, 3);
%% state vatrable decleration
angles                        = zeros(3, 1);
T_Collision                   = zeros(3, 1);
Possition                     = zeros(6, 1);
Possition_OLD                =  zeros(6, 1);
Desired_Possition_OLD =  zeros(6, 1);
 Motor_Speed                 =  zeros(6, 1);
Zeta                          =  zeros(6, 1);
E_Frame_Linear_possition        = zeros(3, 1);
E_Frame_Linear_velocity         = zeros(3, 1);
E_Frame_angulare_velocity   = zeros(3,1);
B_Frame_angulare_velocity   = zeros(3,1);
eta_2_d_old                       = [0;0;0];
eta_2_d_dot_old                   = [0;0;0];
%% for motor propeller dynamics
Motor_Speed_to_Control_Input = [
PROP_Thrust_Coef,PROP_Thrust_Coef,PROP_Thrust_Coef,PROP_Thrust_Coef,PROP_Thrust_Co
ef,PROP_Thrust_Coef;
                                        -PROP_Thrust_Coef*copterradius, -
PROP_Thrust_Coef*Copter_Radius_Along_Y_Axis,
PROP_Thrust_Coef*Copter_Radius_Along_Y_Axis, PROP_Thrust_Coef*copterradius,
PROP_Thrust_Coef*Copter_Radius_Along_Y_Axis, -
PROP_Thrust_Coef*Copter_Radius_Along_Y_Axis;
                                        0, -PROP_Thrust_Coef*Copter_Radius_Along_X_Axis, -
PROP_Thrust_Coef*Copter_Radius_Along_X_Axis, 0,
PROP_Thrust_Coef*Copter_Radius_Along_X_Axis,
PROP_Thrust_Coef*Copter_Radius_Along_X_Axis;
                                        PROP_Rotation_Coef*copterradius,-
PROP_Rotation_Coef*copterradius,PROP_Rotation_Coef*copterradius,-
PROP_Rotation_Coef*copterradius,PROP_Rotation_Coef*copterradius,-
PROP_Rotation_Coef*copterradius ] ;
Control_Input_to_Motor_Speed  = [ 1/(6*PROP_Thrust_Coef), -
1/(2*PROP_Thrust_Coef*(copterradius + Copter_Radius_Along_Y_Axis)),          0,
Copter_Radius_Along_Y_Axis/(2*PROP_Rotation_Coef*copterradius*(copterradius +
Copter_Radius_Along_Y_Axis)) ;
                                        1/(6*PROP_Thrust_Coef), -1/(4*PROP_Thrust_Coef*(copterradius
+ Copter_Radius_Along_Y_Axis)), -1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef),  -
copterradius/(4*PROP_Rotation_Coef*copterradius*(copterradius + Copter_Radius_Along_Y_Axis));
                                        1/(6*PROP_Thrust_Coef),  1/(4*PROP_Thrust_Coef*(copterradius
+ Copter_Radius_Along_Y_Axis)), -1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef),
copterradius/(4*PROP_Rotation_Coef*copterradius*(copterradius + Copter_Radius_Along_Y_Axis));
                                        1/(6*PROP_Thrust_Coef),  1/(2*PROP_Thrust_Coef*(copterradius
+ Copter_Radius_Along_Y_Axis)),          0, -
Copter_Radius_Along_Y_Axis/(2*PROP_Rotation_Coef*copterradius*(copterradius +
Copter_Radius_Along_Y_Axis));
                                        1/(6*PROP_Thrust_Coef),  1/(4*PROP_Thrust_Coef*(copterradius
+ Copter_Radius_Along_Y_Axis)),  1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef),
copterradius/(4*PROP_Rotation_Coef*copterradius*(copterradius + Copter_Radius_Along_Y_Axis));
```

```
                                        1/(6*PROP_Thrust_Coef), -1/(4*PROP_Thrust_Coef*(copterradius
+ Copter_Radius_Along_Y_Axis)),  1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef),  -
copterradius/(4*PROP_Rotation_Coef*copterradius*(copterradius + Copter_Radius_Along_Y_Axis))] ;

Control_Input_to_Motor_Speed_2 = [ 1/(4*PROP_Thrust_Coef), -
1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef),
1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef),
1/(4*Copter_Radius*PROP_Rotation_Coef) ;
                                        1/(4*PROP_Thrust_Coef),
1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef),
1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef), -
1/(4*Copter_Radius*PROP_Rotation_Coef) ;
                                        1/(4*PROP_Thrust_Coef),
1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef), -
1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef),
1/(4*Copter_Radius*PROP_Rotation_Coef) ;
                                        1/(4*PROP_Thrust_Coef),  -
1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef),  -
1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef), -
1/(4*Copter_Radius*PROP_Rotation_Coef)];

Max_Motor_Speed =14.7*1100*2*pi/60;
Min_PWM            = 2350;
PWM_range          = 4800 - Min_PWM;

%% results_1 gathering deffinitions
Lyapunov_bound = zeros(2, Time_steps);
%results_1                    = zeros(31, Time_steps);
Refference_Signal_Time_Steps =20;
Input                    = zeros(4, Time_steps);
Input_rate              = zeros(4, Time_steps);
Refference_signal                    = zeros(6, Refference_Signal_Time_Steps );
%% control and observer gain declaration
xi_1 = [0; 0; 0];
K_1 = [ 10 0 0;
        0 10 0;
        0 0 2] ;
K_2 = [10 0 0;
        0 10 0;
        0 0 2] ;
K_d = [ 10 0 0;
        0 10 0;
        0 0 10];

K_1 = [ 12.1 0 0;
        0 12.5 0;
        0 0 4.5];%2.5] ;
K_2 = [6.1 0 0;
        0 6.1 0;
        0 0 2.35];%0*12]%2.5] ;
K_d = [ 0.9 0 0;
        0 0.9 0;
        0 0 1];%2];
Epsilon_1 = .010;
sigma_1 = 5*10^-4;
tau_aero_dot_max = 1
gamma_1 = 2 * (norm(0.5*inv(I_H )*inv(K_2)*inv(I_H)+sigma_1 * K_d) - sigma_1 * Epsilon_1);
a = [ 1,min(eig(K_1)), min(eig(K_2)), gamma_1/2];
b = [ 1 , sigma_1 ];
c = min(a) /max(b);
```

```
lambda = sigma_1 * (tau_aero_dot_max^2)/(4*Epsilon_1)
lambda / c;
Control_Input = [0;0;0;0];
Loop_Count            = 1;
Loop_Count_initial   = 1;

K_1 =.38838782*K_1;
K_2 =0.86974920* K_2;
K_d =0.9*K_d;
initial_state = 1;
%yaw_initial = 1
%pause(10)
q_old=0;
p_old=0;
r_old=0;
roll = 0;
Angles = [0;0;0];
while (time <=1.0)
   if initial_loop_start == 9
      fprintf(s1, '%s','1');
      info = fscanf(s1)
     while  (info(1) ~='#')
        info = fscanf(s1);
        %yaw_initial =1;
     end
     a=datevec(now);
     TIME =(60*a(5)+a(6)) -TIME_START;
     old_time= TIME -0.02;
   end
%      flushinput(s2);
   a=datevec(now);

TIME =(60*a(5)+a(6)) -TIME_START;

      a2=datevec(now);
TIME2 =(60*a2(5)+a2(6)) -TIME_START;
   if(initial_loop_start == 10)
      info = fscanf(s1);

   info_1 = info;
   i=0;
   j=0;
  coma_count =0;

size_of_info =size(info);
     size_of_info = size_of_info(1,2);
  while  ( size_of_info<37)
     info = fscanf(s1);
     size_of_info =size(info);
     size_of_info = size_of_info(1,2);
     count_3 =count_3 + 1;
  end
   end
   Angles_last_2  = Angles;
        a3=datevec(now);
TIME3 =(60*a3(5)+a3(6)) -TIME_START;

  ed = TIME3 - TIME2;
  if (ed >= 0.025)
     ed;
```

```
  end
flushinput(s1);
%fprintf(s1, '%s','1');

        size_of_info =size(info);
       size_of_info = size_of_info(1,2);

full_stop_count =0;
first_num = 1;
i=0;
     for i = 1:5
         if (info(i) =='#')
             first_num = first_num+1;
         end
         if (info(i) =='Y')
             first_num = first_num+1;
         end
         if (info(i) =='P')
             first_num = first_num+1;
         end
         if (info(i) =='R')
             first_num = first_num+1;
         end
         if (info(i) =='=')
             first_num = first_num+1;
         end
     end

    for i =first_num :size_of_info
        if (coma_count == 0) && (info(i) ~= ',')
            Yaw(i-(first_num-1)) = info(i);
             Yaw_count = i;
        end
        if (coma_count == 1) && (info(i) ~= ',')
            Pitch(i- Yaw_count-1) = info(i);
            Pitch_count = i;
        end
        if (coma_count == 2) && (info(i) ~= ',')
            Roll(i-Pitch_count-1) = info(i);
            Roll_count = i;
        end
        if (coma_count == 3) && (info(i) ~= ',')
            P(i-Roll_count-1) = info(i);
           P_count = i;
        end
        if (coma_count == 4) && (info(i) ~= ',')
            Q(i-P_count -1) = info(i);
           Q_count = i;
        end
        if (coma_count == 5) && (info(i) ~= ',')
            R_sensor(i-Q_count-1) = info(i);
            R_sensor_count = i;
            if info(i) == '!'
                R_sensor(i-Q_count-1)  ='0';
                i=size_of_info;
                coma_count=6;
            end
        end
        if info(i) == ','
            coma_count = coma_count+1;
```

```
        end
    end

    Delta_time = str2double(Delta_time );
    Delta_time = Delta_time /1000;
    pitch = str2double(Pitch);
    roll = str2double(Roll);
    yaw = str2double(Yaw);

    if(yaw>1000)
        yaw
        Yaw
        info
    end

    if (initial_loop_start == 9)
        yaw_temp= yaw;
        initial_loop_start = 10;

    end
   % yaw= yaw  - yaw_temp;
    p = str2double(P);
    q = str2double(Q);
    r = str2double( R_sensor);

    Delta_time = TIME -  old_time;
    if initial_state == 1
        %yaw_initial = yaw
        %pitch_initial = pitch;
        roll_initial = roll;
        p_initial = p;
        q_initial = q;
        r_initial = r;
        initial_state = 0;
    end
    %yaw = yaw - yaw_initial;
    p      = p ;%- p_initial ;
    q      = q ;%q_initial ;
    r      = r ;% r_initial
%    yaw = 0;
   %p=0;
  % q=0;
  %  r=0;
    if (abs(p))>10
        p =p_old;
    end
    if(abs(q))>10
        q=q_old;
    end
    if (abs(r))>10
        r= r_old;
    end
    %yaw= 0;
   %roll= 0;
%pitch=0;
    if (Delta_time  >=0.05)
        TIME ;
         old_time ;
        Delta_time;
        count_2 = count_2 +1;
```

```
        num;
    %   info_1;
    end
    count_3 = 0;
    old_time = TIME;
                Pitch = ['0' '0' '0' '0' '0' '0'];
                Roll =['0' '0' '0' '0' '0' '0'];
                Yaw = ['0' '0' '0' '0' '0' '0'];
                P = ['0' '0' '0' '0' '0' '0'];
                Q = ['0' '0' '0' '0' '0' '0'];
                R = ['0' '0' '0' '0' '0' '0'];
                %Delta_time = ['0' '0' '0' '0' '0' '0'];
                info =['0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0'
'0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0'];

                Yaw_count = 0;
                Pitch_count = 0;
                Roll_count = 0;
                P_count = 0;
                Q_count = 0;
                R_count = 0;
                time_count =0;
                Angles= [pitch;roll;yaw];
    results_initial(:,Loop_Count_initial) = [pitch; roll ;yaw; time; p;q;r];

     time = time + Delta_time;
     %results_initial(:,Loop_Count_initial) = [p; q ;r];
     Loop_Count_initial= Loop_Count_initial +1;
end
i = 0;
%Loop_Count_initial = Loop_Count_initial - 1
for i=(Loop_Count_initial-1):Time_steps
    results_initial(:,i)=  results_initial(:,(Loop_Count_initial-1)) ;
end
sum = [0 ; 0 ; 0];
for i=1:Loop_Count_initial
    sum = sum + [results_initial(5,Loop_Count_initial)
;results_initial(6,Loop_Count_initial);results_initial(7,Loop_Count_initial)];
end

figure
plot(results_initial(4,:) ,results_initial(1,:), ...
                'LineWidth',2)
title('Attitude initial Vs Time')
hold on
plot(results_initial(4,:) ,results_initial(2,:),'g', ...
                'LineWidth',2)
hold on
plot(results_initial(4,:) ,results_initial(3,:),'r', ...
                'LineWidth',2)
legend({'\it{\phi} \rm{(Pitch)}','\it{\theta} \rm{(Roll)}','\it{\psi} \rm{(Yaw)}'}, 'FontSize',16)
xlabel('Time(s)')
ylabel('Radians')
%axis([0, End_Time, - 3, 9])
hold off

Angles_old  =        [pitch; roll; yaw];
Angulare_rate_offset = sum /Loop_Count_initial ;
B_Frame_angulare_velocity_old =Angulare_rate_offset;
yaw_initial  = yaw;
```

```
pitch_initial = pitch
roll_initial = roll;
time = 0
q=9
if(abs(yaw_initial  )>=180)
   yaw_initial
   break
   time = 1000000;
end
tester  = 1;
pitch_old = pitch;
 Angles_last =Angles;
 asdasdasdasdasdasd = Angles
 Angles_new_last = Angles_last
 Angles_new_last_2 = Angles_last_2

 a=datevec(now);
       asqw =  (60*a(5)+a(6)) -TIME_START;

IMU_time_last = (60*a2(5)+a2(6)) -TIME_START-0.02;

 a=datevec(now);
       TIME_START=  (60*a(5)+a(6)) ;
       IMU_time_last =0
       old_time=0;
while (time <= End_Time)
   if initial_loop_start == 9
       %fprintf(s1, '%s','1');
       info = fscanf(s1);
      while  (info(1) ~='#')
          info = fscanf(s1);
          %yaw_initial =1;
      end
      initial_state = 1;
      a=datevec(now);
      TIME =(60*a(5)+a(6)) -TIME_START;
      old_time= TIME -0.02;
   end

   a2=datevec(now);
   TIME2 =(60*a2(5)+a2(6)) -TIME_START;
   imu_read_time  = TIME2 - IMU_time_last;
   initial_loop_start = 10;
   if initial_loop_start == 10
       while imu_read_time < 0.018
           a2=datevec(now);
           TIME2 =(60*a2(5)+a2(6)) -TIME_START;
           imu_read_time  = TIME2 - IMU_time_last;
       end
       [imu_read_time, IMU_time_last];
       if imu_read_time >= 0.018
           info = fscanf(s1);
            %flushinput(s1);
           info_1 = info;
           i=0;
           j=0;
           coma_count =0;
           size_of_info =size(info);
           size_of_info = size_of_info(1,2);
           while  ( size_of_info<35)
```

```
            info = fscanf(s1);
            size_of_info =size(info);
            size_of_info = size_of_info(1,2);
            count_3 =count_3 + 1;
         end
         a=datevec(now);
         IMU_time_last =(60*a(5)+a(6)) -TIME_START;
      end
   end
   a3=datevec(now);
   TIME3 =(60*a3(5)+a3(6)) -TIME_START;
   ed = TIME3 - TIME2;
   if (ed >= 0.025)
      ed;
   end
   if (tester  == 1)
      tester  = 2;
      info_3 = info;
   end
   size_of_info =size(info);
   size_of_info = size_of_info(1,2);
   full_stop_count =0;
   first_num = 1;
   for i = 1:5
      if (info(i) =='#')
         first_num = first_num+1;
      end
      if (info(i) =='Y')
         first_num = first_num+1;
      end
      if (info(i) =='P')
         first_num = first_num+1;
      end
      if (info(i) =='R')
         first_num = first_num+1;
      end
      if (info(i) =='=')
         first_num = first_num+1;
      end
   end
   info;
   first_num;
   for i =first_num:size_of_info
      if (coma_count == 0) && (info(i) ~= ',')
         Yaw(i-first_num+1) = info(i);
          Yaw_count = i;
      end
      if (coma_count == 1) && (info(i) ~= ',')
         Pitch(i- Yaw_count-1) = info(i);
         Pitch_count = i;
      end
      if (coma_count == 2) && (info(i) ~= ',')
         Roll(i-Pitch_count-1) = info(i);
         Roll_count = i;
      end
      if (coma_count == 3) && (info(i) ~= ',')
         P(i-Roll_count-1) = info(i);
        P_count = i;
      end
      if (coma_count == 4) && (info(i) ~= ',')
```

```
            Q(i-P_count -1) = info(i);
           Q_count = i;
        end
        if (coma_count == 5) && (info(i) ~= ',')
            R_sensor(i-Q_count-1) = info(i);
            R_sensor_count = i;
            if info(i) == '!'
                R_sensor(i-Q_count-1)  ='0';
                i=size_of_info;
                coma_count=6;
            end
        end
        if info(i) == ','
            coma_count = coma_count+1;
        end
    end

    Delta_time = str2double(Delta_time );
    Delta_time = Delta_time /1000;
    pitch = str2double(Pitch);
    roll = str2double(Roll);
    yaw = str2double(Yaw);
    if initial_loop_start == 9
        yaw_temp= yaw;
        yaw_initial = yaw_temp;
        initial_loop_start = 10;
    end
    q = str2double(P);
    p = str2double(Q);
    r = str2double( R_sensor);
    Delta_time = TIME -  old_time;
    if initial_state == 1
        p_initial = p;
        q_initial = q;
        r_initial = r;
        initial_state = 0;
    end
    yaw = yaw - yaw_initial;
    if (abs(p))>10
        p =p_old;
    end
    if(abs(q))>10
        q=q_old;
    end
    if (abs(r))>10
        r= r_old;
    end
    if (Delta_time  >=0.05)
        TIME ;
         old_time ;
        Delta_time;
        count_2 = count_2 +1;
        num;
        info_1;
    end
    count_3 = 0;
    old_time = TIME;
    Pitch = ['0' '0' '0' '0' '0' '0'];
    Roll =['0' '0' '0' '0' '0' '0'];
    Yaw = ['0' '0' '0' '0' '0' '0'];
```

```
    P = ['0' '0' '0' '0' '0' '0'];
    Q = ['0' '0' '0' '0' '0' '0'];
    R = ['0' '0' '0' '0' '0' '0'];
    %Delta_time = ['0' '0' '0' '0' '0' '0'];
    info =['0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0'
'0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0'];

    Yaw_count = 0;
    Pitch_count = 0;
    Roll_count = 0;
    P_count = 0;
    Q_count = 0;
    R_count = 0;
    time_count =0;

     time = time + Delta_time;
     if(isnan(p) == 1)
         break
         time = 100000000;
         info
     end
    if(isnan(q) == 1)
        break
        time = 100000000;
        info
    end
    if(isnan(r) == 1)
        break
        time = 100000000;
        info
    end
    if(isnan(pitch) == 1)
        break
        time = 100000000;
        info
    end
    if(isnan(roll) == 1)
        break
        time = 100000000;
        info
    end
    if(isnan(yaw) == 1)
        break
        time = 100000000;
        info
    end
    B_Frame_angulare_velocity = [ p;
                                  q;
                                  r];
                                alpha = 0.3;
    if yaw <-180
        yaw =yaw +360;
    end
    wn = 2*pi;
    zeta =0.0001;

    w_d = pi*0.1;
    alpha = tanh(w_d/2);
    k = ((alpha)^2)/(1+(2^0.5)*alpha +alpha ^2);
    alpha_1 = 2*(alpha^2-1)/(1+(2^0.5)*alpha+alpha^2);
```

```
    alpha_2 = (1-(2^0.5)*alpha+alpha^2)/(1+(2^0.5)*alpha+alpha^2);
    b_1 = 2;
    b_2= 1;
    Angles                              = [ pitch;
                                            roll;
                                           yaw];
    Angles_new =k*Angles +k*b_1*Angles_last + k*b_2*Angles_last_2 - alpha_1* Angles_new_last-
alpha_2*Angles_new_last_2;

    Angles_last_2  = Angles_last;
    Angles_last   = Angles;
    Angles_new_last_2 = Angles_new_last;
    Angles_new_last = Angles_new;
     Angles  = Angles  *pi/180;
          Angles(1,1)*180/pi
    Possition  = [ 0; 0 ;0;
                     Angles];
    Desired_Possition(4,1) =0;%(pi/18)*tanh(t)  ;
    Desired_Possition(5,1) = 0;%-(pi/18) *tanh(t);
    Desired_Possition(6,1) =0;% 0.1*t ;




    Desired_Possition(4,1) =0;% -pitch_initial *(tanh(0.05*time) - 1)*pi/180;%pitch*pi/180;%
    Desired_Possition(5,1) = 0;%-roll_initial *(tanh(0.05*time) - 1)*pi/180;%-roll_initial
*(tanh(0.5*time*abs(pitch_initial /roll_initial )) - 1)*pi/180;%roll*pi/180;
    Desired_Possition(6,1) =0; % Angles(3,1);
%0;%yaw_initial*pi/180;%yaw*pi/180;%0.5*(tanh(0.05*time));% 0.1*t ;
    angles = [  Possition(4,1);
                Possition(5,1);
                Possition(6,1) ];

    R       = rotation(Angles );

    E_Frame_angulare_velocity    =
B_FRAME_ANGULARE_Velocity_TO_E_FRAME_Angulare_velocity(B_Frame_angulare_velocity,
Angles );
    Velocity              = (Possition-Possition_OLD) / Delta_time;
    Desired_Velocity       = (Desired_Possition-Desired_Possition_OLD)/Delta_time;

    if ( Loop_Count  == 1)
       %Velocity =[0;0;0;0;0;0];
       Desired_Velocity = [0;0;0;0;0;0];
    end


    Position_ERROR      = (Possition - Desired_Possition );
    Velocity_ERRPR      = (Desired_Possition-Desired_Possition_OLD)/Delta_time-Velocity;

    eta_2_d_dot          = [ Desired_Velocity(4,1);
                             Desired_Velocity(5,1);
                             Desired_Velocity(6,1) ];
    eta_2_d              = [ Desired_Possition(4,1);
                              Desired_Possition(5,1);
                              Desired_Possition(6,1) ];
    eta_2_d_double_dot    = (eta_2_d_dot -eta_2_d_dot_old) / Delta_time;

    %% Second stage set up converssions between frames of refference
    eta_2_error             =  [ Position_ERROR(4,1);
```

```
                                    Position_ERROR(5,1);
                                    Position_ERROR(6,1)];
   eta_2                = Angles ;
   control_time = time ;
   U = Attitude_Backstepping_Controller_1(I_H,  K_1, K_2, K_d, eta_2, B_Frame_angulare_velocity,
eta_2_d, eta_2_d_dot, eta_2_d_double_dot, Delta_time, xi_1,Loop_Count);
   Controller_time_steps = 0;
   xi_1 = [U(1,1) ; U(2,1); U(3,1)];
   tau_aero_hat = [U(7,1) ; U(8,1); U(9,1)];
   alpha_w = [U(10,1) ; U(11,1); U(12,1)];
   alpha_w_dot  = [0*U(13,1) ; U(14,1);0* U(15,1)];
   u =[ 0.1*m*9.81/R(3,3) ;U(4,1); U(5,1); 1*U(6,1) ];
   U=u;
   w_error                 =   B_Frame_angulare_velocity - alpha_w;


   Motor_Signal = Control_Input_to_Motor_Speed_2 * U;
   for i = 1:4
      if (Motor_Signal(i,1)<0)
         Motor_Signal(i,1) = 0;
      end
      Motor_Signal(i,1)=0*(1/(4*PROP_Thrust_Coef))*m*9.81/R(3,3)+Motor_Signal(i,1);
      Motor_Signal(i,1) =  Min_PWM + PWM_range*(Motor_Signal(i,1)^0.5/Max_Motor_Speed);
      Motor_Signal(i,1) = round(Motor_Signal(i,1) );
     %Motor_Signal(i,1) = 2330;
   end

   flushinput(s2);
   Motor_Signal_String = mat2str(Motor_Signal);
   fprintf(s2,'%s', Motor_Signal_String);
   results_1(:, Loop_Count) = [Angles;
                         time;
                         p;
                         q;
                         r;
                         Motor_Signal];
%% motor spped to propeller thrust force and rotation toruqe
   Control_Input = Motor_and_Propeller_Dynamics(U, Control_Input_to_Motor_Speed,
Motor_Speed_to_Control_Input, Motor_Speed,Loop_Count);
    Control_Input  = U;

       eta_2_error_0 = [0;0;0];
      w_error_0 = [0.2;0.2;0.1];
      tau_error_0 = [ 0.2; 0.2; 0.1];

      V_sum_bound = ((norm(eta_2_error_0))^2 + norm(w_error_0)^2 + sigma_1 *
norm(tau_error_0)^2 -  lambda/c)*exp(-c * time) +   lambda / c;
      V_sum = (norm(eta_2 - eta_2_d))^2 +   (norm(w_error))^2 + sigma_1 * (norm(tau_error))^2;
       Lyapunov_bound(1,Loop_Count) = V_sum_bound;
       Lyapunov_bound(2,Loop_Count) = V_sum;

      if (E_Frame_Linear_possition(3,1) <= 0)
         E_Frame_Linear_possition(3,1) =0;
      end

      Zeta  = [ E_Frame_Linear_velocity;
                 B_Frame_angulare_velocity            ];

   Possition_OLD   = Possition;
   Possition = [ 0;0;0%E_Frame_Linear_possition;
```

```
                        angles                              ];
   Desired_Possition_OLD =     Desired_Possition;

%% For Plotting system information
   Input(:,Loop_Count) = Control_Input;

   if Loop_Count > 1
      Input_rate(:,Loop_Count)  = (Input(:,Loop_Count) - Input(:,(Loop_Count-Controller_time_steps)))
/ Delta_time;
   end
   if Loop_Count == 1
      Input_rate(:,Loop_Count)  = 0;
   end
   alpha = 0.1;
   exponentialMA  = (1-alpha)*pitch_old  +alpha*pitch;
   pitch_old = exponentialMA ;

   results(:,Loop_Count) = [ Possition;
                             time ;
                             B_Frame_angulare_velocity ;
                             tau_aero_hat;
                             w_error;
                              eta_2_error;
                               alpha_w_dot
                               tau_aero_hat;
                               tau_aero_hat;
                               eta_2_d;%   ];%eta_2_noise];
                               exponentialMA ];

   eta_2_d_dot_old          = eta_2_d_dot;
   p_old = p;
   q_old = q;
   r_old = r;
   for i = 1:3
      Refference_signal(i,Loop_Count) = Desired_Possition(i,1) ;
   end
   Refference_signal(4,Loop_Count) = eta_2_d(1,1);
   Refference_signal(5,Loop_Count) = eta_2_d(2,1);
   Refference_signal(6,Loop_Count) = eta_2_d(3,1);
   Loop_Count             = Loop_Count + 1;
   Controller_time_steps = Controller_time_steps + 1;
   %% END OF SIMULATION LOOP%%
   time                        ;
end
 a=datevec(now);
      asqw2 =  (60*a(5)+a(6)) -TIME_START;
      asqw2  - asqw
for i =(Loop_Count-1):Time_steps
   results_1(:, i) = results_1(:, (Loop_Count-1));
end

   a=datevec(now);

TIME_END =(60* a(5)+a(6))  ;% cputime;

delta = TIME_END-TIME_START

   for i = 1:4
      Motor_Signal(i,1) =1500;
   end
```

```
    flushinput(s2);
    Motor_Signal_String = mat2str(Motor_Signal);
    fprintf(s2,'%s', Motor_Signal_String);



time/num
%close the port and delete it
fclose(s1);
delete(s1)
clear('s1')

fclose(s2);
delete(s2)
clear('s2')

delete(instrfindall)

figure
plot(results_1(4,:) ,(180/pi)*results(17,:),'b', ...
            'LineWidth',2)
title('Attitude Error  Vs Time')
hold on
plot(results_1(4,:) ,(180/pi)*results(18,:),'g', ...
            'LineWidth',2)
hold on
plot(results_1(4,:) ,(180/pi)*results(19,:),'r', ...
            'LineWidth',2)
legend({'\it{\phi}_e \rm{(Pitch)}','\it{\theta}_e \rm{(Roll)}','\it{\psi}_e \rm{(Yaw)}'}, 'FontSize',16)
xlabel('Time(s)')
ylabel('Degrees')
hold off

figure
plot(results_1(4,:) ,results_1(1,:), ...
            'LineWidth',2)
title('Attitude  Vs Time')
hold on
plot(results_1(4,:) ,results(32,:),'g', ...
            'LineWidth',2)
         hold off

figure
plot(results_1(4,:) ,(180/pi)*results_1(1,:), ...
            'LineWidth',2)
title('Attitude  Vs Time')
hold on
plot(results_1(4,:) ,(180/pi)*results_1(2,:),'g', ...
            'LineWidth',2)
hold on
plot(results_1(4,:) ,(180/pi)*results_1(3,:),'r', ...
            'LineWidth',2)

hold on
plot(results_1(4,:) ,(180/pi)*results(29,:) ,'b', ...
            'LineWidth',4)
hold on
plot(results_1(4,:) ,(180/pi)*results(30,:) ,'g', ...
            'LineWidth',4)
```

```
hold on
plot(results_1(4,:) ,(180/pi)*results(31,:) ,'r', ...
        'LineWidth',4)
legend({'\it{\phi} \rm{(Pitch)}','\it{\theta} \rm{(Roll)}','\it{\psi} \rm{(Yaw)}'}, 'FontSize',16)
xlabel('Time(s)')
ylabel('Degrees')
%axis([0, End_Time, - 60, 20])
hold off



figure
plot(results(7,:) ,results(11,:), ...
        'LineWidth',2)
title('disturbance observer   Vs Time')
hold on
plot(results(7,:) ,results(12,:),'g', ...
        'LineWidth',2)
hold on
plot(results(7,:) ,results(13,:),'r', ...
        'LineWidth',2)
 set(legend({'$\hat{\tau}_p$','$\hat{\tau}_q$','$\hat{\tau}_r$'}, 'FontSize',20),'interpreter','latex')
xlabel('Time(s)')
ylabel('Distrubance Torque Estimate (Nm)')
hold off
```

```
function U = Attitude_Backstepping_Controller_1(I_H,  K_1, K_2, K_d, eta_2,
B_Frame_angulare_velocity, eta_2_d, eta_2_d_dot, eta_2_d_double_dot, Delta_Time, xi_1,
Loop_Count)
angles                  = eta_2;
T_inv                   = Angular_velocity_cordinant_transform(angles);
T                       = inv(T_inv);
eta_2_dot                 = T * B_Frame_angulare_velocity    ;
eta_2_error              = eta_2 - eta_2_d;
alpha_w                  = T_inv * ( eta_2_d - K_1 * eta_2_error);
w_error                  =   B_Frame_angulare_velocity - alpha_w;
if Loop_Count ==1;
   xi_1=- K_d * I_H * w_error;
end
tor_aero_hat             = xi_1+ K_d * I_H * w_error;
eta_2_error_dot          = K_1 * eta_2_error + T * w_error;
alpha_w_dot              = Calculate_alpha_w_dot_Attitude_Only(K_1, eta_2, eta_2_error, eta_2_d_dot,
eta_2_dot, eta_2_error_dot, eta_2_d_double_dot);
tor                      = I_H*( -T * eta_2_error   -K_2 * w_error + alpha_w_dot ) - tor_aero_hat +
cross(B_Frame_angulare_velocity, (I_H*B_Frame_angulare_velocity));
xi_1_dot                 = -K_d *( -cross(B_Frame_angulare_velocity,
(I_H*B_Frame_angulare_velocity))+tor +tor_aero_hat);
xi_1                     = xi_1_dot * Delta_Time + xi_1;
U                        = [xi_1; tor(1,1); tor(2,1) ; tor(3,1) ; tor_aero_hat; alpha_w];



function alpha_w_dot           = Calculate_alpha_w_dot_Attitude_Only(K_1, eta_2, eta_2_error,
eta_2_d_dot, eta_2_dot, eta_2_error_dot, eta_2_d_double_dot)
phi = eta_2(1,1);
theta = eta_2(2,1);
PHI_DOT = eta_2_dot(1,1);
THETA_DOT = eta_2_dot(2,1);
angles                  = eta_2;
T_inv   = Angular_velocity_cordinant_transform(angles);

T_inverse_Theta_derivative  = [ 0,0 ,-cos(theta);
                                0,0 , -sin(theta)*sin(phi);
                                0,0 ,-sin(theta)*cos(phi) ];
T_inverse_Phi_derivative    = [ 0,0 ,0;
                              0,-sin(phi) , cos(theta)*cos(phi);
                              0,-cos(phi) ,-cos(theta)*sin(phi)   ];
T_inv_dot                = (T_inverse_Phi_derivative*PHI_DOT +
T_inverse_Theta_derivative*THETA_DOT);

alpha_w_dot   = T_inv_dot * (eta_2_d_dot -K_1 * eta_2_error) + T_inv     * (eta_2_d_double_dot -K_1
* eta_2_error_dot


function T_inv          = Angular_velocity_cordinant_transform(angles)
   phi = angles(1);
   theta = angles(2);
   psi = angles(3);
      T_inv =[
      1, 0, -sin(theta)
      0, cos(phi), cos(theta)*sin(phi)
      0, -sin(phi), cos(theta)*cos(phi)
   ];
end
```

```
function Control_Input = Motor_and_Propeller_Dynamics(Control_Input,
Control_Input_to_Motor_Speed, Motor_Speed_to_Control_Input, Motor_Speed,Loop_Count)
    Motor_Speed = Control_Input_to_Motor_Speed * Control_Input ;
    for i = 1:6
        if Motor_Speed(i, 1) >= ((8208*2*pi/60)^2)
            Motor_Speed(i, 1) = (8208*2*pi/60)^2;
        end
            if Motor_Speed(i, 1) <0
                Motor_Speed(i, 1) = 0;
            end
    end
    for i = 1 : 6
        Motor_Speed(i, 1) = Motor_Speed(i, 1)^0.5;
        %MOTOR(i, Loop_Count)  = Motor_Speed(i, 1);
        Motor_Speed(i, 1) = Motor_Speed(i, 1)^2;
    end
    Control_Input = Motor_Speed_to_Control_Input * Motor_Speed ;
    %% motor spped to propeller thrust force and rotation toruqe
    for i = 1:6;
        Blade_pitch_angle = 4;
        A_of_attack = Blade_pitch_angle*pi/180;
    %   [Prop_Thrust(i), Prop_Torque(i)]    =Prop_Aero_Thrust_Rotation_Calc (
total_Propeller_surface_area, Motor_spped(i), air_density , Effective_blade_leangth, A_of_attack);
    end
end
```

Appendix O –Physical Implementation Code, Stochastic Backstepping Controller for Quadrotor Attitude Control

```
clear
%close all
clc
K_1 = [    6.7141         0         0;
                 0    6.9360         0;
                 0         0    4.7165];
%close all
End_Time = 100
results_1= zeros(11,78000);
Time_steps = 78000
results                  = zeros(44, Time_steps);
results_initial          = zeros(7, Time_steps);

delete(instrfindall);
Yaw_count = 0;
Pitch_count = 0;
Roll_count = 0;
time_count =0;
coma_count =0;
Pitch = ['0' '0' '0' '0' '0' '0'];
Roll =['0' '0' '0' '0' '0' '0'];
Yaw = ['0' '0' '0' '0' '0' '0'];
Delta_time = ['9' '9' '9' '9' '9' '9'];
P = ['0' '0' '0' '0' '0' '0'];
Q = ['0' '0' '0' '0' '0' '0'];
R = ['0' '0' '0' '0' '0' '0'];
R_sensor = ['0' '0' '0' '0' '0' '0'];
info =['0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0'
'0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0'];
time = 0;
TIME = 0;
 old_time =0;
%connect to serial port and set baud rate
%s1 = serial('COM4','BaudRate', 115200);
s1 = serial('COM9','BaudRate',115200);%57600);
s1.BytesAvailableFcnCount=35;
fopen(s1);
fprintf(s1, '*IDN?');

s2 = serial('COM10','BaudRate',115200);%115200)%57600);
fopen(s2);
fprintf(s2, '*IDN?');
```

```
% open connection and send identification (to initalize it or something???)



Delta_time = 10*10^-3;
%number of points to be plotted
numberOfPoints = 50000;

%initialize with zeros, to hold yvalues
yVal= zeros(1,numberOfPoints);


flushinput(s1);
flushinput(s2);
%delay or pause so serial data doesnt stutter
%pause(1);
    a=datevec(now);
flushoutput(s1);
flushoutput(s2);
fprintf(s1, '%s','1');
fprintf(s2, '%s','1');
%pause (10)
TIME_START =(60*60*a(4)+60* a(5)+a(6))  ;% cputime;
%pause (2)
initial_loop_start=9;
num = 0;

count_2 = 0;
count_3 = 0;






%%
PROP_Thrust_Coef          =  2.5569*10^-5;%2*  5.83368*10^-6;  %0.583368*10^-6;
PROP_Rotation_Coef     =5.7768*10^-6;% PROP_Thrust_Coef /10 ; %0.27 * 10^-6;%0.00623535 *
10^-6;
%PROP_Thrust_Coef          =  9.885*10^-6;
%PROP_Power_Coef           =
%PROP_Rotation_Coef     = 1.77*10^-6;%PROP_Power_Coef   / (2 * pi);
%% model paramters physical
Copter_Radius              = 0.30;
Copter_Radius_Along_X_Axis  = 0.26;
Copter_Radius_Along_Y_Axis  = 0.15;
Copter_Radius              = 0.225;
Copter_Radius_Along_X_Axis  =  0.225/2^0.5;
Copter_Radius_Along_Y_Axis  = 0.225/2^0.5;
g   = 9.81;
IXX =     0.0823;
IYY =     0.0539;
IZZ =     0.2169;
IXX = 0.022;
IYY = 0.022;
IZZ = 0.08
IXX = 0.016507;
```

```
IYY = 0.016507;
IZZ = 0.016284
m = 2.23;
%m = 1;
JTP =  2.5172e-006;
copterradius = Copter_Radius;
pitch =     4;
Propeller_radius =        (7.8/2)*.75;
alpha = atan(pitch/(2*pi*Propeller_radius ));
alpha = alpha * 180/pi;
Blade_pitch_angle = alpha ;
INCH_to_meter          = 0.0254    ;
BLade_diametre         = 6         ;
L                      = BLade_diametre * INCH_to_meter/2  ;
 p=1.1839; %     wikipedia at 25 degrees C
Blade_leangth          = L;
Effective_blade_leangth = 0.75*Blade_leangth     ;
air_density     =1.1839;
Hex_copter_height   =   1;
I_H         = [ IXX, 0, 0;
               0, IYY, 0;
               0, 0, IZZ];
%% mathamtical deffinitions
T_inverse_Phi_derivative    = zeros(3, 3);
T_inverse_Theta_derivative  = zeros(3, 3);
Plot_info   = zeros(6, 1);
T_inverse                  = zeros(3, 3);
T_inverse_inverse          = zeros(3, 3);
%% state vatrable decleration
angles                       = zeros(3, 1);
T_Collision                  = zeros(3, 1);
Possition                    = zeros(6, 1);
Possition_OLD                =  zeros(6, 1);
Desired_Possition_OLD =  zeros(6, 1);
 Motor_Speed                 =  zeros(6, 1);
Zeta                         =  zeros(6, 1);
E_Frame_Linear_possition       = zeros(3, 1);
E_Frame_Linear_velocity        = zeros(3, 1);
E_Frame_angulare_velocity   = zeros(3,1);
B_Frame_angulare_velocity   = zeros(3,1);
eta_2_d_old                    = [0;0;0];
eta_2_d_dot_old                = [0;0;0];
%% for motor propeller dynamics
Motor_Speed_to_Control_Input = [
PROP_Thrust_Coef,PROP_Thrust_Coef,PROP_Thrust_Coef,PROP_Thrust_Coef,PROP_Thrust_Co
ef,PROP_Thrust_Coef;
                                          -PROP_Thrust_Coef*copterradius, -
PROP_Thrust_Coef*Copter_Radius_Along_Y_Axis,
PROP_Thrust_Coef*Copter_Radius_Along_Y_Axis, PROP_Thrust_Coef*copterradius,
PROP_Thrust_Coef*Copter_Radius_Along_Y_Axis, -
PROP_Thrust_Coef*Copter_Radius_Along_Y_Axis;
                                          0, -PROP_Thrust_Coef*Copter_Radius_Along_X_Axis, -
PROP_Thrust_Coef*Copter_Radius_Along_X_Axis, 0,
PROP_Thrust_Coef*Copter_Radius_Along_X_Axis,
PROP_Thrust_Coef*Copter_Radius_Along_X_Axis;
                                          PROP_Rotation_Coef*copterradius,-
PROP_Rotation_Coef*copterradius,PROP_Rotation_Coef*copterradius,-
PROP_Rotation_Coef*copterradius,PROP_Rotation_Coef*copterradius,-
PROP_Rotation_Coef*copterradius ] ;
```

```
Control_Input_to_Motor_Speed  = [ 1/(6*PROP_Thrust_Coef), -
1/(2*PROP_Thrust_Coef*(copterradius + Copter_Radius_Along_Y_Axis)),          0,
Copter_Radius_Along_Y_Axis/(2*PROP_Rotation_Coef*copterradius*(copterradius +
Copter_Radius_Along_Y_Axis)) ;
                                    1/(6*PROP_Thrust_Coef), -1/(4*PROP_Thrust_Coef*(copterradius
+ Copter_Radius_Along_Y_Axis)), -1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef),  -
copterradius/(4*PROP_Rotation_Coef*copterradius*(copterradius + Copter_Radius_Along_Y_Axis));
                                    1/(6*PROP_Thrust_Coef),  1/(4*PROP_Thrust_Coef*(copterradius
+ Copter_Radius_Along_Y_Axis)), -1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef),
copterradius/(4*PROP_Rotation_Coef*copterradius*(copterradius + Copter_Radius_Along_Y_Axis));
                                    1/(6*PROP_Thrust_Coef),  1/(2*PROP_Thrust_Coef*(copterradius
+ Copter_Radius_Along_Y_Axis)),          0, -
Copter_Radius_Along_Y_Axis/(2*PROP_Rotation_Coef*copterradius*(copterradius +
Copter_Radius_Along_Y_Axis));
                                    1/(6*PROP_Thrust_Coef),  1/(4*PROP_Thrust_Coef*(copterradius
+ Copter_Radius_Along_Y_Axis)),  1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef),
copterradius/(4*PROP_Rotation_Coef*copterradius*(copterradius + Copter_Radius_Along_Y_Axis));
                                    1/(6*PROP_Thrust_Coef), -1/(4*PROP_Thrust_Coef*(copterradius
+ Copter_Radius_Along_Y_Axis)),  1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef),  -
copterradius/(4*PROP_Rotation_Coef*copterradius*(copterradius + Copter_Radius_Along_Y_Axis))] ;

Control_Input_to_Motor_Speed_2 = [ 1/(4*PROP_Thrust_Coef), -
1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef),
1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef),
1/(4*Copter_Radius*PROP_Rotation_Coef) ;
                                        1/(4*PROP_Thrust_Coef),
1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef),
1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef), -
1/(4*Copter_Radius*PROP_Rotation_Coef) ;
                                        1/(4*PROP_Thrust_Coef),
1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef), -
1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef),
1/(4*Copter_Radius*PROP_Rotation_Coef) ;
                                        1/(4*PROP_Thrust_Coef),  -
1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef),  -
1/(4*Copter_Radius_Along_X_Axis*PROP_Thrust_Coef), -
1/(4*Copter_Radius*PROP_Rotation_Coef)];

Max_Motor_Speed =14.7*1100*2*pi/60;

Max_Motor_Speed =17.4*1100*2*pi/60;
Min_PWM            = 2350;
PWM_range          = 4800 - Min_PWM;

%% results_1 gathering deffinitions
Lyapunov_bound = zeros(2, Time_steps);
%results_1                  = zeros(31, Time_steps);
Refference_Signal_Time_Steps =20;
Input                      = zeros(4, Time_steps);
Input_rate                = zeros(4, Time_steps);
Refference_signal                    = zeros(6, Refference_Signal_Time_Steps );
%% control and observer gain declaration
xi_1 = [0; 0; 0];
K_1 = [ 10 0 0;
        0 10 0;
        0 0 2] ;
K_2 = [10 0 0;
        0 10 0;
        0 0 2] ;
K_d = [ 10 0 0;
```

```
          0 10 0;
          0 0 10];

K_1 = [ 12.1 0 0;
          0 12.5 0;
          0 0 8.5];%2.5] ;
K_2 = [6.1 0 0;
          0 6.1 0;
          0 0 2.35];%2.35];%0*12]%2.5] ;
K_d = [ 0.9 0 0;
          0 0.9 0;
          0 0 1];%2];

Epsilon_1 = .010;
sigma_1 = 5*10^-4;
tau_aero_dot_max = 1
gamma_1 = 2 * (norm(0.5*inv(I_H )*inv(K_2)*inv(I_H)+sigma_1 * K_d) - sigma_1 * Epsilon_1);
a = [ 1,min(eig(K_1)), min(eig(K_2)), gamma_1/2];
b = [ 1 , sigma_1 ];
c = min(a) /max(b);
lambda = sigma_1 * (tau_aero_dot_max^2)/(4*Epsilon_1)
lambda / c;
Control_Input = [0;0;0;0];
Loop_Count           = 1;
Loop_Count_initial   = 1;
%{
K_1 =.65048838782*K_1;
K_2 =0.204900996974920* K_2;
K_d = 5.2*K_d;
%}

K_1 =0.5548838782*K_1;
K_2 =0.550* K_2;%0.504900996974920* K_2;
K_d =0.6*K_d;
K_1 = 1*K_1
K_2 = 1*K_2

K_1 = [ 12.1 0 0;
          0 12.5 0;
          0 0 4.5];%2.5] ;
K_2 = [6.1 0 0;
          0 6.1 0;
          0 0 2.35];%0*12]%2.5] ;
K_d = [ 0.9 0 0;
          0 0.9 0;
          0 0 1];%2];

K_1 = [ 12.0 0 0;
          0 12.1 0;
          0 0 4.5];%2.5] ;
%K_1 =.38838782*K_1;
      K_1 =.340000005838782*K_1;
K_2 =0.5000006974920* K_2;
K_d =0.9*K_d;




initial_state = 1;
```

```
%yaw_initial = 1
%pause(10)
q_old=0;
p_old=0;
r_old=0;
roll = 0;
Angles = [0;0;0];

tor_aero_hat_deterministic         = [0 ; 0; 0];
tor_aero_hat_deterministic_dot   = [0 ; 0; 0];
sigma_hat_Stochastic                = 0;
sigma_hat_Stochastic_dot          = 0;

   K_1 =1*[ 10 0 0;
           0 10 0;
           0 0 10];%2.5] ;
K_2 = 0.1*[5 0 0;
           0 5 0;
           0 0 5];%0*12]%2.5] ;
 K_2 = [6.1 0 0;
           0 6.1 0;
           0 0 0*2.35];%2.35];%0*12]%2.5] ;
K_2 =0.550* K_2;%0.504900996974920* K_2;
K_2 = [6.1 0 0;
           0 6.1 0;
           0 0 2.35];%0*12]%2.5] ;
        K_2 =0.86974920* K_2;



D_1 = 0*eye(3);
D_2 = 0*eye(3) ;
D_2 = 0.001 * [ 2.5 0 0;
                     0 2.5 0;
                     0 0 0.3];

                K_1 = 0.79*[ 3.0 0 0;
                        0 3.0 0;
                        0 0 2.2];
K_2 =0.68* [ 5 0 0;
           0 5 0;
           0 0 3];

K_1 = 2.2*K_1;

K_2 = 1.0*K_2;
K_1 = 10*eye(3);
K_2 =5*eye(3);
ROE =0.15*[50 0 0; 0 50 0; 0 0 50];%K_d;
ROE =5*0.115* ROE;
mew_1 = 10;
mew_2 = 57000000;
tor_aero_deterministic_MAX=[1;1;0.1];
tor_aero_Stochastic_MAX_1=0.1^4;
tor_aero_Stochastic_MAX_2= 1;
xi_aero_deterministic=0.005*[10;10;1];
xi_aero_Stochastic_1=0.1*tor_aero_Stochastic_MAX_1;
xi_aero_Stochastic_2=0.8*tor_aero_Stochastic_MAX_2;
gamma_2 = 0.1;
gamma_1 = 100;
```

```
epsilon_1 = 9;
epsilon_2 = 10;
epsilon_3 = 10;
epsilon_4 = 10;
sigma_hat_Stochastic_1 = 0;
sigma_hat_Stochastic_2 = 0;
sigma_hat_Stochastic_dot_1 =0;
sigma_hat_Stochastic_dot_2 = 0;


while (time <=10.0)
    if initial_loop_start == 9
        fprintf(s1, '%s','1');
        info = fscanf(s1)
       while  (info(1) ~='#')
           info = fscanf(s1);
           %yaw_initial =1;
       end
       a=datevec(now);
       TIME =(60*60*a(4)+60*a(5)+a(6)) -TIME_START;
       old_time= TIME -0.02;
    end
%      flushinput(s2);
    a=datevec(now);

TIME =(60*60*a(4)+60*a(5)+a(6)) -TIME_START;

        a2=datevec(now);
TIME2 =(60*60*a2(4)+60*a2(5)+a2(6)) -TIME_START;
    if(initial_loop_start == 10)
        info = fscanf(s1);

    info_1 = info;
    i=0;
    j=0;
   coma_count =0;

size_of_info =size(info);
       size_of_info = size_of_info(1,2);
   while  ( size_of_info<37)
       info = fscanf(s1);
       size_of_info =size(info);
       size_of_info = size_of_info(1,2);
       count_3 =count_3 + 1;
   end
    end
    Angles_last_2  = Angles;
           a3=datevec(now);
TIME3 =(60*60*a3(4)+60*a3(5)+a3(6)) -TIME_START;

   ed = TIME3 - TIME2;
   if (ed >= 0.025)
       ed;
   end
%flushinput(s1);
%fprintf(s1, '%s','1');

        size_of_info =size(info);
       size_of_info = size_of_info(1,2);
```

```
full_stop_count =0;
first_num = 1;
i=0;
    for i = 1:5
        if (info(i) =='#')
            first_num = first_num+1;
        end
        if (info(i) =='Y')
            first_num = first_num+1;
        end
        if (info(i) =='P')
            first_num = first_num+1;
        end
        if (info(i) =='R')
            first_num = first_num+1;
        end
        if (info(i) =='=')
            first_num = first_num+1;
        end
    end

   for i =first_num :size_of_info
       if (coma_count == 0) && (info(i) ~= ',')
           Yaw(i-(first_num-1)) = info(i);
            Yaw_count = i;
       end
       if (coma_count == 1) && (info(i) ~= ',')
           Pitch(i- Yaw_count-1) = info(i);
           Pitch_count = i;
       end
       if (coma_count == 2) && (info(i) ~= ',')
           Roll(i-Pitch_count-1) = info(i);
           Roll_count = i;
       end
       if (coma_count == 3) && (info(i) ~= ',')
           P(i-Roll_count-1) = info(i);
          P_count = i;
       end
       if (coma_count == 4) && (info(i) ~= ',')
           Q(i-P_count -1) = info(i);
          Q_count = i;
       end
       if (coma_count == 5) && (info(i) ~= ',')
           R_sensor(i-Q_count-1) = info(i);
           R_sensor_count = i;
           if info(i) == '!'
               R_sensor(i-Q_count-1)  ='0';
               i=size_of_info;
               coma_count=6;
           end
       end
       if info(i) == ','
           coma_count = coma_count+1;
       end
   end

   Delta_time = str2double(Delta_time );
   Delta_time = Delta_time /1000;
   pitch = str2double(Pitch);
   roll = str2double(Roll);
```

```
    yaw = str2double(Yaw);

    if(yaw>1000)
        yaw
        Yaw
        info
    end

    if (initial_loop_start == 9)
        yaw_temp= yaw;
        initial_loop_start = 10;

    end
   % yaw= yaw  - yaw_temp;
    p = str2double(P);
    q = str2double(Q);
    r = str2double( R_sensor);

    Delta_time = TIME -  old_time;
    if initial_state == 1
        %yaw_initial = yaw
        %pitch_initial = pitch;
        roll_initial = roll;
        p_initial = p;
        q_initial = q;
        r_initial = r;
        initial_state = 0;
    end
    %yaw = yaw - yaw_initial;
    p      = p ;%- p_initial ;
    q      = q ;%q_initial ;
    r      = r ;% r_initial
%    yaw = 0;
   %p=0;
  % q=0;
  %  r=0;
    if (abs(p))>10
        p =p_old;
    end
    if(abs(q))>10
        q=q_old;
    end
    if (abs(r))>10
        r= r_old;
    end
    %yaw= 0;
   %roll= 0;
%pitch=0;
    if (Delta_time  >=0.05)
        TIME ;
         old_time ;
        Delta_time;
        count_2 = count_2 +1;
        num;
     %   info_1;
    end
    %count_3 = 0;
    old_time = TIME;
                Pitch = ['0' '0' '0' '0' '0' '0'];
                Roll =['0' '0' '0' '0' '0' '0'];
```

```
            Yaw = ['0' '0' '0' '0' '0' '0'];
            P = ['0' '0' '0' '0' '0' '0'];
            Q = ['0' '0' '0' '0' '0' '0'];
            R = ['0' '0' '0' '0' '0' '0'];
            %Delta_time = ['0' '0' '0' '0' '0' '0'];
            info =['0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0'
'0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0'];

            Yaw_count = 0;
            Pitch_count = 0;
            Roll_count = 0;
            P_count = 0;
            Q_count = 0;
            R_count = 0;
            time_count =0;
            Angles= [pitch;roll;yaw];
   results_initial(:,Loop_Count_initial) = [pitch; roll ;yaw; time; p;q;r];

      [ pitch roll yaw p q r]

   for i = 1:4
     Motor_Signal(i,1) = 2430;
   end

   flushinput(s2);
   Motor_Signal_String = mat2str(Motor_Signal);
   fprintf(s2,'%s', Motor_Signal_String);


    time = time + Delta_time;
    %results_initial(:,Loop_Count_initial) = [p; q ;r];
    Loop_Count_initial= Loop_Count_initial +1;
end
i = 0;
%Loop_Count_initial = Loop_Count_initial - 1
for i=(Loop_Count_initial-1):Time_steps
   results_initial(:,i)=  results_initial(:,(Loop_Count_initial-1)) ;
end
sum = [0 ; 0 ; 0];
for i=1:Loop_Count_initial
   sum = sum + [results_initial(5,Loop_Count_initial)
;results_initial(6,Loop_Count_initial);results_initial(7,Loop_Count_initial)];
end
%
figure
plot(results_initial(4,:) ,results_initial(1,:), ...
            'LineWidth',2)
title('Attitude  Vs Time')
hold on
plot(results_initial(4,:) ,results_initial(2,:),'g', ...
            'LineWidth',2)
hold on
plot(results_initial(4,:) ,results_initial(3,:),'r', ...
            'LineWidth',2)
legend({'\it{\phi} \rm{(Pitch)}','\it{\theta} \rm{(Roll)}','\it{\psi} \rm{(Yaw)}'}, 'FontSize',16)
xlabel('Time(s)')
ylabel('Radians')
%axis([0, End_Time, - 3, 9])
hold off
%}
```

```
Angles_old  =         [pitch; roll; yaw];
Angulare_rate_offset = sum /Loop_Count_initial ;
B_Frame_angulare_velocity_old =Angulare_rate_offset;
yaw_initial  = yaw;
pitch_initial = pitch
roll_initial = roll;
time = 0
q=9
if(abs(yaw_initial  )>=180)
    yaw_initial
    break
    time = 1000000;
end
tester  = 1;
pitch_old = pitch;
 Angles_last =Angles;
 asdasdasdasdasdasd = Angles
 Angles_new_last = Angles_last
 Angles_new_last_2 = Angles_last_2

 a=datevec(now);
        asqw =  (60*60*a(4)+60*a(5)+a(6)) -TIME_START;
old_time =  asqw -0.01;
clc
a2=datevec(now);

IMU_time_last = (60*60*a2(4)+60*a2(5)+a2(6)) -TIME_START%-0.02;

 a=datevec(now);
        TIME_START=  (60*60*a(4)+60*a(5)+a(6)) ;
        IMU_time_last =-0.2
        old_time=0;
        temp_time = TIME_START
        time = 0;
        initial_loop_start = 10;
        Loop_Count              = 0;
         a=datevec(now);
        TIME_START=  (60*60*a(4)+60*a(5)+a(6)) ;
while (time <= End_Time)
    Loop_Count              = Loop_Count + 1;
    info_2 = info;
    if initial_loop_start == 9
        %fprintf(s1, '%s','1');
        info = fscanf(s1);
       while  (info(1) ~='#')
           info = fscanf(s1);
           %yaw_initial =1;
       end
       initial_state = 1;
       a=datevec(now);
       TIME =(60*60*a(4)+60*a(5)+a(6)) -TIME_START;
       old_time= TIME -0.02;
    end
%      flushinput(s2);
initial_loop_start;

        a2=datevec(now);
TIME2 =(60*60*a2(4)+60*a2(5)+a2(6)) -TIME_START;
TIME4 =(60*60*a2(4)+60*a2(5)+a2(6)) -TIME_START;
            TIME2X= TIME2 ;
```

```
imu_read_time  = TIME2 - IMU_time_last;
initial_loop_start = 10;
IMU_GOOD_READ = 0;
count_3 =0;
if initial_loop_start == 10
while IMU_GOOD_READ ~=1
   %imu_read_time =0.018
            a2=datevec(now);
         TIME5 =(60*60*a2(4)+60*a2(5)+a2(6)) -TIME_START
   if IMU_GOOD_READ == 0;

      imu_read_time

      while imu_read_time < 0*0.015%0.0168
         a2=datevec(now);
         TIME2 =(60*60*a2(4)+60*a2(5)+a2(6)) -TIME_START;
         TIME2X= TIME2 ;
         imu_read_time  = TIME2 - IMU_time_last;
      end

         a2=datevec(now);
         TIME2 =(60*60*a2(4)+60*a2(5)+a2(6)) -TIME_START;
         TIME2X= TIME2 ;
         imu_read_time  = TIME2 - IMU_time_last;
      imu_read_time
   end
            a2=datevec(now);
         TIME6 =(60*60*a2(4)+60*a2(5)+a2(6)) -TIME_START;
   imu_read_time;
   [imu_read_time, IMU_time_last];
   if imu_read_time >= 0*0.015
      info = fscanf(s1);
      info_1 = info;
      i=0;
      j=0;
      coma_count =0;
      size_of_info =size(info);
      size_of_info = size_of_info(1,2);
      if size_of_info>=35
         if (info(1) ~='#')
            size_of_info=6;
         end
         if (info(2) ~='Y')
            size_of_info=6;
         end
         if (info(3) ~='P')
            size_of_info=6;
         end
         if (info(4) ~='R')
            size_of_info=6;
         end
         if (info(5) ~='=')
            size_of_info=6;
         end
      end
      count_commas=0;
      if size_of_info>6
         for i = 1:(size_of_info -2)
            if (info(i) ==',')
               count_commas = count_commas+1;
```

```
            end
          end
          if (count_commas ~=5)
            size_of_info=6;
          end
        end

        reread =0;
          while  ( size_of_info<37|| size_of_info>45)
            info =['0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0'
'0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0'];
            info = fscanf(s1);
            size_of_info =size(info);
            size_of_info = size_of_info(1,2);
            count_3 =count_3 + 1;
            count_commas=0;
            for i = 1:(size_of_info -2)
              if (info(i) ==',')
                count_commas = count_commas+1;
              end
            end
            if (count_commas ~=5)
              size_of_info=6;
            end
            reread =1;
          end
        count_commas;
        a=datevec(now);
        if reread == 1
          IMU_time_last =(60*60*a(4) + 60*a(5)+a(6)) -TIME_START;
        end
        if reread == 0
          IMU_time_last = TIME2X ;
        end

    end
    a=datevec(now);
    TIME =(60*60*a(4)+60*a(5)+a(6)) -TIME_START;

        if (old_time == TIME)
         % flushinput(s1);
        end
          a3=datevec(now);
TIME3 =(60*60*a3(4)+60*a3(5)+a3(6)) -TIME_START;

   ed = TIME3 - TIME2;
   if (ed >= 0.025)
       ed;
   end


%fprintf(s1, '%s','1');
% flushinput(s1);
if (tester  == 1)
    tester  = 2;
    info_3 = info;
end
        size_of_info =size(info);
       size_of_info = size_of_info(1,2);
       full_stop_count =0;
```

```
       first_num = 1;
        for i = 1:5
            if (info(i) =='#')
                first_num = first_num+1;
            end
            if (info(i) =='Y')
                first_num = first_num+1;
            end
            if (info(i) =='P')
                first_num = first_num+1;
            end
            if (info(i) =='R')
                first_num = first_num+1;
            end
            if (info(i) =='=')
                first_num = first_num+1;
            end
        end
info_2 = info;
size_of_info = size_of_info-2;
first_num;
    for i =first_num:size_of_info
        if (coma_count == 0) && (info(i) ~= ',')
            Yaw(i-first_num+1) = info(i);
             Yaw_count = i;
        end
        if (coma_count == 1) && (info(i) ~= ',')
            Pitch(i- Yaw_count-1) = info(i);
            Pitch_count = i;
        end
        if (coma_count == 2) && (info(i) ~= ',')
            Roll(i-Pitch_count-1) = info(i);
            Roll_count = i;
        end
        if (coma_count == 3) && (info(i) ~= ',')
            P(i-Roll_count-1) = info(i);
           P_count = i;
        end
        if (coma_count == 4) && (info(i) ~= ',')
            Q(i-P_count -1) = info(i);
           Q_count = i;
        end
        if (coma_count == 5) && (info(i) ~= ',')
            R_sensor(i-Q_count-1) = info(i);
            R_sensor_count = i;
            if info(i) == '!'
                R_sensor(i-Q_count-1)  ='0';
                i=size_of_info;
                coma_count=6;
            end
        end
        if info(i) == ','
            coma_count = coma_count+1;
        end
    end



info;
```

```
    pitch = str2double(Pitch);
    roll = str2double(Roll);
    yaw = str2double(Yaw);
    if initial_loop_start == 9
        yaw_temp= yaw;
        yaw_initial = yaw_temp;
        initial_loop_start = 10;
    end

    %yaw= yaw  - yaw_temp;
    q = str2double(P);
    p = str2double(Q);
    r = str2double( R_sensor);
    Delta_time =TIME -  old_time;%imu_read_time;%TIME -  old_time;
    if initial_state == 1
        p_initial = p;
        q_initial = q;
        r_initial = r;
        initial_state = 0;
    end

    yaw = yaw - yaw_initial;
%    yaw = 0;
    if ( yaw >40)
        info_1;
    end

    old_time = TIME;
                Pitch = ['0' '0' '0' '0' '0' '0'];
                Roll =['0' '0' '0' '0' '0' '0'];
                Yaw = ['0' '0' '0' '0' '0' '0'];
                P = ['0' '0' '0' '0' '0' '0'];
                Q = ['0' '0' '0' '0' '0' '0'];
                R = ['0' '0' '0' '0' '0' '0'];
                %Delta_time = ['0' '0' '0' '0' '0' '0'];
                info =['0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0'
'0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0' '0'];
                Yaw_count = 0;
                Pitch_count = 0;
                Roll_count = 0;
                P_count = 0;
                Q_count = 0;
                R_count = 0;
                time_count =0;

     time =TIME ;% time + Delta_time;
    %yaw =0;
    IMU_GOOD_READ = 1;
     if(isnan(p) == 1)
         flushinput(s1);
         1
         info_2
         IMU_GOOD_READ = 2;
    %     break
    %     time = 100000000;
     end
    if(isnan(q) == 1)
        flushinput(s1);
        2
     info_2
```

```
     IMU_GOOD_READ = 2;
    end
  if(isnan(r) == 1)
      flushinput(s1);
      3
     info_2
     IMU_GOOD_READ = 2;
  end
    if(isnan(pitch) == 1)
      flushinput(s1);
      4
     info_2
     IMU_GOOD_READ = 2;
    end
    if(isnan(roll) == 1)
        flushinput(s1);
        5
     info_2
     IMU_GOOD_READ = 2;
    end
      if(isnan(yaw) == 1)
          flushinput(s1);
          6
         info_2
         IMU_GOOD_READ = 2;
      end
    end
end
initial_loop_start = 10;
        Time_to_read_seriel = TIME6-TIME5;
    B_Frame_angulare_velocity =10*0.1* [ p;
                                         q;
                                         r];
    B_Frame_angulare_velocity =B_Frame_angulare_velocity -0* [0.004;0.006;-0.013];
                                     alpha = 0.3;

    if yaw <-180
        yaw =yaw +360;
    end

pitch =pitch;
roll = roll;

                                     wn = 2*pi;
zeta =0.0001;
    Angles                          = [ pitch;
                                         roll;
                                        yaw];

     Angles  = Angles  *pi/180;
             Angles(1,1)*180/pi;
    Possition  = [ 0; 0 ;0;
                          Angles];
    Desired_Possition(4,1) =0;%(pi/18)*tanh(t)  ;
    Desired_Possition(5,1) = 0;%-(pi/18) *tanh(t);
    Desired_Possition(6,1) =0;% 0.1*t ;
     Desired_Possition(4,1) =0;% -pitch_initial *(tanh(0.05*time) - 1)*pi/180;%pitch*pi/180;%
    Desired_Possition(5,1) = 0;%-roll_initial *(tanh(0.05*time) - 1)*pi/180;%-roll_initial
*(tanh(0.5*time*abs(pitch_initial /roll_initial )) - 1)*pi/180;%roll*pi/180;
```

```
   Desired_Possition(6,1) =0; % Angles(3,1);
%0;%yaw_initial*pi/180;%yaw*pi/180;%0.5*(tanh(0.05*time));% 0.1*t ;
   angles = [  Possition(4,1);
               Possition(5,1);
               Possition(6,1) ];

   R      = rotation(Angles );



   E_Frame_angulare_velocity    =
B_FRAME_ANGULARE_Velocity_TO_E_FRAME_Angulare_velocity(B_Frame_angulare_velocity,
Angles );
   Velocity              = (Possition-Possition_OLD) / Delta_time;
   Desired_Velocity       = (Desired_Possition-Desired_Possition_OLD)/Delta_time;

   if ( Loop_Count  == 1)
      Desired_Velocity = [0;0;0;0;0;0];
   end


   Position_ERROR      = (Possition - Desired_Possition );
   Velocity_ERRPR      = (Desired_Possition-Desired_Possition_OLD)/Delta_time-Velocity;
t=time;


   eta_2_d                = [ -pitch_initial *(tanh(0.05*time) - 1)*pi/180;%pitch*pi/180;%
                              -roll_initial *(tanh(0.05*time) - 1)*pi/180;%-roll_initial
*(tanh(0.5*time*abs(pitch_initial /roll_initial )) - 1)*pi/180;%roll*pi/180;
                               0;];%yaw_initial*pi/180];%yaw*pi/180;%0.5*(tanh(0.05*time));% 0.1*t ;


         eta_2_d_dot           = [ -pitch_initial *0.05*(1-(tanh(0.05*time))^2)*pi/180;
                                    -roll_initial *0.05*(1-(tanh(0.05*time))^2)*pi/180;
                                    0];
   eta_2_d_double_dot    = [ -pitch_initial *0.05*(pi/180)*(0-2*0.05*tanh(0.05*time)*(1-
(tanh(0.05*time))^2));
                                    -roll_initial *(pi/180)* 0.05 *(0-2*0.05*tanh(0.05*time)*(1-
(tanh(0.05*time))^2));
                                    0];
                                 eta_2_d_dot_old =eta_2_d_dot;
                                 eta_2_d_old = eta_2_d;
if t>50
      eta_2_d                = [ -0.5*pitch_initial *(tanh(0.05*time-50*0.05) )*pi/180;%pitch*pi/180;%
                              -0.5*roll_initial *(tanh(0.05*time-50*0.05) )*pi/180;%-roll_initial
*(tanh(0.5*time*abs(pitch_initial /roll_initial )) - 1)*pi/180;%roll*pi/180;
                               0;];%yaw_initial*pi/180];%yaw*pi/180;%0.5*(tanh(0.05*time));% 0.1*t ;


         eta_2_d_dot           = [ -0.5*pitch_initial *0.05*(1-(tanh(0.05*time-50*0.05))^2)*pi/180;
                                    -0.5*roll_initial *0.05*(1-(tanh(0.05*time-50*0.05))^2)*pi/180;
                                    0];
   eta_2_d_double_dot    = [ -.5*pitch_initial *0.05*(pi/180)*(0-2*0.05*tanh(0.05*time-50*0.05)*(1-
(tanh(0.05*time-50*0.05))^2));
                                    -0.5*roll_initial *(pi/180)* 0.05 *(0-2*0.05*tanh(0.05*time-50*0.05)*(1-
(tanh(0.05*time-50*0.05))^2));
                                    0];
                                 eta_2_d_dot_old =eta_2_d_dot;
                                 eta_2_d_old = eta_2_d;

end
```

```
    %% Second stage set up converssions between frames of refference

    eta_2               = angles ;
    eta_2_error  = eta_2 - eta_2_d ;
    control_time = time ;
       q= Euler_to_Modified_rodrigues_paramater(eta_2 );
       alpha_q = Euler_to_Modified_rodrigues_paramater(eta_2_d);

%       U = Attitude_Backstepping_Controller_Stochastic(I_H,  K_1, K_2, K_d, D_2, eta_2, B_Frame_angulare_velocity, eta_2_d, eta_2_d_dot, eta_2_d_double_dot, Delta_time, tor_aero_hat_deterministic, tor_aero_hat_deterministic_dot, sigma_hat_Stochastic, sigma_hat_Stochastic_dot  ,Loop_Count  );
       U = Attitude_Backstepping_Controller_Stochastic(I_H, I_A, K_1, K_2, K_d, gamma_1, gamma_2, epsilon_1, epsilon_2, epsilon_3, epsilon_4, D_2, eta_2, B_Frame_angulare_velocity, eta_2_d, eta_2_d_dot, eta_2_d_double_dot, Delta_time, tor_aero_hat_deterministic, tor_aero_hat_deterministic_dot, sigma_hat_Stochastic_1, sigma_hat_Stochastic_2,sigma_hat_Stochastic_dot_1  ,sigma_hat_Stochastic_dot_2, Loop_Count  , ROE, tor_aero_deterministic_MAX, tor_aero_Stochastic_MAX_1, tor_aero_Stochastic_MAX_2, mew_1, mew_2, xi_aero_deterministic, xi_aero_Stochastic_1, xi_aero_Stochastic_2);

       tor_aero_hat_deterministic         = [U(10,1) ; U(11,1); U(12,1)];
       tor_aero_hat_deterministic_dot  = [U(13,1) ; U(14,1); U(15,1)];
       sigma_hat_Stochastic_1            = U(16,1) ;
       sigma_hat_Stochastic_dot          = U(17,1) ;
       sigma_hat_Stochastic_2            = U(18,1) ;
       tau_aero_hat  = tor_aero_hat_deterministic;
       alpha_w = [U(4,1) ; U(5,1); U(6,1)];
       alpha_w_dot  = [U(7,1) ; U(8,1); U(9,1)];
       u =[ m*9.81 ; U(1,1); U(2,1); U(3,1) ];
       U=u;
       w_error                  =   B_Frame_angulare_velocity - alpha_w;
       Controller_time_steps = 0;
    Motor_Signal = Control_Input_to_Motor_Speed_2 * U;

    for i = 1:4
       if (Motor_Signal(i,1)<0)
          Motor_Signal(i,1) = 0;
       end
       Motor_Signal(i,1)=0*(1/(4*PROP_Thrust_Coef))*m*9.81/R(3,3)+Motor_Signal(i,1);
       Motor_Signal(i,1) =  Min_PWM + PWM_range*(Motor_Signal(i,1)^0.5/Max_Motor_Speed);
       Motor_Signal(i,1) = round(Motor_Signal(i,1) );
      %Motor_Signal(i,1) = 2630;
      %Motor_Signal(i,1)  = 2340;
    end
%    Motor_Signal
    flushinput(s2);
    Motor_Signal_String = mat2str(Motor_Signal);
    fprintf(s2,'%s', Motor_Signal_String);



results_1(:, Loop_Count) = [Angles;
                            time;
                             B_Frame_angulare_velocity;
                            Motor_Signal];
%% motor spped to propeller thrust force and rotation toruqe
    Control_Input = Motor_and_Propeller_Dynamics(U, Control_Input_to_Motor_Speed, Motor_Speed_to_Control_Input, Motor_Speed,Loop_Count);
     Control_Input  = U;
```

```
%% Disturbance dynamics
         Aero_Disturbance = [0;0;0;0;0;0];

         %Collision_Disturbance =Collision_Disturbance_calc(t, Collision_time,
Collision_Point,F_Collision_x, F_Collision_y, F_Collision_z, E_Frame_Linear_possition)
;;
         Torque_Disturbance = [ -0.2*sin(2*time);
                                      + 0.2*sin(2*time);
                                       0*cos(time)];
                                    Torque_Disturbance = 0*Torque_Disturbance;
          Collision_Disturbance= [ 0;
                                       0;
                                       0;
                                       Torque_Disturbance];
         tau_error = Torque_Disturbance - tau_aero_hat;

      eta_2_error_0 = [0;0;0];
      w_error_0 = [0.2;0.2;0.1];
      tau_error_0 = [ 0.2; 0.2; 0.1];


      if (E_Frame_Linear_possition(3,1) <= 0)
         E_Frame_Linear_possition(3,1) =0;
      end

      Zeta  = [ E_Frame_Linear_velocity;
                B_Frame_angulare_velocity          ];

Possition_OLD   = Possition;
      Possition = [ 0;0;0%E_Frame_Linear_possition;
                    angles                          ];
      Desired_Possition_OLD =     Desired_Possition;

%% For Plotting system information
   Input(:,Loop_Count) = Control_Input;

      if Loop_Count > 1
         Input_rate(:,Loop_Count)  = (Input(:,Loop_Count) - Input(:,(Loop_Count-
Controller_time_steps))) / Delta_time;
      end
   if Loop_Count == 1
      Input_rate(:,Loop_Count)  = 0;
   end
   alpha = 0.1;
   %exponentialMA = filter(alpha, [1 alpha-1],  pitch);
   exponentialMA  = (1-alpha)*pitch_old  +alpha*pitch;
   pitch_old = exponentialMA ;

   results(:,Loop_Count) = [ Possition;%1-6
                             time ;%7
                             B_Frame_angulare_velocity ;%8-10
                             tau_aero_hat;%111 - 13
                             w_error;
                              eta_2_error;
                               alpha_w_dot
                               Torque_Disturbance
                               tau_error;
                               eta_2_d;%   ];%eta_2_noise];
                               exponentialMA ;
```

```
                                    Delta_time;
                                    q;
                                    alpha_q;
                                    count_3;
                                    reread;
                                    Time_to_read_seriel;
                                    sigma_hat_Stochastic_1;
                                    sigma_hat_Stochastic_2];

   eta_2_d_dot_old         = eta_2_d_dot;
   p_old = p;
   q_old = q;
   r_old = r;
for i = 1:3
   Refference_signal(i,Loop_Count) = Desired_Possition(i,1) ;
end
Refference_signal(4,Loop_Count) = eta_2_d(1,1);
Refference_signal(5,Loop_Count) = eta_2_d(2,1);
Refference_signal(6,Loop_Count) = eta_2_d(3,1);

Controller_time_steps = Controller_time_steps + 1;
%% END OF  LOOP%%

   time                    ;
   for i =1:100
      if (Loop_Count == i*200)
         [Loop_Count/10, time/10, Loop_Count/ time, imu_read_time]%, imu_time_last]
      end
   end
end
Loop_Count
time
a=datevec(now);
      asqw2 =  (60*60*a(4)+60*a(5)+a(6)) -TIME_START;
      asqw2  - asqw
for i =(Loop_Count-1):Time_steps
   results_1(:, i) = results_1(:, (Loop_Count-1));
   results(:, i) = results(:, (Loop_Count-1));
end

   a=datevec(now);

TIME_END =(60*60*a(4)+60* a(5)+a(6))  ;% cputime;

delta = TIME_END-TIME_START

time_on_num = time/num
   for i = 1:4
      Motor_Signal(i,1) =1500;
   end

   flushinput(s2);
   Motor_Signal_String = mat2str(Motor_Signal);
   fprintf(s2,'%s', Motor_Signal_String);

%close the port and delete it
fclose(s1);
delete(s1)
clear('s1')
fclose(s2);
```

```
delete(s2)
clear('s2')

delete(instrfindall)

figure
plot(results(7,:) ,results(33,:), ...
            'LineWidth',2)
set(gca,'FontSize',12)
title('Delta timer   Vs Time','FontSize',16)
set(legend({'$\hat{\Delta}_time$'}, 'FontSize',14),'interpreter','latex')
xlabel('Time(s)','FontSize',16)
ylabel('Delta time (s)','FontSize',16)
hold off


figure
plot(results(7,:) ,results(33,:), ...
            'LineWidth',2)
set(gca,'FontSize',12)
title('Delta timer   Vs Time','FontSize',16)
set(legend({'$\hat{\Delta}_time$'}, 'FontSize',14),'interpreter','latex')
xlabel('Time(s)','FontSize',16)
ylabel('delta time (s)','FontSize',16)
hold off


figure
get(0,'Factory');
set(0,'defaultfigurecolor',[1 1 1]);
subplot(2,1,1)
plot(results(7,:) ,results(43,:),'g', ...
            'LineWidth',2)
set(title({'$\hat{\delta}_1 $'}, 'FontSize',16),'interpreter','latex')
set(ylabel({'$\hat{\delta}_1 $'}, 'FontSize',16),'interpreter','latex')
xlabel('Time(s)', 'FontSize',14)
hold on
subplot(2,1,2)
plot(results(7,:) ,results(44,:),'b', ...
            'LineWidth',2)
set(title({'$\hat{\delta}_2 $'}, 'FontSize',16),'interpreter','latex')
hold off
xlabel('Time(s)', 'FontSize',14)
set(ylabel({'$\hat{\delta}_2 $'}, 'FontSize',16),'interpreter','latex')
%axis([0, End_Time,-0.4,0.4])
hold off

figure
plot(results(7,:) ,results(41,:), ...
            'LineWidth',2)
title('reread   Vs Time','FontSize',16)
set(legend({'$\hat{\delta}_time$'}, 'FontSize',14),'interpreter','latex')
xlabel('Time(s)','FontSize',16)
ylabel('delta time (s)','FontSize',16)
hold off
figure
plot(results(7,:) ,results(42,:), ...
            'LineWidth',2)
set(gca,'FontSize',12)
title('Time to read seriel Vs Time','FontSize',16)
```

```
set(legend({'$\hat{\delta}_time$'}, 'FontSize',20),'interpreter','latex')
xlabel('Time(s)','FontSize',16)
ylabel('delta time (s)','FontSize',16)
hold off
figure
plot(results(7,:) ,results(11,:), ...
            'LineWidth',2)
set(gca,'FontSize',12)
set(title({'\textbf{${\hat{\tau}}$ Vs Time}'}, 'FontSize',20),'interpreter','latex')
hold on
plot(results(7,:) ,results(12,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,results(13,:),'r', ...
            'LineWidth',2)
hold on
axis([0, End_Time, - 2, 2])
set(legend({'\textbf{$\hat{\tau}_p$}','\textbf{$\hat{\tau}_q$}','\textbf{$\hat{\tau}_r$}},
'FontSize',16),'interpreter','latex')
set(xlabel('\textbf{Time(s)}', 'FontSize',16),'interpreter','latex')
set(ylabel('\textbf{${\hat{\tau}}$ (Nm)}', 'FontSize',16),'interpreter','latex')
%axis([0, End_Time,-0.4,0.4])
hold off


RAD_to_DEG = 180/pi;

figure
plot(results(7,:) ,RAD_to_DEG*results(4,:), ...
            'LineWidth',2)
set(gca,'FontSize',12)
title('Attitude Euler Representation Vs Time', 'FontSize',16)
hold on
plot(results(7,:) ,RAD_to_DEG*results(5,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,RAD_to_DEG*results(6,:),'r', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,RAD_to_DEG*results(29,:),'--b' , ...
            'LineWidth',1)
hold on
plot(results(7,:) ,RAD_to_DEG*results(30,:),'--g' , ...
            'LineWidth',1)
 hold on
plot(results(7,:) ,RAD_to_DEG*results(31,:),'--r' , ...
            'LineWidth',1)
axis([0, End_Time, - 20, 30])
set(legend({'\it{\phi}','\it{\theta}','\it{\psi}','\it{\phi_d}','\it{\theta_d}','\it{\psi_d}},
'FontSize',14,'Location','eastoutside','Orientation','vertical'))
xlabel('Time(s)', 'FontSize',16)
ylabel('Attitude (Degrees)', 'FontSize',16)
%axis([0, End_Time,-0.4,0.4])
hold off
```

```
RAD_to_DEG = 180/pi;
figure
plot(results(7,:) ,RAD_to_DEG*results(17,:), ...
            'LineWidth',2)
set(gca,'FontSize',12)
title('Attitude error Euler Representation Vs Time', 'FontSize',16)
hold on
plot(results(7,:) ,RAD_to_DEG*results(18,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,RAD_to_DEG*results(19,:),'r', ...
            'LineWidth',2)
hold on
axis([0, End_Time, - 10, 10])
set(legend({'\it{\phi_e}','\it{\theta_e}','\it{\psi_e}'}, 'FontSize',14))
xlabel('Time(s)', 'FontSize',16)
ylabel('Attitude (Degrees)', 'FontSize',16)
hold off


figure
plot(results(7,:) ,Input(2,:),'b', ...
            'LineWidth',2)
set(gca,'FontSize',12)
hold on
plot(results(7,:) ,Input(3,:),'g', ...
            'LineWidth',2)
hold on
plot(results(7,:) ,Input(4,:),'r', ...
            'LineWidth',2)
hold on
title('Control torque input Vs Time', 'FontSize',16)
legend({'\it{U_2}','\it{U_3}','\it{U_4}'}, 'FontSize',16)
xlabel('Time(s)', 'FontSize',16)
ylabel('Control torque(Nm)', 'FontSize',16)
axis([0, End_Time,-2,2])
hold off


figure
plot(results_1(4, :) ,results_1(8, :),'b', ...
            'LineWidth',2)
set(gca,'FontSize',12)
title('PWM Signal Vs Time', 'FontSize',14)
hold on
plot(results_1(4, :) ,results_1(9, :),'g', ...
            'LineWidth',2)
hold on
plot(results_1(4, :) ,results_1(10, :),'r', ...
            'LineWidth',2)
hold on
plot(results_1(4, :) ,results_1(11, :),'k', ...
            'LineWidth',2)
hold on
 axis([0, End_Time, 2340, 3000])
legend({'\rm{PWM1}','\rm{PWM2}','\rm{PWM3}','\rm{PWM4}'}, 'FontSize',16)
```

```
xlabel('Time(s)', 'FontSize',16)
ylabel('Motor PWM step signal', 'FontSize',16)
%axis([0, End_Time,-0.4,0.4])
hold off
 figure
plot(results(7,:) ,results(11,:), ...
                'LineWidth',2)
            set(gca,'FontSize',12)
title('Torque Distrubance estimate deterministic Vs Time', 'FontSize',16)
hold on
plot(results(7,:) ,results(12,:),'g', ...
                'LineWidth',2)
hold on
plot(results(7,:) ,results(13,:),'r', ...
                'LineWidth',2)
hold on
axis([0, End_Time, - 2, 2])
set(legend({'\textbf{$\hat{\tau}_{aero,p}$}','\textbf{$\hat{\tau}_{aero,q}$}','\textbf{$\hat{\tau}_{aero,r}$}'},
'FontSize',20),'interpreter','latex')
xlabel('Time(s)', 'FontSize',16)
ylabel('Estimated torque disturbance (Nm)', 'FontSize',16)
axis([0, End_Time,-0.5,0.5])
hold off
```

```
function U = Attitude_Backstepping_Controller_Stochastic(I_H, I_A, K_1, K_2, K_d, gamma_1,
gamma_2, epsilon_1, epsilon_2, epsilon_3, epsilon_4, D_2, eta_2, B_Frame_angulare_velocity,
eta_2_d, eta_2_d_dot, eta_2_d_double_dot, Delta_time, tor_aero_hat_deterministic,
tor_aero_hat_deterministic_dot, sigma_hat_Stochastic_1,
sigma_hat_Stochastic_2,sigma_hat_Stochastic_dot_1  ,sigma_hat_Stochastic_dot_2, Loop_Count ,
ROE, tor_aero_deterministic_MAX, tor_aero_Stochastic_MAX_1, tor_aero_Stochastic_MAX_2,
mew_1, mew_2, xi_aero_deterministic, xi_aero_Stochastic_1, xi_aero_Stochastic_2)

n_2                     =  eta_2;
q                       = Euler_to_Modified_rodrigues_paramater(n_2);
alpha_q                 = Euler_to_Modified_rodrigues_paramater(eta_2_d);
q_e                     = q - alpha_q;
R_2_alpha_q             = R_2_MRP_calc( alpha_q);
angles                  = eta_2;
T_inv                   = Angular_velocity_cordinant_transform(angles);
R_2_q                   = R_2_MRP_calc(q);
R_2_q_inverse           = R_2_q.'*(16/(1+q.'*q)^2);
w                       = B_Frame_angulare_velocity;
alpha_w                 = R_2_q_inverse*(-K_1 * q_e + R_2_alpha_q * T_inv * eta_2_d_dot );
w_e                     =   B_Frame_angulare_velocity - alpha_w;
[Partial_alpha_w_wrt_q, Partial_alpha_w_wrt_eta_2_d,Partial_alpha_w_wrt_eta_2_d_dot] =
Partial_derivatives_of_alpha_w( K_1, n_2, q, alpha_q, eta_2_d, eta_2_d_dot);
alpha_w_dot                     = Partial_alpha_w_wrt_q * R_2_q * w + Partial_alpha_w_wrt_eta_2_d *
eta_2_d_dot + Partial_alpha_w_wrt_eta_2_d_dot * eta_2_d_double_dot ;
INV_I_H_plus_I_A                        = inv(I_H+I_A);

tor_aero_deterministic_p          = Projection_algorithm(gamma_2*((norm(w_e))^2*([1 0 0
]*INV_I_H_plus_I_A)*w_e)/(1+(w_e.'*w_e)^2)^0.5,[1 0 0]*tor_aero_hat_deterministic,[1 0
0]*tor_aero_hat_deterministic_dot, [1 0 0]*xi_aero_deterministic, [1 0 0]*tor_aero_deterministic_MAX,
ROE(1,1), Delta_time);
tor_aero_deterministic_q          = Projection_algorithm(gamma_2*((norm(w_e))^2*([0 1 0
]*INV_I_H_plus_I_A)*w_e)/(1+(w_e.'*w_e)^2)^0.5,[0 1 0]*tor_aero_hat_deterministic,[0 1
0]*tor_aero_hat_deterministic_dot, [0 1 0]*xi_aero_deterministic, [0 1
0]*tor_aero_deterministic_MAX,ROE(2,2), Delta_time);
tor_aero_deterministic_r          = Projection_algorithm(gamma_2*((norm(w_e))^2*([0 0 1
]*INV_I_H_plus_I_A)*w_e)/(1+(w_e.'*w_e)^2)^0.5,[0 0 1]*tor_aero_hat_deterministic,[0 0
1]*tor_aero_hat_deterministic_dot, [0 0 1]*xi_aero_deterministic, [0 0 1]*tor_aero_deterministic_MAX,
ROE(3,3), Delta_time);
tor_aero_deterministic            = [ tor_aero_deterministic_p;
                                      tor_aero_deterministic_q;
                                      tor_aero_deterministic_r ];

tor_aero_hat_deterministic        =[ tor_aero_deterministic(1,1);
                                      tor_aero_deterministic(3,1);
                                      tor_aero_deterministic(5,1)];
tor_aero_hat_deterministic_dot = [ tor_aero_deterministic(2,1);
                                      tor_aero_deterministic(4,1);
                                      tor_aero_deterministic(6,1)];
if Loop_Count   == 1
   tor_aero_hat_deterministic  =[0;0;0];
end

sigma_Stochastic= Projection_algorithm(gamma_2^2*
(9*epsilon_3)*(norm(inv(I_H+I_A)))^4*(norm(w_e))^4/(1+(norm(w_e))^4)^1.5,sigma_hat_Stochastic_1
,sigma_hat_Stochastic_dot_1  , xi_aero_Stochastic_1, tor_aero_Stochastic_MAX_1, mew_1,
Delta_time);
sigma_hat_Stochastic        = sigma_Stochastic(1,1);

sigma_hat_Stochastic_1 = sigma_hat_Stochastic;
```

```
I_XX_A = I_A(1,1);
I_YY_A = I_A(2,2);
I_ZZ_A = I_A(3,3);

Coriolis_A = [ 0, I_ZZ_A * B_Frame_angulare_velocity(3,1), -I_YY_A *
B_Frame_angulare_velocity(2,1);
                    -I_ZZ_A * B_Frame_angulare_velocity(3,1), 0, I_XX_A *
B_Frame_angulare_velocity(1,1);
                    I_YY_A * B_Frame_angulare_velocity(2,1), -I_XX_A *
B_Frame_angulare_velocity(1,1), 0 ];
skew_w = [ 0, -B_Frame_angulare_velocity(3,1),  B_Frame_angulare_velocity(2,1);
                    B_Frame_angulare_velocity(3,1), 0, -B_Frame_angulare_velocity(1,1);
                    -B_Frame_angulare_velocity(2,1), B_Frame_angulare_velocity(1,1), 0];
                Multiplicitiveand_aditive_noise = inv(I_H+I_A)*(D_2*eye(3)+0*0.05*eye(3)+ Coriolis_A
+skew_w *I_A);
sigma_Stochastic= Projection_algorithm(gamma_2^2*
(9*epsilon_4)*(norm(Multiplicitiveand_aditive_noise
))^4*(norm(w_e))^4/(1+(norm(w_e))^4)^1.0,sigma_hat_Stochastic_2,sigma_hat_Stochastic_dot_2  ,
xi_aero_Stochastic_2, tor_aero_Stochastic_MAX_2, mew_2, Delta_time);
sigma_hat_Stochastic        =1*sigma_Stochastic(1,1);
sigma_hat_Stochastic_dot  =  sigma_Stochastic(2,1);
sigma_hat_Stochastic_2  = sigma_hat_Stochastic;
 w_e_dot             = - K_2 * w_e/ (1+0*(w_e.'*w_e)^2)^0.25 ;
tor                          = (I_H+I_A)*(D_2 * alpha_w + w_e_dot -
gamma_2*((9*epsilon_3/4)*(norm(inv(I_H + I_A)))^4*sigma_hat_Stochastic_1)*w_e /
(1+1*(w_e.'*w_e)^2)^0.5 -
gamma_2*((9*epsilon_4/4)*(norm(Multiplicitiveand_aditive_noise))^4*sigma_hat_Stochastic_2)*w_e/(
1+1*(w_e .'*w_e )^2)^0.5 + alpha_w_dot -
(1/gamma_2)*epsilon_2*((gamma_1/(4*epsilon_1))^2)*norm(R_2_alpha_q)^4/4
*w_e/(1+(w_e.'*w_e)^2)^3.25-
gamma_1*(1/gamma_2)*(q_e.'*R_2_alpha_q*w_e)*w_e/(1+(w_e.'*w_e)^2)^0.25 ) -
0*tor_aero_hat_deterministic + cross(B_Frame_angulare_velocity, ((I_H +
I_A)*B_Frame_angulare_velocity));

U                          = [ tor(1,1); tor(2,1) ; tor(3,1); alpha_w;alpha_w_dot; tor_aero_hat_deterministic;
tor_aero_hat_deterministic_dot; sigma_hat_Stochastic_1; sigma_hat_Stochastic_dot;
sigma_hat_Stochastic_2  ];
```

```
function [Partial_alpha_w_wrt_q, Partial_alpha_w_wrt_eta_2_d,Partial_alpha_w_wrt_eta_2_d_dot] =
Partial_derivatives_of_alpha_w( K_1, n_2, q, alpha_q, eta_2_d, eta_2_d_dot)

T_inv                  = Angular_velocity_cordinant_transform(eta_2_d);
T_inverse_eta_2_d = T_inv;
T_eta_2_d_inverse = T_inverse_eta_2_d;
R_2_alpha_q = R_2_MRP_calc(alpha_q);

q_e = q - alpha_q;


vector_0 =( -K_1 *q_e  + R_2_alpha_q*T_eta_2_d_inverse*eta_2_d_dot);
vector_1 = T_eta_2_d_inverse*eta_2_d_dot;
vector_2 = eta_2_d_dot;

Partial_R_2_alpha_q_wrt_partial_alpha_q_times_vector_1 = R_2_derivitive_times_vector (alpha_q,
R_2_alpha_q, vector_1);
Partial_T_inverse_eta_2_d_wrt_eta_2_d_times_vector_2 =
Partial_T_eta_2_d_inverse_wrt_eta_2_times_vector(eta_2_d, eta_2_d_dot);

R_2_q = R_2_MRP_calc(q);
R_2_inverse_dot_times_x = R_2_inverse_derivative_times_vector(R_2_q,  q, vector_0);
R_2_q_inverse  = R_2_q.'*(16/(1+q.'*q)^2);
P_1 = R_2_inverse_dot_times_x ;

P_2 =R_2_q_inverse * (-(K_1  ));
P_3 =(K_1) ;

P_4 =Partial_R_2_alpha_q_wrt_partial_alpha_q_times_vector_1;
P_5 =R_2_alpha_q*Partial_T_inverse_eta_2_d_wrt_eta_2_d_times_vector_2;
P_6 =R_2_alpha_q*T_inverse_eta_2_d;

Partial_alpha_w_wrt_q = P_1+ R_2_q_inverse *P_2  ;
Partial_alpha_w_wrt_alpha_q = R_2_q_inverse *( P_3  + P_4);
Partial_alpha_w_wrt_eta_2_d = R_2_q_inverse * P_5 + Partial_alpha_w_wrt_alpha_q;
Partial_alpha_w_wrt_eta_2_d_dot = R_2_q_inverse * P_6;

End


function R_2_MRP = R_2_MRP_calc(q)
skew_q = [ 0, -q(3,1), q(2,1);
           q(3,1), 0, -q(1,1);
           -q(2,1), q(1,1), 0];
R_2_MRP = 0.5*(eye(3) + q*q.' + skew_q - ((1+(q.'*q))/2)*eye(3) );
end
```

```
function Modified_rodrigues_paramater = Euler_to_Modified_rodrigues_paramater(Euler_angle)

Phi    = Euler_angle(1,1);
Theta = Euler_angle(2,1);
Psi    = Euler_angle(3,1);
 q_0 = (1 + cos(Phi/2)*cos(Theta/2)*cos(Psi/2) + sin(Phi/2)*sin(Theta/2)*sin(Psi/2) );
q_1 = (sin(Phi/2)*cos(Theta/2)*cos(Psi/2) - cos(Phi/2)*sin(Theta/2)*sin(Psi/2))/q_0;
q_2 = (cos(Phi/2)*sin(Theta/2)*cos(Psi/2) + sin(Phi/2)*cos(Theta/2)*sin(Psi/2))/q_0;
q_3 = (cos(Phi/2)*cos(Theta/2)*sin(Psi/2) - sin(Phi/2)*sin(Theta/2)*cos(Psi/2))/q_0;

Modified_rodrigues_paramater = [ q_1;
                                 q_2;
                                 q_3];

end


function  Partial_R_2_alpha_q_wrt_partial_q_times_vector  = R_2_derivitive_times_vector (q, R_2,
vector)
Partial_R_2_alpha_q_wrt_partial_q_times_vector = 1/2*((q.'*vector*eye(3)+q*(vector).'-vector*q.')-
([0,0,0;0,0,1;0,-1,0]*(vector*[1,0,0])+[0,0,-1;0,0,0;1,0,0]*(vector*[0,1,0])+[0,1,0;-
1,0,0;0,0,0]*(vector*[0,0,1])));
End


function R_2_inverse_dot_times_vector = R_2_inverse_derivative_times_vector(R_2,  q, vector)

R_2_inverse_dot_times_vector=8/(1+q.'*q)^2 * ((q.'*vector*eye(3)+q*vector.'-
vector*q.')+([0,0,0;0,0,1;0,-1,0]*(vector*[1,0,0])+[0,0,-1;0,0,0;1,0,0]*(vector*[0,1,0])+[0,1,0;-
1,0,0;0,0,0]*(vector*[0,0,1])))-64/(1+q.'*q)^3  * R_2.'*vector*q.';

End


function  Partial_T_eta_2_d_inverse_wrt_eta_2_times_vector_x =
Partial_T_eta_2_d_inverse_wrt_eta_2_times_vector (alpha_2, vector)
   alpha_phi    = alpha_2(1,1);
   alpha_theta = alpha_2(2,1);
   Partial_T_eta_2_d_inverse_wrt_eta_2_times_vector_x = ([0,0,0;0,-
sin(alpha_phi),cos(alpha_theta)*cos(alpha_phi);0,-cos(alpha_phi),-
cos(alpha_theta)*sin(alpha_phi)]*vector*[1;0;0].'+[0,0,-cos(alpha_theta);0,0,-
sin(alpha_theta)*sin(alpha_phi);0,0,-sin(alpha_theta)*cos(alpha_phi)]*vector*[0;1;0].');
end
```

```
function  Projection = Projection_algorithm(omega_Macron,omega_hat, omega_hat_dot, xi,
omega_MAX, gamma, Delta_time)
   XI =( (norm(omega_hat))^2 - omega_MAX^2) / (xi^2 + 2*xi*omega_MAX);
   Derivative_XI = 2*omega_hat'/ (xi^2 + 2*xi*omega_MAX);
   if(XI < 0)
      omega_hat_dot = gamma * omega_Macron;
      Derivative_XI = 2*omega_hat'/ (xi^2 + 2*xi*omega_MAX);
   end
   if((XI >=  0) && ((Derivative_XI * omega_Macron)<=0))
      omega_hat_dot = gamma * omega_Macron;
   end
   if((XI >=  0) && ((Derivative_XI * omega_Macron)>0))
      omega_hat_dot = gamma * (1-XI)*omega_Macron;
   end
    omega_hat = omega_hat + Delta_time * omega_hat_dot ;
   Projection  = [omega_hat; omega_hat_dot ];
end


function Control_Input = Motor_and_Propeller_Dynamics(Control_Input,
Control_Input_to_Motor_Speed, Motor_Speed_to_Control_Input, Motor_Speed,Loop_Count)

   Motor_Speed = Control_Input_to_Motor_Speed * Control_Input ;
   for i = 1:6
      if Motor_Speed(i, 1) >= ((8208*2*pi/60)^2)
         Motor_Speed(i, 1) = (8208*2*pi/60)^2;
      end
         if Motor_Speed(i, 1) <0
            Motor_Speed(i, 1) = 0;
         end
   end
   for i = 1 : 6
      Motor_Speed(i, 1) = Motor_Speed(i, 1)^0.5;
      %MOTOR(i, Loop_Count)  = Motor_Speed(i, 1);
      Motor_Speed(i, 1) = Motor_Speed(i, 1)^2;
   end
   Control_Input = Motor_Speed_to_Control_Input * Motor_Speed ;
   %% motor spped to propeller thrust force and rotation toruqe
   for i = 1:6;
      Blade_pitch_angle = 4;
      A_of_attack = Blade_pitch_angle*pi/180;
   %   [Prop_Thrust(i), Prop_Torque(i)]    =Prop_Aero_Thrust_Rotation_Calc (
total_Propeller_surface_area, Motor_spped(i), air_density , Effective_blade_leangth, A_of_attack);
   end
end
```

This Page Has Been Left Blank