

12-2016

Development of Fault Tolerant Adaptive Control Laws for Aerospace Systems

Andres E. Perez Rocha

Follow this and additional works at: <https://commons.erau.edu/edt>



Part of the [Systems Engineering and Multidisciplinary Design Optimization Commons](#)

Scholarly Commons Citation

Rocha, Andres E. Perez, "Development of Fault Tolerant Adaptive Control Laws for Aerospace Systems" (2016). *Dissertations and Theses*. 289.

<https://commons.erau.edu/edt/289>

This Dissertation - Open Access is brought to you for free and open access by Scholarly Commons. It has been accepted for inclusion in Dissertations and Theses by an authorized administrator of Scholarly Commons. For more information, please contact commons@erau.edu.

DEVELOPMENT OF FAULT TOLERANT ADAPTIVE CONTROL LAWS FOR
AEROSPACE SYSTEMS

A Dissertation

Submitted to the Faculty

of

Embry-Riddle Aeronautical University

by

Andres E. Perez Rocha

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy in Aerospace Engineering

December 2016

Embry-Riddle Aeronautical University

Daytona Beach, Florida

Development of Fault Tolerant Adaptive Control Laws for Aerospace Systems

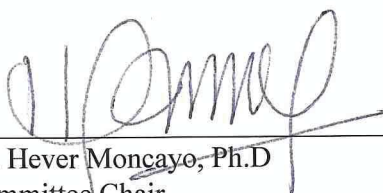
By

Andres E. Perez Rocha


This Dissertation was prepared under the direction of the candidate's Dissertation Committee Chair, Dr. Hever Moncayo, Department of Aerospace Engineering, and has been approved by the members of the dissertation committee. It was submitted to College of Engineering and was accepted in partial fulfillment of the requirements for the

Degree of

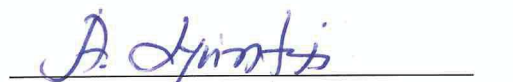
Doctor of Philosophy in Aerospace Engineering




Dr. Hever Moncayo, Ph.D
Committee Chair



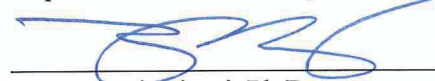
Dr. Mark Balas, Ph.D
Committee Co-Chair



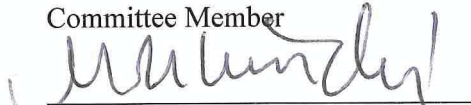
Dr. Tasos Lyrintzis, Ph.D
Department Chair, Aerospace Engineering




Dr. Sergey Drakunov, Ph.D
Committee Member



Dr. Maj Mirmirani, Ph.D
Dean, College of Engineering



Dr. Mario Perhinschi, Ph.D
Committee Member



Dr. Michael Hickey, Ph.D
Dean of Research and Graduate Studies



Dr. Richard Prazenica, Ph.D
Committee Member

12/08//2016

Date

ACKNOWLEDGMENTS

First and foremost I would like to thank everyone who contributed in one way or another and made this dissertation possible. I would like to give special thanks to my parents Jaime and Martha for giving me courage in difficult moments and supporting me in many different ways during this doctorate. Many thanks to my beloved girlfriend Juanita for her unconditional support, patience and infinite courage.

To Dr. Moncayo I would like to extend my gratitude for allowing me be part of his research projects and the opportunity to do this doctorate. I appreciate his guidance and advise during the last years. To Dr. Balas I would like to extend my gratitude for his insight and wisdom, invaluable for many different aspects of this dissertation. To all the committee members I would like to extend my gratitude for their support and insight.

Infinite thanks to all the students of the Flight Dynamics and Control Research Lab, special thanks to Israel, Brendon, Sergio, Diego and Chirag, for their friendship and good times and because without their help and support this dissertation wouldn't have been possible. Finally I would like to extend my gratitude to Mike DuPuis, Tom Ebert and Rob Mueller and all the NASA team members of the STTR project, many of this dissertation results were possible because of their constant and immense support.

TABLE OF CONTENTS

LIST OF TABLES	vii
LIST OF FIGURES	viii
ABBREVIATIONS	xiii
ABSTRACT.....	xiv
1. Introduction.....	1
2. Literature Review	8
2.1. Overview of Adaptive Control	8
2.2. Bio-Inspired and Immunity Based Adaptive Control	9
2.2.1. T-B Artificial Immune Controllers.....	12
2.2.2. Double-Cell Immune Controller	15
2.3. Stability of Non-Linear Systems	17
2.3.1. Equilibrium Points and Nominal Trajectories	18
2.3.2. General Definitions of Stability.....	20
2.3.3. Lyapunov's Direct Method.....	22
2.3.4. La Salle Invariant Principle	23
2.3.5. Barbalat's Lemma	24
2.3.6. Absolute Stability and Circle Criterion.....	25
2.4. Feedback Linearization	30
2.4.1. Non-Linear Dynamic Inversion.....	30
3. General Aerospace System Equations of Motion.....	32
3.1. Rigid Body Dynamics.....	32
3.1.1. Moment Equations	33
3.1.2. Force Equations	35
3.2. Euler Angle Rotation.....	36
3.3. Rigid body Attitude Kinematics - Euler	39
3.4. Navigation Equations	40
4. Development of Adaptive Control Laws Based on Euler Angle NLDI	41
4.1. Angular Rate NLDI Control	41
4.2. Angular Rate NLDI augmented with AIS Adaptive Control	44
4.3. Boundedness of Angular Rate Tracking Errors	47
4.4. Euler Angle Exact NLDI Based on Lie Derivative.....	51
4.5. Euler Angle Incremental Non-Linear Dynamic Inversion	54
4.5.1. Inner Loop - Fast Mode Dynamic Inversion	54
4.5.2. Inner Loop - Slow Mode Dynamic Inversion.....	55
4.6. Exact NLDI Augmented with Novel Model Reference Adaptive AIS	57
4.6.1. Proof of Absolute Stability Based on Circle Criterion.....	60
4.6.2. Robustness on Presence of Bounded Uncertainties	63
4.7. Model Reference AIS Simulation on a LTI Plant	72
4.7.1. Simulation Results for Step Tracking	73

4.7.2.	Simulation Results for Cycloid Tracking	75
4.7.3.	Estimate of Radius of Convergence	76
5.	Application to Aircraft Control	77
5.1.	WVU Fighter Aircraft Simulation Framework	77
5.1.1.	Actuator Failure	79
5.1.2.	Structural Failure	80
5.2.	Control Architectures	80
5.2.1.	NLDI Baseline Controller.....	80
5.2.2.	NLDI Augmented with Artificial Immune System.....	82
5.2.3.	NLDI Augmented with Artificial Neural Networks	83
5.3.	Performance Metrics Definition	85
5.3.1.	Pilot Activity Metric.....	85
5.3.2.	Control Surface Activity Metric	86
5.3.3.	Angular Rate Tracking Error Metric.....	86
5.4.	Motion Based Flight Simulator Results	87
6.	Application to Vectoring Thrust Spacecraft Concept Vehicle	92
6.1.	Concept Vehicle Description.....	94
6.2.	Forces and Moments that Act on the VT Spacecraft Prototype	95
6.3.	Development of Control Allocation for the VT Prototype	97
6.4.	Guidance Control Architecture.....	99
6.5.	Definition of Stability and Performance Metrics	102
6.6.	Simulation Results.....	104
6.7.	HIL Setup and Results	116
6.7.1.	Real Time Environment.....	117
6.7.2.	Data Communication	118
6.7.3.	HIL XAFF Results.	121
6.8.	Implementation Results on Gimbaled Mini-Free Flyer	122
6.8.1.	Results for Failure in EDF 3.....	124
6.8.2.	Results for Added Mass in one Arm.....	126
7.	Quaternion Based Adaptation for Spacecraft Attitude Control.....	128
7.1.	Quaternion Based Partial Feedback Linearized Control.....	130
7.2.	Quaternion Partial NLDI Control Augmented with Time-Varying Gains.....	135
7.3.	Proof of Boundedness of Adaptive Augmentation System.....	139
8.	Implementation of Quaternion Based Adaptation on a Cold Gas Spacecraft ...	142
8.1.	Vehicle and Experimental Framework Description.....	143
8.1.1.	Hardware and Test Setup	144
8.2.	Forces and Moments that Act on ACGFF Spacecraft Prototype	146
8.3.	Control Allocation for ACGFF Spacecraft Prototype.....	147
8.4.	Definition of Stability and Performance Metrics	149
8.4.1.	Angular Rate Tracking Error Activity	149
8.4.2.	Quaternion Tracking Activity.....	150
8.4.3.	Solenoid Activity.....	150

8.4.4.	Global Performance Index	150
8.5.	Implementation Results on ACGFF Spacecraft Prototype	151
8.5.1.	Description of Tests Performed	151
8.5.2.	Preliminary Implementations of NLDIQ Cascade Configuration	152
8.5.3.	Implementation Results for Nominal Conditions	155
8.5.4.	Implementation Results for Abnormal Conditions	157
8.5.5.	Global Performance Controller Assessment	160
9.	Conclusions and Recommendations	162
9.1.	Conclusions on Euler Angle Based Adaptive Controller	162
9.2.	Quaternion Based Adaptive Controller	164
9.3.	Additional Remarks	165
A.	Implementation Results on 3DR-X8 Quadrotor	167
10.	References	169

LIST OF TABLES

Table 2.1: Different Type of Immune Controllers.....	15
Table 4.1 Controller Parameters	73
Table 5.1 Performed Tests in WVU Motion Based Simulator.	88
Table 6.1 Baseline Control Gains.	105
Table 6.2 Attitude Adaptive Augmentation Parameters.	105
Table 6.3 Performance Metrics for Simulation Results.....	115
Table 6.4 Inertial and Geometrical Parameters of the MFF.	123
Table 8.1 Characteristics of Microstrain IMU	145
Table 8.2: Nominal Maneuver	151
Table 8.3 Tests Performed	160
Table 8.4 Average Performance Metrics and Performance Index per Case.	161
Table A.1 Tracking Error Activity for X8 Quadrotor Results.....	168

LIST OF FIGURES

Figure 2.1 Humoral Feedback Mechanism.....	13
Figure 2.2: General Architecture for T-B Type Immune Controller.....	15
Figure 2.3 Perturbed Motion Trajectories.....	20
Figure 2.4 Basic Definitions of Stability.....	22
Figure 2.5 Feedback Connection of a Linear System and a Non-Linear Feedback Element.....	26
Figure 2.6 Sector Non-Linearity.....	27
Figure 2.7 Disk in the Complex Plane.....	29
Figure 2.8 Simple Non-Linear Dynamic Inversion.....	31
Figure 3.1 Representation of a Rigid Body in General Motion.....	33
Figure 3.2 Rotation from Fixed Frame to Frame 1.....	37
Figure 3.3 Rotation from Frame 1 to Frame 2.....	38
Figure 3.4 Rotation from Frame 2 to Body Frame.....	38
Figure 4.1: Angular Rate Non-Linear Dynamic Inversion.....	44
Figure 4.2 Angular Rate NLDI Augmented with AIS Control.....	46
Figure 4.3 Euler Angle Output Feedback Linearization.....	53
Figure 4.4 Incremental Non-Linear Dynamic Inversion Block diagram.....	56
Figure 4.5 Model Reference AIS Adaptive Augmentation.....	58
Figure 4.6 Closed Loop Roll Error Dynamics Seen as a Lur'e Type System.....	60
Figure 4.7 Sector Non-Linearity.....	62
Figure 4.8 Model Reference AIS Adaptive Augmentation with Disturbance.....	64
Figure 4.9 System Adaptive Function.....	68
Figure 4.10 Baseline Closed Loop Control Architecture.....	72
Figure 4.11 Baseline Controller Augmented with MRAIS.....	73
Figure 4.12 Disturbance Rejection for $x_1(t)$	74
Figure 4.13 Disturbance Rejection for $x_2(t)$	74
Figure 4.14 Disturbance Rejection for $x_1(t)$	75
Figure 4.15 Disturbance Rejection for $x_2(t)$	75
Figure 5.1 Simulink Model of WVU Fighter Aircraft.....	78
Figure 5.2 WVU Motion Based Simulator Interfaced with MATLAB/SIMULINK.....	79
Figure 5.3 Baseline Control Architecture.....	81

Figure 5.4 NLDI+AIS Angular Rate Control	82
Figure 5.5 NLDI+ANN Angular Rate Control.	83
Figure 5.6 Test Outline Performed in Motion Based Simulator.....	87
Figure 5.7 Aileron Block Failure.....	88
Figure 5.8 Stabilator Block Failure.....	89
Figure 5.9 Average Angular Rate Tracking Error PI for all Failures.	90
Figure 5.10 Average Pilot Activity PI for all Failures.....	90
Figure 5.11 Average Control Surface Activity PI for all Failures.....	91
Figure 6.1 Possible Interior of Martian Lava Tube.....	92
Figure 6.2 Mars Moon Phobos.....	93
Figure 6.3 Electric Ducted Fan Version of the XAFF	94
Figure 6.4 Concept CAD of the XAFF	95
Figure 6.5 Forces and Moments that Act on the XAFF Spacecraft.....	95
Figure 6.6 XAFF Full Guidance and Control Architecture.	99
Figure 6.7 XAFF Simulation Environment	105
Figure 6.8 Euler Rates Tracking for NLDI and NLDI+MRAIS Nominal Condition.....	106
Figure 6.9 Attitude Tracking for NLDI and NLDI+MRAIS Nominal Condition.	107
Figure 6.10 Velocities Tracking for NLDI and NLDI+MRAIS Nominal Condition.	108
Figure 6.11 Position Tracking for NLDI Nominal Condition.	109
Figure 6.12 Position Tracking for NLDI+MRAIS Nominal Condition.	109
Figure 6.13 Euler Rates Tracking, NLDI and NLDI+MRAIS, Abnormal Condition. ...	110
Figure 6.14 Attitude Tracking for NLDI and NLDI+MRAIS Abnormal Condition.....	111
Figure 6.15 Velocities Tracking for NLDI and NLDI+MRAIS Abnormal Condition..	112
Figure 6.16 Position Tracking for NLDI Abnormal Condition.	113
Figure 6.17 Position Tracking for NLDI+MRAIS Abnormal Condition.	113
Figure 6.18 MRAIS Adaptive Gains for Roll Dynamics.....	114
Figure 6.19 MRAIS Adaptive Gains for Pitch Dynamics.	114
Figure 6.20 MRAIS Adaptive Gains in Yaw Dynamics.	115
Figure 6.21 Global Performance Index for the Tests Performed.....	116
Figure 6.22 Simulation Environment Used for HIL Simulation.....	117
Figure 6.23 The Steps to Compile Simulation using SRT on to Target Hardware.	118
Figure 6.24 Simulink-Real Time Target and JME Displaying Spacecraft System	119
Figure 6.25 HIL Setup at KSC Swamp Works Laboratories.....	119

Figure 6.26 Path Designation for HIL Missions.....	120
Figure 6.27 HIL Global PI Histogram Results.	122
Figure 6.28 Mini-Free Flyer Mounted in Gimbal Setup.....	122
Figure 6.29 Mini-Free Flyer Torque and Thrust vs PWM.....	123
Figure 6.30 Angular Rates Tracking Performance for a Failure in EDF 3.....	124
Figure 6.31 Roll and Yaw Tracking Performance for a Failure in EDF 3.....	125
Figure 6.32 Time History of Adaptive Gains for a Failure in EDF 3.....	125
Figure 6.33 Angular Rate Tracking Performance After Adding 130g on Left Arm.	126
Figure 6.34 Roll and Yaw Tracking Performance after Adding 130g on Left Arm.....	126
Figure 6.35 Time History of Adaptive Gains after Adding 130g on Left Arm.	127
Figure 7.1 Quaternion Attitude Representation.	129
Figure 7.2 NLDI+Quaternion Attitude Control.....	134
Figure 7.3 Adaptive Base Function.	139
Figure 7.4 Adaptive NLDI _Q +AIS Schematic.....	141
Figure 8.1 Gimbale Asteroid Cold Gas Free Flyer (ACGFF) Prototype.....	142
Figure 8.2: Cold Gas System onboard ACGFF.	143
Figure 8.3: Athena II SBC by Diamond Systems®.....	144
Figure 8.4 GX3-45 IMU by Microstrain.....	145
Figure 8.5 Schematic of Test Bed and Hardware Used on the ACGFF Prototype.....	146
Figure 8.6 Actuators Activated to Produce Roll Motion.	147
Figure 8.7 Actuators Activated to Produce Altitude Change Motion.....	148
Figure 8.8 Actuators Activated to produce Yaw and Forward Motion.	148
Figure 8.9 GUI Panel for Real Time Code Operation.	152
Figure 8.10 Implementation Results for NLDI _Q Cascade for Quaternion Tracking.	153
Figure 8.11 Angular Rate Tracking Performance for NLDI _Q Cascade.	154
Figure 8.12 Implementation Tracking Performance for Nominal Conditions.....	155
Figure 8.13 Angular Rate Activity Metric at Nominal Conditions.	156
Figure 8.14 Quaternion Activity Metric at Nominal Conditions.....	156
Figure 8.15 Total Solenoid Activity Performance Metric at Nominal Conditions.....	156
Figure 8.16 Implementation Tracking Performance for Abnormal Conditions.	157
Figure 8.17 Angular Rate Activity Metric for Thruster 2 Failure.	158
Figure 8.18 Quaternion Activity Metric for Thruster 2 Failure.....	158
Figure 8.19 Solenoid Activity Metric for Thruster 2 Failure.....	159

Figure 8.20 Quaternion Adaptive Gains for Thruster 2 Failure.....	159
Figure 8.21 Angular Rate Adaptive Gains for Thruster 2 Failure.	160
Figure 8.22 Global Performance Index Histogram Comparison.	161

SYMBOLS

$\tilde{\epsilon}_\Omega$	Angular rate performance metric
$C\Omega$	Angular rate metric cut-off value
\bar{H}	Angular momentum
ω	Angular rate vector
u, v, w	Body frame velocities
$\delta_x, \delta_y, \delta_z$	Bounded disturbance Signals
ξ	Damping ratio
ϕ, θ, ψ	Euler angles
$\phi_{ref}, \theta_{ref}, \psi_{ref}$	Euler angles reference inputs
$\dot{\phi}, \dot{\theta}, \dot{\psi}$	Euler rates
$\dot{\phi}_{ref}, \dot{\theta}_{ref}, \dot{\psi}_{ref}$	Euler rates reference inputs
\mathbf{x}^*	Equilibrium state
J	Inertia matrix
$\dot{x}, \dot{y}, \dot{z}$	Inertial reference frame velocities
m	mass
ω_n	Natural Frequency
$\Psi(\cdot)$	Non-Linear feedback element of Lur'e Network
Λ	Non-Linear function that appears after Lie Derivative in NLDI
$\eta_i, \mu_i, \gamma_i, \epsilon_i$	Parameters of AIS adaptive controller
$\tilde{\delta}\mathbf{q}$	Quaternion error performance metric
$C\Delta Q$	Quaternion error metric cut-off value
$\omega_x, \omega_y, \omega_z$	Roll rate, pitch rate, yaw rate
$\omega_{xref}, \omega_{yref}, \omega_{zref}$	Roll rate, pitch rate, yaw rate reference inputs
\tilde{s}	Solenoid activity performance metric
$C\Delta S$	Solenoid activity metric cut-off value
T_h	T-helper cells
T_s	T-suppressing cells
γ	Thrust vectoring swiveling angle
\bar{M}	Vector of total moments
\bar{F}	Vector of total forces
${}^B\bar{V}$	Velocity vector in body frame
Θ	Vector of Euler angles

ABBREVIATIONS

ACGFF	Asteroid Cold Gas Free Flyer
AIS	Artificial Immune System
ANN	Artificial Neural Networks
APP	Adaptive Pole Placement
AS	Asymptotic Stability
BIBO	Bounded Input Bounded Output
CAD	Computer Aided Design
CG	Center of Gravity
CGT	Command Generator Tracker
EDF	Electric Ducted Fan
ERAU	Embry-Riddle Aeronautical University
GAS	Global Asymptotic Stability
HIL	Hardware in the Loop
HMS	Hierarchical Multi-Self Strategy
INLDI	Incremental Non-Linear Dynamic Inversion
JME	Java Monkey Engine
KSC	Kennedy Space Center
MFF	Mini Free Flyer
MRAC	Model Reference Adaptive Control
MRAIS	Model Reference Artificial Immune System
NASA	National Aeronautics and Space Administration
NEO	Near-Earth Objects
NEA	Near-Earth Asteroids
NED	North East Down
NLDI	Non-Linear Dynamic Inversion
NLDI _Q	Non-Linear Dynamic Inversion with Quaternions
PID	Proportional Integral Derivative
PWM	Pulse Width Modulation
RTOS	Real Time Operating System
SAC	Simple Adaptive Control
SISO	Single Input Single Output
SPR	Strictly Positive Real
TV	Thrust Vectoring
UDP	User Datagram Protocol
WVU	West Virginia University
XAFF	Extreme Access Free Flyer
3 DOF	Three Degrees of Freedom
6 DOF	Six Degrees of Freedom

ABSTRACT

The main topic of this dissertation is the design, development and implementation of intelligent adaptive control techniques designed to maintain healthy performance of aerospace systems subjected to malfunctions, external parameter changes and/or unmodeled dynamics. The dissertation is focused on the development of novel adaptive control configurations that rely on non-linear functions that appear in the immune system of living organisms as main source of adaptation. One of the main goals of this dissertation is to demonstrate that these novel adaptive control architectures are able to improve overall performance and protect the system while reducing control effort and maintaining adequate operation outside bounds of nominal design. This research effort explores several phases, ranging from theoretical stability analysis, simulation and hardware implementation on different types of aerospace systems including spacecraft, aircraft and quadrotor vehicles.

The results presented in this dissertation are focused on two main adaptivity approaches, the first one is intended for aerospace systems that do not attain large angles and use exact feedback linearization of Euler angle kinematics. A proof of stability is presented by means of the circle Criterion and Lyapunov's direct method. The second approach is intended for aerospace systems that can attain large attitude angles (e.g. space systems in gravity-less environments), the adaptation is incorporated on a baseline architecture that uses partial feedback linearization of quaternions kinematics. In this case, the closed loop stability was analyzed using Lyapunov's direct method and Barbalat's Lemma. It is expected that some results presented in this dissertation can contribute towards the validation and certification of direct adaptive controllers.

1. Introduction

Recent research efforts have been directed towards the development of novel control techniques to increase the safety and operation requirements of manned and unmanned aerospace systems to compensate for internal system malfunctions as well as external upset conditions (Belcastro & Jacobson, 2010) (Edwards, Lombaerts, & Smaili, 2010). These technologies aim to increase the intelligence of the flight control system by detecting when an upset condition is present and reacting efficiently to maintain the system stability (Perez A. E., Moncayo, Perhinschi, Al Azzawi, & Togayev, 2015). The equations of motion that describe the dynamics of most aerospace systems are highly non-linear, and when subjected to external unknown disturbances (i.e. wind gusts, actuator, or structural failures, etc.) out of nominal behavior can be dangerously triggered. Classical linear control techniques have been widely studied and used to control aerospace systems dynamics, however these approaches rely on linear approximations of the plant around a nominal equilibrium condition. As a consequence, linear techniques might have limited capabilities to guarantee stability of the system when it is out of the nominal (trimmed) conditions.

Increased research efforts have been devoted over the last decades to explore novel control techniques to design compensation laws that are able to adapt to unknown external conditions and still guarantee stability of the system. Although a major feat and in fact still a matter of extensive research (Jacklin, 2008) (Falkena, van Oort, & Chu, 2001), adaptive control theory has been developed to increase the versatility of flight control systems providing safer and more robust control architectures whenever the

system experiences unknown perturbations or malfunctions.

In general, any adaptation scheme or algorithm is designed to allow the controller parameters and gains to change in time depending on the information of the states of the plant and/or the knowledge of previous control inputs. In fact, adaptation laws are often introduced into baseline control configurations designed for nominal behavior and are used as stability augmentation loops to increase robustness when the system experiences out of nominal conditions. Adaptive control theory can be majorly classified as direct, indirect and hybrid adaptive control (Nguyen & Boskovic, 2008). Direct adaptive control relies primarily on tracking errors or states of the system by reconfiguring parameters within a baseline control scheme. Indirect adaptive control relies on online estimation of plant parameters to update a baseline controller. Finally, hybrid adaptive control combines the capabilities of both, direct and indirect, approaches.

One important question that arises when dealing with adaptation laws is the possibility to describe and predict if the system will behave in a stable manner after being augmented by a non-linear/adaptive controller. Stability can be thought of as the capability of the system to remain or go back to an equilibrium condition after a disturbance. This is in fact a not trivial question to answer when dealing with the inherent complexity of adaptive controlled systems and in some instances it might not be possible to answer it in a direct closed manner.

There are some powerful analytical resources such as Lyapunov's stability, La Salle's invariant principles (or variants), Barbalat's Lemma, Circle Criterion, Describing Functions, etc., that can help draw stability conclusions about the system without actually solving the system equations of motion (Slotine & Li, 1991). In this scenario, it

is of interest to know if the adaptive controller will be capable of stabilizing the system by recovering the system to a steady state condition after a persistent or temporary perturbation, and determine the maximum boundaries of the perturbations to guarantee stable behavior.

In recent years, one promising avenue towards increasing safety of aerospace systems, and that has received special attention is the development of intelligent bio-inspired fault tolerant control laws that can accommodate a variety of failures and malfunctions of actuators, sensors, and other aerospace subsystems. Assessing adverse interactions with intelligent control laws, and developing mechanisms that can mitigate their effects through design could become a major component of future aerospace vehicle operation safety. An intelligent control system is expected to be capable to perform an assessment of the overall system health and accommodate for upset conditions. Typically, adaptive control laws for aerospace systems do not yet benefit from a comprehensive design and validation methodologies that guarantee stability and performance. In some particular cases, proves for the boundedness of the controlled system exist under certain conditions but there are no algorithms allowing the implementation of these results into the practical design process while guaranteeing stability.

Biological-based mechanisms such as the immune system have been of particular interest for the design of intelligent adaptive systems. The immune system is known to have strong robustness, self-adaptiveness, highly distributed cognitive capabilities and fast response to hostile invasions. These characteristics combined with other intelligence techniques can provide the general conceptual basis for the development of intelligent

integrated, comprehensive and robust systems for treatment of the aerospace system abnormal condition accommodation problem.

The main contributions of this dissertation are outlined and summarized below:

- Development of two different novel bio-inspired direct adaptive control configurations tailored for aerospace systems. The control laws were designed to accommodate for abnormal conditions and persistent disturbance mitigation.
- A mathematical analysis of stability and robustness of the control architecture(s) developed by means of analytical tools such as Absolute Stability, Circle Criterion, Lyapunov's stability method and extensions of Barbalat's Lemma.
- Implementation and preliminary verification of the novel adaptive controllers in simulation.
- Hardware in the Loop (HIL) implementation of the adaptive controllers in a high fidelity Six Degree of Freedom (6 DOF) simulation environment of the Extreme Access Free Flyer (XAFF). This is a concept unmanned spacecraft developed by NASA intended for Mars exploration. Implementation on a motion based simulator that incorporates the dynamics of a supersonic fighter aircraft modeled and developed at West Virginia University (WVU).
- Hardware implementation of the novel controllers on different prototype vehicles: a cold gas thruster concept prototype built by NASA (Asteroid Cold Gas Free Flyer), an Electric Ducted Fan (EDF) thrust vectoring Mini Free Flyer (MFF) designed and built by NASA and an eight motor quadrotor.
- Confirmation of improved stability of the designed controllers compared to baseline non-adaptive controllers through simulation, HIL and implementation.

The research effort presented in this dissertation has resulted in a number of publications or submissions which are outlined as follows:

Published Journals:

1. Al Azzawi D., Moncayo H., Perhinschi M. G., **Perez, A.**, Togayev, A., *Comparison of Immunity-Based Schemes for Aircraft Failure Detection and Identification*, Journal of Engineering Applications of Artificial Intelligence, Vol. 52, pp. 181-193, June 2016.
2. Moncayo, H., Moguel, I., Perhinschi, I., Al Azzawi, D., Togayev, **A.**, **Perez, A.**, *Structured Non-Self Approach for Aircraft Failure Identification within an Immunity-Based Fault Tolerance Architecture*, The Aeronautical Journal, Volume 120, No. 1225, pp. 415-434; March, 2016.
3. Perhinschi M. G., Al Azzawi D., Moncayo H., **Perez A.**, Togayev A., *Immunity-based Aircraft Actuator Failure Evaluation*, Aircraft Engineering and Aerospace Technology, Vol. 88, no. 6, 2016, DOI: 10.1108/AEAT-07-2014-0117.R1.
4. **Perez A.**, Moncayo H., Perhinschi M. G., Al Azzawi D., Togayev A., *A Bio-Inspired Adaptive Control Compensation System for an Aircraft Outside Bounds of Nominal Design*, Journal of Dynamic Systems, Measurement and Control, ASME, Vol. 137, Sep, 2015.
5. M. G. Perhinschi, D. Al Azzawi, H. Moncayo, **A. Perez.**, *Evaluation of Aircraft Sensor Failures Effects Using the Artificial Immune System Paradigm*, International Review of Aerospace Engineering, Vol. 8, N.2, (2015), pp. 71-80.
6. Al Azzawi D., Perhinschi M. G., Moncayo, H., **Perez, A.**, *A Dendritic Cell Mechanism for Detection, Identification, and Evaluation of Aircraft Failures*, Journal of Control Engineering Practice, Elsevier, Vol. 41, August 2015, pp. 134-148.
7. Togayev A., Perhinschi M. G., Al Azzawi D., Moncayo H., **Perez A.**, *Immunity-based Accommodation of Aircraft Subsystem Failures*, Aircraft Engineering and Aerospace Technology, Accepted Sep, 2015.
8. Perhinschi M. G., Al Azzawi D., Moncayo H., **Perez A.**, Togayev A., *Immunity-based Aircraft Actuator Failure Evaluation*, Aircraft Engineering and Aerospace Technology, pp. 264-272., Aug. 3, 2015.
9. Perhinschi M. G., Al Azzawi D., Moncayo H., Togayev A., **Perez A.**, Moguel I., *Immunity-based Flight Envelope Prediction at Post-failure Conditions*, Aerospace Science and Technology, Aug. 2105, pp. 264-272.

10. Al Azzawi D., Moncayo H., Perhinschi M. G., **Perez A.**, Togayev, A., *Comparison of Immunity-Based Schemes for Aircraft Failure Detection and Identification*, International Scientific Journal on Engineering Applications of Artificial Intelligence, under review since Sep. 17, 2014

Journals under Preparation:

1. **Perez A.**, Moncayo H., Balas M., Development and Stability Analysis of a Novel Model Reference Adaptive Controller Based on Immune System Approach. On progress, to be Submitted to the Elsevier Journal of Aerospace Science and Technology, November 2016.
2. **Perez A.**, Moncayo H., Design and HIL Implementation of a Feedback Linearized Bio-Inspired Adaptive Control for Guidance and Control of Unmanned Spacecraft in Extreme Environments. On progress, to be Submitted to the AIAA Journal of Guidance Control and Dynamics, December 2016.
3. **Perez A.**, Moncayo H., Development and Implementation of a Novel Fault Tolerant Adaptive Controller for Spacecraft Attitude Control. On progress, to be Submitted to the AIAA Journal of Guidance Control and Dynamics, December 2016.

Conference Papers Published:

1. Garcia, D., Moncayo, H., **Perez, A.**, Dupuis, M., Mueller, R., *Spacecraft Health Monitoring Using a Biomimetic Fault Diagnosis Scheme*, AIAA Science and Technology Forum and Exposition, 2017, Grapevine, TX.
2. Garcia, D., Moncayo, H., **Perez, A.**, Jain, C., *Low Cost Implementation of a Biomimetic Approach for UAV Health Management*, American Control Conference, Boston, MA. July 3 to 6, 2016.
3. **Perez, A.**, Moncayo, H., Prazenica, R., Zacny, K., Mueller, R., Dupuis, M., Ebert, T., *Free-Flying Robotic System for Interplanetary Prospecting and In Situ Resource Utilization*, ASCE Earth and Space 2016 Conference, Orlando, FL.
4. **Perez, A.**, Moncayo, H., Prazenica, R., Zacny, K., Mueller, R., Dupuis, M., Ebert, T., *Control Laws Development for a Free-Flying Unmanned Robotic System to Support Interplanetary Bodies Prospecting and Characterization Missions*, AIAA SciTech Conference, GNC, Aerospace Robotics and Unmanned/Autonomous Systems IV, January 5, 2016.
5. **Perez, A.**, Moncayo, H., Togayev, A., Perhinschi, M., Al Azzawi, D., *Pilot-in-the-Loop Evaluation of a Bio-Inspired Adaptive Fault Tolerant Control System in a Motion Based Flight Simulator*, AIAA Guidance, Navigation, and Control

- Conference, Science and Technology Forum, Orlando, FL., January 2015.
6. Moguel, I., Moncayo, H., **Perez, A.**, Perhinschi, M., *In-Flight Testing of a Bio-Inspired Approach for Assessment of an UAV Outside Bounds of Nominal Design*, AIAA Infotech at Aerospace, AIAA Science and Technology Forum Orlando, FL., January 2015.
 7. Chan, M., Moncayo, H., **Perez, A.**, Prazenica, R, Kim, D., Azizi, B., *Development and Flight Testing of an Unmanned Aerial System with Micro-Fiber Composite Actuators*, AIAA Infotech at Aerospace, AIAA Science and Technology Forum Orlando, FL., January 2015.
 8. **Perez A.**, Moncayo H., Moguel I., Perhinschi M. G., Al Azawii D., Togayev A., *Immunity-based Adaptive Control Laws for Aircraft Fault Tolerance*, ASME Annual Dynamic Systems and Control Conference, San Antonio, Texas, Oct. 2014.
 9. Togayev A., Perhinschi M. G, Al Azzawii D., Moncayo H., Moguel I., **Perez A.**, *Immunity-Based Abnormal Condition Accommodation of Aircraft Sub-System Failures*, ASME Annual Dynamic Systems and Control Conference, San Antonio, Texas, Oct. 2014.
 10. **Perez A.**, Moguel I., Moncayo H., Chan C. May; *Low Cost Autopilot System for an Autonomous Unmanned Aerial System*, International Conference and Exhibition on Mechanical & Aerospace Engineering, Philadelphia, Pennsylvania, Sept. 2014.
 11. Al Azawii D., Perhinschi M. G., Togayev A., Moncayo H., Moguel I., **Perez A.**, *Evaluating Aircraft Abnormal Conditions Using an Artificial Dendritic Cell Mechanism*, International Conference and Exhibition on Mechanical & Aerospace Engineering, Philadelphia, Pennsylvania, Sept. 2014.
 12. Moguel I., Moncayo H., Perhinschi M. G., Al Azawii D., **Perez A.**, Togayev A., *Biologically-Inspired Approach for Aircraft Management under Upset Conditions*, International Conference and Exhibition on Mechanical & Aerospace Engineering, Philadelphia, Pennsylvania, Sept. 2014.

2. Literature Review

2.1. Overview of Adaptive Control

Efforts to develop systems capable of controlling unknown plants or making self-adjustments to unpredictable changes in operation conditions have a rich history (Krstic, Kanellakopoulos, & Kokotovic, 1995). Starting in the early 1950's and 1960's the requirement to design autopilots for high-performance aircraft that undergo drastic changes in their dynamics when flying from one operating point to another motivated intense research efforts in adaptive control theory (Ioannou & Sun, 1995). At this time, advanced controllers were required to be able to "self-adjust" and make corresponding internal changes in the controller gains so that the closed loop dynamics behaved in stable manner even when intrinsic dynamic characteristics or aerodynamic parameters changed over time due to changes in the flight envelope.

Model reference adaptive control was first proposed by Whitaker and Keezer (Kaufman, Barkana, & Sobel, 1997) to solve the autopilot problem; this controller used a performance index minimization approach which was later known as the MIT rule (Ioannou & Sun, 1995). An alternative and similar approach known as the sensitivity method was developed in the 1960's to control systems with uncertainties (Narendra & Annaswamy, 2005). This method relies on adaptive estimation of plant parameters in a way such that a performance index is minimized. In the 1960's Kalman proposed a technique based on the optimal linear quadratic problem to design an Adaptive Pole Placement (APP) adjusting mechanism. Despite some important efforts and initial development, the adaptive flight control research was still heuristic and some bad implementations were performed as some adaptive algorithms still lacked formal stability

arguments. This triggered the NASA X-15 disaster in a flight test performed in 1967, (Ioannou & Sun, 1995) (Taylor & Adkins, 1965) which resulted in lack of interest and reduced support for research and development of non-linear and adaptive controllers in the aerospace industry.

In the late 1970's the development of positivity concepts allowed a more solid and well established stability theory that led to development and foundation of Model Reference Adaptive Control techniques (MRAC). Many authors such as Monopoli, Morse and Narendra made outstanding contributions to Single Input Single Output (SISO) adaptive model reference adaptive control (Kaufman, Barkana, & Sobel, 1997). However the 70's advancements in adaptive control still didn't account for stable performance in the presence of disturbances. Many subsequent research efforts in the 1980's and 1990's were noticeable in this direction such the work of Rohr, Kaufman, Mabus, Sobel and Balas (Kaufman, Barkana, & Sobel, 1997), which set the foundation of what is known as simple adaptive control (SAC), Command Generator Tracker (CGT) and robust extensions in infinite dimensional systems (Balas & Frost, 2014).

2.2. Bio-Inspired and Immunity Based Adaptive Control

One of the major contributions of this dissertation is to introduce the design and implementation of novel direct adaptive controller(s) that rely on the biological immune system metaphor on generic aerospace systems. The immune system of living organisms is a highly evolved and complex network that protects the body from hazardous external intruders such as viruses, bacteria, parasites, etc. (Benjamini, 1992). In order to successfully eliminate external agents, the immune system counts on an vast line of defense formed by different types of cells specialized in combating and eliminating many

different types of external intruders. The immune system can be viewed as a self-regulated feedback dynamic network capable of automatically producing the correct amount of specialized cells required to overcome a specific infection (or anomaly) and recover the correct functionality of the organism. Due to its vast complexity and intricate behavior, the immune system is still a matter of study in biological and medical sciences. However some characteristics such as robustness, adaptability, memory and fast response to repel external agents are highly desirable for application on aerospace system operations (Perez A. E., Moncayo, Perhinschi, Al Azzawi, & Togayev, 2015).

The artificial immune system (AIS) metaphor has been applied successfully to a variety of problems ranging from anomaly detection and pattern recognition, to data mining and computer security (Castro & Von Zuben, 2001). Krishna Kumar and Dasgupta (Kumar, 2003) (Dasgupta, 1999) have pioneered the application of the AIS paradigm to fault detection in aerospace systems. In addition, research efforts have been extended by WVU and ERAU researchers using the AIS paradigm integrated with a Hierarchical Multi-Self (HMS) Strategy to perform failure detection, identification, and evaluation of aerospace systems (Moncayo, Perhinschi, & Davis, 2011) (Moncayo & Perhinschi, 2011) (Perhinschi, Moncayo, & Al Azzawi, 2013) (Perez A. E., Moncayo, Perhinschi, Al Azzawi, & Togayev, 2015). However, the theory is still in evolution regarding the development of direct adaptive compensation techniques inspired by the immune system metaphor, and the application of this type of adaptation architectures in aerospace systems is in fact a novel contribution of this dissertation.

The idea of formulating an adaptive controller that mimics the interaction of *T-cells* in a living organism was first introduced by (Takahashi & Yamada, 1998). In this study, the AIS feedback mechanism was successfully applied to the velocity tracking control of a DC servo motor. Increased robustness is demonstrated through different simulations when nonlinear disturbances such as dead-zone and solid friction at the bearings are incorporated. In other relevant works, such as (Zhao, Shen, & Zhang, 2013), the immune mechanism is used to control the response of a Computer Numerical Control machine, showing increased robustness and faster error mitigation when compared to a traditional Proportional Integral Derivative (PID) controller. Other significant research results are shown in (Jie & Jiong, 2009) in which an immune feedback controller is used for the super-heated- steam temperature control in power plants. In this work the stability of a novel kind of immune controller is discussed from the point of view of the small gain theorem and bounded input bounded output (BIBO) stability. Other applications range from the control of an electric ship power system (Mitra & Venayagamoorthy, 2008) to car cruise control (Huang Jinying, Ma Bo, & Wang Haojing, 2009), in which promising results were observed.

An extended compendium of published works and literature review regarding the development and application of immune controllers is presented in (Mo, 2008). In this source different techniques and approaches (i.e. discrete and continuous time) are discussed and analyzed after being implemented into different types of dynamic systems. Some initial efforts to establish preliminary mathematical foundations to address the stability analysis of immune controllers is also presented.

2.2.1. T-B Artificial Immune Controllers

The biological immune system is composed primarily of lymphocytes and antibodies, these can be either *T-cells* or *B-cells* (Benjamini, 1992). The *T-cells* are produced in the thymus gland and are primarily composed of assistant *T_h-cells* and suppressing *T_s-cells*; they both are important in controlling the right amount of *B-cells* in the bloodstream, which is directly related to the current balance between antibodies and antigens. The *B-cells* are produced by the bone marrow and are in charge of recognizing and eliminating the antigens by increasing the production of antibodies (Moncayo H. , Perhinschi, Wilburn, & Wilburn, 2012). In the event of an infection, the number of *T_h-cells* will be superior to the amount of *T_s-cells* and hence the *B-cells* count will increase to try to reduce the number of antigens present in the organism to control the infection (Perez A. E., Moncayo, Perhinschi, Al Azzawi, & Togayev, 2015).

On the other hand, if the infection has passed, the number of *T_s-cells* produced by the thymus gland will increase. This will result in a reduction of the number of *B-cells* which is directly related to the amount of antibodies in the bloodstream. After a certain period of time, the immune system will self-regulate the production of *B-cells*, and therefore a dynamic balance will be achieved. Additionally, it is known that some of the *B-cells* are differentiated into memory cells that can establish a faster and more aggressive secondary response in future encounters with the same pathogens, achieving a form of immunity memory (Moncayo & Perhinschi, 2012). These interactions are known as the humoral immune feedback mechanism, a conceptual description of these interactions is presented in Figure 2.1. In which IL+ represents the interleukin secreted by *T_h-cells*, and IL- represents the interleukin secreted by *T_s-cells*. The interleukin is in

charge of modulating the correct amount of T_h -cells and T_s -cells depending on the required demand of B -cells.

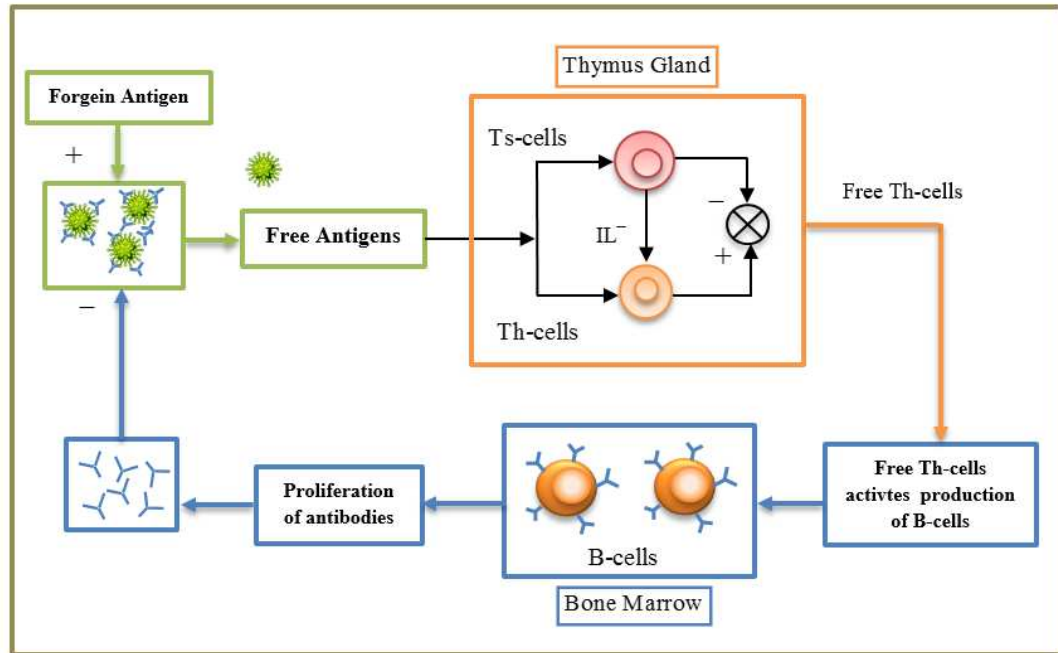


Figure 2.1 Humoral Feedback Mechanism.

A simple mathematical model to represent the interactions of the immune system can be obtained from Figure 2.1. First, the total stimulation of the T -cells received by the Bone Marrow (which will represent the total quantity of B -cells) can be expressed as the difference between T-helper cells $T_h(k)$ and the T-suppressing cells $T_s(k)$ (Takahashi & Yamada, 1998):

$$B(k) = T_h(k) - T_s(k) \quad (2.1)$$

The following set of equations can be used to relate the response of the $T_h(k)$ cells and $T_s(k)$ cells respectively (Chen & Wei, 2006) (Sun & Xu, 2010) (Yu, Cai, Jiang, & Hu, 2007) (Song, Fang, & Wang, 2009).

$$T_h(k) = c_1 \lambda(k) \quad (2.2)$$

$$T_s(k) = c_2 f(\Delta B(k)) \lambda(k) \quad (2.3)$$

where the total amount of antigens at instant of time k is defined as $\lambda(k)$, c_1 is a stimulation constant of the T_h -cells and c_2 is a suppression constant of the T_s -cells. The instantaneous change of concentration on the B -cells is defined as $\Delta\beta(k)$. Additionally, $f(\Delta B(k))$ is a function (generally non-linear) that correlates this change with the amount of T_s -cells. Substituting Eq.(2.2) and Eq.(2.3) into Eq.(2.1) yields:

$$B(k) = K [1 - \eta f(\Delta u(k))] \lambda(k) \quad (2.4)$$

where $K = c_1$ represents the system reaction rate and $\eta = c_2/c_1$ is a proportionality factor that describes the interaction between the T_h and T_s -cells. The stability of the system will generally depend on the value of η and the non-linear function that describes the change on B -cells. If $B(k)$ and $\Delta B(k)$ are replaced by a conventional control law $u(k)$ and change in control law $\Delta u(k)$ respectively and $\lambda(k)$ by an error signal $e(k)$ then we will have a feedback mechanism that is analogous to the immune system behavior. The control law will take the form (Perez A. E., Moncayo, Perhinschi, Al Azzawi, & Togayev, 2015):

$$u(k) = K [1 - \eta f(\Delta u(k))] e(k) \quad (2.5)$$

The continuous time analogous control law will be (Mo, 2008):

$$u(t) = K [1 - \eta f(\dot{u}(t))] e(t) \quad (2.6)$$

The previous expression is a special case of a more general architecture that relies in the T-B cell immune controller (Mo, 2008) and that is represented in Figure 2.2.

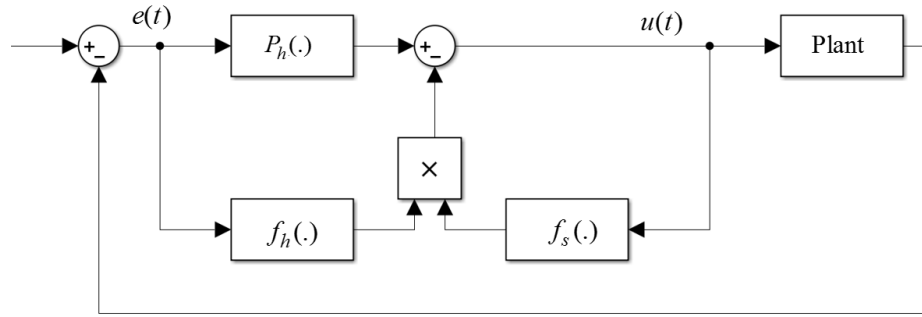


Figure 2.2: General Architecture for T-B Type Immune Controller.

In Figure 2.2 $P_h(\cdot)$ is a generic function (controller) that stimulates the production of antibodies (control signal) as a function of the error $e(t)$, $f_h(\cdot)$ is a function that describes the interaction of helper cells with antigens (error) and $f_s(\cdot)$ is a function that describes the interaction of the suppressing cells as a function of the current amount of antibodies. Table 2.1 illustrates different examples of previous published work for different types of immune controllers (Mo, 2008):

Table 2.1 Different Type of Immune Controllers.

$P_h(\cdot)$	$f_h(\cdot)$	$f_s(\cdot)$
$P_h(\cdot) = K_p$	$f_h(e(k)) = \begin{cases} 0 & e(k) \leq \frac{e_0}{k_p} \\ k_1 e(k) & \frac{e_0}{k_p} < e(k) \leq \frac{f_0}{k_p k_1} \\ f_0 & e(k) > \frac{f_0}{k_p k_1} \end{cases}$	$f_s(i(t)) = k_2 \left(\frac{1}{1 + e^{-i(t)}} - 0.5 \right)$
$P_h(\cdot) = K_p$	$f_h(\cdot) = \gamma$	$f_s(\cdot) = 1 - \frac{2}{e^{-ai(t)} + e^{ai(t)}}$
$P_h(\cdot) = K_p$	$f_h(\cdot) = \eta$	$f_s(\cdot) = 1 - \exp(-u(t)^2 / a)$

2.2.2. Double-Cell Immune Controller

An alternative way to model the complex interactions of the immune system is

considering a macro-approach. This model was proposed by (Gutnikov & Melnikov, 2003) based on the interaction of recognizers and killers. The model is bilinear in the sense it can describe better the interaction between recognizers, killer cells and antigens. The model is described by the following set of differential equations (Mo, 2008).

$$\begin{cases} \dot{e}(t) = \alpha e(t) - \eta e(t)u(t) \\ \dot{R}(t) = ke(t)^2 \quad R(0) = R_0 \\ \dot{u}(t) = vR(t) - \lambda e(t) - \mu u(t) \end{cases} \quad (2.7)$$

where $e(t)$ represents the intruder cells (error), $u(t)$ the killer cells (control input), and $R(t)$ are the recognizers. After some simplifications the following control law mimics the killer cell interaction as dependence of the number of intruder cells (Mo, 2008) :

$$\dot{u}(t) = \left[\gamma_e e^2(t) + \beta_e |u(t)| \right] e(t), \quad u(0) = u_0 \quad (2.8)$$

where γ_e denotes the error recognition coefficient, β_e denotes the immune feedback coefficient and μ_e denotes the stability coefficient of the immune controller. This control law can be used as augmentation of a nominal baseline controller.

2.3. Stability of Non-Linear Systems

The first step in the search for reliable and safe control approaches relies on the availability to study the behavior of a non-linear system. One important question that arises when dealing with any non-linear system is the possibility to predict if the system will behave in a stable manner or not. Non-linear dynamic systems can behave very differently from linear ones; sometimes linearization results can yield approximate solutions but great care should be taken since the stability results from the linearized approximations do not necessarily apply for the full non-linear plant. Predicting stability is indeed a non-trivial task when dealing with the inherent complexity of aerospace systems and in some instances it might actually not be possible to answer. Moreover, it is important to study how the system will react when subject to external perturbations and evaluate how robust it is. It would also be desirable to predict if the system will be able to turn back to a steady state condition after a consistent or temporary perturbation or if it will behave erratically and enter into a dangerous unstable behavior.

Fortunately, there exist some powerful mathematical tools that might answer these questions. One of the most useful and general approaches for studying the stability of nonlinear control systems is the theory introduced in the late 19th Century by the Russian mathematician Aleksandr Mikhailovich Lyapunov (Slotine & Li, 1991).

In Lyapunov's work, *The General Problem of Motion Stability*, he includes two methods to analyze the stability of non-linear systems, the so called *Lyapunov's Direct method* and the so called *Lyapunov's Indirect Method*. Lyapunov's direct method is actively used in this dissertation to study the stability of non-linear systems since it is a more general and vast tool. Lyapunov's type of analysis is carried out by constructing

some “energy like” functions called *Lyapunov’s functions* with the purpose of characterizing the inherit stable behavior of the system.

The following notation can be used to describe a continuous time non-linear system of differential equations, where $\mathbf{x} \in \mathfrak{R}^n$ is the state vector of the system.

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t) \quad (2.9)$$

In closed loop feedback form this system can be written as follows.

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t, \mathbf{u}(\mathbf{x})) \quad (2.10)$$

where $u(\mathbf{x})$ is a control input that depends on the state.

Non-Linear systems can be further classified as autonomous and non-autonomous systems. The generic system in Eq.(2.9) is said to be autonomous if the system dynamics are not explicitly dependent on time. In this case the system will have the form:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) \quad (2.11)$$

On the other hand, the system in Eq.(2.9) is said to be non-autonomous if it explicitly depends on time. These basic definitions will be important later in the development of the dissertation to specify what stability theorems can be applied to study the stability of the proposed adaptive architectures.

2.3.1. Equilibrium Points and Nominal Trajectories

Many of the stability problems are directly related to the so called *equilibrium points or equilibrium conditions* of a non-linear system, therefore is very important to state a formal mathematical definition of an equilibrium point. Formally a state $\mathbf{x}^* \in \mathfrak{R}^n$ is an equilibrium point or equilibrium state of the system in Eq.(2.9) if and only if after $\mathbf{x}(t^*) = \mathbf{x}^*$ the state will remain on \mathbf{x}^* always $\forall t \geq t^*$, where t^* is the time in which the

system reaches the equilibrium point \mathbf{x}^* . One might bear in mind that a non-linear system might have more than one equilibrium point and most of the analytical tools available to analyze the stability of a non-linear system will directly imply the equilibrium state of the non-linear system.

When dealing with these type of problems one might want to investigate the stability of a system around a nominal trajectory or motion instead of an equilibrium point; such a problem often arises in aerospace systems when studying the stability of an aerospace system respect to its nominal motion. In order to investigate such problems a useful transformation can be applied as follows (Slotine & Li, 1991). Let's assume $\mathbf{x}_s(t) \in \mathcal{R}^n$ is the solution of the system described by Eq.(2.11), let this system have an initial condition $\mathbf{x}_s(0) = \mathbf{x}_0$, and let the associated solution to this system be a nominal aerospace system trajectory.

Let's define some perturbed initial condition as $\mathbf{x}_\delta(0) = \mathbf{x}_s(0) + \delta\mathbf{x}_0$ and let's define the solution to this new initial condition to be given by $\mathbf{x}_\delta(t)$. One simple way to study the associated variation error of $\mathbf{x}_\delta(t)$ with respect to the nominal state trajectory $\mathbf{x}_s(t)$ is to analyze the variation of the disturbed motion with respect to the nominal motion. One can define the difference between $\mathbf{x}_\delta(t)$ and $\mathbf{x}_s(t)$ as the motion error or error dynamics as follows (see Figure 2.3):

$$\mathbf{e}(t) = \mathbf{x}_\delta(t) - \mathbf{x}_s(t) \quad (2.12)$$

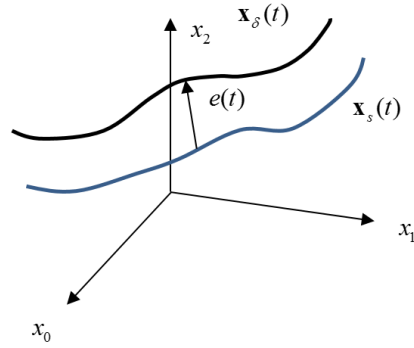


Figure 2.3 Perturbed Motion Trajectories.

Since $\mathbf{x}_\delta(t)$, $\mathbf{x}_s(t)$ are both solutions of Eq.(2.9), the following is true:

$$\begin{cases} \dot{\mathbf{x}} = f(\mathbf{x}_s) \\ \mathbf{x}_s(0) = \mathbf{x}_0 \end{cases} \rightarrow \begin{cases} \dot{\mathbf{x}} = f(\mathbf{x}_\delta) \\ \mathbf{x}_\delta(0) = \mathbf{x}_0 + \delta\mathbf{x}_0 \end{cases} \quad (2.13)$$

Therefore $\mathbf{e}(t)$ satisfies the following non-autonomous differential equation (Slotine & Li, 1991) :

$$\dot{\mathbf{e}} = f(\mathbf{x}_s + \mathbf{e}, t) - f(\mathbf{x}_s, t) = \mathbf{g}(\mathbf{e}, t) \quad (2.14)$$

System Eq.(2.14) will have the initial condition, $\mathbf{e}(0) = \delta\mathbf{x}_0$. This transformation allows us to study the dynamic behavior of the system with respect to the equilibrium point $\mathbf{0}$ which lies at the origin of the state space rather than studying the deviation of $\mathbf{x}_\delta(t)$ with respect to $\mathbf{x}_s(t)$. However, it should be noted that the error dynamics of Eq. (2.14) will correspond to a non-autonomous system.

2.3.2. General Definitions of Stability

Non-linear systems are by essence more complex than linear systems, and therefore it is important to outline some mathematical definitions of stability applicable to non-linear systems; these definitions are very important to characterize and describe the behavior of the system.

Stability:

Let A_r be a spherical region in the state space defined by $\|\mathbf{x}\| < R$. The equilibrium point \mathbf{x}^* (usually the origin) is said to be stable if and only if there exists an initial state condition $\|\mathbf{x}(0)\| < r$ such that $\|\mathbf{x}(t)\| < R$ for all $t \geq 0$, otherwise the equilibrium point is unstable (Narendra & Annaswamy, 2005), (Slotine & Li, 1991).

Asymptotic Stability:

In engineering, usually general stability of an equilibrium point might not be enough for the intended application. Often it is also required that the system actually goes back to the origin as $t \rightarrow \infty$. This concept implies that the equilibrium is stable and additionally, it is required that the system converge to the equilibrium point after some time. Formally, this concept is defined as Asymptotic Stability (AS). An equilibrium point is asymptotically stable if it is stable and in addition for an initial condition state $\|\mathbf{x}(0)\| < r$ implies that $x(t) \rightarrow 0$ as $t \rightarrow \infty$.

Global Asymptotic Stability:

If asymptotic stability holds for any initial state no matter how far away from the origin, then the equilibrium point is said to be asymptotically stable “in the large” or Globally Asymptotically Stable (GAS). The basic definitions of stability can be depicted in Figure 2.4, in which $\mathbf{x}_S(0)$, $\mathbf{x}_{AS}(0)$ and $\mathbf{x}_{GAS}(0)$ are examples of initial state conditions of stable, asymptotically stable and globally asymptotically stable trajectories respectively.

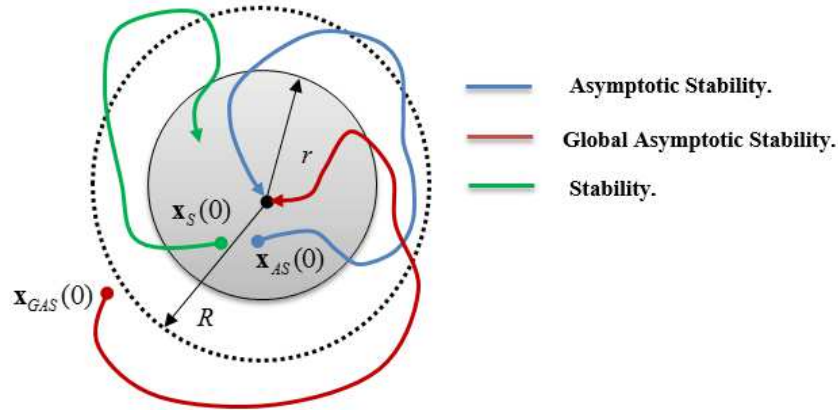


Figure 2.4 Basic Definitions of Stability.

2.3.3. Lyapunov's Direct Method

Lyapunov developed his direct method to examine the stability of a dynamic system without needing to explicitly solve the non-linear differential equation that describes the dynamics of the system. To develop his theory, he intuitively thought about the global amount of energy that is dissipated or conserved by a system. If the system is losing energy it will eventually “damp out” and will remain stable. On the other hand if the total energy of the system increases, the system will behave in an unstable manner. Lyapunov's direct method is based on the formal mathematical generalization of these concepts for any kind of non-linear system. The method is based on finding an energy like function of the state, and then showing that the derivative along trajectories of this function is always negative. In this manner conclusions may be drawn on the stability of the system without solving the system equations.

Theorem 2.1:

The equilibrium state of Eq.(2.9) will be globally asymptotically stable if there exists a scalar function of the state trajectories and time $V(\mathbf{x}, t)$ with continuous first

partial derivatives with respect to \mathbf{x} and t such that $V(\mathbf{0}, t) = 0$ and the following conditions are satisfied (Narendra & Annaswamy, 2005):

- i. $V(\mathbf{x}, t)$ is positive-definite; that is, there exists a continuous non-decreasing scalar function $\alpha(\mathbf{x})$ such that $\alpha(\mathbf{0}) = 0$ and $V(\mathbf{x}, t) \geq \alpha(\|\mathbf{x}\|) > 0$ for all t and $\mathbf{x} \neq \mathbf{0}$.
- ii. $V(\mathbf{x}, t)$ is decrescent; that is, there exists a continuous non-decreasing scalar function $\beta(\mathbf{x})$ such that $\beta(\mathbf{0}) = 0$ and $\beta(\|\mathbf{x}\|) \geq V(\mathbf{x}, t)$ for all t .
- iii. $V(\mathbf{x}, t)$ is radially unbounded, that is, $\alpha(\|\mathbf{x}\|) \rightarrow \infty$ with $\|\mathbf{x}\| \rightarrow \infty$
- iv. The derivative along trajectories of $V(\mathbf{x}, t)$ is negative-definite; that is:

$$\dot{V}(\mathbf{x}, t) = \frac{\partial V(\mathbf{x}, t)}{\partial t} + \frac{\partial V(\mathbf{x}, t)}{\partial \mathbf{x}} f(\mathbf{x}, t) \leq -\Upsilon(\|\mathbf{x}\|) < 0 \quad (2.15)$$

Where Υ is a continuous non-decreasing scalar function with $\Upsilon(\mathbf{0}) = 0$.

2.3.4. La Salle Invariant Principle

Asymptotic stability of a control system is usually the most important property to be determined prior to implementation of an adaptive controller. However, often Lyapunov's direct methods might be difficult to apply in order to assert this property. The reason is that usually $\dot{V}(\mathbf{x})$ is only a negative semi-definite function and hence no conclusions can be drawn except for local stability and bounded trajectories. In order to overcome this situation La-Salle contributed his well-known invariance principle theorem applicable only for autonomous systems (Slotine & Li, 1991) (Krstic, Kanellakopoulos, & Kokotovic, 1995), where it is still possible to draw conclusions on the asymptotic stability of a system even if $\dot{V}(\mathbf{x})$ is only negative semi-definite.

A set Ω is called a positive invariant set of the system if any solution $\mathbf{x}(t)$ that starts from a point in Ω will remain in Ω for all future time after $t \geq t_0$. Mathematically this can be written as:

$$x(t_0) \in \Omega \Rightarrow x(t) \in \Omega, \forall t \geq 0 \quad (2.16)$$

Using LaSalle's invariant set theorem, it is possible to guarantee the convergence to a desired invariant set in the state space. Let's consider the autonomous system in Eq. (2.11), and let $V(\mathbf{x})$ be a scalar function with continuous first partial derivatives that has the following properties.

- i. $V(\mathbf{x}) \rightarrow \infty$ as $\|\mathbf{x}\| \rightarrow \infty$, radially unbounded.
- ii. $\dot{V}(\mathbf{x}) \leq 0, \forall \mathbf{x} \in \Omega$.

Let E be the set of all points where $\dot{V}(\mathbf{x}) = 0$. This set can be defined as follows:

$$E = \{\mathbf{x} \mid \mathbf{x} \in \Omega, \dot{V}(\mathbf{x}) = 0\} \quad (2.17)$$

Let M be the largest invariant set in E . Then every solution of $\mathbf{x}(t)$ starting in Ω will asymptotically converge to M as $t \rightarrow \infty$. The convergence properties of the designed controller will be stronger if the dimension of M is lower. The most favorable case occurs when the largest invariant set M in E is the origin $\mathbf{x} = 0$. If that is the case, then asymptotic stability can be proven for the system (Narendra & Annaswamy, 2005).

2.3.5. Barbalat's Lemma

An extension of La Salle theorem can be used to generalize these invariant principles for non-autonomous systems. Let $\mathbf{x}^* = 0$ be an equilibrium point of Eq.(2.9) and suppose f is locally Lipschitz in \mathbf{x} and uniformly in t . Let $V(\mathbf{x}, t)$ be a continuously

differentiable, positive definite and decrescent function (conditions (i) and (ii) in Theorem 2.1) and let its derivative along trajectories be such that:

$$\dot{V}(\mathbf{x}, t) = \frac{\partial V}{\partial t} + \frac{\partial V}{\partial \mathbf{x}} f(\mathbf{x}, t) \leq -W(\mathbf{x}) \leq 0, \quad \forall t \geq 0, \forall \mathbf{x} \in \mathfrak{R}^n \quad (2.18)$$

If $\dot{W}(\mathbf{x})$ is a bounded function $\forall \mathbf{x}(t)$ then $W(\mathbf{x})$ will be uniformly continuous.

If $W(\mathbf{x})$ is uniformly continuous then all solutions of Eq.(2.9) will be globally uniformly bounded and satisfy:

$$\lim_{t \rightarrow \infty} W(\mathbf{x}(t)) = 0 \quad (2.19)$$

In addition if $W(\mathbf{x})$ is positive definite, then the equilibrium \mathbf{x}^* will be globally uniformly asymptotically stable (Krstic, Kanellakopoulos, & Kokotovic, 1995) (Harvey, 2008).

2.3.6. Absolute Stability and Circle Criterion

This section describes a powerful technique to analyze the stability of a special type of non-linear dynamic system that possess linear characteristics in the open loop but with a non-linear feedback element for stabilization purposes. The problem of analyzing this type of system often arises in many electrical and aerospace engineering applications. One of the first ones to formally address the stability of this type of systems was the former Soviet Union's scholar and control expert A.I Lur'e (Liao & Yu, 2008) (Khalil, 1996) (Passino & Yurkovich, 1998). In 1944 he proposed the following dynamic description of the problem:

$$\begin{cases} \dot{\mathbf{x}} = A\mathbf{x} + B(\Psi(\mathbf{y}, t)) \\ y = C\mathbf{x} \end{cases} \quad (2.20)$$

where $x \in \mathfrak{R}^{nx1}$, $A \in \mathfrak{R}^{nxn}$, $y \in \mathfrak{R}^{px1}$, $B \in \mathfrak{R}^{px1}$. This type of dynamic system is depicted in Figure 2.5.

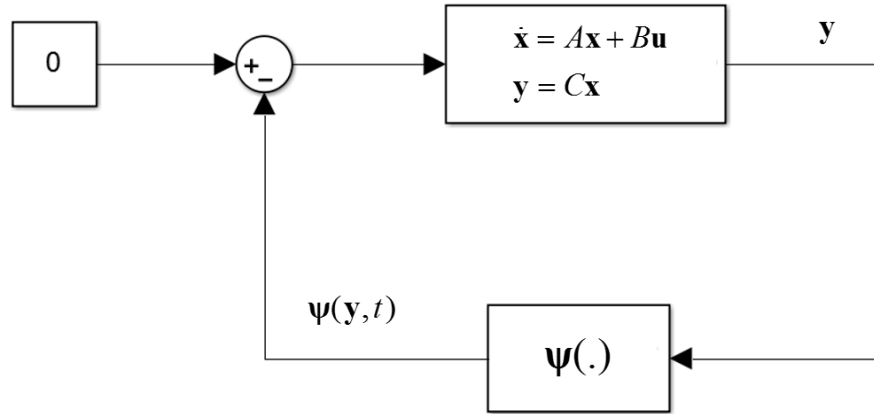


Figure 2.5 Feedback Connection of a Linear System and a Non-Linear Feedback Element.

Most of Lur'e work was known later as the “*Isolation Method*” (Liao & Yu, 2008) since it tried to fully isolate the non-linear element and tried to answer the question under what conditions the nonlinear feedback element $\Psi(y, t)$ could be designed such that the closed loop system is rendered globally asymptotically stable. Lur'e work was enriched later by other prominent researchers such as V.M. Popov who developed a well-founded frequency criterion for absolute stability theory. The main results from Lur'e and other researchers was to show under what conditions the system in (2.20) is globally asymptotically stable and can actually be summarized by means of the “*Circle Criterion*”. To start, it is important to outline some special characteristics of $\Psi(y, t)$ that are required so that the Circle Criterion of Stability can be applied.

Sector Condition (SISO Case):

In order to apply the Circle Criterion, the non-linearity $\Psi(y,t)$ needs to hold the sector condition. For a SISO system the possibly time invariant, memoryless and piecewise continuous scalar non-linearity $\Psi(y,t):[0,\infty)\times R^1 \rightarrow R^1$ is said to lie in a sector if there exists constants α, β, a and b with $a < 0 < b$ and $\beta > \alpha$ such that (Khalil, 1996) (Passino & Yurkovich, 1998):

$$\alpha y^2 \leq y\psi(t,y) \leq \beta y^2, \quad \forall t \geq 0, \forall y \in [a,b] \quad (2.21)$$

If Eq.(2.21) holds for $y \in (-\infty, \infty)$ then the sector condition holds globally. For illustrative purposes an example of a non-linearity that holds the sector condition is depicted in Figure 2.6.

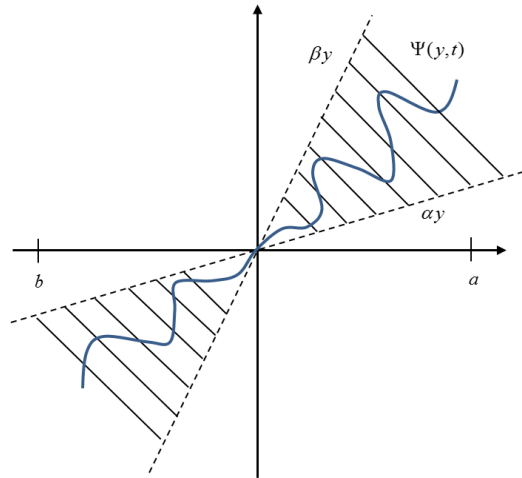


Figure 2.6 Sector Non-Linearity.

Sector Condition (MIMO Case):

For the multivariable case ($p > 1$) the form of the sector condition is more complicated. Let's start with the case in which the non-linearity $\Psi(y,t)$ is decoupled as follows.

diameter is the line segment between the points $[-1/\alpha + 0j, -1/\beta + 0j]$.

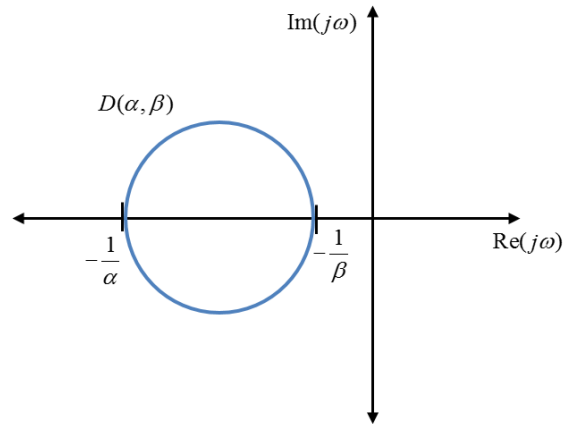


Figure 2.7 Disk in the Complex Plane

The system in Eq.(2.20) will be absolutely stable if one of the following three conditions are met (Khalil, 1996) (Passino & Yurkovich, 1998).

- i. If $0 < \alpha < \beta$, the Nyquist plot of $G(j\omega)$ does not enter the disk $D(\alpha, \beta)$ and encircles it m times in the counterclockwise direction where m is the number of poles of $G(s)$ with positive real parts.
- ii. If $0 = \alpha < \beta$, $G(s)$ is Hurwitz and the Nyquist plot of $G(j\omega)$ lies to the right of the vertical line defined by $\text{Re}(j\omega) = -1/\beta$
- iii. If $\alpha < 0 < \beta$, $G(s)$ is Hurwitz and the Nyquist plot of $G(j\omega)$ lies in the interior of the disk $D(\alpha, \beta)$.

Circle Criterion, (MIMO Case):

Considering the system Eq.(2.20) for the multivariable case ($p > 1$), where the non-linear vector feedback element $\psi(\mathbf{y}, t)$ satisfies the sector condition in Eq.(2.23) globally, the closed loop system will be absolutely stable if the following conditions are

satisfied (Khalil, 1996):

- i. The pair (A, B) is controllable and the pair (A, C) is observable.
- ii. $G_T(s) = G(s)[I + K_{\min}G(s)]^{-1}$ is Hurwitz.
- iii. $Z_T(s) = [I + K_{\max}G(s)][I + K_{\min}G(s)]^{-1}$ is Strictly Positive Real.

2.4. Feedback Linearization

Most of the literature and work associated with feedback linearization was developed around the mid 1980's, and a deep treatment of this theory was given by Isidori in his seminal book (Krstic, Kanellakopoulos, & Kokotovic, 1995). Feedback Linearization has been successfully used to address different type of control problems ranging from control of helicopters, high performance aircrafts, serial manipulators and parallel robots (Slotine & Li, 1991). The fundamental idea of feedback linearization is to transform nonlinear system dynamics into fully or partly linear ones using feedback control. By rendering the closed loop system linear, one can use linear or classical control techniques (i.e., pole placement, root locus, etc.) to stabilize the new system.

2.4.1. Non-Linear Dynamic Inversion

Let's define a generic affine non-linear system as follows:

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})\mathbf{u}(t) \quad (2.24)$$

where $\mathbf{x} \in \mathfrak{R}^n$ is the state vector, $f(\mathbf{x}) \in \mathfrak{R}^n$ and $g(\mathbf{x}) \in \mathfrak{R}^{n \times m}$ are invertible a non-linear vector functions and $\mathbf{u}(t) \in \mathfrak{R}^m$ is the vector of the inputs of the system. Since the non-linear functions are invertible it is possible to formulate a direct feedback linearizing control law for the system in Eq.(2.24) as follows:

$$u(\mathbf{x}) = g^{-1}(\mathbf{x})[v(\mathbf{x}) - f(\mathbf{x})] \quad (2.25)$$

where $v(\mathbf{x}) \in \mathfrak{R}^n$ is a virtual controller designed to guarantee stability of the closed loop system. If the function $g^{-1}(\mathbf{x})$ is invertible $\forall t$ then an exact feedback linearizing control law can be found. After inserting Eq. (2.25) into Eq. (2.24) the new closed loop dynamics of the system will yield:

$$\dot{\mathbf{x}} = v(\mathbf{x}) \quad (2.26)$$

The virtual controller $v(\mathbf{x})$ can be arbitrarily chosen in order to guarantee global asymptotic stability of the closed loop system. For convenience and simplicity the virtual controllers can be chosen as PID controllers or conventional linear state-feedback controllers. A schematic diagram that illustrates the general idea of the dynamic inversion system is shown in Figure 2.8.

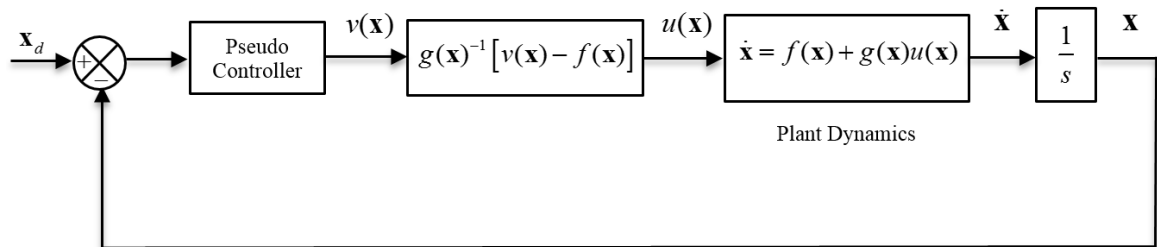


Figure 2.8 Simple Non-Linear Dynamic Inversion.

3. General Aerospace System Equations of Motion

This chapter outlines the most relevant equations of motion that describe the dynamics of any type of aerospace system that can be modeled as a rigid body. In this dissertation four main vehicles were studied: an F-15 aircraft, a thrust vectoring concept spacecraft (intended for Mars exploration), a cold gas thruster spacecraft (intended for asteroid exploration) and a quadrotor.

The equations that will be derived in this chapter can be applied to each of the vehicles under study and each of the subsequent chapters will deepen in the specific contribution of forces and moments of the actuators, aerodynamic forces or other applicable external forces and moments that apply for each vehicle.

3.1. Rigid Body Dynamics

In order to fully understand where the equations of motion come from, it is useful to understand the concept of finding the derivative of a vector in two different reference frames. This procedure is usually accomplished by means of the Transport Theorem (Sidi, 1997). Let's consider the diagram in Figure 3.1 in which the frame E is fixed. On the other hand frame B is attached to the body of a spacecraft and it is in general motion. Let's suppose we need to find the derivative of a vector in the moving frame B with respect to the fixed frame E and let's name this vector $\bar{B} = b_1\hat{i}_b + b_2\hat{j}_b + b_3\hat{k}_b$. Using the Transport Theorem or operator equation (Sidi, 1997), the derivative of vector \bar{B} with respect to frame E will be:

$$\frac{{}^E d\bar{B}}{dt} = \frac{{}^B d\bar{B}}{dt} + {}^E \bar{\omega}^B \times \bar{B} \quad (3.1)$$

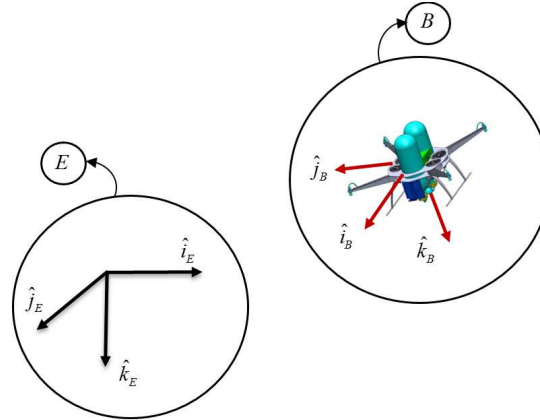


Figure 3.1 Representation of a Rigid Body in General Motion.

where ${}^E\bar{\omega}^B$ is the relative angular velocity of the rigid body in frame B with respect to Frame E .

3.1.1. Moment Equations

Newton's second law of motion can be applied to describe the rotational dynamics that result from applying a set of moments on the rigid body. The total sum of moments will be equal to the change of angular momentum over time (Yechout, 2003).

This relation can be written as:

$$\bar{M}(t) = \begin{bmatrix} \sum M_x \\ \sum M_y \\ \sum M_z \end{bmatrix} = \frac{{}^E d\bar{H}}{dt} \quad (3.2)$$

Usually, the reference point for which the external moments and angular momentum are calculated is located at the center of mass of the rigid body. The angular momentum of a rigid body can be defined as the product of its inertia tensor and the angular velocity respect to an inertial frame:

$$\bar{H} = J\omega \quad (3.3)$$

where $\boldsymbol{\omega} = [\omega_x \ \omega_y \ \omega_z]^T$ is the angular velocity vector of the rigid body with respect to the inertial reference frame and $J \in \mathfrak{R}^3$ is the inertia tensor (dyadic) of the rigid body and is usually defined as follows:

$$J = \begin{bmatrix} J_{xx} & J_{xy} & J_{xz} \\ J_{xy} & J_{yy} & J_{yz} \\ J_{xz} & J_{yz} & J_{zz} \end{bmatrix} \quad (3.4)$$

By means of the Transport Theorem, it is possible to find an expression for the angular momentum with respect to the inertial reference frame as follows:

$$\frac{{}^E d\bar{H}}{dt} = \frac{{}^B d\bar{H}}{dt} + \boldsymbol{\omega} \times \bar{H} \quad (3.5)$$

Assuming that the inertia tensor is time invariant and from the definition of angular momentum in Eq.(3.3), the first term on right hand side of Eq.(3.5) yields:

$$\frac{{}^B d\bar{H}}{dt} = J \frac{d\boldsymbol{\omega}}{dt} = \begin{bmatrix} J_{xx} & J_{xy} & J_{xz} \\ J_{xy} & J_{yy} & J_{yz} \\ J_{xz} & J_{yz} & J_{zz} \end{bmatrix} \begin{bmatrix} \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{bmatrix} = J\dot{\boldsymbol{\omega}} \quad (3.6)$$

Also, from Eq.(3.3) the second term on the right hand side of Eq.(3.5) yields:

$$\boldsymbol{\omega} \times \bar{H} = \boldsymbol{\omega} \times (J\boldsymbol{\omega}) \quad (3.7)$$

Finally, the following vector differential equation of motion for the rotational dynamics is obtained.

$$\dot{\boldsymbol{\omega}} = J^{-1} \{ -\boldsymbol{\omega} \times (J\boldsymbol{\omega}) + \bar{M}(t) \} \quad (3.8)$$

It should be noted that the angular acceleration term is arranged on the left hand side.

3.1.2. Force Equations

Newton's second law of motion can also be applied to describe the translational dynamics of a rigid body. The total sum of forces applied on the CG of the rigid body will be equal to its mass times the acceleration with respect to the inertial reference frame.

$$\bar{F}(t) = \begin{bmatrix} \sum F_x \\ \sum F_y \\ \sum F_z \end{bmatrix} = m \frac{{}^E d {}^B \bar{V}}{dt} \quad (3.9)$$

Although Newton's second law is only valid with respect to an inertial reference frame, the equations can be related to the body axis system by means of the Transport Theorem.

$$\frac{{}^E d {}^B \bar{V}}{dt} = \frac{d {}^B \bar{V}}{dt} + \boldsymbol{\omega} \times {}^B \bar{V} \quad (3.10)$$

where ${}^B \bar{V}$ is the relative velocity of the center of mass of the rigid body with respect to the inertial frame E in components of the body frame B . This vector is defined as:

$${}^B \bar{V} = u \hat{i}_b + v \hat{j}_b + w \hat{k}_b \quad (3.11)$$

Expanding terms of Eq.(3.10) yields:

$$\frac{{}^E d {}^B \bar{V}}{dt} = \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} + \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \times \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} \dot{u} + \omega_y w - \omega_z v \\ \dot{v} + \omega_z u - \omega_x w \\ \dot{w} + \omega_x v - \omega_y u \end{bmatrix} \quad (3.12)$$

Finally replacing Eq.(3.12) into Eq.(3.9) and leaving the first order derivative terms on the left hand side the rigid body, translational dynamics can be written as:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} \omega_z v - \omega_y w \\ \omega_x w - \omega_z u \\ \omega_y u - \omega_x v \end{bmatrix} + \frac{1}{m} \begin{bmatrix} \sum F_x \\ \sum F_y \\ \sum F_z \end{bmatrix} \quad (3.13)$$

3.2. Euler Angle Rotation

In order to derive the navigation equations which are useful to represent the velocities and positions of a rigid body respect to the inertial reference frame, the relative orientation of the rigid body must be known. The orientation of a moving object can be obtained by a sequence of three successive orthogonal frame rotations. If the convention in Figure 3.1 is adopted, in which the body frame consists of three orthogonal axes $\hat{i}_b, \hat{j}_b, \hat{k}_b$ and the fixed inertial frame consists of the $\hat{i}_E, \hat{j}_E, \hat{k}_E$ orthogonal axes, then there are multitude possible combinations by which three successive rotations can be performed. There are two distinct types of rotations (Sidi, 1997):

- i. Successive rotations about each of the three axes. There are six possible combinations for successive rotations namely: 1-2-3, 1-3-2, 2-1-3, 2-3-1, 3-1-2 and 3-2-1. (e.g. 1,2,3 represent successive rotation orders around roll, pitch and yaw respectively).
- ii. Non-Successive rotations. First and third rotations about the same axis with the second rotation about one of the two remaining axes: 1-2-1, 1-3-1, 2-1-2, 2-3-2, 3-1-3 and 3-2-3.

In this dissertation the rotation order of 3-2-1 will be used as it is a very well-known and common rotation convention for aircraft and other similar vehicles.

Rotation from Earth Frame to Frame 1:

The first step is to define a rotation matrix that transforms vectors from the inertial (e.g. Earth) frame to an orthogonal rotated frame around ψ ; this frame will be called frame one. This is achieved by relating unit vectors of frame 1 with unit vectors of the fixed inertial frame E .

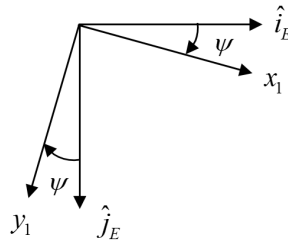


Figure 3.2 Rotation from Fixed Frame to Frame 1.

The rotation is obtained by means of the matrix \mathbf{A}_ψ that transforms vectors from the fixed frame E to frame 1.

$$\begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} = \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{i}_E \\ \hat{k}_E \\ \hat{z}_E \end{bmatrix} = \mathbf{A}_\psi \begin{bmatrix} \hat{i}_E \\ \hat{k}_E \\ \hat{z}_E \end{bmatrix} \quad (3.14)$$

Rotation from Frame 1 to Frame 2:

The next step is to form a rotation matrix that transforms vectors from the frame 1 to frame 2 around θ (see Figure 3.3). This can be achieved by means of the matrix \mathbf{A}_θ defined as follows:

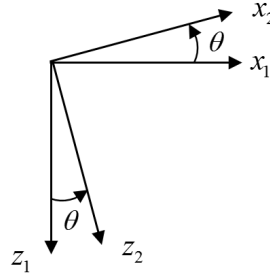


Figure 3.3 Rotation from Frame 1 to Frame 2.

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} = \mathbf{A}_\theta \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} \quad (3.15)$$

Rotation from Frame 2 to Body Frame:

The final step is to develop a rotation matrix that transforms vectors from frame two to the body frame around the roll angle ϕ (see Figure 3.4). This can be achieved by means of the rotation matrix \mathbf{A}_ϕ defined as follows:

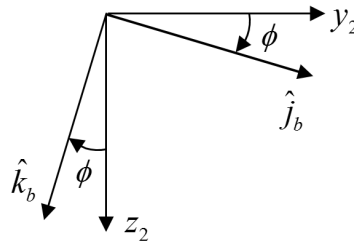


Figure 3.4 Rotation from Frame 2 to Body Frame.

$$\begin{bmatrix} \hat{i}_b \\ \hat{j}_b \\ \hat{k}_b \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} = \mathbf{A}_\phi \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} \quad (3.16)$$

If a transformation matrix that rotates any vector in the fixed frame \mathbf{E} into the body frame \mathbf{B} is required, then this can be achieved by post-multiplication of all the previous rotation matrices in Eq.(3.14)-Eq.(3.16).

$$\begin{bmatrix} \hat{i}_b \\ \hat{j}_b \\ \hat{k}_b \end{bmatrix} = \mathbf{A}_\phi \mathbf{A}_\theta \mathbf{A}_\psi \begin{bmatrix} \hat{i}_E \\ \hat{k}_E \\ \hat{z}_E \end{bmatrix} \quad (3.17)$$

Moreover, if a full transformation matrix is required to rotate a vector in the body frame to a vector in the fixed frame, it can be achieved by pre-multiplication of the transpose of the previous rotation matrices.

$$\begin{bmatrix} \hat{i}_E \\ \hat{k}_E \\ \hat{z}_E \end{bmatrix} = (\mathbf{A}_\psi)^T (\mathbf{A}_\theta)^T (\mathbf{A}_\phi)^T \begin{bmatrix} \hat{i}_b \\ \hat{j}_b \\ \hat{k}_b \end{bmatrix} \quad (3.18)$$

Therefore a Direction Cosine Matrix (DCM) that transforms vectors from a body reference frame to a inertial reference frame can be defined as:

$$DCM_b^E = \begin{bmatrix} c\theta c\psi & s\phi s\theta c\psi - c\phi s\psi & c\phi s\theta c\psi + s\phi s\psi \\ c\theta s\psi & s\phi s\theta s\psi + c\phi c\psi & c\phi s\theta s\psi - s\phi c\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix} \quad (3.19)$$

3.3. Rigid body Attitude Kinematics - Euler

With the successive matrices defined in the previous section it is possible to find a relationship between the angular rate vector $\boldsymbol{\omega}$ of the rigid body and Euler angle rates as follows (Sidi, 1997) :

$$\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \mathbf{A}_\phi \mathbf{A}_\theta \mathbf{A}_\psi \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} + \mathbf{A}_\phi \mathbf{A}_\theta \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + \mathbf{A}_\phi \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} \quad (3.20)$$

After performing the matrix multiplications, the following result is obtained:

$$\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \phi & \sin \phi \cos \theta \\ 0 & -\sin \phi & \cos \phi \cos \theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (3.21)$$

The previous vector equation can be solved for $[\dot{\phi} \ \dot{\theta} \ \dot{\psi}]^T$ as follows:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi \sec \theta & \cos \phi \sec \theta \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \quad (3.22)$$

The previous result is usually called in the literature the Euler Angle-Kinematic equations.

3.4. Navigation Equations

The navigation equations can be obtained multiplying the $[u \ v \ w]^T$ (velocities in the body frame) by the DCM_b^E to obtain the velocities components of the aerospace system in the fixed reference frame.

$$\begin{aligned} \dot{x} &= (c\theta c\psi)u + (s\phi s\theta c\psi - c\phi s\psi)v + (c\phi s\theta c\psi + s\phi s\psi)w \\ \dot{y} &= (c\theta c\psi)u + (s\phi s\theta s\psi + c\phi c\psi)v + (c\phi s\theta s\psi - s\phi c\psi)w \\ \dot{z} &= -s\theta u + (s\phi c\theta)v + (c\phi c\theta)w \end{aligned} \quad (3.23)$$

Equations (3.8), (3.13), (3.22) and (3.23) form a complete set of twelve solvable nonlinear state equations required to simulate the dynamics of any aerospace system or vehicle that behaves as a rigid body.

4. Development of Adaptive Control Laws Based on Euler Angle NLDI

This chapter discusses different baseline non-linear control architectures that are further augmented by means of direct bio-inspired adaptive controllers. The baseline controllers rely primarily on feedback linearization (exact or approximate). The feedback linearization is the first layer of the control design. It will allow the closed loop system to behave very close to a linear system for which its control gains can be designed to achieve specific performance or time response characteristics such as damping ratio or natural frequency. Since exact feedback linearization is usually impossible to achieve in real implementation, the second layer of the control design includes an adaptive augmentation system to try to eradicate uncertainties and unmodeled dynamics of the system.

A successful adaptive augmentation controller should have the ability to mitigate uncertainties in the dynamic model by increasing its overall robustness. It is also expected that the adaptive augmentation can compensate at some point for the effect of external disturbances such as system malfunctions or abrupt changes in inertial or aerodynamic characteristics.

4.1. Angular Rate NLDI Control

Most aerospace systems require some sort of angular rate control to operate correctly, not only because angular rate control is essential for overall system stability but also because angular rate sensors (gyroscopes) are one of the most reliable and easy to operate sensors to measure angular rates. As discussed in Chapter 3 the rotational dynamics of a rigid body can be defined in vector form as follows:

$$\dot{\boldsymbol{\omega}} = J^{-1} \{ -\boldsymbol{\omega} \times (J\boldsymbol{\omega}) + \bar{M}(t) \} \quad (4.1)$$

where $J \in \mathfrak{R}^{3 \times 3}$ is the inertia matrix, $\bar{M}(t) \in \mathfrak{R}^{3 \times 1}$ is the sum of external moments that act on the rigid body and $\boldsymbol{\omega} \in \mathfrak{R}^{3 \times 1}$ is the vector of angular rates. If the inertia matrix is constant in time, the rotational dynamics reduce to:

$$\dot{\boldsymbol{\omega}} = \mathbf{f}_{\omega}(\boldsymbol{\omega}) + J^{-1} \bar{M}(t) \quad (4.2)$$

where:

$$\mathbf{f}_{\omega}(\boldsymbol{\omega}) = J^{-1} \{ -\boldsymbol{\omega} \times (J\boldsymbol{\omega}) \} \quad (4.3)$$

These equations can be found on any rotating rigid body subject to external moments acting over its center of mass. One way to implement an angular rate feedback linearizing controller to stabilize the generic system in Eq.(4.2) is to use the following control law:

$$\begin{aligned} \bar{M}(t) &= \mathbf{u}_{DI}(t) = \boldsymbol{\omega} \times (J\boldsymbol{\omega}) + J\mathbf{u}_v(t) \\ &= J[\mathbf{u}_v(t) - \mathbf{f}_{\omega}(\boldsymbol{\omega})] \end{aligned} \quad (4.4)$$

Inserting control law Eq.(4.4) into the system in Eq.(4.1) will yield the following closed loop dynamics:

$$\dot{\boldsymbol{\omega}} = \mathbf{u}_v(\boldsymbol{\omega}) \quad (4.5)$$

where $\mathbf{u}_v(t) = [u_{v\omega_x}(t) \quad u_{v\omega_y}(t) \quad u_{v\omega_z}(t)]^T$ is a virtual controller. If the goal is to achieve global asymptotic tracking of specified angular rate commands, the following control law can be used:

$$\mathbf{u}_v(t) = \mathbf{K}_{p_{\omega}} (\boldsymbol{\omega}_{ref} - \boldsymbol{\omega}) + \mathbf{K}_{p_I} \int (\boldsymbol{\omega}_{ref} - \boldsymbol{\omega}) dt + \dot{\boldsymbol{\omega}}_{ref} \quad (4.6)$$

where $\mathbf{K}_{p_{\omega}}$ and \mathbf{K}_{p_I} are defined as follows:

$$\mathbf{K}_{p\omega} = \begin{bmatrix} k_{p\omega_x} & 0 & 0 \\ 0 & k_{p\omega_y} & 0 \\ 0 & 0 & k_{p\omega_z} \end{bmatrix}, \quad \mathbf{K}_{I\omega} = \begin{bmatrix} k_{I\omega_x} & 0 & 0 \\ 0 & k_{I\omega_y} & 0 \\ 0 & 0 & k_{I\omega_z} \end{bmatrix} \quad (4.7)$$

For convenience this control law can be written in scalar form for each channel:

$$\begin{aligned} u_{v\omega_x}(t) &= k_{p\omega_x}(\omega_{xref} - \omega_x) + k_{pI\omega_x} \int (\omega_{xref} - \omega_x) dt + \dot{\omega}_{xref} \\ u_{v\omega_y}(t) &= k_{p\omega_y}(\omega_{yref} - \omega_y) + k_{pI\omega_y} \int (\omega_{yref} - \omega_y) dt + \dot{\omega}_{yref} \\ u_{v\omega_z}(t) &= k_{p\omega_z}(\omega_{zref} - \omega_z) + k_{pI\omega_z} \int (\omega_{zref} - \omega_z) dt + \dot{\omega}_{zref} \end{aligned} \quad (4.8)$$

Using this control law the closed loop dynamics will yield:

$$\begin{cases} \dot{\omega}_x = k_{p\omega_x}(\omega_{xref} - \omega_x) + k_{I\omega_x} \int (\omega_{xref} - \omega_x) dt + \dot{\omega}_{xref} \\ \dot{\omega}_y = k_{p\omega_y}(\omega_{yref} - \omega_y) + k_{I\omega_y} \int (\omega_{yref} - \omega_y) dt + \dot{\omega}_{yref} \\ \dot{\omega}_z = k_{p\omega_z}(\omega_{zref} - \omega_z) + k_{I\omega_z} \int (\omega_{zref} - \omega_z) dt + \dot{\omega}_{zref} \end{cases} \quad (4.9)$$

It can be seen that this NLDI control law decouples the rotational dynamics and renders the closed loop system linear. The stability of this new system of differential equations can be treated separately as each of the angular rate channels is totally decoupled from each other. To analyze the error dynamics let's choose the following error state variables:

$$\begin{cases} e_{1\omega_x} = \int (\omega_{xref} - \omega_x) dt \\ e_{2\omega_x} = (\omega_{xref} - \omega_x) \end{cases} \quad \begin{cases} e_{1\omega_y} = \int (\omega_{yref} - \omega_y) dt \\ e_{2\omega_y} = (\omega_{yref} - \omega_y) \end{cases} \quad \begin{cases} e_{1\omega_z} = \int (\omega_{zref} - \omega_z) dt \\ e_{2\omega_z} = (\omega_{zref} - \omega_z) \end{cases} \quad (4.10)$$

Taking the derivative of the previous error terms will lead to the following decoupled state space definition for the tracking errors:

$$\begin{aligned} \begin{bmatrix} \dot{e}_{1\omega_x} \\ \dot{e}_{2\omega_x} \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -k_{I\omega_x} & -k_{p\omega_x} \end{bmatrix} \begin{bmatrix} e_{1\omega_x} \\ e_{2\omega_x} \end{bmatrix} \\ \begin{bmatrix} \dot{e}_{1\omega_y} \\ \dot{e}_{2\omega_y} \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -k_{I\omega_y} & -k_{p\omega_y} \end{bmatrix} \begin{bmatrix} e_{1\omega_y} \\ e_{2\omega_y} \end{bmatrix} \\ \begin{bmatrix} \dot{e}_{1\omega_z} \\ \dot{e}_{2\omega_z} \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -k_{I\omega_z} & -k_{p\omega_z} \end{bmatrix} \begin{bmatrix} e_{1\omega_z} \\ e_{2\omega_z} \end{bmatrix} \end{aligned} \quad (4.11)$$

The tracking errors might also be expressed in vector state-space compact form as follows:

$$\begin{bmatrix} \dot{\mathbf{e}}_1 \\ \dot{\mathbf{e}}_2 \end{bmatrix} = \begin{bmatrix} [0]_{3 \times 3} & I_{3 \times 3} \\ -\mathbf{K}_{I\omega} & -\mathbf{K}_{p\omega} \end{bmatrix} \begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \end{bmatrix} \quad (4.12)$$

where the error vectors are conveniently defined as:

$$\mathbf{e}_1(t) = [e_{1\omega_x}, e_{1\omega_y}, e_{1\omega_z}]^T, \quad \mathbf{e}_2(t) = [e_{2\omega_x}, e_{2\omega_y}, e_{2\omega_z}]^T \quad (4.13)$$

With a Routh Hurwitz or eigenvalue analysis it is easy to check that the error dynamics will be Globally Asymptotically Stable as long as the controller gains are all positive. The only two assumptions here are that there is perfect knowledge of the plant inertias (including cross terms) and that the control system has access to all angular rate measurements. Figure 4.1 shows the resultant Non-Linear Dynamic Inversion controller for angular rates in block diagram form.

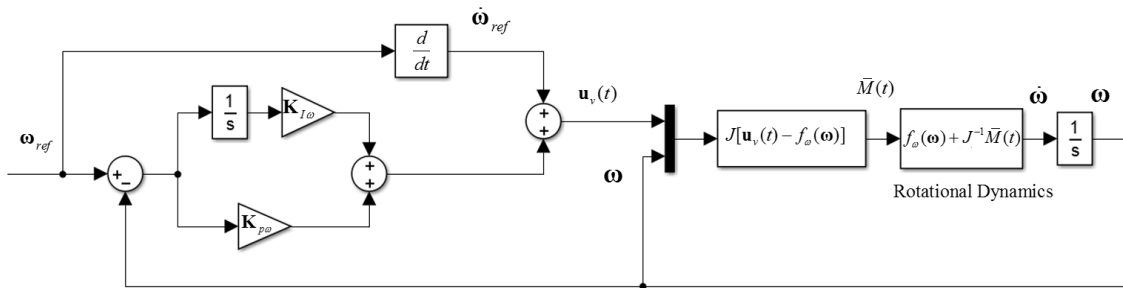


Figure 4.1: Angular Rate Non-Linear Dynamic Inversion.

4.2. Angular Rate NLDI augmented with AIS Adaptive Control

This section describes the implementation of an adaptive control configuration based on the immune system mechanism described in Chapter 2 used to augment the angular rate NLDI controller presented in Section 4.1. As mentioned before, one contribution of this dissertation is to combine the idea of the immune feedback

mechanism with the Non-Linear Dynamic Inversion approach; so far there is no formal proof of stability or any attempt to incorporate this controller into a complex non-linear aerospace system.

The adaptive laws designed and implemented here aim to incorporate adaptivity in the virtual controller proportional and integral gains. The main adaptive functions are dependent on $\Delta u_{v\omega_i}(t)$, or change in virtual control input for each of the roll, pitch and yaw rate channels. The time varying adaptation gains are defined as follows:

$$\begin{aligned} k_{p\omega_x}(t) &= k_{p\omega_x} \left\{ 1 + \eta_{\omega_x} f[\Delta u_{v\omega_x}(t)] \right\} \\ k_{p\omega_y}(t) &= k_{p\omega_y} \left\{ 1 + \eta_{\omega_y} f[\Delta u_{v\omega_y}(t)] \right\} \\ k_{p\omega_z}(t) &= k_{p\omega_z} \left\{ 1 + \eta_{\omega_z} f[\Delta u_{v\omega_z}(t)] \right\} \end{aligned} \quad (4.14)$$

$$\begin{aligned} k_{I\omega_x}(t) &= k_{I\omega_x} \left\{ 1 + \eta_{\omega_x} f[\Delta u_{v\omega_x}(t)] \right\} \\ k_{I\omega_y}(t) &= k_{I\omega_y} \left\{ 1 + \eta_{\omega_y} f[\Delta u_{v\omega_y}(t)] \right\} \\ k_{I\omega_z}(t) &= k_{I\omega_z} \left\{ 1 + \eta_{\omega_z} f[\Delta u_{v\omega_z}(t)] \right\} \end{aligned} \quad (4.15)$$

where $\eta_{\omega_x}, \eta_{\omega_y}, \eta_{\omega_z}$ are proportional constants. The non-linear adaptation basis function will take the form (Takahashi & Yamada, 1998):

$$f[\Delta u_{v\omega_i}(t)] = \left[1 - \frac{2}{e^{\gamma_i [\Delta u_{v\omega_i}(t)]^2} + e^{-\gamma_i [\Delta u_{v\omega_i}(t)]^2}} \right] \quad (4.16)$$

where γ_i are scalar proportional constants that might be different for each of the roll, pitch and yaw channels ($i = x, y, z$). The change in input is defined as follows:

$$\begin{aligned} \Delta u_{v\omega_x}(t) &= u_{v\omega_x}(t) - u_{v\omega_x}(t - \tau) \\ \Delta u_{v\omega_y}(t) &= u_{v\omega_y}(t) - u_{v\omega_y}(t - \tau) \\ \Delta u_{v\omega_z}(t) &= u_{v\omega_z}(t) - u_{v\omega_z}(t - \tau) \end{aligned} \quad (4.17)$$

The closed loop system augmented with the adaptive law will have the following closed loop dynamics:

$$\begin{cases} \dot{\omega}_x = k_{p\omega_x}(t)(\omega_{xref} - \omega_x) + k_{I\omega_x}(t) \int (\omega_{xref} - \omega_x) dt + \dot{\omega}_{xref} \\ \dot{\omega}_y = k_{p\omega_y}(t)(\omega_{yref} - \omega_y) + k_{I\omega_y}(t) \int (\omega_{yref} - \omega_y) dt + \dot{\omega}_{yref} \\ \dot{\omega}_z = k_{p\omega_z}(t)(\omega_{zref} - \omega_z) + k_{I\omega_z}(t) \int (\omega_{zref} - \omega_z) dt + \dot{\omega}_{zref} \end{cases} \quad (4.18)$$

Furthermore, the adaptive augmentation control law can be written in vector form:

$$\begin{aligned} \mathbf{u}_v(t) &= \mathbf{K}_{I\omega} \{I_{3 \times 3} + \boldsymbol{\eta} \cdot F(\Delta \mathbf{u}_v(t))\} \mathbf{e}_1 + \mathbf{K}_{p\omega} \{I_{3 \times 3} + \boldsymbol{\eta} \cdot F(\Delta \mathbf{u}_v(t))\} \mathbf{e}_2 + \dot{\boldsymbol{\omega}}_{ref} \\ &= \mathbf{K}_{I\omega} \{I_{3 \times 3} + \mathbf{G}_\omega(\Delta \mathbf{u}_v(t))\} \mathbf{e}_1 + \mathbf{K}_{p\omega} \{I_{3 \times 3} + \mathbf{G}_\omega(\Delta \mathbf{u}_v(t))\} \mathbf{e}_2 + \dot{\boldsymbol{\omega}}_{ref} \end{aligned} \quad (4.19)$$

where:

$$\mathbf{G}_\omega(\Delta \mathbf{u}_v(t)) = \boldsymbol{\eta}_\omega \cdot F(\Delta \mathbf{u}_v(t)) \quad (4.20)$$

The scale factor matrix $\boldsymbol{\eta}_\omega \in \mathfrak{R}^{3 \times 3}$ and adaptive function $F(\Delta \mathbf{u}_v(t)) \in \mathfrak{R}^{3 \times 3}$ are defined respectively as follows:

$$\boldsymbol{\eta}_\omega = \begin{bmatrix} \eta_{\omega_x} & 0 & 0 \\ 0 & \eta_{\omega_y} & 0 \\ 0 & 0 & \eta_{\omega_z} \end{bmatrix}, \quad F(\Delta \mathbf{u}_v(t)) = \begin{bmatrix} f(\Delta u_{v\omega_x}(t)) & 0 & 0 \\ 0 & f(\Delta u_{v\omega_y}(t)) & 0 \\ 0 & 0 & f(\Delta u_{v\omega_z}(t)) \end{bmatrix} \quad (4.21)$$

Figure 4.2 shows the block diagram implementation of the angular rate NLDI augmented with AIS adaptation.

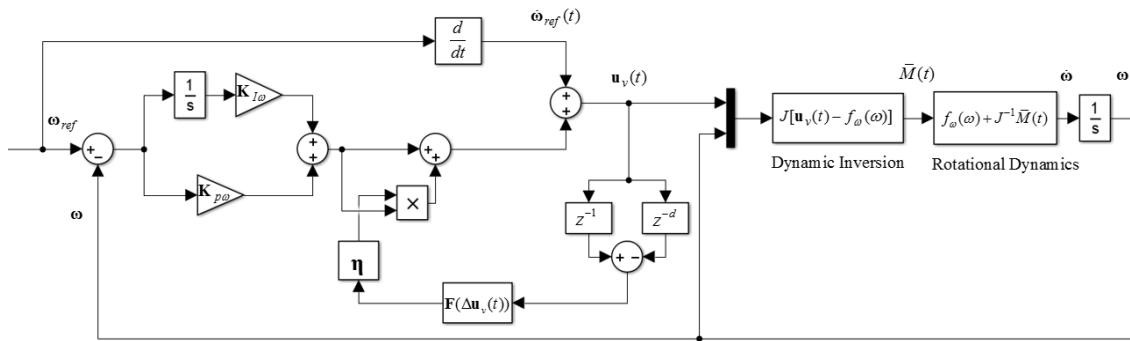


Figure 4.2 Angular Rate NLDI Augmented with AIS Control.

In a similar fashion as in Section 4.1 and keeping the same state space definition convention as in Equations (4.10)-(4.13), the closed loop error dynamics with the adaptive augmentation yield:

$$\begin{bmatrix} \dot{\mathbf{e}}_1 \\ \dot{\mathbf{e}}_2 \end{bmatrix} = \begin{bmatrix} [0]_{3 \times 3} & I_{3 \times 3} \\ -\mathbf{K}_{I\omega} & -\mathbf{K}_{p\omega} \end{bmatrix} \begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \end{bmatrix} + \begin{bmatrix} [0]_{3 \times 3} \\ I_{3 \times 3} \end{bmatrix} [-\mathbf{G}_\omega(t)\mathbf{y}_\omega(t)] \quad (4.22)$$

This system can be written in state space compact form as follows:

$$\dot{\mathbf{e}}_\omega = \mathbf{A}_\omega \mathbf{e}_\omega + \mathbf{B}_\omega [-\mathbf{G}_\omega(t)\mathbf{y}_\omega(t)] \quad (4.23)$$

where the output of the system is defined as:

$$\mathbf{y}_\omega(t) = \mathbf{C}_\omega \mathbf{e}_\omega = \begin{bmatrix} k_{I\omega_x} & 0 & 0 & k_{p\omega_x} & 0 & 0 \\ 0 & k_{I\omega_y} & 0 & 0 & k_{p\omega_y} & 0 \\ 0 & 0 & k_{I\omega_z} & 0 & 0 & k_{p\omega_z} \end{bmatrix} \begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \end{bmatrix} \quad (4.24)$$

4.3. Boundedness of Angular Rate Tracking Errors

After defining the equations of the closed loop angular rate error dynamics, it is possible to come up with a stability analysis of the adaptive augmentation system presented in Section 4.2. To start, it is required that the closed loop system holds the Kalman-Yacubovich conditions (Kaufman, Barkana, & Sobel, 1997):

$$\begin{cases} \mathbf{A}_\omega^T \mathbf{P}_\omega + \mathbf{P}_\omega \mathbf{A}_\omega = -\mathbf{Q}_\omega \\ \mathbf{P}_\omega \mathbf{B}_\omega = \mathbf{C}_\omega^T \end{cases} \quad (4.25)$$

Let's start with the second condition defined in Eq.(4.25):

$$\mathbf{P}_\omega \mathbf{B}_\omega = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} & p_{15} & p_{16} \\ p_{12} & p_{22} & p_{23} & p_{24} & p_{25} & p_{26} \\ p_{13} & p_{23} & p_{33} & p_{34} & p_{35} & p_{36} \\ p_{14} & p_{24} & p_{34} & p_{44} & p_{45} & p_{46} \\ p_{15} & p_{25} & p_{35} & p_{45} & p_{55} & p_{56} \\ p_{16} & p_{26} & p_{36} & p_{46} & p_{56} & p_{66} \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} p_{14} & p_{15} & p_{16} \\ p_{24} & p_{25} & p_{26} \\ p_{34} & p_{35} & p_{36} \\ p_{44} & p_{45} & p_{46} \\ p_{45} & p_{55} & p_{56} \\ p_{46} & p_{56} & p_{66} \end{bmatrix} = \mathbf{C}_\omega^T = \begin{bmatrix} k_{I\omega_x} & 0 & 0 \\ 0 & k_{I\omega_y} & 0 \\ 0 & 0 & k_{I\omega_z} \\ k_{p\omega_x} & 0 & 0 \\ 0 & k_{p\omega_y} & 0 \\ 0 & 0 & k_{p\omega_z} \end{bmatrix} \quad (4.26)$$

From the previous relationship it follows that:

$$\begin{aligned}
p_{14} &= k_{I\omega_x}, p_{15} = 0, p_{16} = 0, p_{24} = 0, p_{25} = k_{I\omega_y}, p_{26} = 0, \\
p_{34} &= 0, p_{35} = 0, p_{36} = k_{I\omega_z}, p_{44} = k_{p\omega_x}, p_{45} = 0, p_{46} = 0, \\
p_{45} &= 0, p_{55} = k_{p\omega_y}, p_{56} = 0, p_{46} = 0, p_{56} = 0, p_{66} = k_{p\omega_z}
\end{aligned} \tag{4.27}$$

Using the terms already solved from Eq.(4.26) and using the first K-Y condition yields:

$$\begin{aligned}
&\mathbf{A}_\omega^T \mathbf{P}_\omega + \mathbf{P}_\omega \mathbf{A}_\omega = -\mathbf{Q}_\omega \\
&= \begin{bmatrix} 2k_{I\omega_x}^2 & 0 & 0 & -p_{11} + 2k_{I\omega_x}k_{p\omega_x} & -p_{12} & -p_{13} \\ 0 & 2k_{I\omega_y}^2 & 0 & -p_{12} & -p_{22} + 2k_{I\omega_y}k_{p\omega_y} & -p_{23} \\ 0 & 0 & 2k_{I\omega_z}^2 & -p_{13} & -p_{23} & -p_{33} + 2k_{I\omega_z}k_{p\omega_z} \\ -p_{11} + 2k_{I\omega_x}k_{p\omega_x} & -p_{12} & -p_{13} & 2k_{p\omega_x}^2 - 2k_{I\omega_x} & 0 & 0 \\ -p_{12} & -p_{22} + 2k_{I\omega_y}k_{p\omega_y} & -p_{23} & 0 & 2k_{p\omega_y}^2 - 2k_{I\omega_y} & 0 \\ -p_{13} & -p_{23} & -p_{33} + 2k_{I\omega_z}k_{p\omega_z} & 0 & 0 & 2k_{p\omega_z}^2 - 2k_{I\omega_z} \end{bmatrix}
\end{aligned} \tag{4.28}$$

One way to make the \mathbf{Q}_ω matrix positive definite is to have all the non-diagonal elements as zeroes and the diagonal entries to be positive. To achieve this, the following should hold:

$$p_{11} = 2k_{I\omega_x}k_{p\omega_x}, \quad p_{12} = 0, \quad p_{13} = 0, \quad p_{22} = 2k_{I\omega_y}k_{p\omega_y}, \quad p_{23} = 0, \quad p_{33} = 2k_{I\omega_z}k_{p\omega_z} \tag{4.29}$$

The final result for the \mathbf{P}_ω and \mathbf{Q}_ω matrices is:

$$\mathbf{P}_\omega = \begin{bmatrix} 2k_{I\omega_x}k_{p\omega_x} & 0 & 0 & k_{I\omega_x} & 0 & 0 \\ 0 & 2k_{I\omega_y}k_{p\omega_y} & 0 & 0 & k_{I\omega_y} & 0 \\ 0 & 0 & 2k_{I\omega_z}k_{p\omega_z} & 0 & 0 & k_{I\omega_z} \\ k_{I\omega_x} & 0 & 0 & k_{p\omega_x} & 0 & 0 \\ 0 & k_{I\omega_y} & 0 & 0 & k_{p\omega_y} & 0 \\ 0 & 0 & k_{I\omega_z} & 0 & 0 & k_{p\omega_z} \end{bmatrix} \tag{4.30}$$

$$\mathbf{Q}_\omega = \begin{bmatrix} 2k_{I\omega_x}^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2k_{I\omega_y}^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2k_{I\omega_z}^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2k_{p\omega_x}^2 - 2k_{I\omega_x} & 0 & 0 \\ 0 & 0 & 0 & 0 & 2k_{p\omega_y}^2 - 2k_{I\omega_y} & 0 \\ 0 & 0 & 0 & 0 & 0 & 2k_{p\omega_z}^2 - 2k_{I\omega_z} \end{bmatrix} \tag{4.31}$$

To satisfy the K - Y conditions the \mathbf{Q}_ω must be positive definite. By simple inspection of the diagonal entries the following restrictions are imposed on the baseline controller gains:

$$\begin{aligned} k_{p\omega_x} &> \sqrt{k_{I\omega_x}} \\ k_{p\omega_y} &> \sqrt{k_{I\omega_y}} \\ k_{p\omega_z} &> \sqrt{k_{I\omega_z}} \end{aligned} \quad (4.32)$$

It is also required that the P_ω matrix be positive definite, which is equivalent to requiring that $\det(P_\omega) > 0$.

$$\det(P_\omega) = k_{I\omega_x} k_{I\omega_y} k_{I\omega_z} (2k_{p\omega_x}^2 - k_{I\omega_x})(2k_{p\omega_y}^2 - k_{I\omega_y})(2k_{p\omega_z}^2 - k_{I\omega_z}) > 0 \quad (4.33)$$

This implies the following conditions on the baseline controller gains:

$$\begin{aligned} k_{I\omega_x} &> 0 \\ k_{I\omega_y} &> 0 \\ k_{I\omega_z} &> 0 \end{aligned} \quad (4.34)$$

$$\begin{aligned} k_{p\omega_x} &> \sqrt{k_{I\omega_x}/2} \\ k_{p\omega_y} &> \sqrt{k_{I\omega_y}/2} \\ k_{p\omega_z} &> \sqrt{k_{I\omega_z}/2} \end{aligned} \quad (4.35)$$

However, it is worth noting that the conditions in Eq.(4.32) are stronger (more conservative) than conditions in Eq.(4.35); thus it is enough to satisfy the conditions in Eq.(4.34) and Eq.(4.32) only. With these restrictions the following Lyapunov candidate function can be postulated:

$$\begin{aligned} V(\mathbf{e}_\omega) = \frac{1}{2} \mathbf{e}_\omega^T P_\omega \mathbf{e}_\omega = &k_{I\omega_x} k_{p\omega_x} e_{1\omega_x}^2 + k_{I\omega_x} e_{1\omega_x} e_{2\omega_x} + \frac{k_{p\omega_x} e_{2\omega_x}^2}{2} + k_{I\omega_y} k_{p\omega_y} e_{1\omega_y}^2 + \dots \\ &\dots + k_{I\omega_y} e_{1\omega_y} e_{2\omega_y} + \frac{k_{p\omega_y} e_{2\omega_y}^2}{2} + k_{I\omega_z} k_{p\omega_z} e_{1\omega_z}^2 + k_{I\omega_z} e_{1\omega_z} e_{2\omega_z} + \frac{k_{p\omega_z} e_{2\omega_z}^2}{2} \end{aligned} \quad (4.36)$$

To find a stability argument the derivative along trajectories of the Lyapunov candidate function must be calculated:

$$\begin{aligned}\dot{V}(\mathbf{e}_\omega, t) &= \frac{1}{2} \dot{\mathbf{e}}_\omega^T P_\omega \mathbf{e}_\omega + \frac{1}{2} \mathbf{e}_\omega^T P_\omega \dot{\mathbf{e}}_\omega \\ &= \frac{1}{2} \{A_\omega \mathbf{e}_\omega + B_\omega [-\mathbf{G}_\omega(t) y_\omega(t)]\}^T P_\omega \mathbf{e}_\omega + \frac{1}{2} \mathbf{e}_\omega^T P_\omega \{A_\omega \mathbf{e}_\omega + B_\omega [-\mathbf{G}_\omega(t) y_\omega(t)]\}\end{aligned}\quad (4.37)$$

Cancelling and factorizing terms, the following result is obtained:

$$\dot{V}(\mathbf{e}_\omega, t) = -\frac{1}{2} \mathbf{e}_\omega^T \mathbf{Q}_\omega \mathbf{e}_\omega - y_\omega^T \mathbf{G}_\omega(t) y_\omega \leq 0 \quad (4.38)$$

The Lyapunov function derivative is negative semi-definite as long as: $\mathbf{G}_\omega(t) = \boldsymbol{\eta}_\omega \cdot F[\Delta \mathbf{u}_v(t)] \succ 0$. From the definition of the non-linear feedback function it can be corroborated that $\sup \{f[\Delta u_{v\omega_i}(t)]\} = 1$ and $\min \{f[\Delta u_{v\omega_i}(t)]\} = 0$, which means that it is bounded above and below; therefore $\boldsymbol{\eta}_\omega > 0$, which requires:

$$\eta_{\omega_x} > 0, \eta_{\omega_y} > 0, \eta_{\omega_z} > 0 \quad (4.39)$$

By means of Lyapunov's stability (Theorem 2.1), the trajectories of the system are bounded. This result is often called Lagrange stability.

4.4. Euler Angle Exact NLDI Based on Lie Derivative

It is possible to implement a full Non-Linear Dynamic Inversion control architecture that can be used to achieve the final goal of controlling the desired attitude of a rigid body (Euler angles) and Euler angle rates. For that, a single step procedure called the *Lie* Derivative can be employed to obtain an exact linear system in closed loop form (Yuan, Guoliang, Yi, & Yu, 2009). The rotational dynamics and kinematics described in Chapter 3 can be conveniently described in vector state space representation as follows:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{\Theta} \\ \dot{\boldsymbol{\omega}} \end{bmatrix} = \begin{bmatrix} \mathbf{g}(\Theta)\boldsymbol{\omega} \\ \mathbf{f}_{\omega}(\boldsymbol{\omega}) + J^{-1}\mathbf{u}(t) \end{bmatrix} \quad (4.40)$$

where $\Theta = [\phi \ \theta \ \psi]^T$ is a vector containing Euler angles, $\boldsymbol{\omega} = [\omega_x, \omega_y, \omega_z]^T$ is a vector containing angular rates and $\mathbf{u}(t) = \bar{M}(t)$ is a control input vector of moments. The functions $\mathbf{g}(\Theta)$ and $\mathbf{f}_{\omega}(\boldsymbol{\omega})$ are defined based on Euler Kinematics and rotational dynamics:

$$\mathbf{g}(\Theta) = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi \sec \theta & \cos \phi \sec \theta \end{bmatrix} \quad (4.41)$$

$$\mathbf{f}_{\omega}(\boldsymbol{\omega}) = -J^{-1}[-\boldsymbol{\omega} \times (J\boldsymbol{\omega})] \quad (4.42)$$

The system in Eq.(4.40) can be rewritten as an affine non-linear system:

$$\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x}) + \mathbf{G}(\mathbf{x})\mathbf{u}(t) = \begin{bmatrix} \mathbf{g}(\Theta)\boldsymbol{\omega} \\ \mathbf{f}_{\omega}(\boldsymbol{\omega}) \end{bmatrix} + \begin{bmatrix} [\mathbf{0}]_{3 \times 3} \\ J^{-1} \end{bmatrix} \mathbf{u}(t) \quad (4.43)$$

where the state vector is $\mathbf{x} = [\phi \ \theta \ \psi \ \omega_x \ \omega_y \ \omega_z]^T$ and $\mathbf{F}(\mathbf{x}) \in \mathfrak{R}^{6 \times 1}$, $\mathbf{G}(\mathbf{x}) \in \mathfrak{R}^{6 \times 3}$. To perform an exact feedback linearization using the *Lie* Derivative technique, a direct relationship is required between our desired output and the input of the system $\mathbf{u}(t)$ (Sieberling, Mulde, & Chu, 2010) (Snell, 1998). Let's assume that the system has

sensors that allow direct measurement of the following output vector defined as:

$$\mathbf{y} = h(\mathbf{x}) = \Theta = [\phi \quad \theta \quad \psi]^T \quad (4.44)$$

The *Lie* Derivative technique is based on finding the time derivative of the output \mathbf{y} respect to time n number of times until the input $\mathbf{u}(t)$ appears explicitly in the resultant equations. Thus a direct inversion of the equations can be established. Taking the derivative of the output once yields:

$$\dot{\mathbf{y}} = \frac{d}{dt} h(\mathbf{x}) = \frac{\partial h(\mathbf{x})}{\partial \mathbf{x}} \dot{\mathbf{x}} = \frac{\partial h(\mathbf{x})}{\partial \mathbf{x}} [\mathbf{F}(\mathbf{x}) + \mathbf{G}(\mathbf{x})\mathbf{u}(t)] \quad (4.45)$$

$$\dot{\mathbf{y}} = \dot{\Theta} = \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = g(\Theta) \boldsymbol{\omega} = \begin{bmatrix} \omega_x + \omega_y \sin \phi \tan \theta + \omega_z \cos \phi \tan \theta \\ \omega_y \cos \phi - \omega_z \sin \phi \\ \omega_y \sin \phi \sec \theta + \omega_z \cos \phi \sec \theta \end{bmatrix} \quad (4.46)$$

Since the input still doesn't appear explicitly in Eq.(4.46), another differentiation with respect to time is required. The second derivative of the output with respect to time will yield:

$$\ddot{\mathbf{y}} = \frac{d^2 h(\mathbf{x})}{dt^2} = \frac{d\dot{\mathbf{y}}}{dt} = \frac{\partial \dot{\mathbf{y}}}{\partial \mathbf{x}} \dot{\mathbf{x}} = \frac{\partial \dot{\mathbf{y}}}{\partial \mathbf{x}} [\mathbf{F}(\mathbf{x}) + \mathbf{G}(\mathbf{x})\mathbf{u}(t)] \quad (4.47)$$

$$\ddot{\mathbf{y}} = \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \underbrace{\begin{bmatrix} \dot{\theta} \dot{\phi} \tan \theta + \dot{\theta} \dot{\psi} \sec \theta \\ -\dot{\psi} \dot{\phi} \cos \theta \\ \dot{\theta} \dot{\phi} \sec \theta + \dot{\theta} \dot{\psi} \tan \theta \end{bmatrix}}_{\Lambda(\Theta, \dot{\Theta})} + \underbrace{\begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi \sec \theta & \cos \phi \sec \theta \end{bmatrix}}_{g(\Theta)} \begin{bmatrix} \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{bmatrix} = \Lambda(\Theta, \dot{\Theta}) + g(\Theta) \dot{\boldsymbol{\omega}} \quad (4.48)$$

Using the angular acceleration expression of Eq.(4.1) into Eq.(4.48) a direct expression for the second derivative of the output with respect to time can be obtained.

Now the input $\mathbf{u}(t)$ appears explicitly:

$$\ddot{\mathbf{y}} = \Lambda(\mathbf{x}) + g(\mathbf{x}) \{ J^{-1} [-\boldsymbol{\omega} \times (J\boldsymbol{\omega}) + \mathbf{u}(t)] \} \quad (4.49)$$

A control law that inverts the output dynamics in Eq.(4.49) is:

$$\mathbf{u}(t) = \boldsymbol{\omega} \times (J\boldsymbol{\omega}) + J \{ g(\mathbf{x})^{-1} [\mathbf{V}(\mathbf{x}) - \Lambda(\mathbf{x})] \} \quad (4.50)$$

Implementing this control law into the system of Eq.(4.49) will render the closed loop linear, and will take the form:

$$\dot{\mathbf{y}} = \mathbf{V}(\mathbf{y}, \dot{\mathbf{y}}) = [u_{v\phi} \quad u_{v\theta} \quad u_{v\psi}]^T \quad (4.51)$$

where $\mathbf{V}(\mathbf{y}, \dot{\mathbf{y}}) \in \mathfrak{R}^{3 \times 1}$ is a virtual controller that can be chosen arbitrarily to stabilize the closed loop system. In this case the following virtual control law was selected:

$$\begin{aligned} \ddot{\phi} = u_{v\phi}(t) &= k_{\phi} k_{D\phi} (\phi_{ref} - \phi) - k_{D\phi} \dot{\phi} \\ \ddot{\theta} = u_{v\theta}(t) &= k_{\theta} k_{D\theta} (\theta_{ref} - \theta) - k_{D\theta} \dot{\theta} \\ \ddot{\psi} = u_{v\psi}(t) &= k_{\psi} k_{D\psi} (\psi_{ref} - \psi) - k_{D\psi} \dot{\psi} \end{aligned} \quad (4.52)$$

The feedback linearization approach requires knowledge of Euler angle and Euler angles rates. For implementation purposes, gyroscopes can provide angular measurements, then Euler rates can be calculated online using Eq.(4.46). This approach is recommended instead of differentiation of angle measurements to avoid noise amplification. The controller gains can be calculated using Eq.(4.53) in order to achieve specific damping ratio and natural frequency requirements, the gains are obtained by comparing Eq.(4.52) with a second order system response. Figure 4.3 shows the block diagram implementation of the attitude output feedback linearization.

$$\begin{cases} k_{D\phi} = 2\xi_{\phi}\omega_{n\phi}, & k_{p\phi} = \omega_{n\phi}/2\xi_{\phi} \\ k_{D\theta} = 2\xi_{\theta}\omega_{n\theta}, & k_{p\theta} = \omega_{n\theta}/2\xi_{\theta} \\ k_{D\psi} = 2\xi_{\psi}\omega_{n\psi}, & k_{p\psi} = \omega_{n\psi}/2\xi_{\psi} \end{cases} \quad (4.53)$$

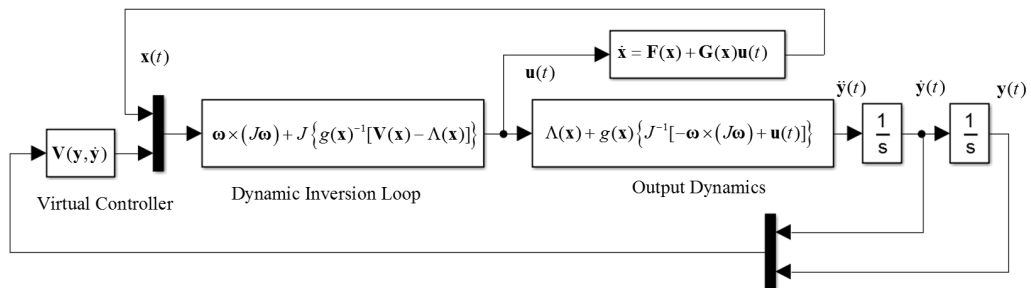


Figure 4.3 Euler Angle Output Feedback Linearization.

4.5. Euler Angle Incremental Non-Linear Dynamic Inversion

Another approach to implement a feedback linearization control in order to achieve desired angular rates and desired orientation (roll, pitch yaw angles) is called Incremental Non-Linear Dynamic Inversion (INLDI) (Acquatella, Falkena, Van Kampen, & Chu, 2012). This is a two-step approximate approach. As before, the rotational dynamics and kinematics can be conveniently described in vector state space representation as follows:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{\Theta} \\ \dot{\boldsymbol{\omega}} \end{bmatrix} = \begin{bmatrix} \mathbf{g}(\Theta)\boldsymbol{\omega} \\ \mathbf{f}_{\boldsymbol{\omega}}(\boldsymbol{\omega}) + \mathbf{J}^{-1}\mathbf{u}(t) \end{bmatrix} \quad (4.54)$$

This time, however the dynamic inversion of the system will be carried out in a two-time scale inversion process that consists on a “slow mode” and a “fast mode”. The slow mode utilizes Euler angles $[\phi_d, \theta_d, \psi_d]^T$ and outputs desired angular rates $[\omega_{xref}, \omega_{yref}, \omega_{zref}]^T$. The fast mode implements a similar dynamic inversion approach as in Section 4.1 to obtain stable inner rotational dynamics.

4.5.1. Inner Loop - Fast Mode Dynamic Inversion

To guarantee asymptotic stability of angular rates, the rotational dynamics shall be inverted in a similar fashion as the angular rate controller, as described at the beginning of this chapter. A control law that will invert the rotational dynamics of the system would be:

$$\mathbf{u}(t) = \mathbf{J}[\mathbf{u}_{vi}(\boldsymbol{\omega}) - \mathbf{f}_{\boldsymbol{\omega}}(\boldsymbol{\omega})] \quad (4.55)$$

where $\mathbf{u}_{vi}(\boldsymbol{\omega})$ is a virtual control vector that can be arbitrarily selected to stabilize the closed loop system. In this case a simple proportional controller can be selected. The

resultant control law that will feedback linearize the inner loop is:

$$\mathbf{u}(t) = \boldsymbol{\omega} \times (J\boldsymbol{\omega}) + J \begin{bmatrix} k_{p\omega_x} (\omega_{xref} - \omega_x) \\ k_{p\omega_y} (\omega_{yref} - \omega_y) \\ k_{p\omega_z} (\omega_{zref} - \omega_z) \end{bmatrix} \quad (4.56)$$

The closed loop inner loop fast dynamics are rendered linear and have the following closed loop dynamics:

$$\begin{aligned} \dot{\omega}_x &= k_{p\omega_x} (\omega_{xref} - \omega_x) \\ \dot{\omega}_y &= k_{p\omega_y} (\omega_{yref} - \omega_y) \\ \dot{\omega}_z &= k_{p\omega_z} (\omega_{zref} - \omega_z) \end{aligned} \quad (4.57)$$

4.5.2. Inner Loop - Slow Mode Dynamic Inversion

Assuming that the fast mode dynamics (Angular Rates) are much faster than the slow mode ones it is reasonable to assume that $\boldsymbol{\omega}(t) \approx \boldsymbol{\omega}_{ref}(t) \forall t$. Moreover, based on Eq. (4.46), the desired angular rates should have the following form so that the attitude kinematic equations are dynamically inverted:

$$\boldsymbol{\omega}_{ref}(t) = \mathbf{g}^{-1}(\Theta) \mathbf{u}_{v_2}(t) \quad (4.58)$$

where, $\mathbf{u}_{v_2}(\boldsymbol{\omega})$ is a vector containing virtual controllers that can be arbitrarily selected to stabilize the slow mode dynamics. The resultant angular rate command is:

$$\boldsymbol{\omega}_{ref}(t) = \begin{bmatrix} \omega_{xref} \\ \omega_{yref} \\ \omega_{zref} \end{bmatrix} = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi \sec \theta & \cos \phi \sec \theta \end{bmatrix}^{-1} \begin{bmatrix} k_\phi (\phi_{ref} - \phi) \\ k_\theta (\theta_{ref} - \theta) \\ k_\psi (\psi_{ref} - \psi) \end{bmatrix} \quad (4.59)$$

The slow mode loop will have the following linear dynamics:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} k_\phi (\phi_{ref} - \phi) \\ k_\theta (\theta_{ref} - \theta) \\ k_\psi (\psi_{ref} - \psi) \end{bmatrix} \quad (4.60)$$

Again, the virtual controllers were conveniently chosen as proportional

controllers. After performing the fast mode and slow mode dynamic inversions, the following system of three second order differential equations will describe the approximate closed loop behavior of the rotational dynamics of the system (Perez, Moncayo, & Prazenica, 2016) (Wang & Zhang, 2014). A block diagram that summarizes the Incremental NLDI approach is shown in Figure 4.4.

$$\begin{aligned}\ddot{\phi} &= k_{p\omega_x} k_\phi (\phi_d - \phi) - k_{p\omega_x} \omega_x \approx -2\xi_\phi \omega_{n\phi} \dot{\phi} - \omega_{n\phi}^2 (\phi - \phi_d) \\ \ddot{\theta} &= k_{p\omega_y} k_\theta (\theta_d - \theta) - k_{p\omega_y} \omega_y \approx -2\xi_\theta \omega_{n\theta} \dot{\theta} - \omega_{n\theta}^2 (\theta - \theta_d) \\ \ddot{\psi} &= k_{p\omega_z} k_\psi (\psi_d - \psi) - k_{p\omega_z} \omega_z \approx -2\xi_\psi \omega_{n\psi} \dot{\psi} - \omega_{n\psi}^2 (\psi - \psi_d)\end{aligned}\quad (4.61)$$

The natural frequency and damping for the system are related to the system fast and slow loop gains as follows:

$$\begin{cases} k_{p\omega_x} = 2\xi_\phi \omega_{n\phi}, & k_\phi = \omega_{n\phi} / 2\xi_\phi \\ k_{p\omega_y} = 2\xi_\theta \omega_{n\theta}, & k_\theta = \omega_{n\theta} / 2\xi_\theta \\ k_{p\omega_z} = 2\xi_\psi \omega_{n\psi}, & k_\psi = \omega_{n\psi} / 2\xi_\psi \end{cases}\quad (4.62)$$

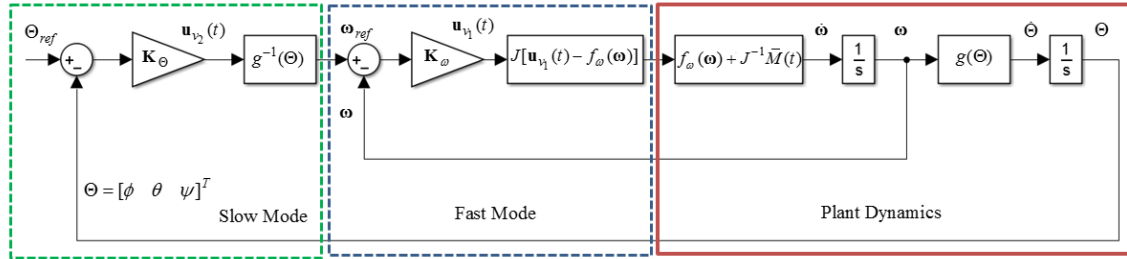


Figure 4.4 Incremental Non-Linear Dynamic Inversion Block diagram.

4.6. Exact NLDI Augmented with Novel Model Reference Adaptive AIS

This section outlines one of the most relevant contributions of this dissertation by combining concepts of model following control, direct adaptive bio-inspired control and Non-Linear Dynamic Inversion. To start, it is worth mentioning that the Lie Derivative exact feedback linearization approach presented in Section 4.3 shall be used instead of the INLDI technique outlined in Section 4.4 for adaptive augmentation purposes. Some authors (Acquatella, Falkena, Van Kampen, & Chu, 2012) have claimed that the dynamic inversion incremental approach produces a more robust closed loop system; however since it involves a time scale separation the closed loop system is approximately linear. To prove global stability of the INLDI controller with the adaptive augmentation is a much harder task due to this fact. To start it is worth recalling that the Lie Derivative feedback linearization approach yields the following system of equations:

$$\begin{aligned}\ddot{\phi} &= u_{v\phi}(t) = k_{D\phi}k_{\phi}(\phi_{ref} - \phi) - k_{D\phi}\dot{\phi} \\ \ddot{\theta} &= u_{v\theta}(t) = k_{D\theta}k_{\theta}(\theta_{ref} - \theta) - k_{D\theta}\dot{\theta} \\ \ddot{\psi} &= u_{v\psi}(t) = k_{D\psi}k_{\psi}(\psi_{ref} - \psi) - k_{D\psi}\dot{\psi}\end{aligned}\quad (4.63)$$

where the state space vector is conveniently redefined as $\mathbf{x} = [\phi \ \theta \ \psi \ \dot{\phi} \ \dot{\theta} \ \dot{\psi}]^T$.

This selection of states is convenient since the control architecture uses Euler angle rates instead of angular rates. Let's consider the adaptive control structure shown in Figure 4.5 for one of the channels, i.e. roll control (since all the channels are decoupled due to feedback linearization the same stability result will hold for all of them). Our goal is to follow a model reference plant dynamics described by the following equations:

$$\ddot{\phi}_m = u_{x_m}(t) = k_{D\phi}[k_{\phi}(\phi_{ref} - \phi_m) - \dot{\phi}_m] = k_{D\phi}k_{\phi}(\phi_{ref} - \phi_m) - k_{D\phi}\dot{\phi}_m \quad (4.64)$$

The controller will incorporate adaptivity as a function of $\Delta u_x(t)$ which is defined

as the difference between the closed loop nominal model reference plant control input $u_{xm}(t)$ and the closed loop non-adaptive control input $u_x(t)$. (Subscript m will be used to refer to the model reference plant and no subscript for the actual plant).

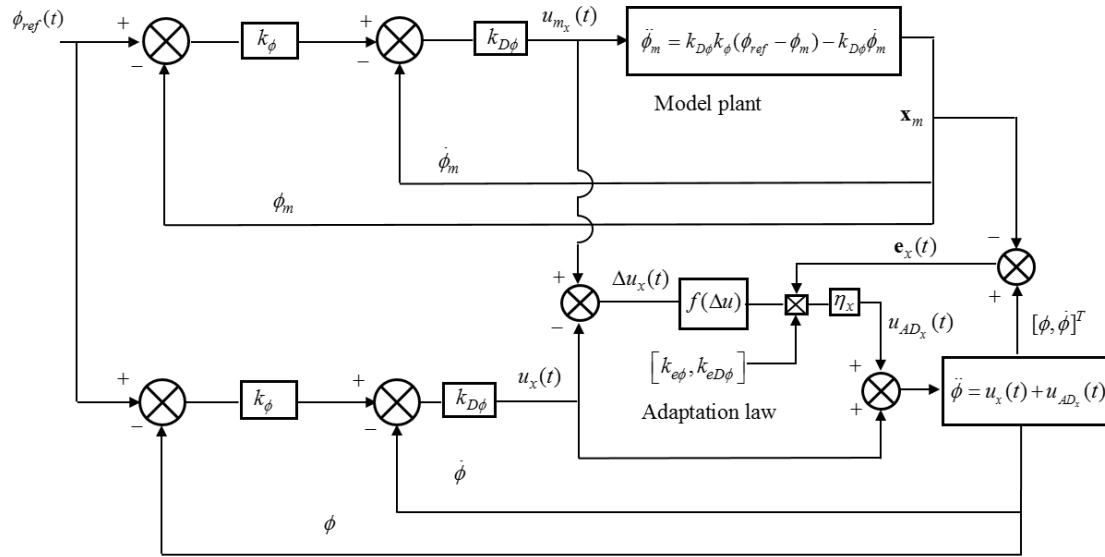


Figure 4.5 Model Reference AIS Adaptive Augmentation.

Based on the proposed control architecture we can establish the closed loop dynamics of the plant as follows:

$$\ddot{\phi} = u_{\psi\phi}(t) = u_x(t) + u_{AD_x}(t) \Rightarrow \ddot{\phi} = \underbrace{k_{D\phi}k_\phi(\phi_{ref} - \phi) - k_{D\phi}\dot{\phi}}_{u_x(t)} + \underbrace{\{-k_{e\phi}(t)(\phi - \phi_m) - k_{eD\phi}(t)(\dot{\phi} - \dot{\phi}_m)\}}_{u_{AD}(t)} \quad (4.65)$$

The adaptive gains will be defined as:

$$\begin{aligned} k_{e\phi}(t) &= k_{D\phi}k_\phi\eta_x f[\Delta u_x(t)] \\ k_{eD\phi}(t) &= k_{D\phi}\eta_x f[\Delta u_x(t)] \end{aligned} \quad (4.66)$$

where $f(\Delta u_x(t))$ is a non-linear positive definite bounded function that incorporates adaptivity. This function can be found in the immune response of some organisms when attacked by intruders and is described by (Takahashi & Yamada, 1998) (Mo, 2008):

$$f(\Delta u_x(t)) = \left[1 - \frac{2}{e^{\gamma_x[\Delta u_x(t)]^2} + e^{-\gamma_x[\Delta u_x(t)]^2}} \right] \quad (4.67)$$

The closed loop plant dynamics can be expanded as follows:

$$\ddot{\phi} = k_{D\phi}k_\phi(\phi_{ref} - \phi) - k_{D\phi}\dot{\phi} - \eta_x f[\Delta u_x(t)] \left[k_{D\phi}k_\phi(\phi - \phi_m) + k_{D\phi}(\dot{\phi} - \dot{\phi}_m) \right] \quad (4.68)$$

The difference control will be defined as:

$$\Delta u_x(t) = u_{m_x}(t) - u_x(t) \quad (4.69)$$

$$\begin{aligned} \Delta u_x(t) &= k_{D\phi}k_\phi(\phi_{ref} - \phi_m) - k_{D\phi}\dot{\phi}_m - \left\{ k_{D\phi}k_\phi(\phi_{ref} - \phi) - k_{D\phi}\dot{\phi} \right\} \\ &= k_{D\phi}k_\phi\phi_{ref} - k_{D\phi}k_\phi\phi_m - k_{D\phi}\dot{\phi}_m - k_{D\phi}k_\phi\phi_{ref} + k_{D\phi}k_\phi\phi + k_{D\phi}\dot{\phi} \\ &= k_{D\phi}k_\phi(\phi - \phi_m) + k_{D\phi}(\dot{\phi} - \dot{\phi}_m) \end{aligned} \quad (4.70)$$

Let's define the error as the difference between the actual plant and the nominal plant:

$$e_\phi = \phi(t) - \phi_m(t); \quad e_{D\phi} = \dot{\phi}(t) - \dot{\phi}_m(t) \quad (4.71)$$

Based on this definition the difference control input takes the form:

$$\Delta u_x(t) = k_{D\phi}k_\phi e_\phi + k_{D\phi}e_{D\phi} \quad (4.72)$$

It is possible to define a state space error dynamics system based on the previous definition of the error:

$$\begin{aligned} \dot{e}_\phi &= \dot{\phi}(t) - \dot{\phi}_m(t) = e_{D\phi} \\ \dot{e}_{D\phi} &= \ddot{\phi}(t) - \ddot{\phi}_m(t) = \left\{ k_{D\phi}k_\phi(\phi_{ref} - \phi) - k_{D\phi}\dot{\phi} - \eta_x f(\Delta u_x(t)) \left[k_{D\phi}k_\phi(\phi - \phi_m) + k_{D\phi}(\dot{\phi} - \dot{\phi}_m) \right] \right\} - \left\{ k_{D\phi}k_\phi(\phi_{ref} - \phi_m) - k_{D\phi}\dot{\phi}_m \right\} \\ \dot{e}_{D\phi} &= -k_{D\phi}k_\phi(\phi - \phi_m) - k_{D\phi}(\dot{\phi} - \dot{\phi}_m) - \eta_x f(\Delta u_x(t)) \left[k_{D\phi}k_\phi(\phi - \phi_m) + k_{D\phi}(\dot{\phi} - \dot{\phi}_m) \right] \end{aligned} \quad (4.73)$$

which reduces to:

$$\begin{aligned} \dot{e}_\phi &= e_{D\phi} \\ \dot{e}_{D\phi} &= -k_{D\phi}k_\phi e_\phi - k_{D\phi}e_{D\phi} - k_{D\phi}k_\phi \eta_x f(\Delta u_x(t)) e_\phi - k_{D\phi} \eta_x f(\Delta u_x(t)) e_{D\phi} \end{aligned} \quad (4.74)$$

The error dynamics can be written in state space form as follows:

$$\begin{bmatrix} \dot{e}_\phi \\ \dot{e}_{D\phi} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k_{D\phi}k_\phi & -k_{D\phi} \end{bmatrix} \begin{bmatrix} e_\phi \\ e_{D\phi} \end{bmatrix} + \begin{bmatrix} 0 \\ -k_{D\phi}k_\phi\eta_x f(\Delta u_x)e_\phi - k_{D\phi}\eta_x f(\Delta u_x)e_{D\phi} \end{bmatrix} \quad (4.75)$$

4.6.1. Proof of Absolute Stability Based on Circle Criterion

The non-linear error dynamics can be conveniently cast as a Lur'e type system such as the one shown in Figure 4.6 if we let the output to be:

$$y_x = [k_{D\phi}k_\phi, \quad k_{D\phi}] \begin{bmatrix} e_\phi \\ e_{D\phi} \end{bmatrix} = \bar{C}_x \mathbf{e}_x = k_{D\phi}k_\phi e_\phi + k_{D\phi} e_{D\phi} = \Delta u_x(t) \quad (4.76)$$

Let's define the non-linear feedback element as:

$$\Psi_x(y_x) = y_x \eta_x f(y_x) = k_{D\phi}k_\phi \eta_x f[\Delta u_x(t)]e_\phi + k_{D\phi} \eta_x f[\Delta u_x(t)]e_{D\phi} \quad (4.77)$$

The closed loop error dynamics can be written in state space form as:

$$\begin{bmatrix} \dot{e}_\phi \\ \dot{e}_{D\phi} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k_{D\phi}k_\phi & -k_{D\phi} \end{bmatrix} \begin{bmatrix} e_\phi \\ e_{D\phi} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} [-\Psi_x(y_x)] \Rightarrow \dot{\mathbf{e}}_x = \bar{A}_x \mathbf{e}_x + \bar{B}_x [-\Psi_x(y_x)] \quad (4.78)$$

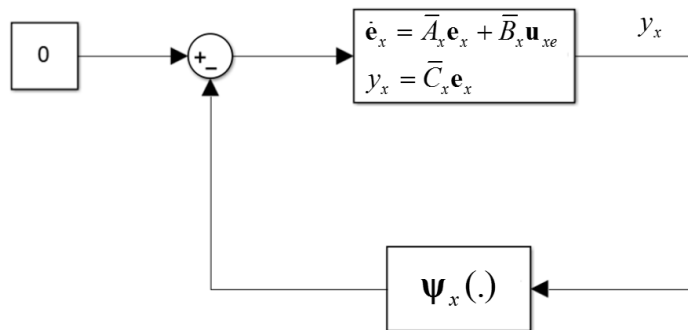


Figure 4.6 Closed Loop Roll Error Dynamics Seen as a Lur'e Type System.

To use the absolute stability criteria, the non-linear feedback element should hold the following sector condition globally (see Section 2.3.6):

$$\alpha_x y_x^2 \leq y_x \Psi_x(y_x) \leq \beta_x y_x^2, \quad y_x \in [-\infty, \infty] \quad (4.79)$$

Some insight can be obtained if the nature of the non-linear feedback function is analyzed:

$$\Psi_x(y_x) = y_x \eta_x f(y_x) = y_x \eta_x \left[1 - \frac{2}{e^{\gamma_x [\Delta u_x(t)]^2} + e^{-\gamma_x [\Delta u_x(t)]^2}} \right] \quad (4.80)$$

where $\eta_x, \beta_x, \alpha_x$ are real numbers. In this scenario $\alpha_x < \beta_x$, represents the minimum and maximum linear sectors in which the non-linear function $\Psi_x(y_x)$ can lie. For this analysis, let's consider the case in which $\alpha_x = 0$. Following the previous definition:

$$\begin{aligned} 0 &\leq y_x \Psi_x(y_x) \leq \beta y_x^2 \\ 0 &\leq y_x^2 \eta_x f(y_x) \leq \beta y_x^2 \end{aligned} \quad (4.81)$$

Dividing both sides of the inequality by y_x^2 :

$$0 \leq \eta_x f(y_x) \leq \beta_x \quad (4.82)$$

Since $\sup\{f(y_x)\} = 1$, the following inequality holds:

$$\begin{aligned} 0 &\leq \eta_x f(y_x) \leq \eta_x \sup[f(y_x)] \leq \beta_x \\ 0 &\leq \eta_x f(y_x) \leq \eta_x \leq \beta_x \end{aligned} \quad (4.83)$$

Therefore, as long as η_x is a real positive scalar, the sector condition will hold with $\eta_x = \beta_x$. Figure 4.7 shows how the nonlinearity behaves for the case in which $\eta_x = 1$. It can be seen that the non-linearity will always hold the sector condition defined in Eq.(4.81).

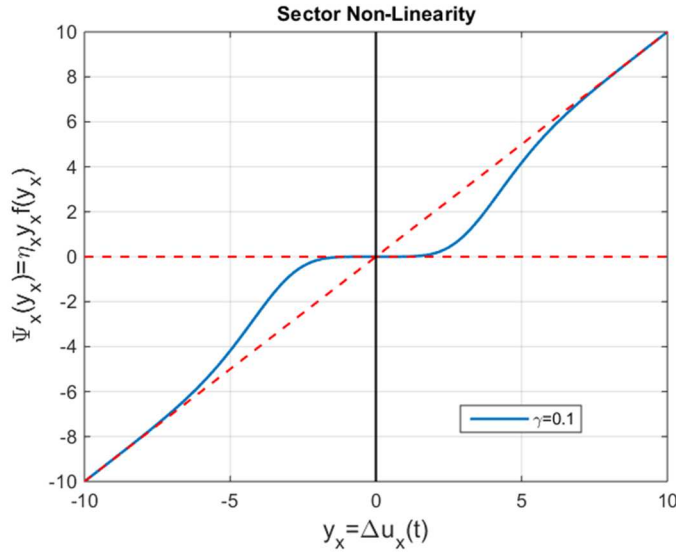


Figure 4.7 Sector Non-Linearity.

In order to proof absolute stability, the Circle Criterion presented in Chapter 2 can be used as an extension of the Popov theorem. This theorem states that the system with a feedback sector non-linearity (that holds globally) will be rendered absolutely stable if (Khalil, 1996) :

$$\operatorname{Re}[1 + \beta_x G_x(s)] > 0, \quad \rightarrow \quad \operatorname{Re}[G_x(s)] > -\frac{1}{\beta_x} \quad \forall \omega \in R \quad (4.84)$$

This means that the Nyquist plot of $G_x(j\omega)$ must lie to the right of the vertical line defined by $\operatorname{Re}(s) = -1/\beta_x$. To prove this condition a more conservative condition can be used:

$$\operatorname{Re}[G_x(s)] > 0, \quad \forall \omega \in R \quad (4.85)$$

First, the transfer function $G_x(s)$ should be specified in the frequency domain:

$$G_x(j\omega) = C(sI - \bar{A}_x)^{-1} \bar{B}_x = \frac{k_{D\phi}(j\omega) + k_{D\phi}k_\phi}{(j\omega)^2 + k_{D\phi}(j\omega) + k_{D\phi}k_\phi} = \frac{k_{D\phi}(j\omega) + k_{D\phi}k_\phi}{-\omega^2 + k_{D\phi}(j\omega) + k_{D\phi}k_\phi} \quad (4.86)$$

The real part of the transfer function yields:

$$\operatorname{Re}\{G_x(j\omega)\} = \frac{k_\phi^2 k_{D\phi}^2 + \omega^2 (k_{D\phi}^2 - k_\phi k_{D\phi})}{k_\phi^2 k_{D\phi}^2 + \omega^2 (k_{D\phi}^2 - 2k_\phi k_{D\phi}) + \omega^4} \quad (4.87)$$

From the previous relationship it is easy to see that in order to have the real part of the transfer function positive the following inequalities must hold:

$$k_{D\phi} > 2k_\phi \quad (4.88)$$

$$k_{D\phi} > k_\phi \quad (4.89)$$

Since condition in (4.88) is stronger than (4.89) this is the one that should be used.

The second condition to prove that the system is absolutely stable requires that transfer function $G_x(s) = \bar{C}_x (sI - \bar{A}_x)^{-1} \bar{B}_x$ must be Hurwitz. To check this, the following Laplace domain analysis can be performed:

$$G_x(s) = \bar{C}_x (sI - \bar{A}_x)^{-1} \bar{B}_x = \frac{k_{D\phi} s + k_{D\phi} k_\phi}{s^2 + k_{D\phi} s + k_{D\phi} k_\phi} \quad (4.90)$$

The gains of the denominator of the transfer function need to be positive, this means that the following inequalities should also hold:

$$\begin{cases} k_{D\phi} > 0 \\ k_\phi k_{D\phi} > 0 \end{cases} \quad (4.91)$$

Therefore if the conditions in Eq.(4.88) and Eq.(4.91) are satisfied, the closed loop system error dynamics will be absolutely stable.

4.6.2. Robustness on Presence of Bounded Uncertainties

As mentioned before, the exact feedback linearization approach requires that the plant dynamics and parameters are well known in order to perform an exact inversion of the system. In a real implementation scenario this is often not the case since there are always parametric modeling errors or unmodeled (unknown) dynamics. Some of these

unknowns could be modeled as external bounded disturbances of time variant nature. It is important to come up with a robustness result that proves that the adaptation law is still able to maintain stable behavior while mitigating the effect of time varying bounded uncertainties.

Let's assume that the closed loop feedback linearized system is perturbed by bounded uncertainties as follows:

$$\begin{aligned}\ddot{\phi} &= u_x(t) + u_{AD_x}(t) + \delta_x(t) = k_\phi k_{D\phi} (\phi_{ref} - \phi) - k_{D\phi} \dot{\phi} + \left\{ -k_{e\phi}(t)(\phi - \phi_m) - k_{eD\phi}(t)(\dot{\phi} - \dot{\phi}_m) \right\} + \delta_x(t) \\ \ddot{\theta} &= u_y(t) + u_{AD_y}(t) + \delta_y(t) = k_\theta k_{D\theta} (\theta_{ref} - \theta) - k_{D\theta} \dot{\theta} + \left\{ -k_{e\theta}(t)(\theta - \theta_m) - k_{eD\theta}(t)(\dot{\theta} - \dot{\theta}_m) \right\} + \delta_y(t) \\ \ddot{\psi} &= u_z(t) + u_{AD_z}(t) + \delta_z(t) = k_\psi k_{D\psi} (\psi_{ref} - \psi) - k_{D\psi} \dot{\psi} + \left\{ -k_{e\psi}(t)(\psi - \psi_m) - k_{eD\psi}(t)(\dot{\psi} - \dot{\psi}_m) \right\} + \delta_z(t)\end{aligned}\quad (4.92)$$

It is convenient to assume that each perturbation term affects each channel individually so that there is no coupling between the states of system. As before, the roll system can still be analyzed as an individual system and the stability result will hold for the other channels. Let's consider the control diagram in Figure 4.8 that incorporates the bounded time varying disturbance $\delta_x(t)$.

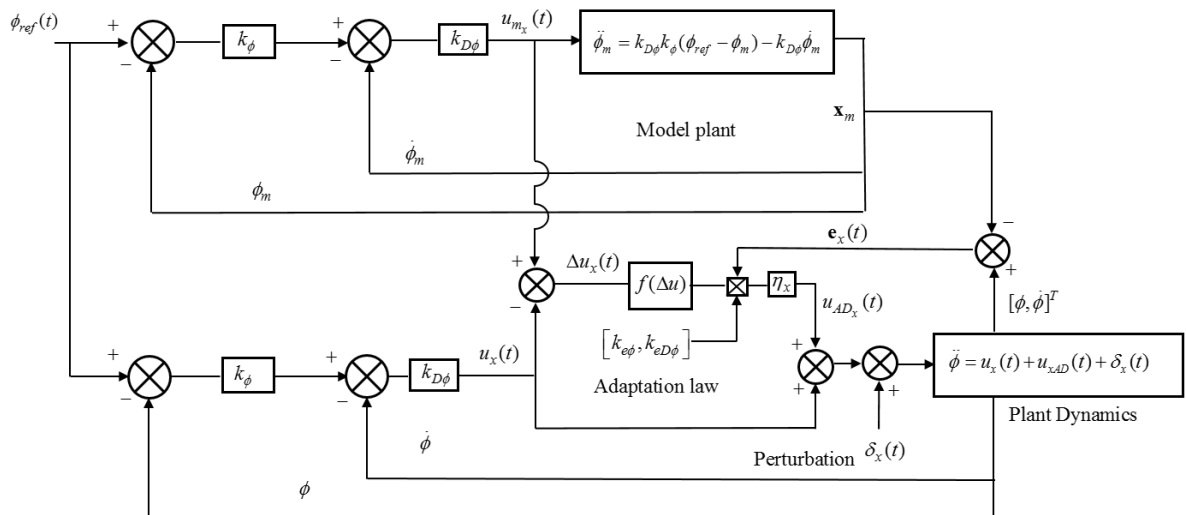


Figure 4.8 Model Reference AIS Adaptive Augmentation with Disturbance.

To consider the additional disturbance term in the closed loop system, it must be

incorporated into the error dynamics as follows:

$$\begin{aligned}
\dot{e}_\phi &= \dot{\phi}(t) - \dot{\phi}_m(t) = e_{D\phi} \\
\dot{e}_{D\phi} &= \ddot{\phi}(t) - \ddot{\phi}_m(t) = \left\{ k_{D\phi} k_\phi (\phi_{ref} - \phi) - k_{D\phi} \dot{\phi} - \eta_x f[\Delta u_x(t)] \left[k_{D\phi} k_\phi (\phi - \phi_m) + k_{D\phi} (\dot{\phi} - \dot{\phi}_m) \right] + \delta_x(t) \right\} + \dots + \\
&\quad - \left\{ k_{D\phi} k_\phi (\phi_{ref} - \phi_m) - k_{D\phi} \dot{\phi}_m \right\} \\
&= -k_{D\phi} k_\phi (\phi - \phi_m) - k_{D\phi} (\dot{\phi} - \dot{\phi}_m) - \eta_x f(\Delta u_x(t)) \left[k_{D\phi} k_\phi (\phi - \phi_m) + k_{D\phi} (\dot{\phi} - \dot{\phi}_m) \right] + \delta_x(t)
\end{aligned} \tag{4.93}$$

which reduces to:

$$\begin{aligned}
\dot{e}_\phi &= e_{D\phi} \\
\dot{e}_{D\phi} &= -k_{D\phi} k_\phi e_\phi - k_{D\phi} e_{D\phi} - k_{D\phi} k_\phi \eta_x f(\Delta u_x(t)) e_\phi - k_{D\phi} \eta_x f(\Delta u_x(t)) e_{D\phi} + \delta_x(t)
\end{aligned} \tag{4.94}$$

In state space form it yields:

$$\begin{bmatrix} \dot{e}_\phi \\ \dot{e}_{D\phi} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k_{D\phi} k_\phi & -k_{D\phi} \end{bmatrix} \begin{bmatrix} e_\phi \\ e_{D\phi} \end{bmatrix} + \begin{bmatrix} 0 \\ -k_{D\phi} k_\phi \eta_x f(\Delta u_x) e_\phi - k_{D\phi} \eta_x f(\Delta u_x) e_{D\phi} \end{bmatrix} + \begin{bmatrix} 0 \\ \delta_x(t) \end{bmatrix} \tag{4.95}$$

Similarly, as in the previous section, let the adaptive gain and the output of the system be defined respectively as:

$$\Psi_x(t) = \eta_x f(\Delta u_x(t)) \tag{4.96}$$

$$y_x(t) = \bar{C}_x \mathbf{e}_x = [k_{D\phi} k_\phi, \quad k_{D\phi}] \begin{bmatrix} e_\phi \\ e_{D\phi} \end{bmatrix} = k_{D\phi} k_\phi e_\phi + k_{D\phi} e_{D\phi} \tag{4.97}$$

The closed loop error dynamics can be written in state space form as follows:

$$\begin{bmatrix} \dot{e}_\phi \\ \dot{e}_{D\phi} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 \\ -k_{D\phi} k_\phi & -k_{D\phi} \end{bmatrix}}_{\bar{A}_x} \begin{bmatrix} e_\phi \\ e_{D\phi} \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ 1 \end{bmatrix}}_{\bar{B}_x} [-\Psi_x(t) y_x(t)] + \underbrace{\begin{bmatrix} 0 \\ \delta_x(t) \end{bmatrix}}_{\Delta_x(t)} \tag{4.98}$$

$$\dot{\mathbf{e}}_x = \bar{A}_x \mathbf{e}_x + \bar{B}_x [-\Psi_x(y_x)] + \Delta_x(t) \tag{4.99}$$

The absolute stability analysis presented in the previous section fails since there is no way to know the behavior of the disturbance $\delta_x(t)$. This means that a Lyapunov type of argument should be employed instead. To complete this proof, the first step is to satisfy the Kalman-Yacubovich conditions for the closed loop and unperturbed state space

system of Eq.(4.78). These conditions require that the two following equations hold (Balas & Frost, 2014) (Kaufman, Barkana, & Sobel, 1997).

$$\bar{A}_x^T P_x + P_x \bar{A}_x = -Q_x \quad (4.100)$$

$$P_x \bar{B}_x = \bar{C}_x^T \quad (4.101)$$

where $P_x \in \mathfrak{R}^{2 \times 2}$ and $Q_x \in \mathfrak{R}^{2 \times 2}$ are respectively symmetric and positive definite matrices. It is easier to first satisfy the condition in Eq.(4.101):

$$P_x \bar{B}_x = \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} p_{12} \\ p_{22} \end{bmatrix} = \bar{C}_x^T = \begin{bmatrix} k_{D\phi} k_\phi \\ k_{D\phi} \end{bmatrix} \rightarrow \begin{bmatrix} p_{12} \\ p_{22} \end{bmatrix} = \begin{bmatrix} k_{D\phi} k_\phi \\ k_{D\phi} \end{bmatrix} \quad (4.102)$$

Using the known values of the P_x matrix, the condition in Eq.(4.100) can be solved:

$$\underbrace{\begin{bmatrix} 0 & -k_{D\phi} k_\phi \\ 1 & -k_{D\phi} \end{bmatrix}}_{\bar{A}_x^T} \underbrace{\begin{bmatrix} p_{11} & k_{D\phi} k_\phi \\ k_{D\phi} k_\phi & k_{D\phi} \end{bmatrix}}_{P_x} + \underbrace{\begin{bmatrix} p_{11} & k_{D\phi} k_\phi \\ k_{D\phi} k_\phi & k_{D\phi} \end{bmatrix}}_{P_x} \underbrace{\begin{bmatrix} 0 & 1 \\ -k_{D\phi} k_\phi & -k_{D\phi} \end{bmatrix}}_{\bar{A}_x} = - \underbrace{\begin{bmatrix} 2(k_{D\phi} k_\phi)^2 & 2(k_{D\phi})^2 k_\phi - p_{11} \\ 2(k_{D\phi})^2 k_\phi - p_{11} & 2k_{D\phi}^2 - 2k_{D\phi} k_\phi \end{bmatrix}}_{Q_x} \quad (4.103)$$

A relatively simple way to make Q_x positive definite is letting all the non-diagonal entries of Q_x to be equal to zero. With this extra requirement p_{11} can be defined as follows:

$$p_{11} = 2(k_{D\phi})^2 k_\phi \Leftrightarrow Q_x = \begin{bmatrix} 2(k_{D\phi} k_\phi)^2 & 0 \\ 0 & 2k_{D\phi}^2 - 2k_{D\phi} k_\phi \end{bmatrix} \quad (4.104)$$

Finally, all the diagonal elements of Q_x must be positive as well; therefore the following condition must hold:

$$2k_{D\phi}^2 > 2k_{D\phi} k_\phi \Rightarrow k_{D\phi} > k_\phi \quad (4.105)$$

Another requirement is that the P_x matrix be positive definite, this is equivalent to $\det(P_x) > 0$. Based on the previous restrictions the P_x matrix will be given by:

$$P_x = \begin{bmatrix} 2(k_{D\phi})^2 k_\phi & k_{D\phi} k_\phi \\ k_{D\phi} k_\phi & k_{D\phi} \end{bmatrix} \quad (4.106)$$

The determinant of the P_x matrix is:

$$\det(P_x) = \det \begin{bmatrix} 2(k_{D\phi})^2 k_\phi & k_{D\phi} k_\phi \\ k_{D\phi} k_\phi & k_{D\phi} \end{bmatrix} = 2(k_{D\phi})^3 k_\phi - (k_{D\phi} k_\phi)^2 > 0 \Rightarrow k_{D\phi} > \frac{k_\phi}{2} \quad (4.107)$$

Condition (4.107) is weaker than condition (4.105), so condition (4.105) alone is sufficient to satisfy the K - Y conditions. The next step is to form a quadratic Lyapunov candidate function that uses the resultant P_x matrix that satisfies the K - Y conditions.

$$V(\mathbf{e}_x) = \frac{1}{2} \mathbf{e}_x^T P_x \mathbf{e}_x = \frac{1}{2} \begin{bmatrix} e_\phi & e_{D\phi} \end{bmatrix}^T \begin{bmatrix} 2(k_{D\phi})^2 k_\phi & k_{D\phi} k_\phi \\ k_{D\phi} k_\phi & k_{D\phi} \end{bmatrix} \begin{bmatrix} e_\phi \\ e_{D\phi} \end{bmatrix} \quad (4.108)$$

$$V(\mathbf{e}_x) = k_\phi k_{D\phi}^2 e_\phi^2 + k_{D\phi} k_\phi e_\phi e_{D\phi} + \frac{k_{D\phi} e_{D\phi}^2}{2}$$

To find a stability argument, it is necessary to find the derivative along trajectories of the Lyapunov candidate function:

$$\begin{aligned} \dot{V}(\mathbf{e}_x, t) &= \frac{1}{2} \dot{\mathbf{e}}_x^T P_x \mathbf{e}_x + \frac{1}{2} \mathbf{e}_x^T P_x \dot{\mathbf{e}}_x \\ &= \frac{1}{2} \{ \bar{A}_x \mathbf{e}_x + \bar{B}_x [-\Psi_x(\mathbf{e}_x, t) y_x(t)] + \Delta_x(t) \}^T P_x \mathbf{e}_x + \dots \\ &\quad + \frac{1}{2} \mathbf{e}_x^T P_x \{ \bar{A}_x \mathbf{e}_x + \bar{B}_x [-\Psi_x(\mathbf{e}_x, t) y_x(t)] + \Delta_x(t) \} \end{aligned} \quad (4.109)$$

Cancelling and factorizing terms, the following result is obtained:

$$\dot{V}(\mathbf{e}_x, t) = -\frac{1}{2} \mathbf{e}_x^T Q_x \mathbf{e}_x - y_x^T \Psi_x(t) y_x + \delta_x(t) y_x \quad (4.110)$$

Now it is possible to use the Sylvester's inequality (Balas & Frost, 2014) (Khalil, 1996):

$$-\frac{1}{2} \mathbf{e}_x^T Q_x \mathbf{e}_x \leq -\frac{1}{2} \lambda_{\min}(Q_x) \|\mathbf{e}_x\|^2 \quad (4.111)$$

where $\lambda_{\min}(Q_x)$ is the minimum eigenvalue of Q_x .

Using Sylvester's inequality, the following inequality holds:

$$\dot{V}(\mathbf{e}_x, t) \leq -\frac{1}{2} \lambda_{\min}(\mathcal{Q}) \|\mathbf{e}_x\|^2 - y_x^T \boldsymbol{\Psi}_x(t) y_x + \delta_x(t) y_x \quad (4.112)$$

It can be seen that the disturbance term is directly coupled with the output, to overcome this difficulty, the non-linear time varying feedback element $\boldsymbol{\Psi}_x(\mathbf{e}_x, t)$ needs to be conveniently designed to be lower bounded as well. To do so, the first step is to incorporate a modification in the original non-linear adaptive function presented in Eq.(4.80) as follows:

$$\boldsymbol{\Psi}_x(y_x) = y_x \eta_x f(\Delta u_x(t)) = y_x \eta_x \left[1 - \frac{\mu_x}{e^{\gamma_x [\Delta u_x(t)]^2} + e^{-\gamma_x [\Delta u_x(t)]^2}} \right]$$

where the factor μ_x will be bounded by $0 < \mu_x < 2$. This variable modulates the bias in the adaptive gain function $f(\Delta u_x(t))$ as seen in Figure 4.9. Both are related as follows:

$$\mu_x = -2\varepsilon_x + 1 \quad (4.113)$$

The lower bound of the system adaptation gain yields:

$$\tilde{\gamma}_x = \eta_x \varepsilon_x \leq \eta_x f(\Delta u_x) = \boldsymbol{\Psi}_x(\mathbf{e}_x, t) \quad (4.114)$$

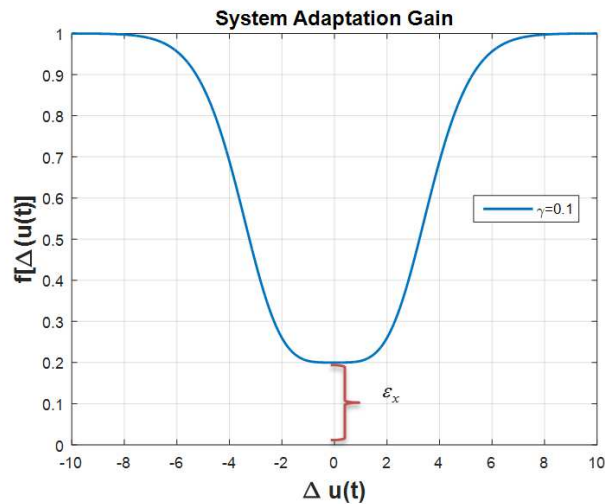


Figure 4.9 System Adaptive Function.

Using the lower bound, the previous result yields the following Lyapunov function derivative inequality:

$$\dot{V}(\mathbf{e}_x, t) \leq -\frac{1}{2} \lambda_{\min}(Q_x) \|\mathbf{e}_x\|^2 - y_x^2 \tilde{\gamma}_x + \delta_x(t) y_x \quad (4.115)$$

Completing the square in Eq.(4.115) yields the following result:

$$\begin{aligned} \dot{V}(\mathbf{e}_x, t) &\leq -\frac{1}{2} \lambda_{\min}(Q_x) \|\mathbf{e}_x\|^2 - \tilde{\gamma}_x \left[y_x - \frac{\delta_x(t)}{2\tilde{\gamma}_x} \right]^2 + \frac{\delta_x^2(t)}{4\tilde{\gamma}_x} \\ &\leq -\frac{1}{2} \lambda_{\min}(Q_x) \|\mathbf{e}_x\|^2 + \frac{\delta_x^2(t)}{4\tilde{\gamma}_x} \end{aligned} \quad (4.116)$$

From the quadratic Lyapunov function candidate definition, the following is also true (Balas & Frost, 2014).

$$\frac{1}{2} \lambda_{\min}(P_x) \|\mathbf{e}_x\|^2 \leq V(\mathbf{e}_x) \leq \frac{1}{2} \lambda_{\max}(P_x) \|\mathbf{e}_x\|^2 \Rightarrow -\|\mathbf{e}_x\|^2 \leq -\frac{2V(\mathbf{e}_x, t)}{\lambda_{\max}(P_x)} \quad (4.117)$$

where $\lambda_{\max}(P_x)$ and $\lambda_{\min}(P_x)$ are the maximum and minimum eigenvalues of the P_x matrix. The following inequalities will hold:

$$\begin{aligned} \dot{V}(\mathbf{e}_x, t) &\leq -\frac{\lambda_{\min}(Q_x)}{\lambda_{\max}(P_x)} V(\mathbf{e}_x, t) + \frac{\delta_x^2(t)}{4\tilde{\gamma}_x} \\ &\leq -\tilde{K}_x V(\mathbf{e}_x) + \frac{\delta_x^2(t)}{4\tilde{\gamma}_x} \end{aligned} \quad (4.118)$$

where $\tilde{K}_x = \frac{\lambda_{\min}(Q_x)}{\lambda_{\max}(P_x)}$.

The previous differential inequality can be solved by multiplying the left and right hand side by the integrating factor $e^{\tilde{K}_x t}$:

$$e^{\tilde{K}_x t} \left[\dot{V}(\mathbf{e}_x, t) + \tilde{K}_x V(\mathbf{e}_x) \right] \leq e^{\tilde{K}_x t} \frac{\delta_x^2(t)}{4\tilde{\gamma}_x} \quad (4.119)$$

Solving the differential equation yields:

$$V(t) \leq V(0)e^{-\tilde{K}_x t} + e^{-\tilde{K}_x t} \int_0^t e^{\tilde{K}_x \tau} \frac{\delta_{x(t)}^2}{4\tilde{\gamma}_x} d\tau \quad (4.120)$$

The disturbance term $\delta_{x(t)}^2$ can be bounded above by:

$$\sup[\delta_{x(t)}^2] \geq \delta_{x(t)}^2 \quad (4.121)$$

This term can be factored from the integral, and the inequality in Eq.(4.117) can be used again to obtain:

$$\frac{1}{2} \lambda_{\min}(P_x) \|\mathbf{e}_x\|^2 \leq V(t) \leq V(0)e^{-\tilde{K}_x t} + \frac{\sup[\delta_{x(t)}^2]}{4\tilde{\gamma}_x} e^{-\tilde{K}_x t} \int_0^t e^{\tilde{K}_x \tau} d\tau \quad (4.122)$$

Moreover, the following holds for the initial condition of the Lyapunov candidate function:

$$V(0) \leq \frac{1}{2} \lambda_{\max}(P_x) \|\mathbf{e}_x(0)\|^2 \quad (4.123)$$

The expression for the tracking error norm will be:

$$\begin{aligned} \|\mathbf{e}_x\|^2 &\leq \frac{2}{\lambda_{\min}(P_x)} V(0)e^{-\tilde{K}_x t} + \frac{\sup[\delta_{x(t)}^2]}{2\tilde{\gamma}_x \lambda_{\min}(P_x)} e^{-\tilde{K}_x t} [e^{\tilde{K}_x t} - 1] \\ &\leq \frac{\lambda_{\max}(P_x)}{\lambda_{\min}(P_x)} \|\mathbf{e}_x(0)\|^2 e^{-\tilde{K}_x t} + \frac{\sup[\delta_{x(t)}^2]}{2\tilde{\gamma}_x \lambda_{\min}(P_x)} [1 - e^{-\tilde{K}_x t}] \end{aligned} \quad (4.124)$$

The tracking error state is bounded by:

$$\|\mathbf{e}_x\| \leq \left\{ \frac{\lambda_{\max}(P_x)}{\lambda_{\min}(P_x)} \|\mathbf{e}_x(0)\|^2 e^{-\tilde{K}_x t} + \frac{\sup[\delta_{x(t)}^2]}{2\tilde{\gamma}_x \lambda_{\min}(P_x)} [1 - e^{-\tilde{K}_x t}] \right\}^{1/2} \quad (4.125)$$

Using the triangle inequality $\sqrt{\|a\|^2 + \|b\|^2} \leq \|a\| + \|b\|$, the following result is obtained:

$$\|\mathbf{e}_x(t)\| \leq \sqrt{\frac{\lambda_{\max}(P_x)}{\lambda_{\min}(P_x)}} \|\mathbf{e}_x(0)\| e^{-\tilde{K}_x t/2} + \sqrt{\frac{\sup[\delta_{x(t)}^2]}{2\tilde{\gamma}_x \lambda_{\min}(P_x)}} [1 - e^{-\tilde{K}_x t}]^{1/2} \quad (4.126)$$

Evaluating the $\limsup_{t \rightarrow \infty} \{\|\mathbf{e}_x(t)\|\}$ and the $\lim_{t \rightarrow \infty}$ on each side of Eq.(4.126) the following region of convergence is obtained for the tracking error trajectories of the system with adaptive augmentation:

$$\|\mathbf{e}_x\| \leq \sqrt{\frac{\sup[\delta_{x(t)}^2]}{2\tilde{\gamma}_x \lambda_{\min}(P_x)}} \quad (4.127)$$

As mentioned previously this result will hold for each attitude channel separately, therefore the following also holds:

$$\|\mathbf{e}_y\| \leq \sqrt{\frac{\sup[\delta_{y(t)}^2]}{2\tilde{\gamma}_y \lambda_{\min}(P_y)}}, \quad \|\mathbf{e}_z\| \leq \sqrt{\frac{\sup[\delta_{z(t)}^2]}{2\tilde{\gamma}_z \lambda_{\min}(P_z)}} \quad (4.128)$$

It is worth empathizing that these norms represent a region of attraction for which global exponential tracking is achieved in the presence of bounded disturbances. It means that the tracking error will exponentially converge to the circle or radius given by the Equations (4.127)- (4.128). The radius of convergence can be numerically specified after the baseline controller gains are obtained considering the constraints presented in Eq.(4.105) and as long the disturbance maximum value is known. In the case in which there is no disturbance, this region will collapse to the origin and thus global asymptotic tracking is achieved. This result was also corroborated by means of the Circle Criterion analysis presented in Section 4.6.1.

4.7. Model Reference AIS Simulation on a LTI Plant

The adaptive control configuration developed in Section 4.6 consists of a Model Reference Artificial Immune System (MRAIS). Before implementing it on a six degree of freedom (6-DOF) simulation environment, is worth assessing the robustness of the controller applied on a simple double integrator plant (which is already linear). Let's consider the following double integrator plant dynamics:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t) \quad (4.129)$$

In open loop this plant is neutrally stable since both of its eigenvalues are located at the origin of the real versus imaginary axis. Let's suppose it is desired to implement a baseline controller that has the following control law:

$$u_b(t) = k_2[k_1(x_{1ref} - x_1) - x_2] = k_2k_1(x_{1ref} - x_1) - k_2x_2 \quad (4.130)$$

This is in fact a cascade controller, very similar to the ones presented in Sections 4.5-4.6. Figure 4.10 illustrates the baseline closed loop control architecture:

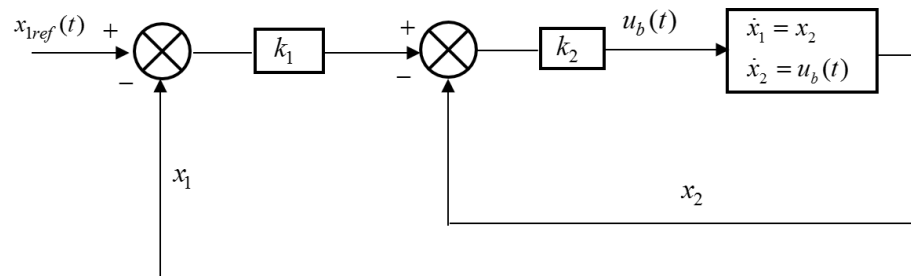


Figure 4.10 Baseline Closed Loop Control Architecture

Then the baseline controller is augmented with the adaptive model reference structure presented in Section 4.6. The resultant control architecture takes the form:

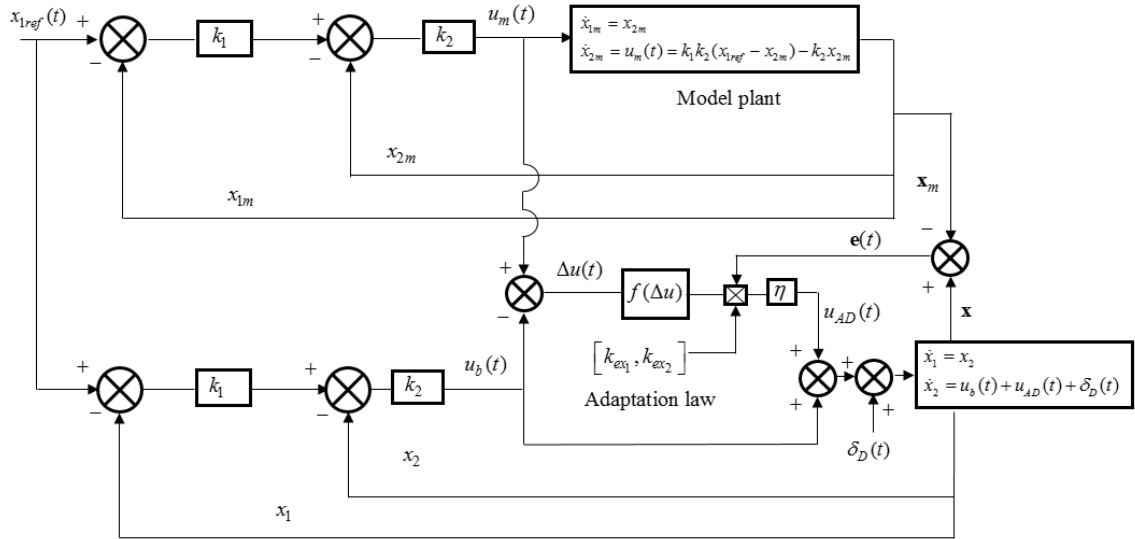


Figure 4.11 Baseline Controller Augmented with MRAIS.

Table 4.1 illustrates some of the controller parameters.

Table 4.1 Controller Parameters

Damping Ratio	$\xi = 0.779$
Natural Frequency (rad/s)	$\omega_n = 5.130$
Settling Time (s)	$T_s = 1.0$
Outer Loop Gain	$k_1 = 3.29$
Inner Loop Gain	$k_2 = 8.0$
Adaptive Power Gain	$\eta = 5$
Adaptive Function Bias	$\varepsilon = 0.1$

4.7.1. Simulation Results for Step Tracking

Figure 4.12 and Figure 4.13 show the tracking results for a step input with a persistent sine disturbance with the following characteristics:

$$\delta_D(t) = 2.1\sin(\pi t) \tag{4.131}$$

The following figures show the tracking results for both the baseline and the adaptive controller.

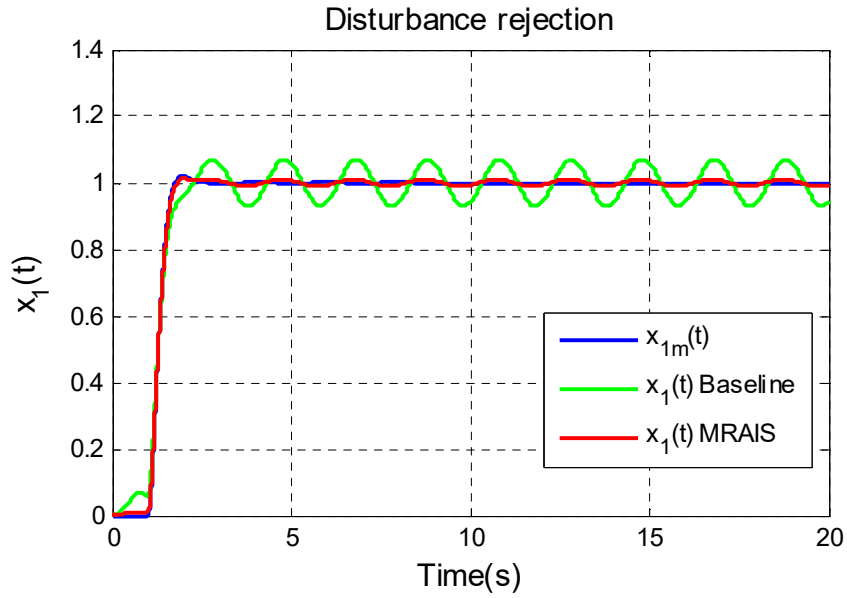


Figure 4.12 Disturbance Rejection for $x_1(t)$.

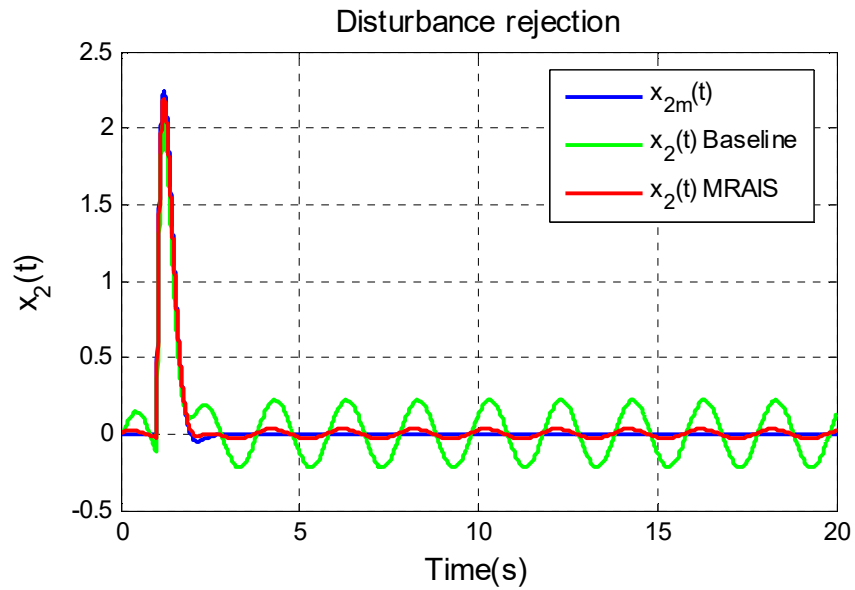


Figure 4.13 Disturbance Rejection for $x_2(t)$

It can clearly be seen that the disturbance is better rejected in terms of amplitude of oscillations on both states when the adaptation is engaged.

4.7.2. Simulation Results for Cycloid Tracking

Figure 4.14 and Figure 4.15 show the tracking results for a cycloid input with the same persistent disturbance specified in Eq.(4.131).

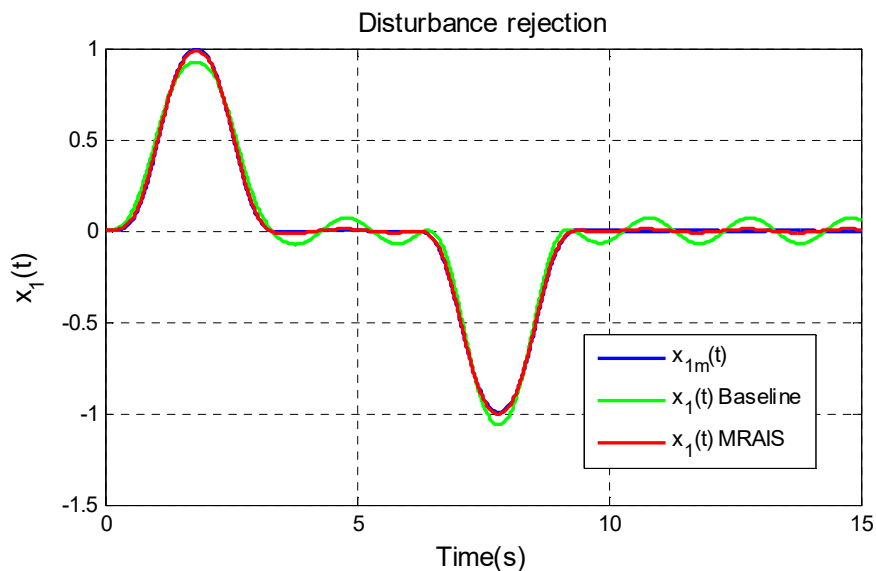


Figure 4.14 Disturbance Rejection for $x_1(t)$.

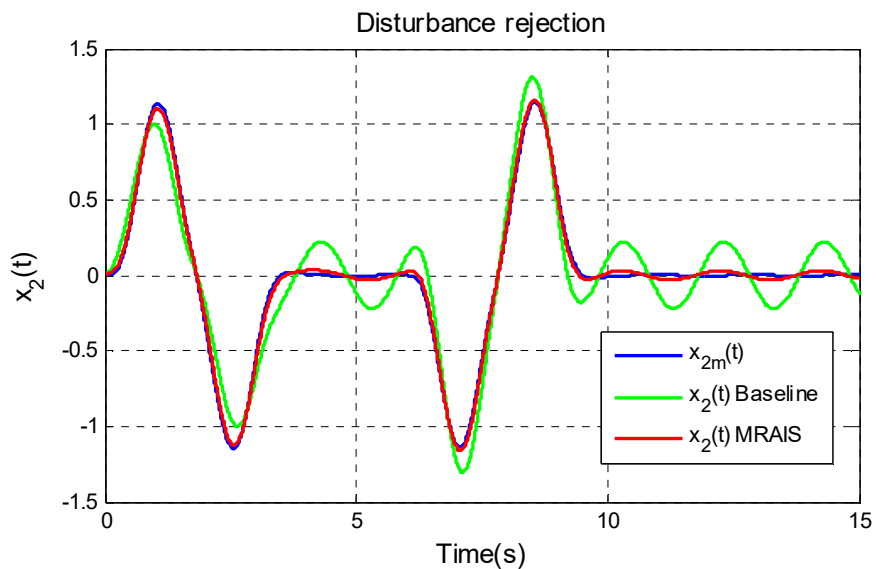


Figure 4.15 Disturbance Rejection for $x_2(t)$.

The disturbance is better rejected in terms of amplitude of oscillations on both states when the adaptation is engaged while tracking a cycloid input.

4.7.3. Estimate of Radius of Convergence

Using the values in table 4.1 and Eq. (4.127) it is possible to estimate the radius of convergence of global tracking for the specific disturbance in this example. The theory guarantees that the tracking error will globally exponentially converge and keep within the radius of attraction presented in Section 4.6.2. The first step is to calculate the P matrix and its minimum eigenvalue using Eq.(4.106):

$$P = \begin{bmatrix} 421.09 & 26.31 \\ 26.31 & 8.0 \end{bmatrix} \rightarrow \lambda_{\min}(P) = 6.33$$

Then the radius of convergence can be estimated calculated using the following equation:

$$\|\mathbf{e}(t)\| = \sqrt{e_1^2(t) + e_2^2(t)} \leq \sqrt{\frac{\sup[\delta_{(t)}^2]}{2\eta\varepsilon\lambda_{\min}(P)}} = \sqrt{\frac{2.1}{2(5)(0.1)(6.33)}} = 0.5760 \quad (4.131)$$

The tracking error Euclidean norm will converge to a ball of radius of 0.576 after reaching the steady state condition.

This chapter presented two major adaptive configurations intended for aerospace systems that do not require large attitude maneuvers. Important theoretical results were obtained for the proposed architectures. A proof of Lagrange stability is presented for the angular rate control adaptive configuration presented in Section 4.2 and a proof of absolute stability and robustness to bounded uncertainties is presented for the novel MRAIS approach (the main results are presented in Section 4.6). Preliminary simulation results on a double integrator plant show the MRAIS adaptive augmentation provides enhanced robustness and stability in the presence of bounded time-varying uncertainties.

5. Application to Aircraft Control

This chapter presents the implementation results of the angular rate NLDI control augmented with AIS adaptation described in Section 4.2 on a Supersonic Fighter Aircraft model developed for research purposes at West Virginia University (WVU) as part of a collaborative effort with ERAU under a research project sponsored by DARPA (Perez A. E., et al., 2014). The capabilities of the proposed adaptive augmentation were examined addressing different types of upset conditions that include control surfaces and structural failures. The proposed adaptive approach was compared with respect to a baseline configuration and the baseline configuration augmented with an Artificial Neural Network (ANN). The control configurations were tested on a motion based simulation environment with a real pilot in the loop.

5.1. WVU Fighter Aircraft Simulation Framework

This part of the dissertation was focused on developing and implementing adaptive control laws for a Fighter aircraft model developed at WVU. This model originated from a high performance military aircraft simulation distributed by NASA to academic institutions in 1990 within a student design competition (Antoniewicz, Duke, & Patterson, 1988). This generic model was entirely developed in MATLAB and Simulink and was further customized through the addition of the aerodynamic canard surfaces (Perhinschi, Napolitano, Campa, & Fravolini, 2003). The aerodynamic and thrust characteristics were incorporated through 42 look-up tables that model the individual contribution of each control surface, aerodynamic surfaces and engines. In this manner is possible to simulate structural damage, control surface failure, and engine malfunctions. Figure 5.1 presents the main broad view of the Simulink model, which is composed of

several different subsystems; as control laws, data recording, aircraft dynamics, aerodynamics and sensor characteristics (Perez A. E., Moncayo, Perhinschi, Al Azzawi, & Togayev, 2015). The Simulink model is used as a benchmark either for desktop simulation or motion based simulation purposes.

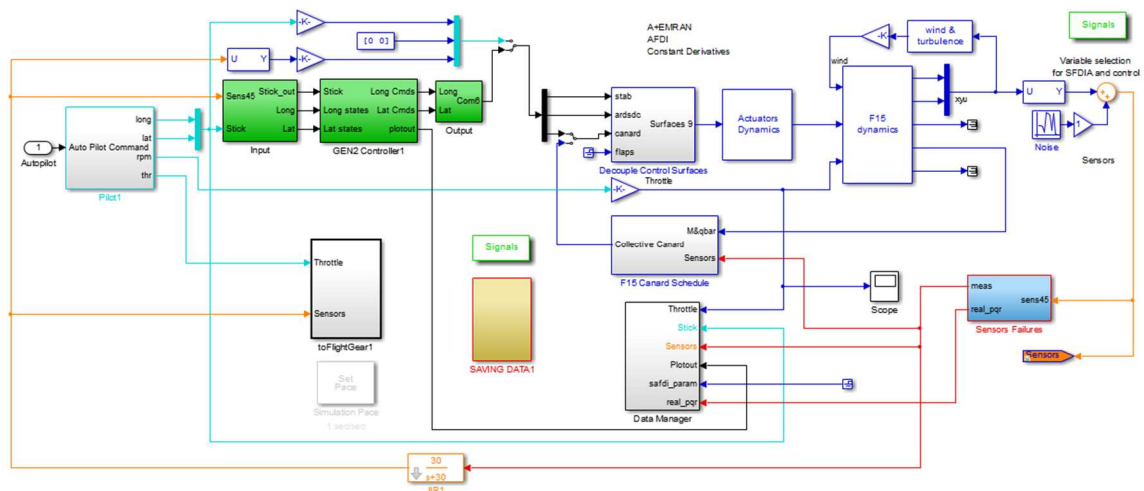


Figure 5.1 Simulink Model of WVU Fighter Aircraft (Perez A. E., Moncayo, Perhinschi, Al Azzawi, & Togayev, 2015).

The experimental data acquisition and implementation of the different control laws was performed on a 6 DOF motion based simulator part of the WVU simulation environment. The flight simulator consists of the following components: A motion platform driven by electrical motors, external visual displays, instructor station, research X-plane flight simulation software, and a server computer (Perhinschi, Napolitano, Campa, & Fravolini, 2003) (see Figure 5.2). The WVU Flight Simulator has been interfaced with an external computer that runs within MATLAB/Simulink environment to drive the motion of the platform (Perez A. E., Moncayo, Perhinschi, Al Azzawi, & Togayev, 2015). The entire system mechanism can be described as follows: First the pilot input signals are transmitted from the cockpit into MATLAB/Simulink model, at the same time the MATLAB/Simulink model is connected to X-Plane software (Meyer &

Van Kampen, 2002), the outputs of the model are transferred to X-Plane to control all the simulator subsystems to generate the visual cues for the pilot.



Figure 5.2 WVU Motion Based Simulator Interfaced with MATLAB/SIMULINK (Perez A. E., Moncayo, Perhinschi, Al Azzawi, & Togayev, 2015).

In order to assess the overall behavior and handling qualities of the control architectures under investigation, the following abnormal conditions were considered within this research effort.

5.1.1. Actuator Failure

Within this effort, failure on left or right individual stabilator, aileron, or rudder have been considered. This type of failure corresponds to locked control surface due to a mechanic failure. The control surface remains fixed in the current position/deflection or moves to a pre-defined position and remains fixed there. It is assumed that a failure involving a blockage of the control surface at a fixed deflection does not alter the aerodynamic properties of the control surface. However, each surface in a pair (left and right) will have different deflections and the resulting moments and forces are computed individually (Perez A. E., Moncayo, Perhinschi, Al Azzawi, & Togayev, 2015).

5.1.2. Structural Failure

For the purpose of this dissertation the damage of the wing is modeled separately. Damages to other aerodynamic surfaces may be considered as failures of the respective actuators (loss of aerodynamic “efficiency”). A simple model of wing damage was developed considering both aerodynamic and gravimetric effects. The failure type corresponds to a total or partial physical destruction and/or deformation of the wing and different percent values along the wing can be selected as damage affected area (Perez A. E., Moncayo, Perhinschi, Al Azzawi, & Togayev, 2015).

5.2. Control Architectures

In this dissertation, a comparison between three main control configurations is performed to assess the capabilities of the proposed AIS-based controller applied to the dynamics of the WVU supersonic fighter aircraft. The three control architectures considered are: NLDI baseline controller, NLDI augmented with AIS and NLDI augmented with ANN. These control architectures are explained in further detail in the following sub-sections.

5.2.1. NLDI Baseline Controller

The baseline control architecture consists mainly of the NLDI angular rate control architecture developed in Section 4.1 (see Figure 4.1). This control architecture is inspired by previous research done by (Perhinschi M. G., Napollitano, Campa, & Fravolini, 2004). Additionally to the control architecture developed in Section 4.1, the system includes a model reference that uses pilot stick inputs $[\delta_a \quad \delta_e \quad \delta_r]^T$ to generate

desired angular rate commands. The general structure of this control architecture is presented in Figure 5.3.

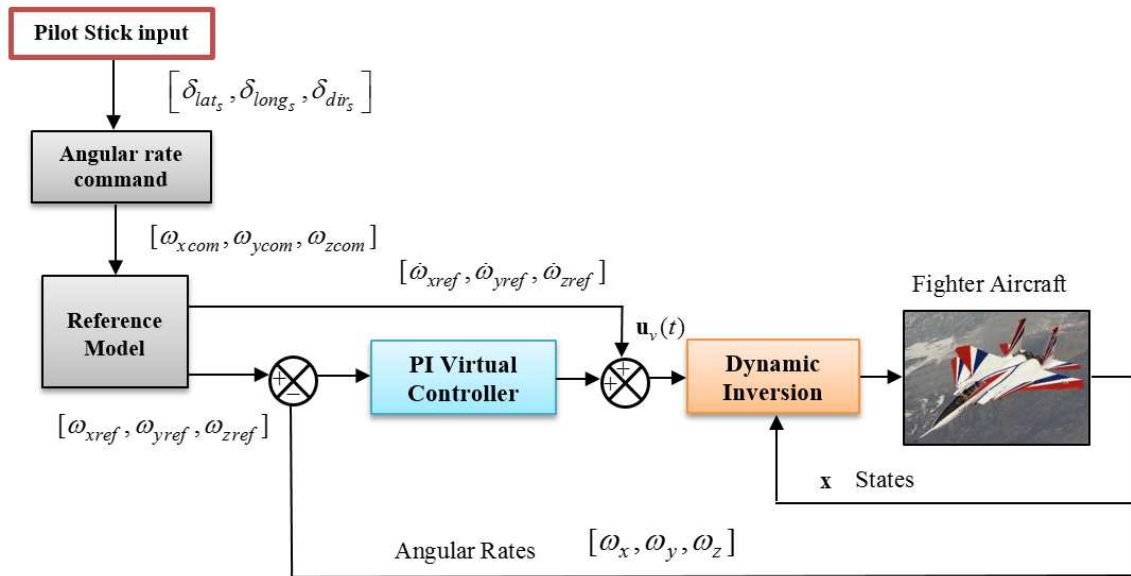


Figure 5.3 Baseline Control Architecture.

The first step is to use the pilot stick inputs and displacements and convert them into angular rate reference commands using Eq.(5.1)-Eq.(5.3). This will ensure a stable transition between stick inputs and commanded angular rates (Perhinschi M. G., Napollitano, Campa, & Fravolini, 2004) (Perez A. E., et al., 2014) (Perhinschi, et al., 2014):

$$\omega_{xcom}(s) = k_{lat} \delta_{lat_{stick}} \quad (5.1)$$

$$\omega_{ycom}(s) = k_{long} \delta_{long_{stick}} \quad (5.2)$$

$$\omega_{zcom}(s) = \frac{g}{V} (k_{dir} \delta_{dir_{pedal}} + \sin \psi) \quad (5.3)$$

After the commanded angular rates are obtained, these are smoothed using first and second order model reference transfer functions. The output of the transfer functions are reference angular rates (Perhinschi, et al., 2014).

$$\omega_{xref}(s) = \frac{1}{1 + \tau_{roll}s} \omega_{xcom}(s) \quad (5.4)$$

$$\omega_{yref}(s) = \frac{\omega_{n\ pitch}^2}{s^2 + 2\zeta_{pitch}\omega_{n\ pitch}s + \omega_{n\ pitch}^2} \omega_{ycom}(s) \quad (5.5)$$

$$\omega_{zref}(s) = \frac{\omega_{n\ yaw}^2}{s^2 + 2\zeta_{yaw}\omega_{n\ yaw}s + \omega_{n\ yaw}^2} \omega_{zcom}(s) \quad (5.6)$$

5.2.2. NLDI Augmented with Artificial Immune System

This control architecture is very similar to the one described in Section 4.2. The only difference is that a model reference is used to generate the angular rate inputs based on pilot commands. The control architecture is depicted in Figure 5.4.

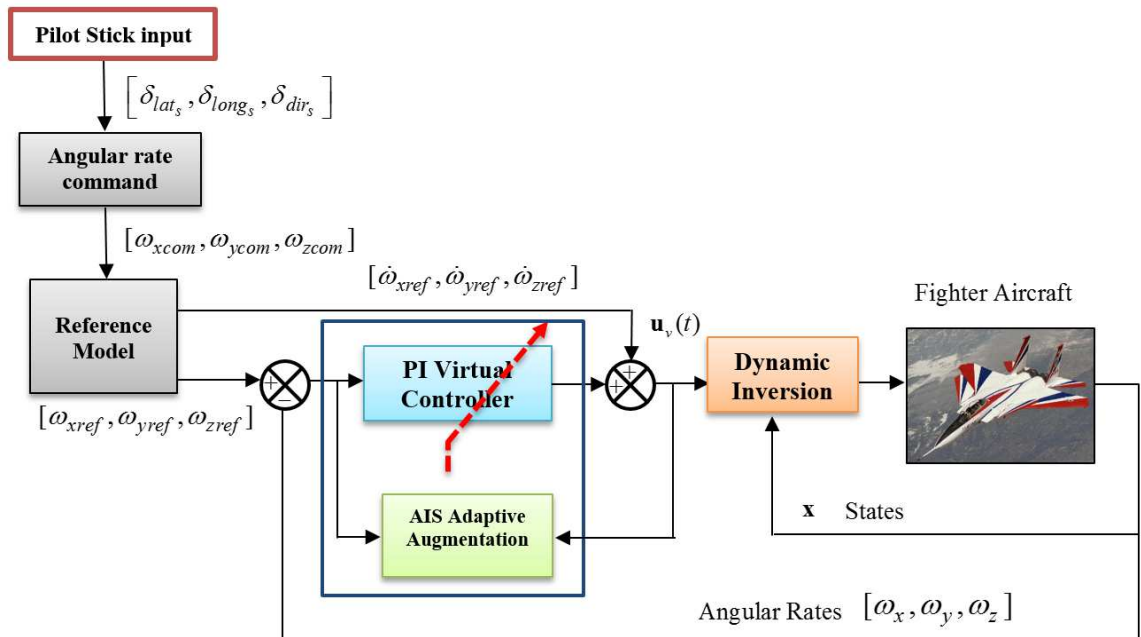


Figure 5.4 NLDI+AIS Angular Rate Control

5.2.3. NLDI Augmented with Artificial Neural Networks

ANN have been used previously (Perhinschi M. G., Napollitano, Campa, & Fravolini, 2004) to augment the baseline controller and improve its adaptability and robustness. The ANN are used in conjunction with the output from the virtual PID compensation, the states of the aircraft, and the angular rates and accelerations references. The general structure of this architecture is shown on Figure 5.5. In this case the contribution of the Neural Networks can be expressed as:

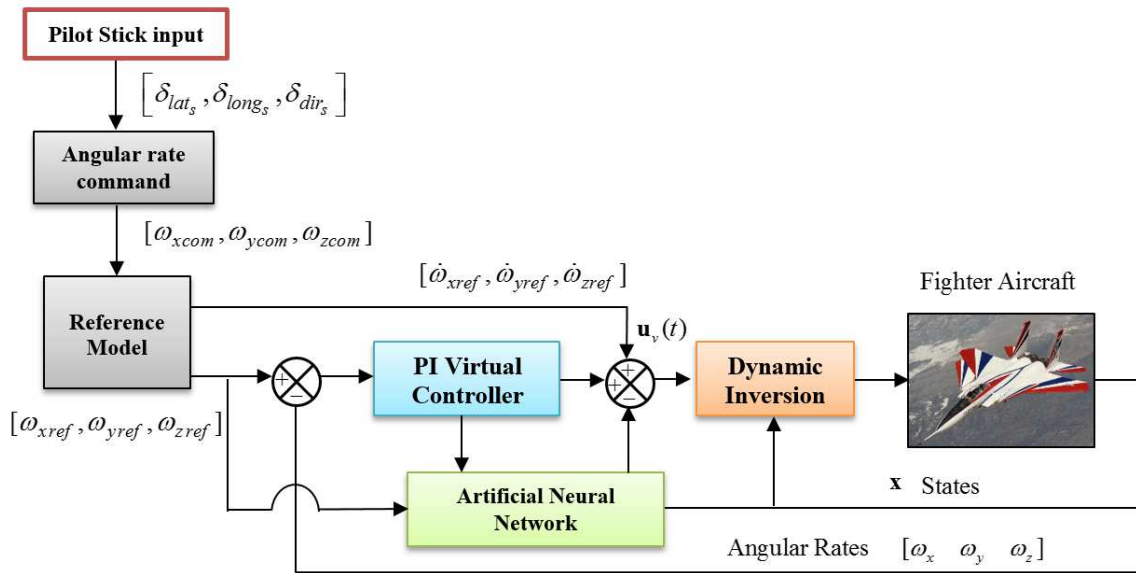


Figure 5.5 NLDI+ANN Angular Rate Control.

$$\begin{bmatrix} \dot{\omega}_{xcom} \\ \dot{\omega}_{ycom} \\ \dot{\omega}_{zcom} \end{bmatrix} = \begin{bmatrix} u_{v\omega_x}(t) \\ u_{v\omega_y}(t) \\ u_{v\omega_z}(t) \end{bmatrix} - \begin{bmatrix} U_{xad} \\ U_{yad} \\ U_{zad} \end{bmatrix} \quad (5.7)$$

where $[U_{xad}, U_{yad}, U_{zad}]^T$ are augmentation commands generated by the ANN in order to compensate for the angular rates tracking errors. These tracking errors are used to provide proportional, integral, and derivative compensation. After including the ANN augmentation, the new virtual controller will yield:

$$\mathbf{u}_{vNN}(t) = \mathbf{K}_{I\omega} \mathbf{e}_1(t) + \mathbf{K}_{p\omega} \mathbf{e}_2(t) + \dot{\mathbf{x}}_{2ref} - \mathbf{v}_{NN} \quad (5.8)$$

The closed loop error dynamics will become:

$$\begin{bmatrix} \dot{\mathbf{e}}_1 \\ \dot{\mathbf{e}}_2 \end{bmatrix} = \begin{bmatrix} [\mathbf{0}]_{3 \times 3} & I_{3 \times 3} \\ -\mathbf{K}_{I\omega} & -\mathbf{K}_{p\omega} \end{bmatrix} \begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \end{bmatrix} + \begin{bmatrix} [\mathbf{0}]_{3 \times 3} \\ [I]_{3 \times 3} \end{bmatrix} \mathbf{v}_{NN} \quad (5.9)$$

The ANN algorithm that was implemented is called the *Extended Minimal Resource Allocation Network* (EMRAN) (Sundararajan, Sartchandran, & Li, 2002). For Gaussian basis functions, the output of the ANN is computed with the expression:

$$\hat{y}(x, \theta) = \sum_{i=1}^M w_i e^{-\left(\frac{|x - \mu_i|^2}{2\sigma_i^2}\right)} \quad (5.10)$$

where x is the input vector, θ is the set of parameters to be tuned by the learning algorithm including the weights w , the Gaussian center positions μ , and the variances σ .

It is worth noticing that the size M of the network is not constant and that the inactive neurons are removed, while new neurons are generated in regions of the state space where the mapping accuracy is poor. New neurons are inserted if the estimation error and the windowed estimation error are large and if the distance from the input to the nearest neuron center is larger than a selected threshold. If one of the three criteria is not met, the tuning parameters are updated using the relationship (Perez A. E., Moncayo, Perhinschi, Al Azzawi, & Togayev, 2015):

$$\theta(k+1) = \theta(k) - \gamma \left. \frac{\partial \hat{y}(k)}{\partial \theta(k)} \right|_{(k)} \cdot e(k) \quad (5.11)$$

where $e(k)$ is the estimation error and γ is the learning rate. The input to the ANN is:

$$x = [V \ H \ \alpha \ \beta \ \omega \ \omega_{ref} \ \bar{U}_\omega]^T \quad (5.12)$$

where V is the aircraft velocity, H is the altitude, α is the angle of attack, and β is the sideslip angle. These four inputs are the same on all three channels. For the longitudinal channel, $\omega = \omega_y$, while for the lateral and directional channels, $\omega = [\omega_x \ \omega_z]$. Only on-channel variables are considered for ω_{ref} and \bar{U}_ω . Finally, \bar{U}_ω is defined as:

$$\bar{U}_\omega = \frac{1 - e^{(\hat{y} - U_\omega)}}{1 + e^{(\hat{y} - U_\omega)}} \quad (5.13)$$

where \hat{y} is the previous output of the ANN.

5.3. Performance Metrics Definition

In order to obtain quantitative measurements of the performance of the different control architectures, some performance characteristics were defined in terms of the total pilot input activity, tracking error of angular rates and the total amount of work used by each of the control surfaces. These performance metrics are also required to tune the parameters of the AIS.

5.3.1. Pilot Activity Metric

One of the most relevant parameters required to assess the overall performance of each control configuration is the total amount of work that the pilot must invest to maintain stable behavior of the aircraft. If the control augmentation is effective enough, then the pilot would need to produce less stick and pedal displacements. Therefore, one direct way to assess the performance of the controller is simply to calculate the accumulated history of the absolute value of the stick and pedal input activity as follows

(Perez A. E., Moncayo, Perhinschi, Al Azzawi, & Togayev, 2015):

$$\mathbf{P}_{act} = \frac{1}{CS_e} \frac{1}{T} \int_0^T |S_e(t)| dt + \frac{1}{CS_a} \frac{1}{T} \int_0^T |S_a(t)| dt + \frac{1}{CS_r} \frac{1}{T} \int_0^T |S_r(t)| dt \quad (5.14)$$

where $S_e(t)$ is the longitudinal stick, $S_a(t)$ is the lateral stick and $S_r(t)$ directional stick time histories, and CS_e , CS_a , CS_r are the corresponding cutoff to normalize each of the pilot activity performance metrics.

5.3.2. Control Surface Activity Metric

Another important factor that determines the quality of the control system is the total amount of work performed by the aircraft control surfaces. The total control surface activity metric was defined as (Perez A. E., Moncayo, Perhinschi, Al Azzawi, & Togayev, 2015):

$$\mathbf{C}_{act} = \frac{1}{C\dot{\delta}_e} \frac{1}{T} \int_0^T |\dot{\delta}_e(t)| dt + \frac{1}{C\dot{\delta}_a} \frac{1}{T} \int_0^T |\dot{\delta}_a(t)| dt + \frac{1}{C\dot{\delta}_r} \frac{1}{T} \int_0^T |\dot{\delta}_r(t)| dt \quad (5.15)$$

where $\dot{\delta}_e(t)$, $\dot{\delta}_a(t)$ and $\dot{\delta}_r(t)$ are the angular velocity of motion time history of each control surface and $C\dot{\delta}_e$, $C\dot{\delta}_a$, $C\dot{\delta}_r$ are corresponding cut off values used to normalize each of the items within the total control surface activity.

5.3.3. Angular Rate Tracking Error Metric

The total angular rate tracking error is a fundamental metric to obtain a direct statistical measurement of the overall inner loop stability augmentation system of the aircraft. The total tracking error performance metric is defined as follows:

$$\mathbf{E}_{act} = \frac{1}{C\omega_x} \sqrt{\frac{1}{T} \int_0^T e_{\omega_x}^2(t) dt} + \frac{1}{C\omega_y} \sqrt{\frac{1}{T} \int_0^T e_{\omega_y}^2(t) dt} + \frac{1}{C\omega_z} \sqrt{\frac{1}{T} \int_0^T e_{\omega_z}^2(t) dt} \quad (5.16)$$

where $e_{\omega_x}(t), e_{\omega_y}(t), e_{\omega_z}(t)$ are the roll, pitch, and yaw angular rates tracking errors, respectively, and $C\omega_x, C\omega_y, C\omega_z$ are their respective cutoff values.

5.4. Motion Based Flight Simulator Results

The purpose of these tests was to assess the performance and resultant handling qualities of the aircraft for three different types of controller architectures: NLDI baseline controller, NLDI + AIS and NLDI + ANN. All controller configurations were tested on five different scenarios: nominal condition, right stabilator failure, left aileron failure, right rudder failure, and left wing structural failure. Figure 5.6 shows the experimental design and the chronological history of the maneuvers carried on by the pilot per each configuration and condition on the WVU simulator. Table 5.1 shows the set of tests that were performed.

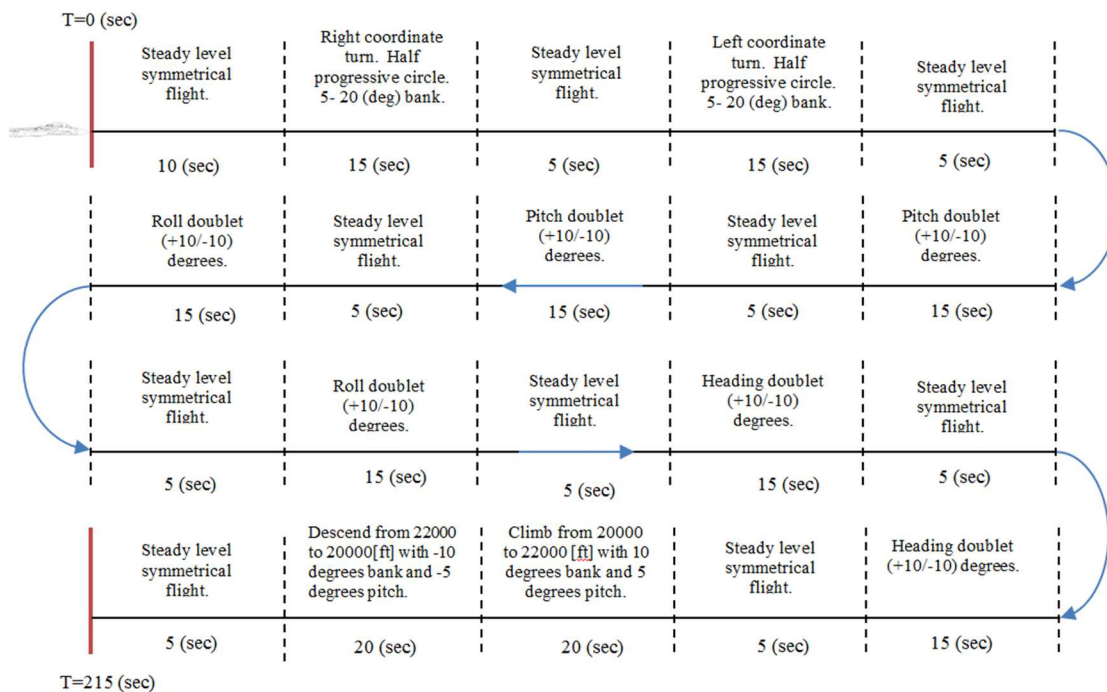


Figure 5.6 Test Outline Performed in Motion Based Simulator (Perez A. E., Moncayo, Perhinschi, Al Azzawi, & Togayev, 2015).

Table 5.1 Performed Tests in WVU Motion Based Simulator.

	NLDI	NLDI + ANN	NLDI + AIS
Nominal	Test 1	Test 6	Test 11
R. Stabilator 8 degrees	Test 2	Test 7	Test 12
L. Aileron 8 degrees	Test 3	Test 8	Test 13
R. Rudder 8 degrees	Test 4	Test 9	Test 14
L. Wing Structural High Magnitude	Test 5	Test 10	Test 15

The results of the implementation are shown in the histogram plots of Figure 5.7 and Figure 5.8 in which a global performance index was calculated from a weighted average of the performance metrics described in Section 5.3 as follows:

$$PI = 1 - [0.7P + 0.05C + 0.25E] \quad (5.17)$$

It is observed that the NLDI+AIS configuration has better global performance than the NLDI+ANN architecture for aileron and stabilator failures.

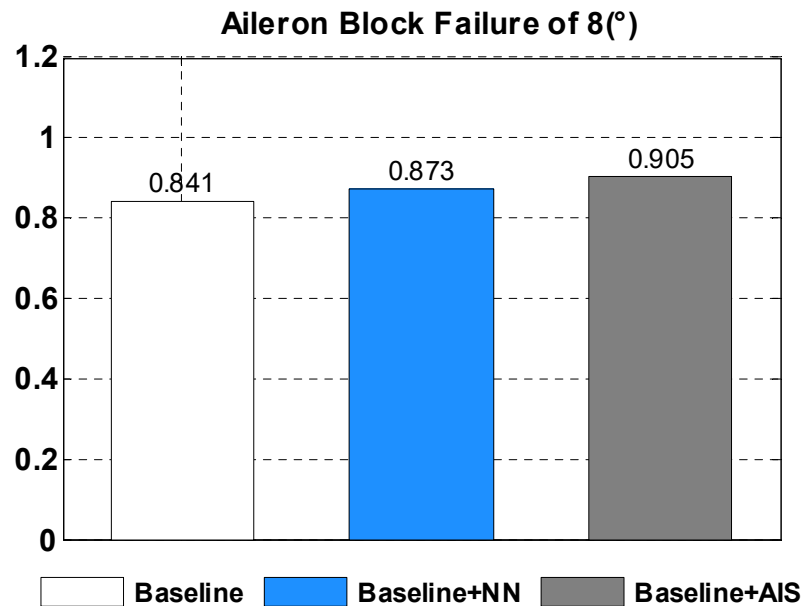


Figure 5.7 Aileron Block Failure.

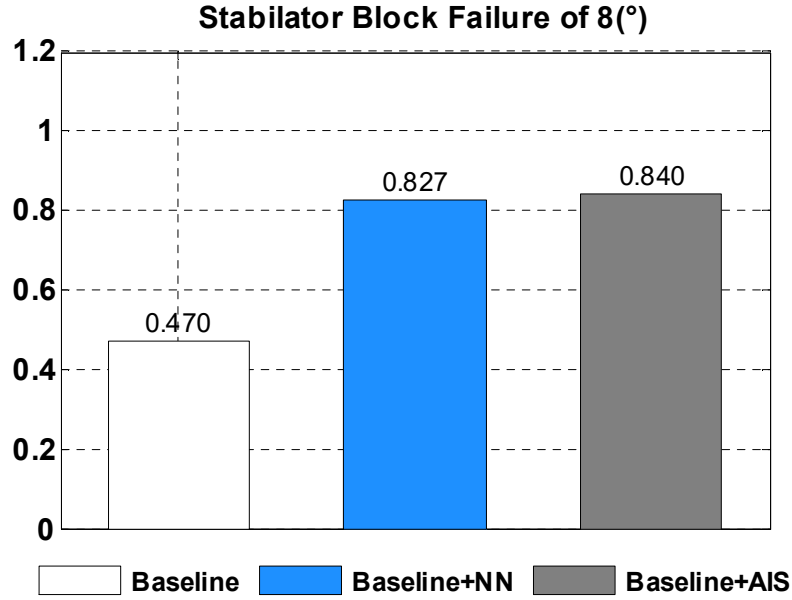


Figure 5.8 Stabilator Block Failure.

A better insight into comparing the different control laws can be achieved by analyzing the individual contribution of the pilot activity, control surface activity, and tracking error activity as an average of all the different tests performed. Un-weighted portions of the global performance index are defined in Eq.(5.18) – Eq.(5.20). These results are presented in the histograms in Figure 5.9 - Figure 5.11.

$$PI_{P_{act}} = 1 - \mathbf{P}_{act} \quad (5.18)$$

$$PI_{C_{act}} = 1 - \mathbf{C}_{act} \quad (5.19)$$

$$PI_{E_{act}} = 1 - \mathbf{E}_{act} \quad (5.20)$$

It can be seen that the AIS has better performance than the other control configurations in terms of angular rate tracking error and total pilot activity; however, there is an increase in the total control surface activity. This result is expected since this adaptive configuration relies on more control surface activity in exchange for robustness.

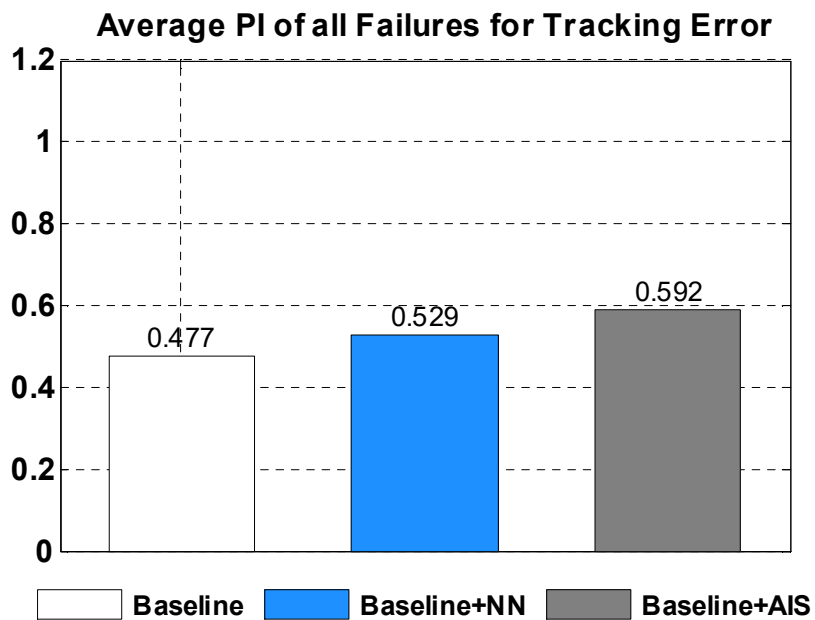


Figure 5.9 Average Angular Rate Tracking Error PI for all Failures.

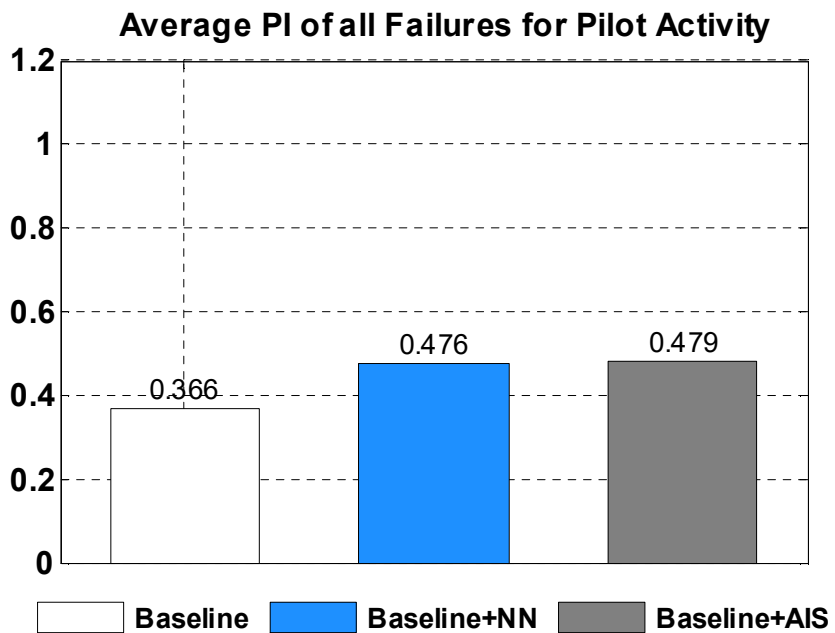


Figure 5.10 Average Pilot Activity PI for all Failures.

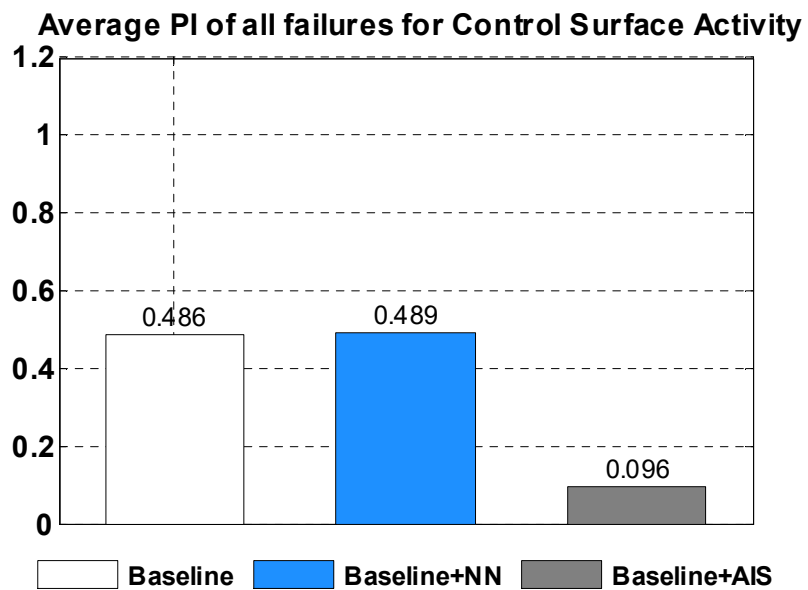


Figure 5.11 Average Control Surface Activity PI for all Failures.

6. Application to Vectoring Thrust Spacecraft Concept Vehicle

Near-Earth Asteroids (NEA) and Near-Earth Objects (NEO) like comets and small interplanetary environments as Martian moons Phobos and Deimos have become of major scientific importance for future space exploration plans due to the potential of extracting consumable resources, such as water, oxygen and Nitrogen (Perez A. , et al., 2016) (Perez, Moncayo, & Prazenica, 2016). As new technologies allow increased capabilities of space vehicles, there has been a renewed interest for exploitation of these resources that would not need to be lifted from the surfaces of the Earth in order to be utilized or studied in Situ (Brophy, et al., 2014). In 2010 the White House recommended that NASA take the lead in conducting research efforts towards the development of technologies that allow NEO detection and characterization (Wie B. , 2015). However, the extreme nature of these environments would require the development of novel advanced unmanned space technologies integrated with sample-capture devices to achieve the ultimate goal of prospecting and studying these resources (Perez A. , et al., 2016).



Figure 6.1 Possible Interior of Martian Lava Tube (Frederick, 1999).

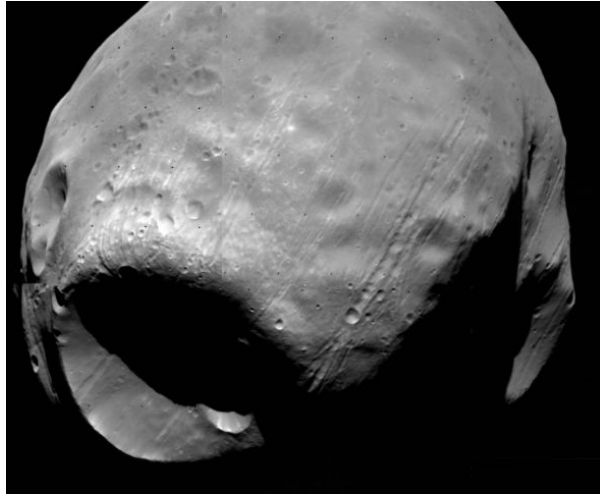


Figure 6.2 Mars Moon Phobos (Bell, 2003).

It is known that some NEO of interest such as Martian lava tubes or craters present in some Asteroids or Moons (see Figures 6.1 and 6.2) are impossible to reach by means of traditional systems such as rovers or other type of vehicles. For such terrains a small prospector free flying vehicle that incorporates vision aided navigation for full autonomous flight could be an ideal solution to investigate and obtain samples in areas that haven't been accessed before.

This chapter presents some of the progress carried out in simulation and implementation of some of the non-linear and adaptive control techniques discussed in Chapter 4 applied into a preliminary autonomous prospector concept prototype designed by NASA. The final goal is to show that the novel controllers developed provide robust attitude and trajectory control for the unmanned platforms so that they handle uncertainties or other system malfunctions while performing autonomous preprogrammed missions in extra-terrestrial environments.

6.1. Concept Vehicle Description

In this part of the dissertation, the main goal is to incorporate guidance, navigation and control for a concept spacecraft vehicle by means of the nonlinear and adaptive controllers developed in Chapter 4. It is desired to implement a guidance system so that the vehicle can go through lava tubes and craters in partial gravity environments without requiring large attitudes angles. Figure 6.3 depicts a concept prototype called Extreme Access Free Flyer (XAFF) built by NASA at Kennedy Space Center. The concept vehicle possesses four thrusters in cross configuration (similar to a quadrotor) to provide roll and pitch motion by means of cold gas thrusters actuated by solenoid valves located on the end of its arms. The main difference between this vehicle with respect to quadrotors is that it incorporates thrust vectoring (TV) by means of a swiveling angle γ ; the swiveling angle allows yaw motion in environments where no aerodynamic forces are present. It is worth recalling that quadrotors require the use of counter propeller rotation and aerodynamic forces to produce yaw. Figure 6.3 shows a preliminary version of the XAFF that uses Electric Ducted Fans (EDFs) instead of thrusters for preliminary testing purposes.

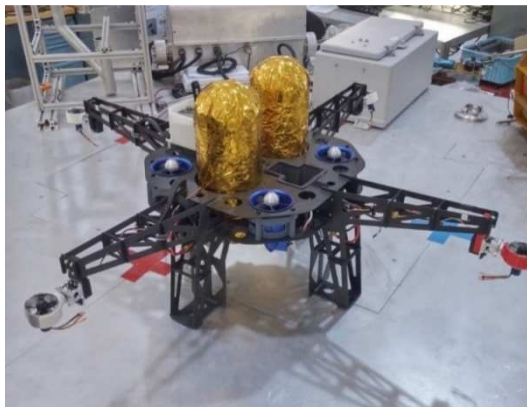


Figure 6.3 Electric Ducted Fan Version of the XAFF (Siceloff, 2015)

Figure 6.4 depicts a realistic CAD version of the XAFF. It can be seen it incorporates two central cold gas tanks in order to operate the solenoid valves and regulate the amount of thrust they can produce.



Figure 6.4 Concept CAD of the XAFF (Perez A. , et al., 2016)

6.2. Forces and Moments that Act on the VT Spacecraft Prototype

The diagram of Figure 6.5 illustrates an isometric and upper view of the forces and moments that act on the body frame of the XAFF spacecraft due to thrusters.

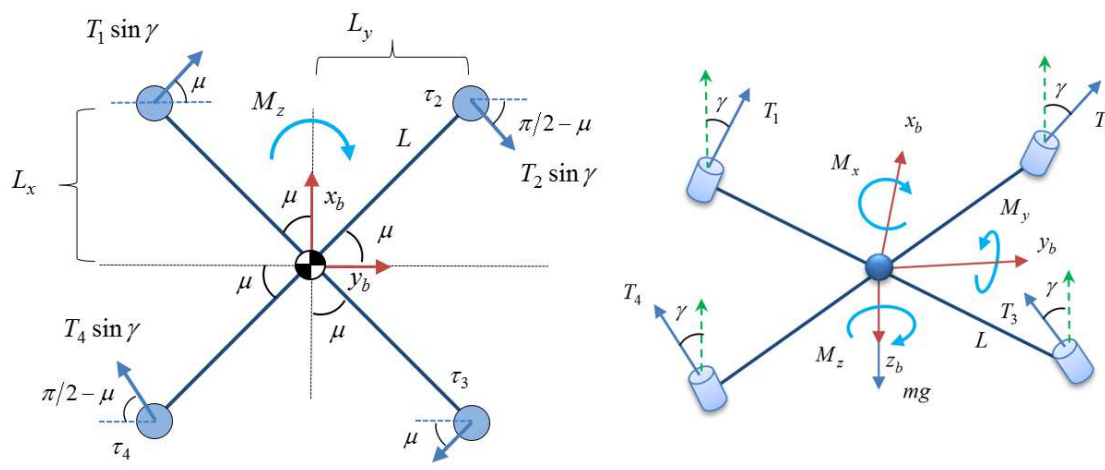


Figure 6.5 Forces and Moments that Act on the XAFF Spacecraft.

Based on Figure 6.5, the following set of forces will be generated by each thruster with respect to the center of gravity on the Body frame of the XAFF:

$$\vec{F}_{1b} = \begin{bmatrix} T_1 \sin \gamma \sin \mu \\ T_1 \sin \gamma \cos \mu \\ -T_1 \cos \gamma \end{bmatrix} \quad (6.1)$$

$$\vec{F}_{2b} = \begin{bmatrix} -T_2 \sin \gamma \sin(\pi/2 - \mu) \\ T_2 \sin \gamma \cos(\pi/2 - \mu) \\ -T_2 \cos \gamma \end{bmatrix} \quad (6.2)$$

$$\vec{F}_{3b} = \begin{bmatrix} -T_3 \sin \gamma \sin \mu \\ -T_3 \sin \gamma \cos \mu \\ -T_3 \cos \gamma \end{bmatrix} \quad (6.3)$$

$$\vec{F}_{4b} = \begin{bmatrix} T_4 \sin \gamma \sin(\pi/2 - \mu) \\ -T_4 \sin \gamma \cos(\pi/2 - \mu) \\ -T_4 \cos \gamma \end{bmatrix} \quad (6.4)$$

The sum of moments produced by the actuator forces with respect to the Center of Gravity can be calculated as follows:

$$\sum \vec{M}_b = \vec{r}_1 \times \vec{F}_{b1} + \vec{r}_2 \times \vec{F}_{b2} + \vec{r}_3 \times \vec{F}_{b3} + \vec{r}_4 \times \vec{F}_{b4} \quad (6.5)$$

where:

$$\vec{r}_1 = \begin{bmatrix} L_{1x} \\ -L_{1y} \\ L_{1z} \end{bmatrix}, \quad \vec{r}_2 = \begin{bmatrix} L_{2x} \\ L_{2y} \\ L_{2z} \end{bmatrix}, \quad \vec{r}_3 = \begin{bmatrix} -L_{3x} \\ L_{3y} \\ L_{3z} \end{bmatrix}, \quad \vec{r}_4 = \begin{bmatrix} -L_{4x} \\ -L_{4y} \\ L_{4z} \end{bmatrix} \quad (6.6)$$

For the specific case of the XAFF, the following applies for the distances from the actuators to the CG: $L_x = L_{1x} = L_{2x} = L_{3x} = L_{4x}$ and $L_y = L_{1y} = L_{2y} = L_{3y} = L_{4y}$. Furthermore, since the distance from the CG to the xy plane where the force of each actuator is applied is very small, then: $L_z = L_{1z} = L_{2z} = L_{3z} = L_{4z} = 0$. The resultant sum of moments in the Body frame will be given by:

$$\sum \vec{M}_b = \begin{bmatrix} L_y \cos \gamma [(T_1 + T_4) - (T_2 + T_3)] \\ L_x \cos \gamma [(T_1 + T_2) - (T_3 + T_4)] \\ L \sin \gamma (T_1 + T_2 + T_3 + T_4) \end{bmatrix} = \begin{bmatrix} M_{xd} \\ M_{yd} \\ M_{zd} \end{bmatrix} \quad (6.7)$$

On the other hand, the total sum of forces in the body frame produces the following result:

$$\sum \vec{F}_b = \begin{cases} \sin \gamma [T_1 \sin \mu - T_2 \cos \mu - T_3 \sin \mu + T_4 \cos \mu] \\ \sin \gamma [T_1 \cos \mu + T_2 \sin \mu - T_3 \cos \mu - T_4 \sin \mu] \\ \cos \gamma [-T_1 - T_2 - T_3 - T_4] \end{cases} = \begin{bmatrix} F_{xd} \\ F_{yd} \\ F_{zd} \end{bmatrix} \quad (6.8)$$

where $[M_{xd} \ M_{yd} \ M_{zd}]^T$ and $[F_{xd} \ F_{yd} \ F_{zd}]^T$ can be viewed as required forces and moments that need to be produced by the nonlinear dynamic inversion approach.

6.3. Development of Control Allocation for the VT Prototype

In order to achieve the required forces and moments commanded from the Euler angle based NLDI controllers developed in Chapter 4 (incremental or exact approach). Equations (6.7) and (6.8) must be solved in order to find the exact thrust forces and swiveling angle inputs $[T_1 \ T_2 \ T_3 \ T_4 \ \gamma]^T$. It can be noticed that there is a total of six equations for five unknowns, therefore the system has one extra redundant equation. In order to overcome this issue the following set of equations was proposed instead.

$$\begin{aligned} M_{xd} &= L_y \cos \gamma [(T_1 + T_4) - (T_2 + T_3)] \\ M_{yd} &= L_x \cos \gamma [(T_1 + T_2) - (T_3 + T_4)] \\ M_{zd} &= L \sin \gamma (T_1 + T_2 + T_3 + T_4) \\ \frac{F_{zd}}{2} &= -T_1 \cos \gamma - T_3 \cos \gamma \\ \frac{F_{zd}}{2} &= -T_4 \cos \gamma - T_2 \cos \gamma \end{aligned} \quad (6.9)$$

The desired F_{zd} vertical force must be divided in two equations: half of the total thrust will be supplied by T_1 and T_3 , while the other half must be supplied by T_2 and T_4 . The simultaneous solution of the system of nonlinear equations presented in Eq.(6.9) yields the following result.

$$\begin{aligned}
T_1 &= \left(\frac{-L_x M_{xd} - L_y M_{yd} + F_{zd} L_x L_y}{4F_{zd} L_x L_y L} \right) \sqrt{(F_{zd}^2 L^2 + M_{zd}^2)} \\
T_2 &= \left(\frac{L_x M_{xd} - L_y M_{yd} + F_{zd} L_x L_y}{4F_{zd} L_x L_y L} \right) \sqrt{(F_{zd}^2 L^2 + M_{zd}^2)} \\
T_3 &= \left(\frac{L_x M_{xd} + L_y M_{yd} + F_{zd} L_x L_y}{4F_{zd} L_x L_y L} \right) \sqrt{(F_{zd}^2 L^2 + M_{zd}^2)} \\
T_4 &= \left(\frac{-L_x M_{xd} + L_y M_{yd} + F_{zd} L_x L_y}{4F_{zd} L_x L_y L} \right) \sqrt{(F_{zd}^2 L^2 + M_{zd}^2)} \\
\gamma &= 2 * \arctan 2 \left(\sqrt{F_{zd}^2 L^2 + M_{zd}^2} + F_{zd} L, M_{zd} \right)
\end{aligned} \tag{6.10}$$

It is worth recalling that due to the convention adopted (NED), F_{zd} (desired force) must be a negative value to produce thrust to overcome gravity. An approximate solution of Eq.(6.9) can be obtained to reduce computational effort by not solving the system of equations in simultaneous manner. The approximation assumes that the swiveling angle γ is small, and that the effect of the yawing moment on the total thrust is negligible.

$$\begin{aligned}
T_1(k) &= \frac{M_{xd}(k)}{4L_y} + \frac{M_{yd}(k)}{4L_x} - \frac{F_{zd}(k)}{4} \\
T_2(k) &= \frac{-M_{xd}(k)}{4L_y} + \frac{M_{yd}(k)}{4L_x} - \frac{F_{zd}(k)}{4} \\
T_3(k) &= \frac{-M_{xd}(k)}{4L_y} - \frac{M_{yd}(k)}{4L_x} - \frac{F_{zd}(k)}{4} \\
T_4(k) &= \frac{M_{xd}(k)}{4L_y} - \frac{M_{yd}(k)}{4L_x} - \frac{F_{zd}(k)}{4} \\
\gamma(k+1) &= \arcsin \left(\frac{M_{zd}(k)}{L[T_1(k) + T_2(k) + T_3(k) + T_4(k)]} \right)
\end{aligned} \tag{6.11}$$

In Eq.(6.11) k denotes one instant of time. The solutions for $T_1 - T_4$ are obtained first and then a value for γ can be calculated for the next time step assuming the change in total thrust in one time step is negligible.

6.4. Guidance Control Architecture

As was mentioned before, it is desired that the XAFF prototype has the capabilities to navigate autonomously through Martian lava tubes or other extra-terrestrial environments. So far Nonlinear adaptive control laws were developed for attitude tracking (see Sections 4.4 - 4.6). Additionally, two different control allocation techniques applicable for the XAFF were discussed in Section 6.2. However, in order to track a predefined path or trajectory for full autonomous navigation a guidance loop that incorporates position and velocity control is required. A cascaded control strategy can be used so that the output of the outer loop are the desired attitude commands for the attitude controller (inner loop). The main view of the full guidance and control architecture designed for the XAFF (with the specific feedback states required per each loop) is shown in Figure 6.6.

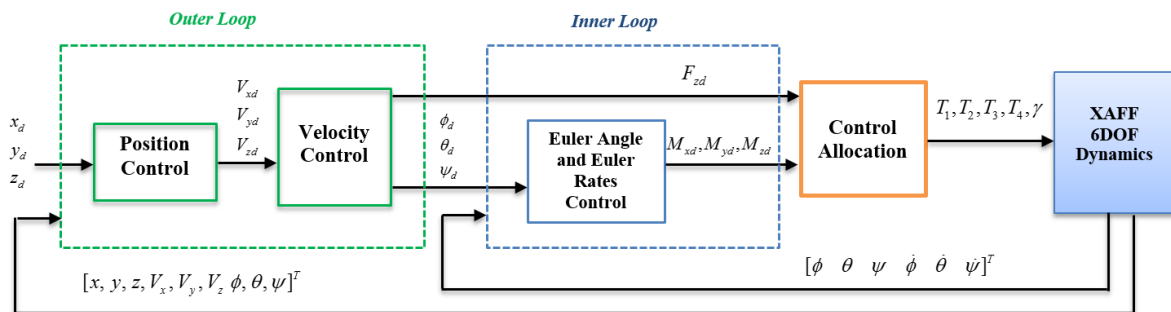


Figure 6.6 XAFF Full Guidance and Control Architecture.

Since the Inner loop and control allocation parts were already discussed, this section will focus on the Outer Loop controller.

Similarly as the Inner Loop, the Outer Loop will rely on a feedback linearization approach. The goal is to cancel out the non-linearity's in the dynamics that govern the

translational equations of the rigid body so that the resultant dynamics are linear and can be conveniently designed using pole placement techniques. To start, it is convenient to recall Newton's second law for the total sum of forces in the z axis:

$$\ddot{z} = g - \frac{F_z \cos \phi \cos \theta}{m} \quad (6.12)$$

The total force F_z in the z upward direction will be provided by the thrusters ($T_1 - T_4$). From section 6.3 any particular required vertical force F_{zd} or moments M_{xd}, M_{yd}, M_{zd} can be achieved using Eq.(6.10) or Eq.(6.11) since the thruster and swiveling angle commands are specifically allocated to generate the required forces and moments coming from the Inner loop. Therefore if the effect or lag of the thruster dynamics is neglected the control allocation will be precise and $F_z = F_{zd}$. With this in mind it is possible to feedback linearize the vertical vehicle dynamics of Eq.(6.12) if the desired force is:

$$F_{zd} = \frac{m[\hat{u}_z(t) - g]}{\cos \phi \cos \theta} \quad (6.13)$$

where $\hat{u}_z(t)$ is a virtual controller that can be conveniently designed to achieve stable vertical closed loop dynamics. The virtual control law was selected as second order system that requires velocity and position in the z direction. After performing the feedback linearization the vertical second order dynamics yield (Ireland, Vargas, & Anderson, 2015).

$$\ddot{z} = \hat{u}_z(t) = k_{Vz} k_{Pz} (z_{ref} - z) - k_{Vz} \dot{z} \quad (6.14)$$

In a similar way, it is possible to feedback linearize the equations of motion that govern the x and y dynamics. For that effect it is more convenient to express the force

equations in the earth reference frame (Ireland, Vargas, & Anderson, 2015).

$$\ddot{x} = \frac{-F_z}{m} (\sin \phi \sin \psi + \cos \phi \sin \theta \cos \psi) \quad (6.15)$$

$$\ddot{y} = \frac{-F_z}{m} (\cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi) \quad (6.16)$$

The inversion of these equations yields the required roll and pitch commands for the inner loop (Ireland, Vargas, & Anderson, 2015).

$$\phi_d = -\arcsin \left\{ \frac{m[\hat{u}_x(t) \sin \psi - \hat{u}_y(t) \cos \psi]}{F_{zd}} \right\} \quad (6.17)$$

$$\theta_d = -\arcsin \left\{ \frac{m[\hat{u}_x(t) \cos \psi + \hat{u}_y(t) \sin \psi]}{F_{zd} \cos \phi} \right\} \quad (6.18)$$

where $\hat{u}_x(t)$ and $\hat{u}_y(t)$ are linear virtual controllers that were designed to produce desirable second order system the closed loop dynamics for the motion in x and y . After performing the feedback linearization the following second order dynamics are obtained:

$$\ddot{x} = \hat{u}_x(t) = k_{V_x} k_{P_x} (x_{ref} - x) - k_{V_x} \dot{x} \quad (6.19)$$

$$\ddot{y} = \hat{u}_y(t) = k_{V_y} k_{P_y} (y_{ref} - y) - k_{V_y} \dot{y} \quad (6.20)$$

The outer loop controller gains can be calculated to obtain specific damping ratio and natural frequency.

$$\begin{cases} k_{V_x} = 2\xi_x \omega_{nx} \\ k_{V_y} = 2\xi_y \omega_{ny} \\ k_{V_z} = 2\xi_z \omega_{nz} \end{cases} \quad (6.21)$$

$$\begin{cases} k_{P_x} = \omega_{nx} / 2\xi_x \\ k_{P_y} = \omega_{ny} / 2\xi_y \\ k_{P_z} = \omega_{nz} / 2\xi_z \end{cases} \quad (6.22)$$

6.5. Definition of Stability and Performance Metrics

In order to evaluate the performance of different control architectures, a set of performance metrics were developed to measure different qualities of interest that include the total activity of the actuators, the ability to follow a predefined path and the ability to maintain appropriate attitude and angular rate tracking control.

Angular Rate/Euler Rate Error Performance Metric:

This performance metric corresponds to the sum of the root mean square of the three angular rates or Euler rates error signals.

$$\tilde{e}_{\Delta\Omega} = \frac{1}{C\Delta\Omega} \left(\sqrt{\int_0^T e_{\phi}^2 dt} + \sqrt{\int_0^T e_{\theta}^2 dt} + \sqrt{\int_0^T e_{\psi}^2 dt} \right) \quad (6.23)$$

where $C\Delta\Omega$ is a cut-off value used to normalize the resultant index $\tilde{e}_{\Delta\Omega}$ with respect to the worst case (biggest value) from the set of tests.

Attitude Error Performance Metric:

This performance metric corresponds to the sum of the *rms* of the error signals of roll, pitch and yaw angles with respect to the desired attitude from the controller.

$$\tilde{e}_{\Theta} = \frac{1}{C\Theta} \left(\sqrt{\int_0^T e_{\phi}^2 dt} + \sqrt{\int_0^T e_{\theta}^2 dt} + \sqrt{\int_0^T e_{\psi}^2 dt} \right) \quad (6.24)$$

where $C\Theta$ is a cut-off value used to normalize the resultant index \tilde{e}_{Θ} respect to the worst case from the set of tests considered.

Velocity Error Performance Metric:

This performance metric corresponds to the sum of the root mean square of the error signals in V_x, V_y, V_z with respect to the desired signals from the controller.

$$\tilde{e}_v = \frac{1}{CV} \left(\sqrt{\frac{1}{T} \int_0^T e_{V_x}^2 dt} + \sqrt{\frac{1}{T} \int_0^T e_{V_y}^2 dt} + \sqrt{\frac{1}{T} \int_0^T e_{V_z}^2 dt} \right) \quad (6.25)$$

where CV is a cut off value used to normalize the resultant index \tilde{e}_v respect to the worst case from the set of tests.

Position Error Performance Metric:

This performance metric corresponds to the sum of the *rms* of the error signals of x, y, z respect to the commanded position from the controller.

$$\tilde{e}_p = \frac{1}{CP} \left(\sqrt{\frac{1}{T} \int_0^T e_x^2 dt} + \sqrt{\frac{1}{T} \int_0^T e_y^2 dt} + \sqrt{\frac{1}{T} \int_0^T e_z^2 dt} \right) \quad (6.26)$$

where CP is a cut off value used to normalize the resultant index \tilde{e}_p with respect to the worst case from the set of tests.

Solenoid Activity Performance Metric:

This performance metric corresponds to the total actuation activity of the solenoid valves. It is calculated as the root mean square of the solenoid PWM commanded signal and is basically the *root mean square* of the total time the solenoid valves remained open during the mission. It is computed using the following expression.

$$\tilde{s} = \frac{1}{C\Delta S} \left(\sum_{i=1}^4 \sqrt{\int_0^T S_i(t) dt} \right) \quad (6.27)$$

where $C\Delta S$ is a cut-off value used to normalize the resultant index \tilde{s} with respect to the worst case from the set of tests.

Global Performance Index:

A global performance index can be computed from the previous set of metrics using different weights for each of the indices in Eq.(6.23) - Eq.(6.27) considering relative importance or weight on the study. For that effect, the following expression is used to calculate a global performance index. In this case, the same weights are assigned to each metric.

$$\begin{aligned} P_T &= 1 - [0.2\tilde{\epsilon}_\Omega + 0.2\tilde{\epsilon}_\theta + 0.2\tilde{\epsilon}_\nu + 0.2\tilde{\epsilon}_p + 0.2\tilde{\epsilon}_s] \\ \tilde{P}_T &= P_T / CP_T \end{aligned} \quad (6.28)$$

where CP_T is a cutoff value to normalize the total performance P_T with respect to the best of all the set of tests considered. Thus, after the tests are performed, the best performance metric will yield a value of 1.0 and the set of performance metrics will have values between [0.0-1.0], where 0.0 is the worst possible value and 1.0 corresponds to the best performance from the set of tests.

6.6. Simulation Results

This section presents an overall overview of the simulation environment developed to test the guidance and control architectures for the autonomous operation of the XAFF. As can be seen in Figure 6.7 the simulation contains a 6 DOF equations of motion module, a sensor module (used to simulate the response and noise of the avionics sensors) and an estimation and control module in which estimation algorithms are implemented to obtain all required states for guidance and control of the system. It also contains a performance metric module that incorporate some of the equations presented in Eq.(6.23) – Eq.(6.27).

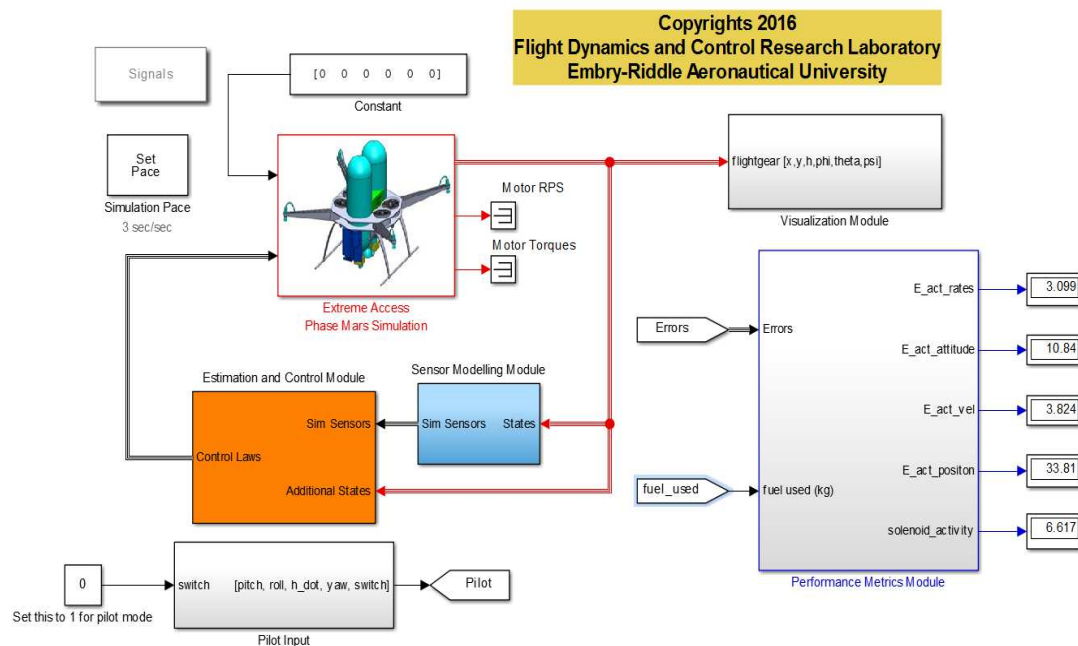


Figure 6.7 XAFF Simulation Environment

Within the Estimation and Control Module there is a switch selector block to conveniently change the control architecture for comparison purposes at the user discretion. A total of three control architectures were tested in two different conditions, a nominal condition that consists of tracking a position trajectory of seven waypoints (see Figure 6.11) and a high magnitude abnormal condition.

Table 6.1 and 6.2 presents the control parameters and gains used within the tests for the baseline controller and for the adaptive augmentation.

Table 6.1 Baseline Control Gains.

Euler Rate Loop		Attitude Loop		Velocities Loop		Position Loop	
$k_{D\phi}$	16	k_{ϕ}	6.58	k_{v_x}	0.4	k_{P_x}	0.16
$k_{D\theta}$	16	k_{θ}	6.58	k_{v_y}	0.4	k_{P_y}	0,16
$k_{D\psi}$	16	k_{ψ}	6.58	k_{v_z}	0.4	k_{P_z}	0.16

Table 6.2 Attitude Adaptive Augmentation Parameters.

Model Reference and Adaptive Control Parameters					
ξ_x	0.78	η_x	5.5	ε_x	0.2
T_{sx} (s)	0.5	η_y	5.5	ε_y	0.2
ξ_y	0.78	η_z	3.0	ε_z	0.2
T_{sy} (s)	0.5				
ξ_z	0.78				
T_{sz} (s)	0.5				

Nominal Conditions Results

Figure 6.8 and Figure 6.9 depict the tracking results of the inner loop controllers tested at nominal conditions.

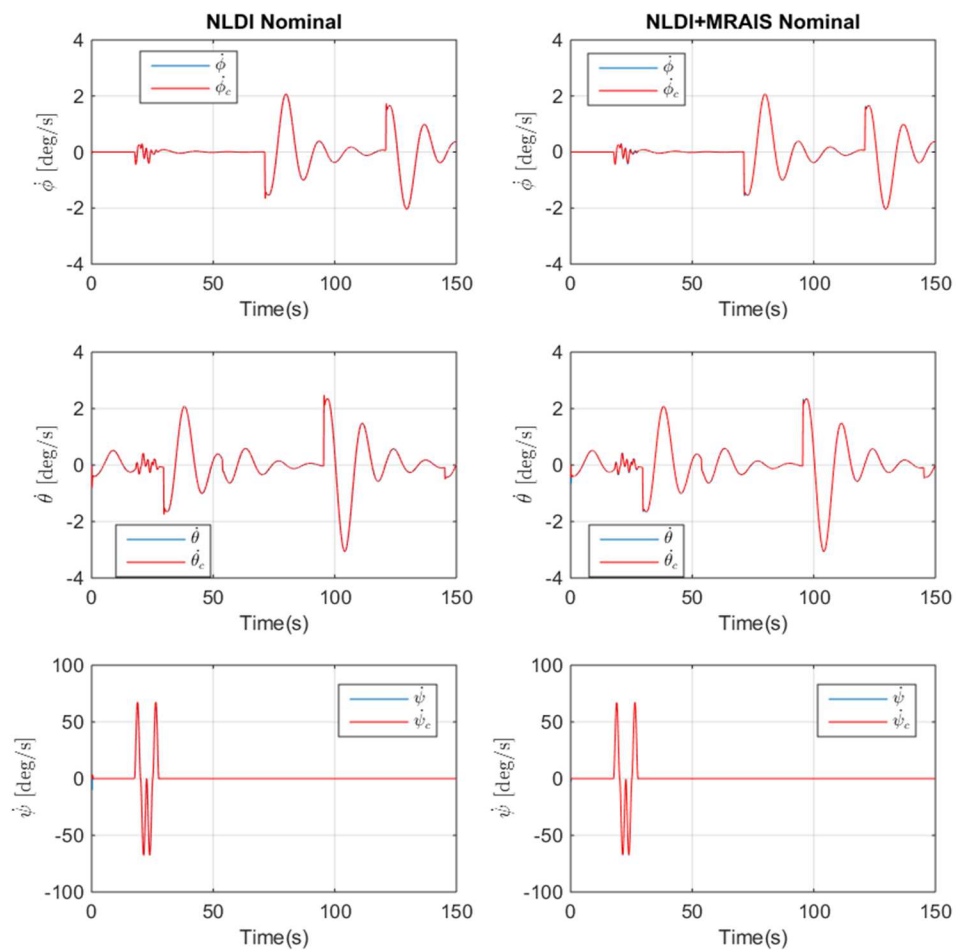


Figure 6.8 Euler Rates Tracking for NLDI and NLDI+MRAIS Nominal Condition.

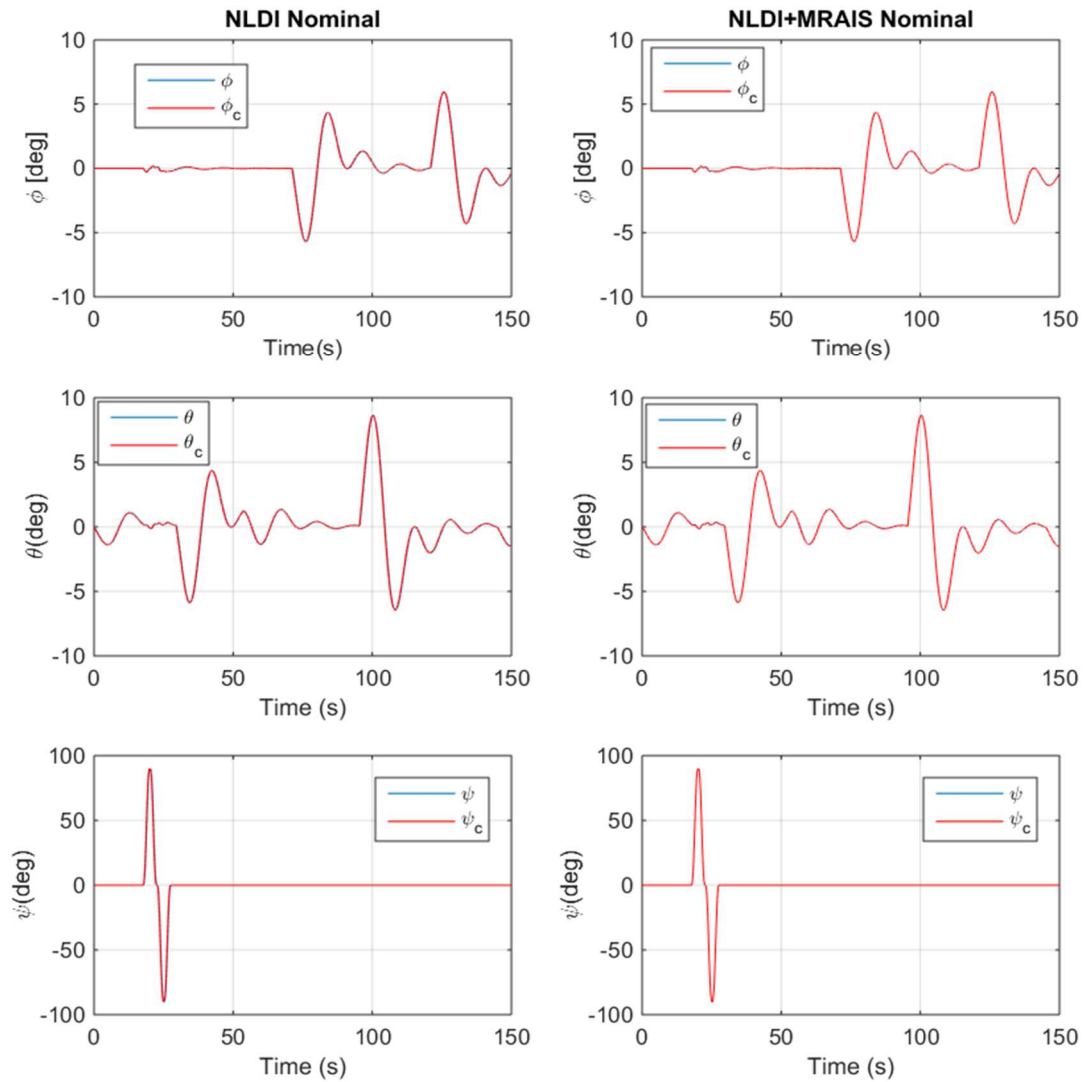


Figure 6.9 Attitude Tracking for NLDI and NLDI+MRAIS Nominal Condition.

Figure 6.8 and Figure 6.9 also depict a direct comparison between the NLDI and NLDI+MRAIS control architectures. It can be seen that for nominal conditions the tracking performance for Euler angles and Euler rates is very similar. Figure 6.10 shows a very similar trend for the tracking performance for velocities (outer loop).

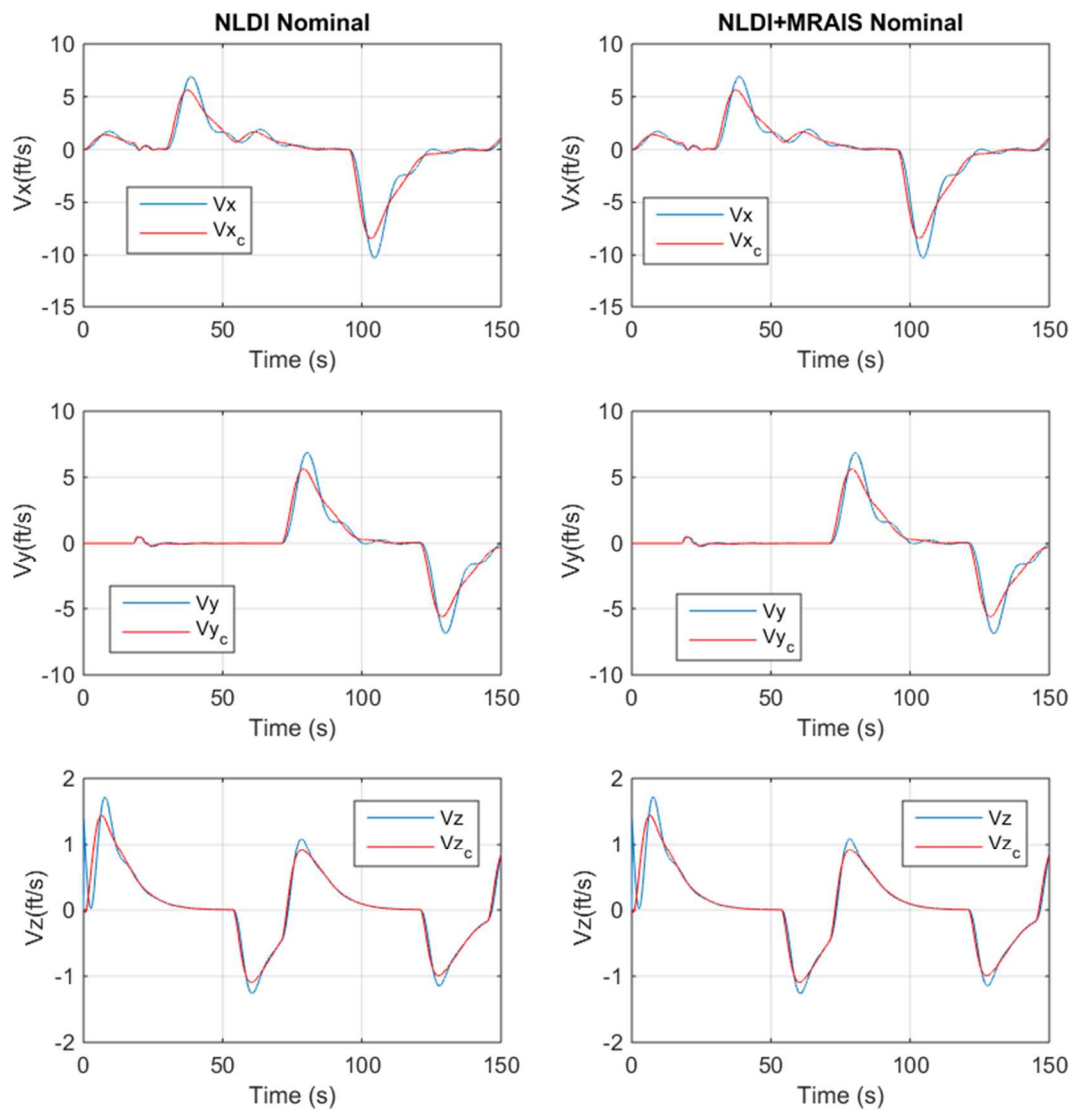


Figure 6.10 Velocities Tracking for NLDI and NLDI+MRAIS Nominal Condition.

Figure 6.11 and Figure 6.12 depict the 3D trajectory and waypoint navigation results for the NLDI and NLDI+MRAIS for nominal conditions. It is very well appreciated that both controllers are able to hit the specified waypoints and successfully complete the mission.

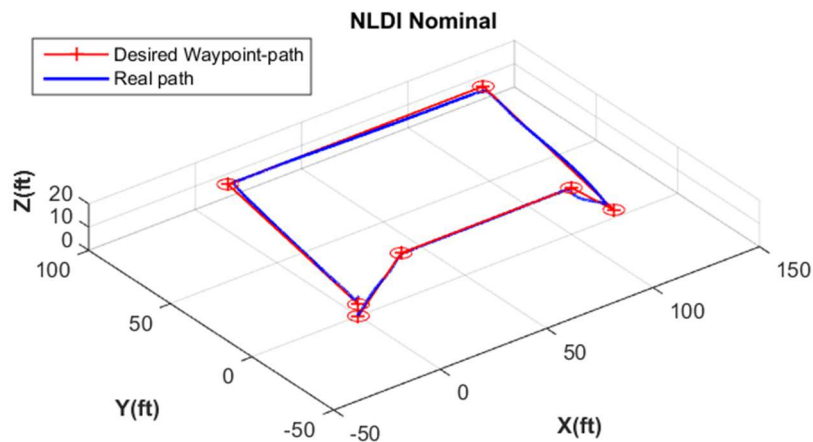


Figure 6.11 Position Tracking for NLDI Nominal Condition.

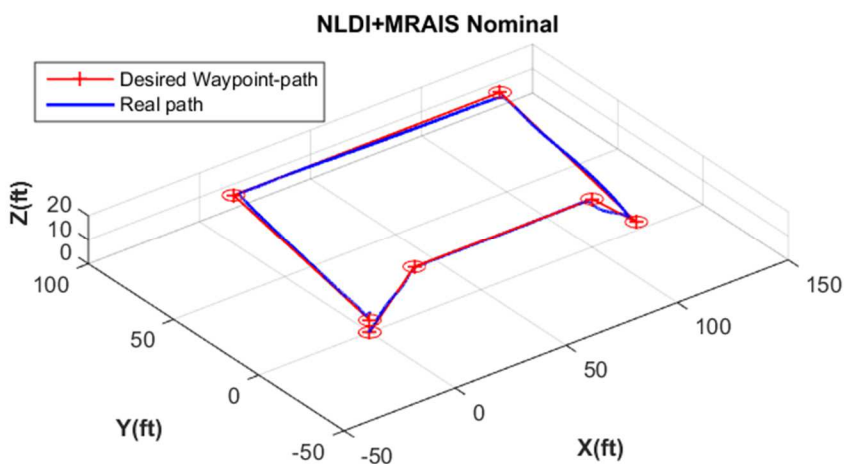


Figure 6.12 Position Tracking for NLDI+MRAIS Nominal Condition.

Abnormal Conditions Results

Figure 6.13 - Figure 6.17 present the tracking performance for all the states of interest for the same waypoints defined for the nominal condition case. However, in this scenario a high magnitude failure is induced. It consists of a thruster limited to 4.6% of its maximum power and the swiveling of the same thruster is blocked to 10 degrees. The failure is injected forty five seconds after the vehicle takes off.

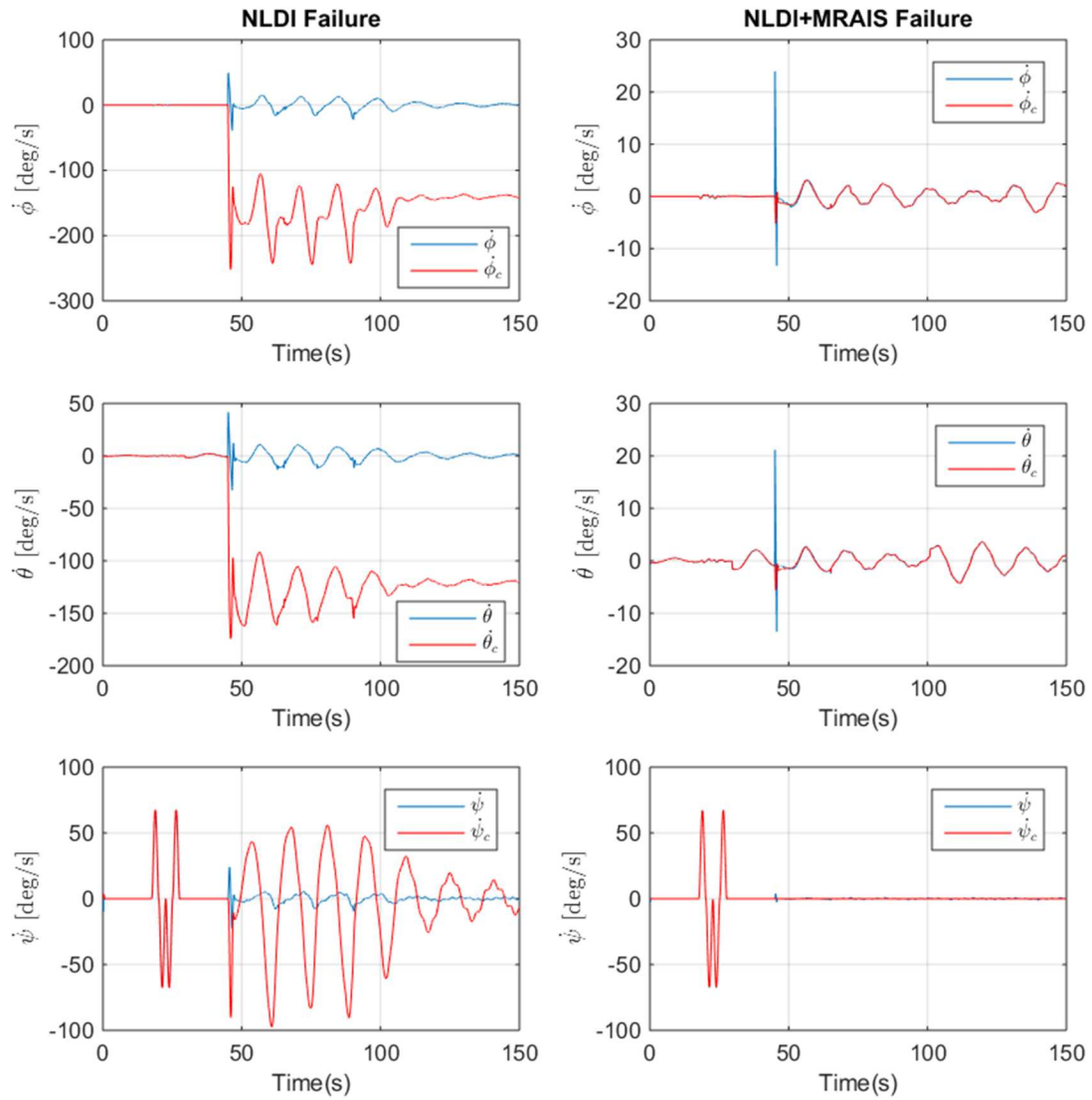


Figure 6.13 Euler Rates Tracking, NLDI and NLDI+MRAIS, Abnormal Condition.

Figure 6.13 and Figure 6.14 show a direct comparison between the NLDI and NLDI+MRAIS for attitude and Euler rates tracking for abnormal conditions. From Figure 6.13 it can clearly be seen that the adaptive augmentation successfully rejects the failure for Euler rates. On the other hand, as shown on Figure 6.14, a significant mitigation of the overshoot and amplitude of undesired oscillations is noticed in favor of the MRAIS.

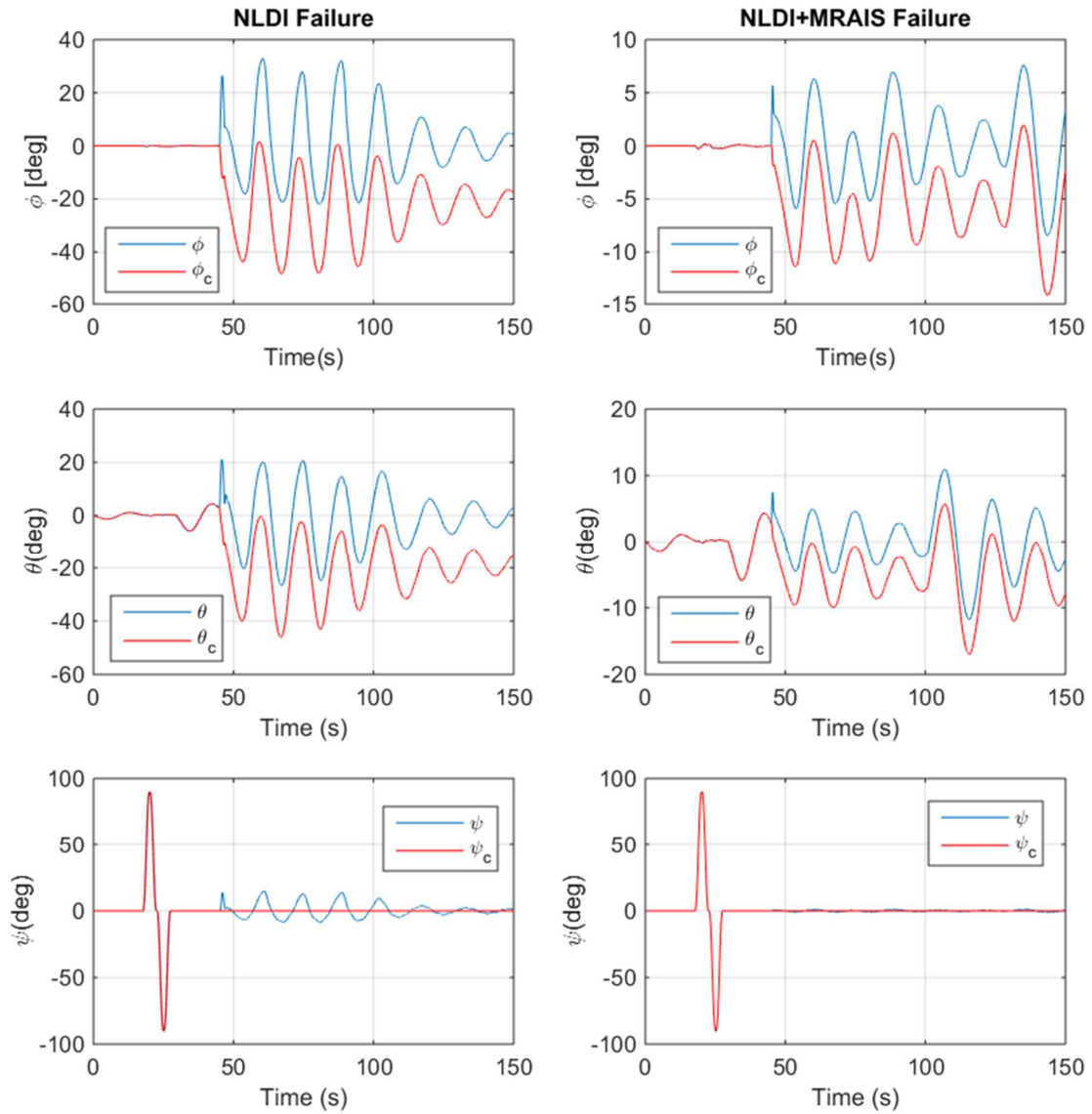


Figure 6.14 Attitude Tracking for NLDI and NLDI+MRAIS Abnormal Condition.

Figure 6.15 depicts the velocities tracking performance for both controllers. Although the adaptive augmentation is only implemented in the inner loop, it can be observed that the adaptation successfully mitigates the effect of the failure.

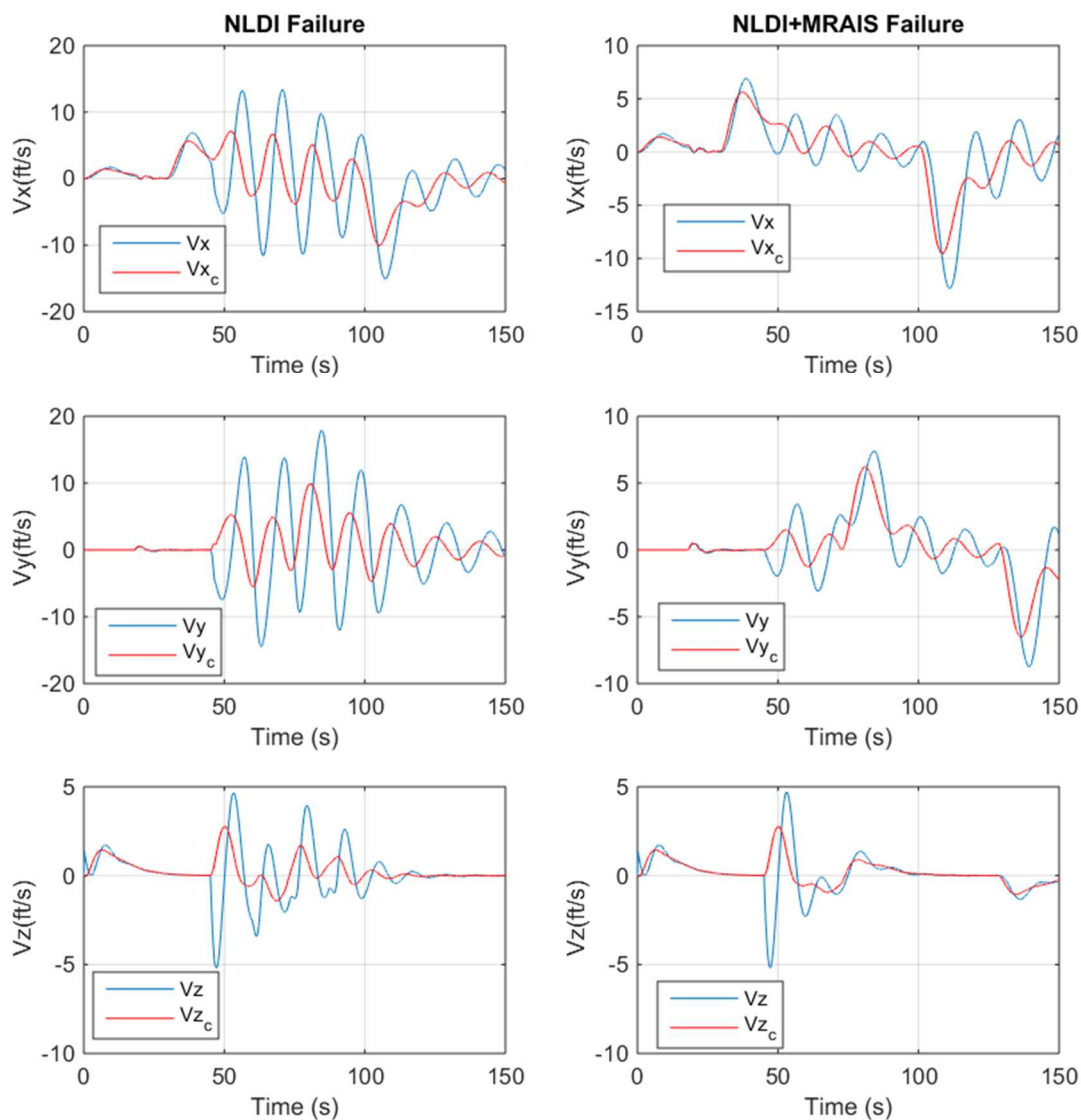


Figure 6.15 Velocities Tracking for NLDI and NLDI+MRAIS Abnormal Condition.

Figure 6.16 and Figure 6.17 present a 3D position trajectory comparison for both controllers. It can be observed that the NLDI+MRAIS possess a higher stability margin with less oscillatory behavior. It can be noticed that the NLDI without the augmentation is not even capable of finalizing the whole maneuver due to the effect of the high magnitude failure.

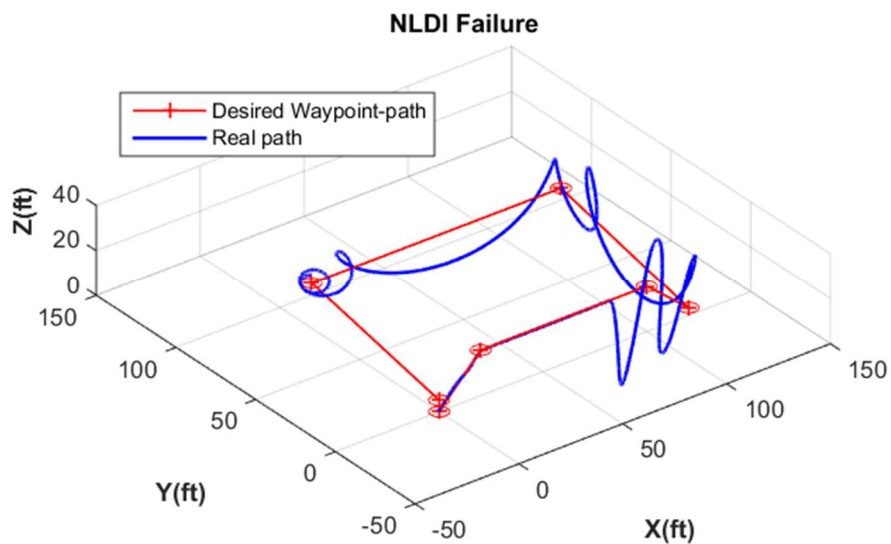


Figure 6.16 Position Tracking for NLDI Abnormal Condition.

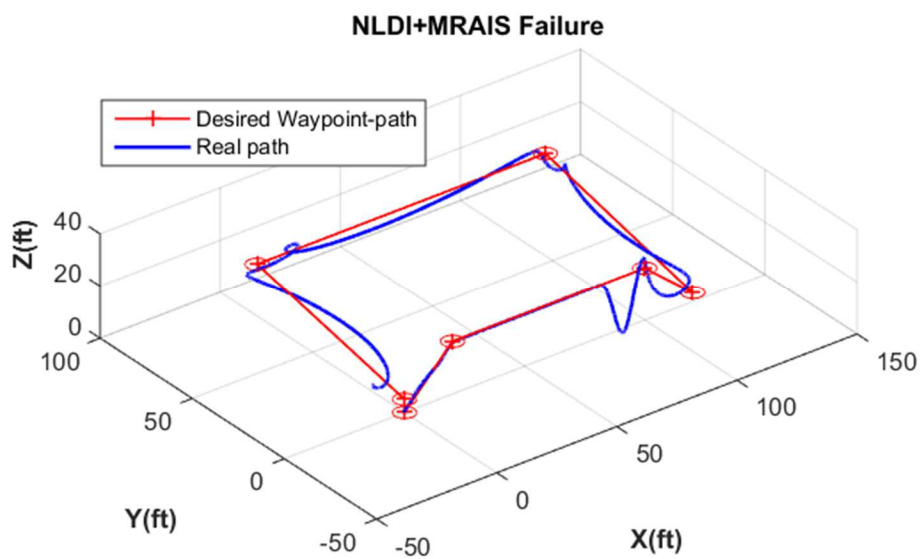


Figure 6.17 Position Tracking for NLDI+MRAIS Abnormal Condition.

Figure 6.18 and Figure 6.19 depict the time history of the adaptive gains for the roll and pitch axis of the NLDI+MRAIS controller. From the Figures is possible to see that as soon the failure is injected (45 seconds after the test starts) the adaptation gains get more active to compensate for the effect of the failure.

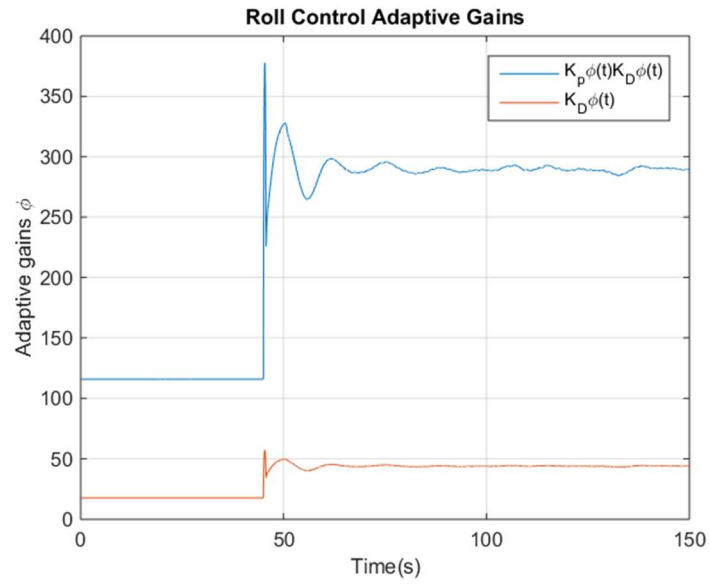


Figure 6.18 MRAIS Adaptive Gains for Roll Dynamics.

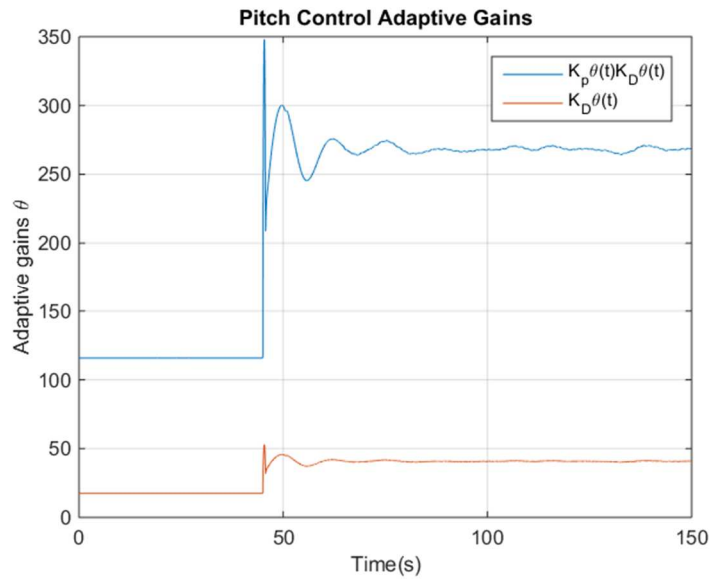


Figure 6.19 MRAIS Adaptive Gains for Pitch Dynamics.

The same effect is noticed for the yaw axis adaptive component; a higher activity of the adaptive gains is noticeable right after the failure is injected.

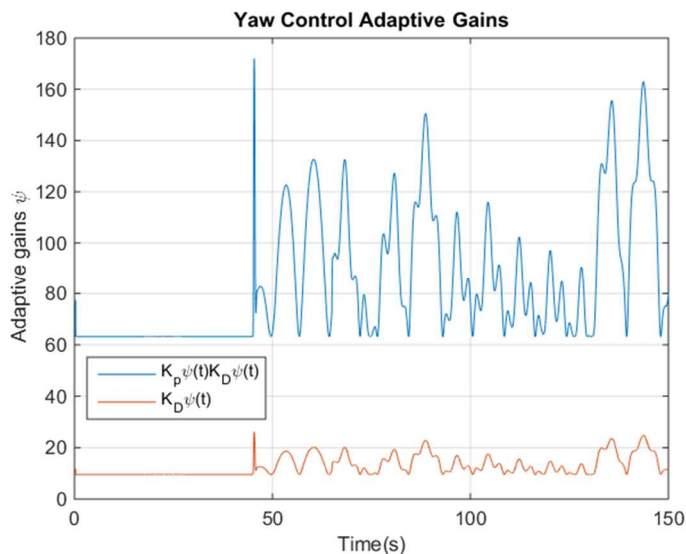


Figure 6.20 MRAIS Adaptive Gains in Yaw Dynamics.

Table 6.3 presents a direct comparison of each of the resultant performance metrics per each case. From the table it is possible to see that the adaptive augmentation greatly outperforms the baseline NLDI controller in all aspects.

Table 6.3 Performance Metrics for Simulation Results.

	Nominal		Failure		CV	Percentages
	NLDI	NLDI + MRAIS	NLDI	NLDI + MRAIS		
rms rates(deg/s)	1.05	0.24	268.98	2.25	268.98	0.2
rms attitude(deg/s)	1.90	0.03	41.28	9.61	41.28	0.2
rms velocities (ft/s)	1.05	1.05	11.76	3.68	11.76	0.2
rms position (ft)	31.61	31.61	42.84	33.67	42.84	0.2
P. Index	0.63	0.67	0.004	0.56		
P. Index Norm	0.95	1.00	0.006	0.86		

The Histogram depicted in Figure 6.21 presents a global performance index comparison obtained from the results of Table 2 using Eq.(6.28). It can clearly be noticed that the NLDI+MRAIS control provides better global performance than the NLDI controller.

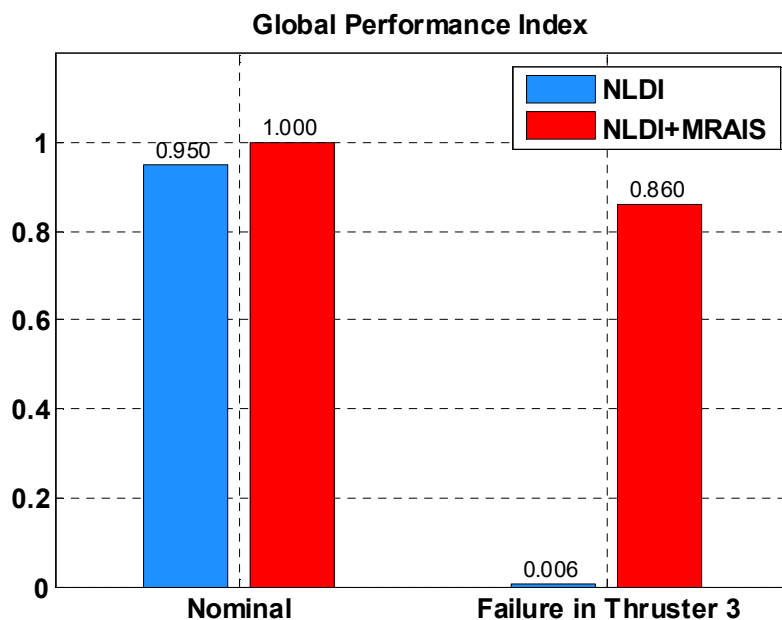


Figure 6.21 Global Performance Index for the Tests Performed.

6.7.HIL Setup and Results

Hardware in the Loop (HIL) simulation is one of the most important stages of control law testing and validation before the final onboard implementation. It is important to test all flight control code functionalities in order to address any relevant issues that might appear in a real flight. It is desirable to design a flight control system that can operate in real-time and make sure that the flight computer can perform all the computations and all data acquisition required to maintain stable flight within the designed control laws. For HIL testing purposes three independent computer systems are commonly used: a primary flight computer, a computer that simulates the system dynamics and a host computer.

Figure 6.22 depicts the most relevant components for HIL simulation and how the different computers interact. The Target PC-2 holds a high fidelity simulation environment to model all the vehicle and actuator dynamics, it also incorporates sensor

models to account for realistic noise and biases that the real onboard sensors might read. The Athena II Flight Computer (Target PC-1) holds all the control laws, guidance algorithms and prerecorded trajectories required for real operation. The host computer is in charge of uploading the code to both targets and displaying and recording the simulation results.

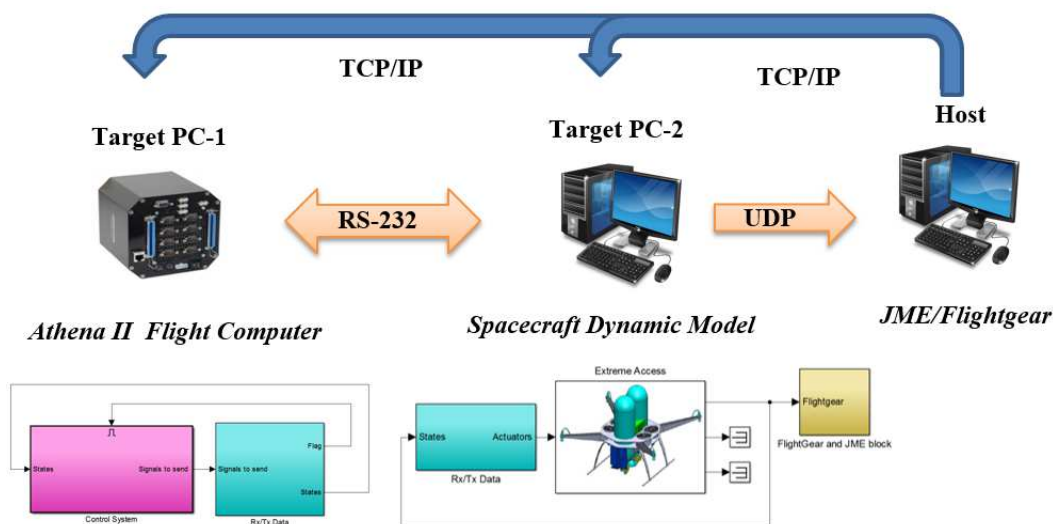


Figure 6.22 Simulation Environment Used for HIL Simulation.

6.7.1. Real Time Environment

The real time environment used is the MathWorks® Simulink Real-time toolset. Simulink Real time is a real-time operating system that enables the user to load Simulink models on to physical systems and execute them in real-time. In the setup shown in Figure 6.22 the host computer uses MATLAB/Simulink® to create and deploy executable code (.dlm) into the target(s) computer(s) to run real time applications. The host computer builds and compiles Simulink Real-Time code using a C++ compiler (usually Microsoft SDK or Microsoft Visual 2010 or higher). Then, the host computer sends this information to the target computers to execute the code in two separate

instances of the Simulink Real-Time operating system (RTOS). The SRT operating system will boot on both targets after startup using pre-configured external USB devices. The steps used to perform this process are shown in Figure 6.23.

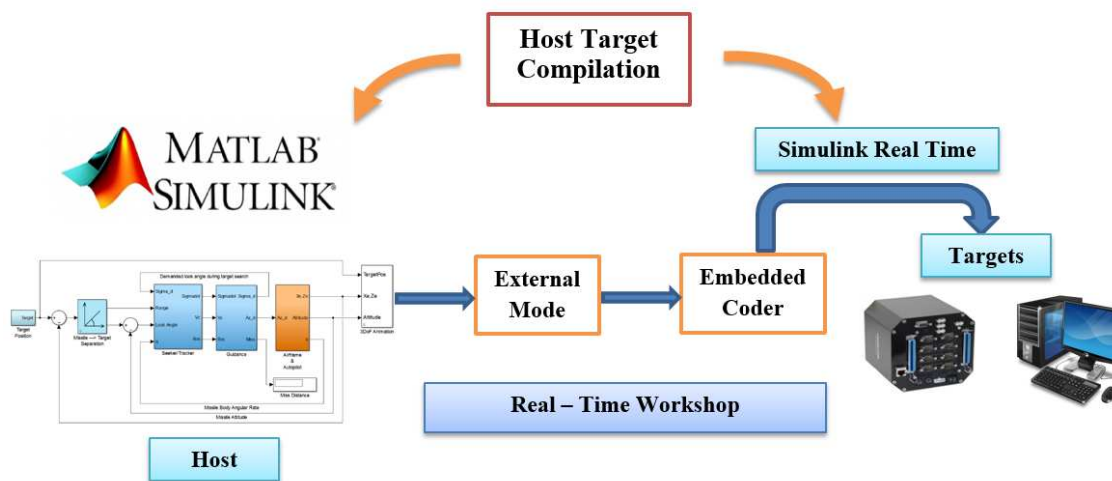


Figure 6.23 The Steps to Compile Simulation using SRT on to Target Hardware.

The real-time environment has the capability of tracking and logging a parameter called Task Execution Time (TET). This parameter is important to analyze the time required by the onboard computer to perform a computer cycle.

6.7.2. Data Communication

The data communication in Figure 6.22 is established as follows: first, the host uploads the Simulink Real Time executable code on both target computers via a stable LAN or Wi-Fi TCP/IP protocol. Once both targets are turned on and linked to the host (using the Simulink real time explorer) the spacecraft simulation computer (Target PC-2) will start exchanging data with the primary flight computer (Target PC-1) via a serial link (RS-232 protocol). The states of the system required from the guidance and control laws

are sent from the simulation computer to the primary flight computer. The required control inputs (thrusters) required for autonomous operation are sent from the primary flight computer to the simulation computer. Simultaneously, the vehicle computer sends selected state data via UDP to the host computer to be visualized in a Java Monkey Engine. The visualization results are shown in Figure 6.24.

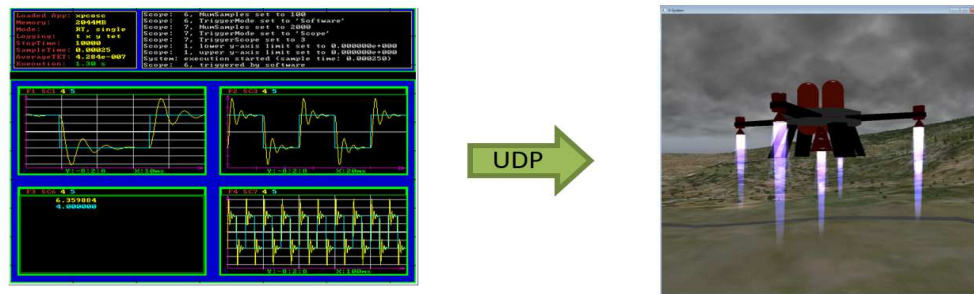


Figure 6.24 Simulink-Real Time Target and JME Displaying Spacecraft System (Perez A. , et al., 2016).

A real HIL Simulation setup is shown in Figure 6.25, where the fundamental sample frequency of the controller target was set to 100 Hz while a 500 Hz sampling frequency is selected for the simulation environment target computer.

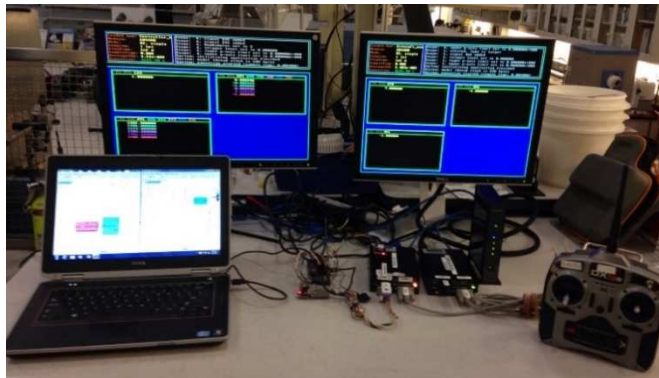


Figure 6.25 HIL Setup at KSC Swamp Works Laboratories (Perez A. , et al., 2016).

Mission 1, Nominal condition:

This test was performed to follow a specific path of nine waypoints $[\bar{p}_1 - \bar{p}_9]$. At

waypoint \bar{p}_5 , the spacecraft lands to drill and take a regolith sample. A total mass of 200g is sampled and placed at a location $\bar{r} = [0 \ 0.5 \ 0.5]ft$ from the center of gravity (CG) of the vehicle. The added mass will produce the following change in inertia and CG shift on the vehicle.

$$\Delta J_1 = \begin{bmatrix} 0.0068 & 0 & 0 \\ 0 & 0.0034 & 0 \\ 0 & 0 & 0.0034 \end{bmatrix} \text{slug} \cdot \text{ft}^2, \quad \Delta \bar{r}_1 = [0 \ 0.0057 \ 0.0057]ft \quad (6.29)$$

The mission sequence is described below.

- Take off from position $\bar{p}_1 = [0 \ 0 \ 0]ft$.
- Advance to position $\bar{p}_2 = [0 \ 0 \ 16.4]ft$.
- Advance to position $\bar{p}_3 = [164 \ 98.4 \ 16.4]ft$.
- Advance to position $\bar{p}_4 = [360.8 \ 131.2 \ 16.4]ft$,
- Advance to position $\bar{p}_5 = [360.8 \ 131.2 \ -213.2]ft$, and collect a sample.
- Advance to position $\bar{p}_6 = [360.8 \ 131.2 \ -32.8]ft$.
- Advance to position $\bar{p}_7 = [229.6 \ 65.6 \ 16.4]ft$.
- Advance to position $\bar{p}_8 = [0 \ 0 \ 16.4]ft$.
- Advance to position $\bar{p}_9 = [0 \ 0 \ 0]ft$.

The sequence of commanded waypoints is illustrated in Figure 6.26.

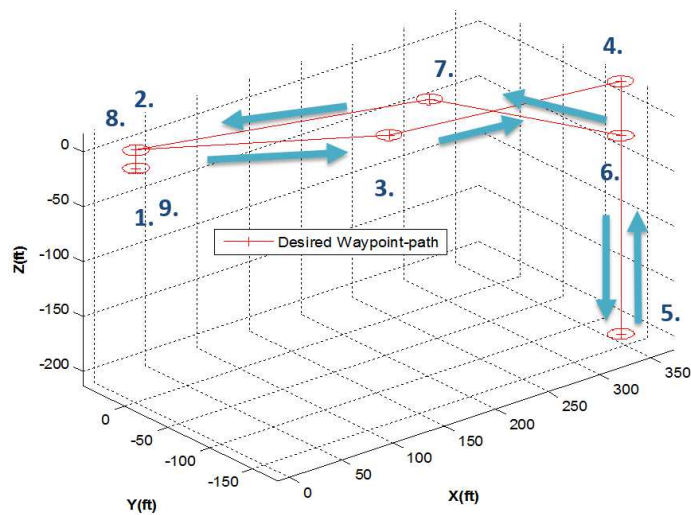


Figure 6.26 Path Designation for HIL Missions.

Mission 2: Sampling Extra Weight

This test is performed to follow the same waypoint path $[\bar{p}_1 - \bar{p}_9]$ of Mission 1, but instead, the system is loaded after reaching \bar{p}_5 with a mass equal to 800 g at a location $\bar{r} = [0 \ 0.5 \ 0.5]ft$ from the CG of the vehicle. The added mass will produce the following change in inertia and CG shift on the vehicle:

$$\Delta J_2 = \begin{bmatrix} 0.0262 & 0 & 0 \\ 0 & 0.0131 & 0 \\ 0 & 0 & 0.0131 \end{bmatrix} \text{slug} \cdot \text{ft}^2, \quad \Delta \bar{r}_2 = [0 \ 0.022 \ 0.022]ft \quad (6.30)$$

Mission 3: Reduced Efficiency on Thruster 1

This test was performed to follow the same waypoint path $[\bar{p}_1 - \bar{p}_9]$ of Mission 1. At \bar{p}_5 the system is loaded with a mass equal to 200 g at a location $\bar{r} = [0 \ 0.5 \ 0.5]ft$ from the CG. Additionally thruster 1 power is slowly reduced up to 40% of its maximum force in a period of 150 seconds; this is intended to simulate the effect of a thruster that is slowly losing efficiency.

6.7.3. HIL XAFF Results.

The Histogram of Figure 6.27 depicts a global performance index for the three missions for each controller under study. The controllers considered for this HIL simulation are: a classic PID controller, INLDI, NLDI and NLDI+MRAIS. From the histogram it is possible to directly assess the performance of each one; it is clear that the NLDI+MRAIS architecture yields improved performance for all the cases. Both versions of the NLDI (incremental and exact approach) yield very similar performance while the PID controller is the least robust of all of them.

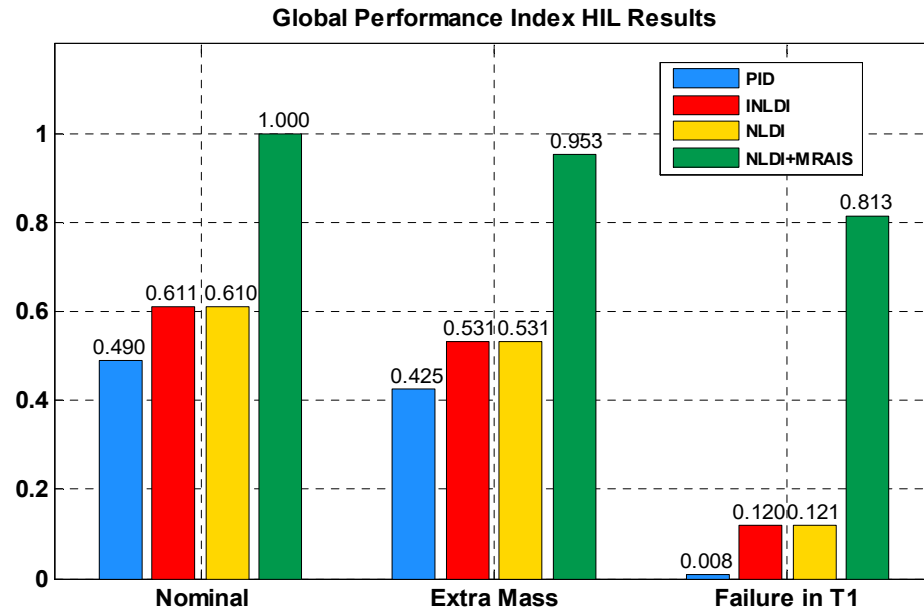


Figure 6.27 HIL Global PI Histogram Results.

6.8. Implementation Results on Gimbaled Mini-Free Flyer

This section presents the real implementation results of NLDI and NLDI+MRAIS control architectures into a concept prototype designed by NASA. This vehicle is called Mini-Free Flyer (MFF). It is a low cost vectoring thrust prototype built for preliminary testing purposes within a 3 DOF gimbaled platform as shown in Figure 6.28.



Figure 6.28 Mini-Free Flyer Mounted in Gimbal Setup.

The MFF is equipped with four EDFs capable of 1.2 kg of thrust each. The motors are powered by a Thunder Power 14.8V 3850 mA Lipo battery and vectoring thrust is incorporated via four Futaba S3152 high quality Servos that allows axial motion of each of the ducted fans. The board used for telemetry and control law development is a 3DR *Pixhawk*, compatible with MATLAB-Simulink code deployment. The inertial characteristics and distance from each motor to the CG of the vehicle are specified in Table 6.4.

Table 6.4 Inertial and Geometrical Parameters of the MFF.

Inertial parameters		Distance from motors to CG	
Ixx (slug-ft ²)	0.0089	Lx (ft)	0.331
Iyy (slug-ft ²)	0.0088	Ly (ft)	0.331
Izz (slug-ft ²)	0.0126	Lz (ft)	0.064
m (slug)	0.14		

The propulsive characteristics of the EDF of the MFF prototype are shown in Figure 6.29.

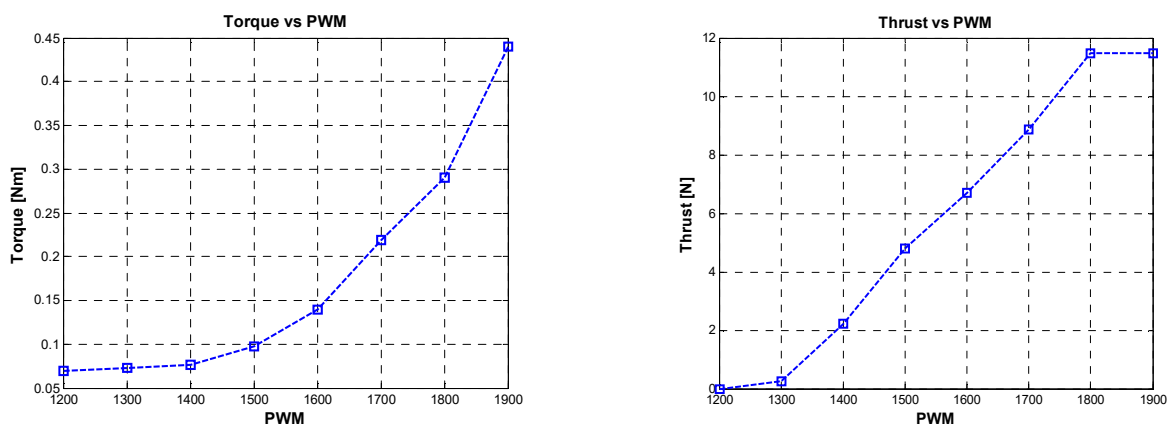


Figure 6.29 Mini-Free Flyer Torque and Thrust vs PWM.

6.8.1. Results for Failure in EDF 3.

This section presents the results of a set of tests in which the EDF 3 (see Figure 6.5 for convention) is fully blocked 15 seconds after the tests start while the MFF tries to track a cycloid type maneuver. The results presented in Figure 6.30 depict an angular rate tracking comparison for both the nominal NLDI controller and the NLDI controller augmented with MRAIS.

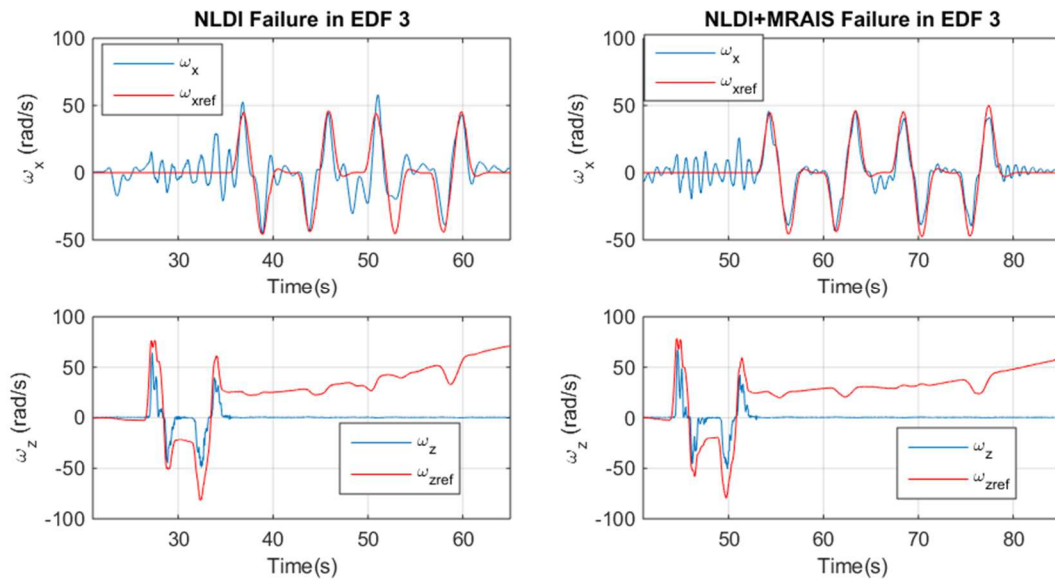


Figure 6.30 Angular Rates Tracking Performance for a Failure in EDF 3.

It is possible to see a noticeable improvement in terms of less oscillatory behavior and more precise tracking performance in favor of the adaptive augmentation. The results in Figure 6.31 show a similar trend for roll angle tracking. It is noticeable that the adaptive augmentation allows the system to perform better tracking control on the roll angle during abnormal conditions.

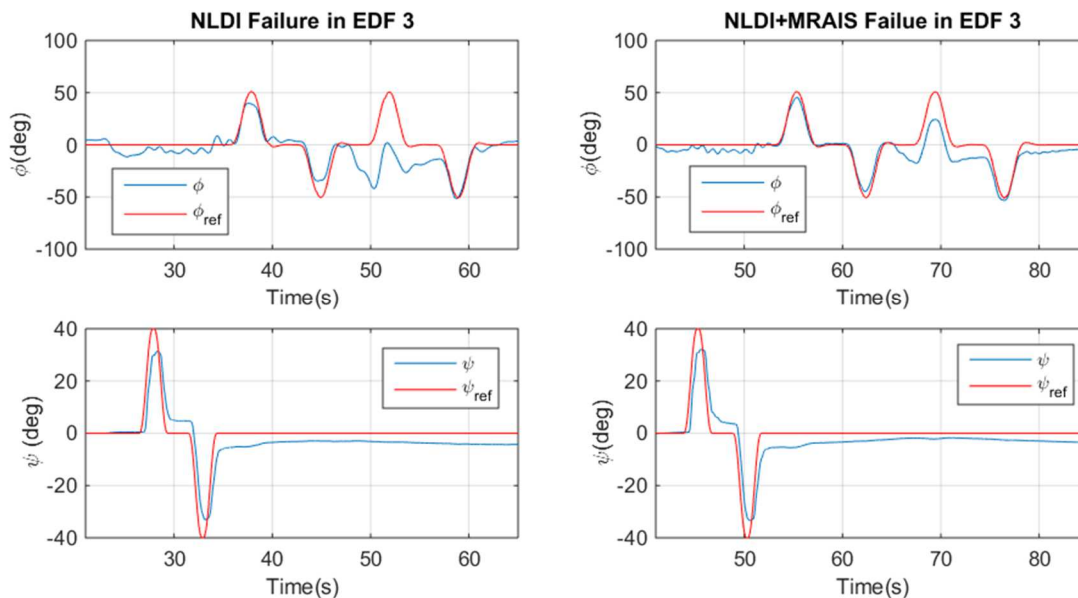


Figure 6.31 Roll and Yaw Tracking Performance for a Failure in EDF 3.

Figure 6.32 depicts the time history of the adaptive gains for the roll axis. It is possible to see that the gains get more active exactly when the failure is injected into the system. This same behavior was observed in the simulations of the XAFF.

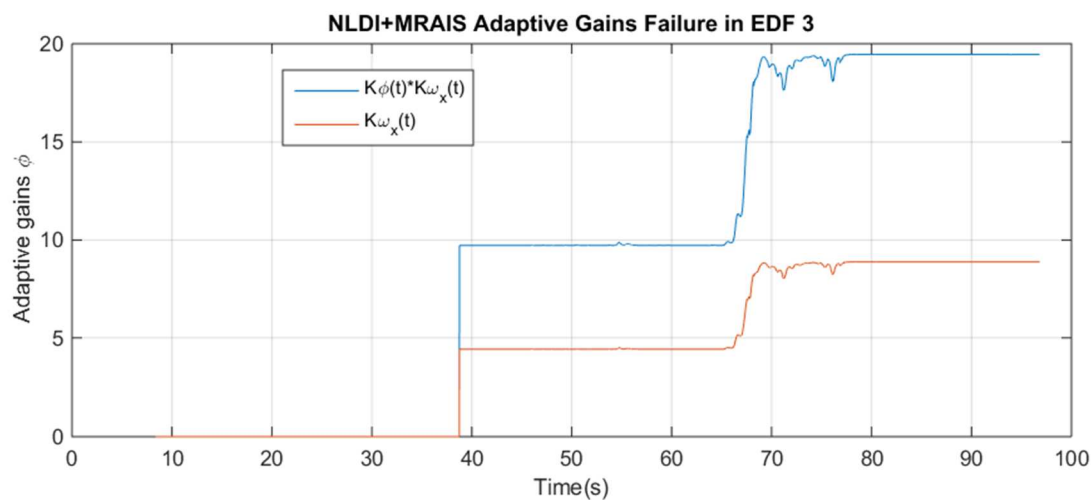


Figure 6.32 Time History of Adaptive Gains for a Failure in EDF 3.

6.8.2. Results for Added Mass in one Arm

This section presents the results of a set of tests in which a mass of 130g was added in one of the arms of the system while the MFF tries to track a cycloid type maneuver.

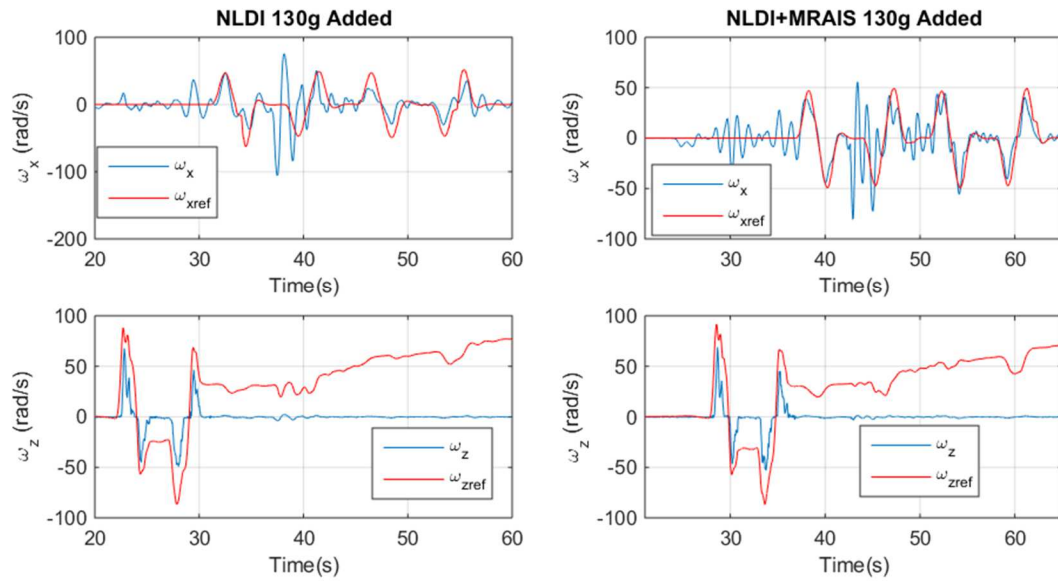


Figure 6.33 Angular Rate Tracking Performance After Adding 130g on Left Arm.

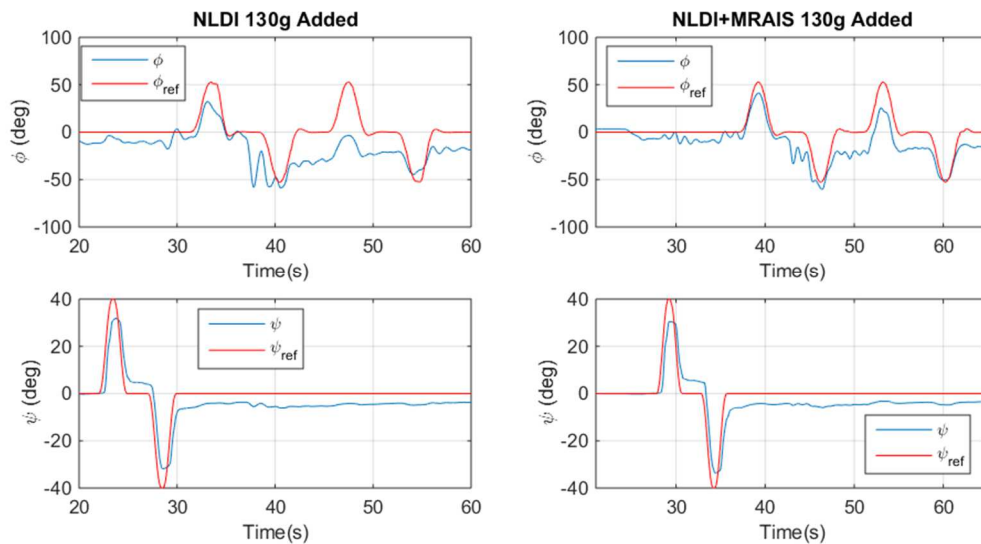


Figure 6.34 Roll and Yaw Tracking Performance after Adding 130g on Left Arm.

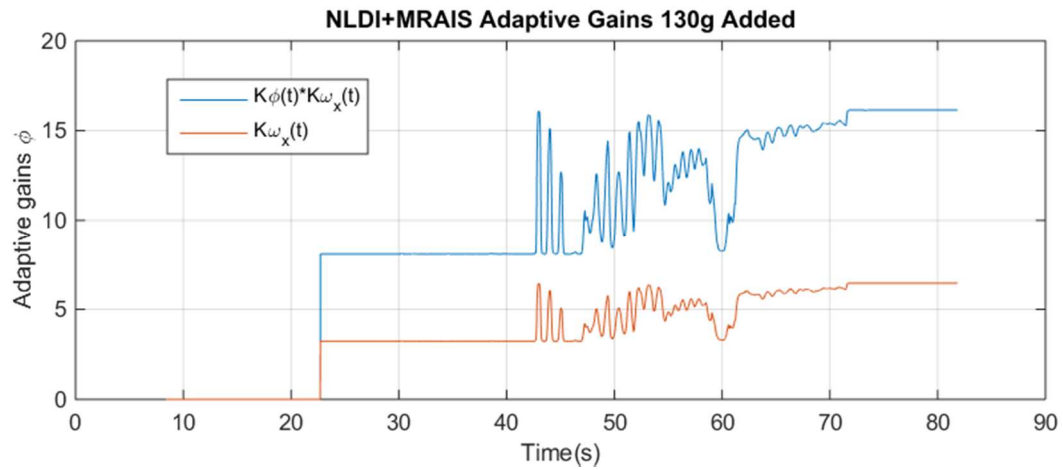


Figure 6.35 Time History of Adaptive Gains after Adding 130g on Left Arm.

Once again, improvement in tracking of angular rate and roll angle is achieved when the adaptive augmentation is engaged. However, from the results it is possible to see that this failure greatly modifies the inertia and dynamics of the system. As a result, it is more difficult for the adaptive augmentation to overcome.

7. Quaternion Based Adaptation for Spacecraft Attitude Control

This chapter presents the theoretical background for the development of an adaptive control architecture proposed in this dissertation applicable to spacecraft vehicles that can attain large attitude angles. Spacecraft control is different than conventional aircraft control; in space or in semi-gravity-less environments where there is no drag or friction forces, large attitude angles are easily attainable. It is known that the main issue with Euler angles for attitude representation and control are the singularities present at 90° and -90° in the pitch angle. To overcome this difficulty, quaternion mathematics can be used.

Quaternions were developed in 1843 by Sir William Hamilton (Yang, 2012) as an alternative approach to represent the orientation of a rigid body. Quaternions can overcome the singularity problem by defining a hyper-complex number with four parameters, a three component vector and an angle rotation around this vector (see Figure 7.1). A quaternion can be described mathematically as follows:

$$\mathbf{q} = \{q_{1,3}, q_4\} = \{q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}, q_4\} \quad (7.1)$$

where $\mathbf{q}_{1,3} = [q_1 \ q_2 \ q_3]^T$ are the complex terms and q_4 is the scalar part of the quaternion. Quaternions can be related to a principal rotation angle Φ using the following relationship:

$$\begin{aligned} q_1 &= \hat{e}_x \sin(\Phi/2) \\ q_2 &= \hat{e}_y \sin(\Phi/2) \\ q_3 &= \hat{e}_z \sin(\Phi/2) \\ q_4 &= \cos(\Phi/2) \end{aligned} \quad (7.2)$$

where $\hat{e} = [e_x \ e_y \ e_z]^T$ is a unit length vector for the rotational axis angle Φ (Yang, 2012).

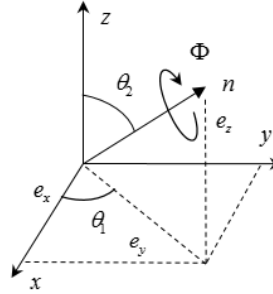


Figure 7.1 Quaternion Attitude Representation.

One important feature about quaternions relies on the way attitude error is described, namely by the product quaternion operator defined in Eq. (7.3) (Wie & Barba, 1985).

$$\delta \mathbf{q} = \mathbf{q} \otimes \mathbf{q}_c^{-1} = \begin{bmatrix} \delta q_1 \\ \delta q_2 \\ \delta q_3 \\ \delta q_4 \end{bmatrix} = \begin{bmatrix} q_{4c} & q_{3c} & -q_{2c} & -q_{1c} \\ -q_{3c} & q_{4c} & q_{1c} & -q_{2c} \\ q_{2c} & -q_{1c} & q_{4c} & -q_{3c} \\ q_{1c} & q_{2c} & q_{3c} & -q_{4c} \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \quad (7.3)$$

where \mathbf{q} and \mathbf{q}_c are the actual and commanded quaternion vectors respectively. Great care should be taken as different notations are used for quaternion operations. Some authors include the scalar component of the quaternion as the first element of the quaternion vector; in this dissertation that is not the case, since the scalar component is taken as the fourth component of the quaternion vector.

As described in Chapter 3, the system of equations formed by Eq.(7.4) and Eq.(7.5) fully describes the rotational dynamics and quaternion kinematics commonly used for spacecraft modeling and control.

$$\dot{\boldsymbol{\omega}} = J^{-1} \{ -\boldsymbol{\omega} \times (J\boldsymbol{\omega}) + \bar{M}(t) \} \quad (7.4)$$

$$\dot{\mathbf{q}} = \frac{1}{2} \Omega(\boldsymbol{\omega}) \mathbf{q} \quad (7.5)$$

where $J \in \mathfrak{R}^{3 \times 3}$ is the inertia matrix, $\bar{M}(t) \in \mathfrak{R}^{3 \times 1}$ is a vector with the sum of external

moments that act on the rigid body and $\boldsymbol{\omega} \in \mathfrak{R}^{3 \times 1}$ is the vector of angular rates. In this case the moments that act on the spacecraft will be the control inputs provided by the spacecraft actuators (thrusters, reaction wheels etc.). The skew symmetric matrix $\Omega(\boldsymbol{\omega})$ will depend on the angular rates and it is defined as follows (Wie B. , 2015):

$$\Omega(\boldsymbol{\omega}) = \begin{bmatrix} 0 & \omega_3 & -\omega_2 & \omega_1 \\ -\omega_3 & 0 & \omega_1 & \omega_2 \\ \omega_2 & -\omega_1 & 0 & \omega_3 \\ -\omega_1 & -\omega_2 & -\omega_3 & 0 \end{bmatrix} \quad (7.6)$$

Using the derivation given in (Crassidis & Markley, 2014) the quaternion error kinematics can be defined as follows:

$$\delta \dot{\mathbf{q}} = \frac{1}{2} \Omega(\boldsymbol{\omega}) \delta \mathbf{q} \quad (7.7)$$

If the inertia matrix is assumed to be time invariant the system of equations can be described in non-linear affine state space form:

$$\dot{\mathbf{x}} = f(t, \mathbf{x}) = \mathbf{f}_q(\mathbf{x}) + \mathbf{G}_q(\mathbf{x}) \mathbf{u}(t) = \begin{bmatrix} \dot{\boldsymbol{\omega}} \\ \delta \dot{\mathbf{q}} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_\omega(\boldsymbol{\omega}) \\ (1/2) \Omega(\boldsymbol{\omega}) \delta \mathbf{q} \end{bmatrix} + \begin{bmatrix} J^{-1} \\ [\mathbf{0}]_{4 \times 3} \end{bmatrix} \bar{M}(t) \quad (7.8)$$

where the state vector is defined as $\mathbf{x} = [\boldsymbol{\omega}, \delta \mathbf{q}]^T \in \mathfrak{R}^{7 \times 1}$, the vector function $\mathbf{f}_q(\mathbf{x}) \in \mathfrak{R}^{7 \times 1}$, the matrix $\mathbf{G}_q(\mathbf{x}) \in \mathfrak{R}^{7 \times 3}$ and $\mathbf{f}_\omega(\boldsymbol{\omega})$ is defined as:

$$\mathbf{f}_\omega(\boldsymbol{\omega}) = -J^{-1}[-\boldsymbol{\omega} \times (J\boldsymbol{\omega})] \quad (7.9)$$

7.1. Quaternion Based Partial Feedback Linearized Control

Quaternion math simplifies the attitude singularity (“gimbal lock”) problem by incorporating the quaternion error operator into the control law. However quaternion error kinematics defined in Eq.(7.7) can’t be fully feedback linearized without incorporating an additional singularity in the principal rotational axis angle Φ when it

approaches 180 degrees (Bang, ShinLee, & JuEun, 2004). Therefore, only the angular rate rotational dynamics of Eq.(7.4) can be fully feedback linearized. A control law that will partially feedback linearize the system (7.8) is:

$$\bar{M}(t) = \mathbf{u}_{NLDI}(t) = \boldsymbol{\omega} \times (J\boldsymbol{\omega}) + J\mathbf{u}_v(\mathbf{q}, \boldsymbol{\omega}) \quad (7.10)$$

where $\mathbf{u}_v(\mathbf{q}, \boldsymbol{\omega})$ is a virtual controller that can be selected arbitrarily. In this case the following time invariant virtual controller can be used to stabilize the system:

$$\mathbf{u}_v(\mathbf{q}, \boldsymbol{\omega}) = -\mathbf{K}_{\omega q}[\boldsymbol{\omega}, \delta\mathbf{q}]^T \quad (7.11)$$

where $\mathbf{K}_{\omega q} \in \mathfrak{R}^{3 \times 7}$ is a matrix containing control gains defined as follows:

$$\mathbf{K}_{\omega q} = [\mathbf{K}_\omega : \mathbf{K}_q : \mathbf{0}_{3 \times 1}] \quad (7.12)$$

The quaternion and angular rate control gains \mathbf{K}_q and \mathbf{K}_ω are defined respectively as follows:

$$\mathbf{K}_\omega = \begin{bmatrix} k_{\omega_x} & 0 & 0 \\ 0 & k_{\omega_y} & 0 \\ 0 & 0 & k_{\omega_z} \end{bmatrix}, \quad \mathbf{K}_q = \begin{bmatrix} k_{q1} & 0 & 0 \\ 0 & k_{q2} & 0 \\ 0 & 0 & k_{q3} \end{bmatrix} \quad (7.13)$$

which yields the following control input:

$$\bar{M}(t) = \mathbf{u}_{NLDI}(t) = \boldsymbol{\omega} \times (J\boldsymbol{\omega}) + J\{-\mathbf{K}_{\omega q}[\boldsymbol{\omega}, \delta\mathbf{q}]^T\} \quad (7.14)$$

After inserting the control law of Eq.(7.14) into the system in Eq.(7.8) the closed loop dynamics will be:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{\boldsymbol{\omega}} \\ \delta\dot{\mathbf{q}} \end{bmatrix} = \begin{bmatrix} -\mathbf{K}_{\omega q}[\boldsymbol{\omega}, \delta\mathbf{q}]^T \\ \frac{1}{2}\Omega(\boldsymbol{\omega})\delta\mathbf{q} \end{bmatrix} = \begin{bmatrix} -\mathbf{K}_\omega\boldsymbol{\omega} - \mathbf{K}_q\delta\mathbf{q}_{1:3} \\ \frac{1}{2}\Omega(\boldsymbol{\omega})\delta\mathbf{q} \end{bmatrix} \quad (7.15)$$

It can be seen that the closed loop system is being partially feedback linearized. However since the system is still not fully linear due to quaternion kinematics, it is worth

linearizing it with respect to an operating point to design \mathbf{K}_q and \mathbf{K}_ω via pole placement. The first step to linearize the closed loop system is to select an operating point. In this case, the operating point is located at the equilibrium condition in which the quaternion error and angular rate vectors are both zero.

$$\boldsymbol{\omega}_e = [0 \ 0 \ 0]^T, \delta \mathbf{q} = [0 \ 0 \ 0 \ 1]^T \quad (7.16)$$

To linearize the system, the Jacobian operator of the closed loop system in Eq.(7.15) must be calculated and evaluated at the operating point. This yields the following result:

$$\mathbf{A}_{cl} = \left. \frac{\partial \mathbf{x}}{\partial \mathbf{f}} \right|_{\mathbf{x}=[\boldsymbol{\omega}_e, \delta \mathbf{q}_e]} = \begin{bmatrix} -\mathbf{K}_\omega & -\mathbf{K}_q & [\mathbf{0}]_{3 \times 1} \\ (1/2)\mathbf{I}_{3 \times 3} & [\mathbf{0}]_{3 \times 3} & [\mathbf{0}]_{3 \times 1} \\ [\mathbf{0}]_{1 \times 3} & [\mathbf{0}]_{1 \times 3} & 0 \end{bmatrix} \quad (7.17)$$

If the eigenvalues of \mathbf{A}_{cl} are calculated, there will be a pole that is uncontrollable. This is due to the limitation of the system to have three inputs and four quaternions to be controlled. From (Yang, 2012), it is known that the controller will be able to successfully control the first three quaternion error components $\delta \mathbf{q}_{1:3}$ and the angular rates $\boldsymbol{\omega}$. It is known that the original system can be reduced to a controllable one in which the last error quaternion component δq_4 is omitted. This can be done and any orientation can still be achieved due to the following quaternion constraint:

$$\delta q_4 = \sqrt{1 - \delta q_1^2 - \delta q_2^2 - \delta q_3^2} \quad (7.18)$$

If $\mathbf{q}_{1:3}$ are actively tracked then they will be constrained and controlled as well.

The reduced system yields:

$$\begin{bmatrix} \dot{\boldsymbol{\omega}} \\ \delta \dot{\mathbf{q}}_{1:3} \end{bmatrix} = \begin{bmatrix} -\mathbf{K}_\omega & -\mathbf{K}_q \\ \frac{1}{2}\mathbf{I}_{3 \times 3} & [\mathbf{0}]_{3 \times 3} \end{bmatrix} \begin{bmatrix} \boldsymbol{\omega} \\ \delta \mathbf{q}_{1:3} \end{bmatrix} \quad (7.19)$$

where the new state space closed loop matrix reduces to :

$$\tilde{\mathbf{A}}_{cl} = \begin{bmatrix} -\mathbf{K}_\omega & -\mathbf{K}_q \\ \frac{1}{2}\mathbf{I}_{3 \times 3} & [\mathbf{0}]_{3 \times 3} \end{bmatrix} \quad (7.20)$$

The characteristic equation of the closed loop system can be obtained as follows:

$$p(\lambda) = \det(\lambda \mathbf{I}_{6 \times 6} - \tilde{\mathbf{A}}_{cl}) = (\lambda^2 + k_{\omega_x} \lambda + k_{q_1}/2) (\lambda^2 + k_{\omega_y} \lambda + k_{q_2}/2) (\lambda^2 + k_{\omega_z} \lambda + k_{q_3}/2) \quad (7.21)$$

Comparing this characteristic equation to a desired one:

$$p_d(\lambda) = (\lambda^2 + 2\zeta_x \omega_{nx} \lambda + \omega_{nx}^2) (\lambda^2 + 2\zeta_y \omega_{ny} \lambda + \omega_{ny}^2) (\lambda^2 + 2\zeta_z \omega_{nz} \lambda + \omega_{nz}^2) \quad (7.22)$$

By direct comparison, the following result is obtained for desired damping and natural frequency on each axis:

$$\begin{aligned} k_{\omega_x} &= 2\zeta_x \omega_{nx}, & k_{q_1} &= 2\omega_{nx}^2 \\ k_{\omega_y} &= 2\zeta_y \omega_{ny}, & k_{q_2} &= 2\omega_{ny}^2 \\ k_{\omega_z} &= 2\zeta_z \omega_{nz}, & k_{q_3} &= 2\omega_{nz}^2 \end{aligned} \quad (7.23)$$

The nonlinear controller of Eq.(7.14) can be proven to be stable for the full closed loop system by means of the Lyapunov candidate function in Eq.(7.24). However, the angular rate and quaternion gains need to be positive and all the quaternion gains will need to have the same scalar value, this is $k_{q_1} = k_{q_2} = k_{q_3} = k_q$:

$$V(\mathbf{x}) = \frac{1}{2} \omega_x^2 + \frac{1}{2} \omega_y^2 + \frac{1}{2} \omega_z^2 + \delta q_1^2 + \delta q_2^2 + \delta q_3^2 + (\delta q_4 - k_q)^2 \quad (7.24)$$

This is a positive definite function. The time derivative along trajectories of $V(\mathbf{x})$ will be:

$$\begin{aligned} \dot{V}(\mathbf{x}) &= \frac{\partial V}{\partial t} + \nabla V f(t, \mathbf{x}) = [\omega_x, \omega_y, \omega_z, 2\delta q_1, 2\delta q_2, 2\delta q_3, 2(\delta q_4 - k_q)] f(t, \mathbf{x}) \\ \dot{V}(\mathbf{x}) &= -\boldsymbol{\omega}^T \mathbf{K}_\omega \boldsymbol{\omega} = -k_{\omega_x} \omega_x^2 - k_{\omega_y} \omega_y^2 - k_{\omega_z} \omega_z^2 \end{aligned}$$

The derivative along trajectories of the Lyapunov candidate function in Eq.(7.24)

is negative semi-definite, the system trajectories will be bounded. However, to proof global asymptotic stability, one can use Lyapunov-Barbalat's Lemma by taking the second derivative of the Lyapunov function candidate:

$$\begin{aligned}\dot{V}(\mathbf{x}) &= -2\boldsymbol{\omega}^T \mathbf{K}_\omega \dot{\boldsymbol{\omega}} = -2\boldsymbol{\omega}^T \mathbf{K}_\omega (-k_q \delta \mathbf{q}_{1:3} - \mathbf{K}_\omega \boldsymbol{\omega}) \\ \dot{V}(\mathbf{x}) &= 2k_q \boldsymbol{\omega}^T \mathbf{K}_\omega \delta \mathbf{q}_{1:3} + 2\boldsymbol{\omega}^T (\mathbf{K}_\omega)^2 \boldsymbol{\omega}\end{aligned}\quad (7.25)$$

The previous function will be necessarily bounded since all the states (trajectories) are bounded and the gains are fixed. This means \dot{V} is uniformly continuous and thus $\dot{V} \rightarrow 0$. This can only be true if $\boldsymbol{\omega} \rightarrow 0$ and $\delta \mathbf{q}_{1:3} \rightarrow 0$ as time goes to infinity. Thus the non-linear control law in Eq.(7.14) reorients the spacecraft to the desired attitude from any arbitrary initial orientation. Note that δq_4 can be ± 1 , but it is not an inconvenience since both signs produce the same attitude. However, this control law doesn't guarantee that the shortest path is provided to the final orientation. To overcome this issue a slight modification can be included by the following control law (Crassidis & Markley, 2014):

$$\bar{M}(t) = \tilde{\mathbf{u}}_{NLDI}(t) = \boldsymbol{\omega} \times (J\boldsymbol{\omega}) + J[-k_q \text{sign}(\delta q_4) \delta \mathbf{q}_{1:3} - \mathbf{K}_\omega \boldsymbol{\omega}] \quad (7.26)$$

Figure 7.2 shows a schematic of the quaternion partially feedback linearized controller described by Eq.(7.26).

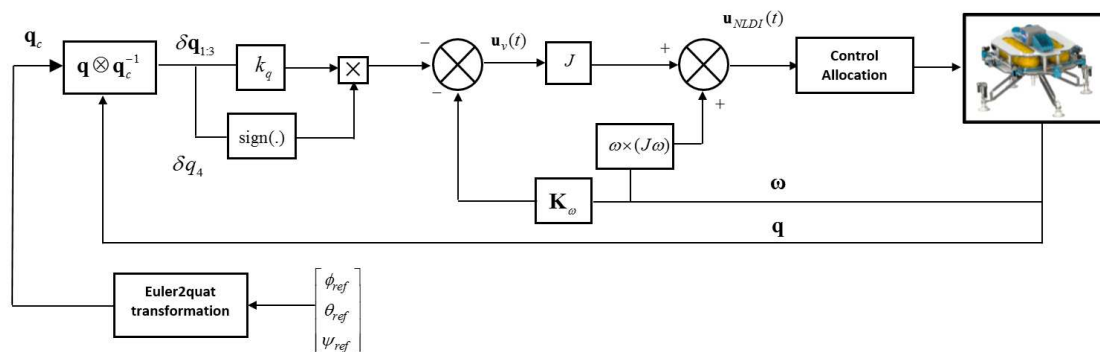


Figure 7.2 NLDI+Quaternion Attitude Control.

7.2. Quaternion Partial NLDI Control Augmented with Time-Varying Gains

Similar as before, let's define the following control input designed to partially feedback linearize the system in Eq.(7.8):

$$\bar{M}(t) = \mathbf{u}_{NLDI}(t) = \boldsymbol{\omega} \times (J\boldsymbol{\omega}) + J\mathbf{u}_v(\mathbf{q}, \boldsymbol{\omega}, t) \quad (7.27)$$

In this case, $\mathbf{u}_v(\mathbf{q}, \boldsymbol{\omega}, t)$ will be a time-varying virtual adaptive controller that will be specifically designed to meet Lyapunov's stability criteria. The closed loop system of equations after implementing the control law of Eq.(7.27) into the system in Eq.(7.8) yields:

$$\begin{cases} \dot{\boldsymbol{\omega}} = \mathbf{u}_v(\mathbf{q}, \boldsymbol{\omega}, t) \\ \delta \dot{\mathbf{q}} = \frac{1}{2} \Omega(\boldsymbol{\omega}) \delta \mathbf{q} \end{cases} \quad (7.28)$$

For clarity and further derivation purposes it is useful to write the system in scalar form as follows:

$$\begin{aligned} \dot{\omega}_x &= u_{v1}(\delta \mathbf{q}, \boldsymbol{\omega}, t) \\ \dot{\omega}_y &= u_{v2}(\delta \mathbf{q}, \boldsymbol{\omega}, t) \\ \dot{\omega}_z &= u_{v3}(\delta \mathbf{q}, \boldsymbol{\omega}, t) \\ \delta \dot{q}_1 &= (\omega_x \delta q_4 - \omega_y \delta q_3 + \omega_z \delta q_2) / 2 \\ \delta \dot{q}_2 &= (\omega_x \delta q_3 + \omega_y \delta q_4 - \omega_z \delta q_1) / 2 \\ \delta \dot{q}_3 &= (-\omega_x \delta q_2 + \omega_y \delta q_1 + \omega_z \delta q_4) / 2 \\ \delta \dot{q}_4 &= (-\omega_x \delta q_1 - \omega_y \delta q_2 - \omega_z \delta q_3) / 2 \end{aligned} \quad (7.29)$$

The final goal is to be able to prove stability of the closed loop system by means of time varying adaptive virtual controllers $u_{v1}(\delta \mathbf{q}, \boldsymbol{\omega}, t)$, $u_{v2}(\delta \mathbf{q}, \boldsymbol{\omega}, t)$ and $u_{v3}(\delta \mathbf{q}, \boldsymbol{\omega}, t)$. For convenience let's define the following state vector:

$$\mathbf{x} = [\omega_x, \omega_y, \omega_z, \delta q_1, \delta q_2, \delta q_3, \delta q_4]^T \quad (7.30)$$

which leads to the following non-linear and non-autonomous state space

representation:

$$\dot{\mathbf{x}} = f(\mathbf{x}, t) = \begin{bmatrix} u_{v1}(\mathbf{x}, t) \\ u_{v2}(\mathbf{x}, t) \\ u_{v3}(\mathbf{x}, t) \\ (\omega_x \delta q_4 - \omega_y \delta q_3 + \omega_z \delta q_2) / 2 \\ (\omega_x \delta q_3 + \omega_y \delta q_4 - \omega_z \delta q_1) / 2 \\ (-\omega_x \delta q_2 + \omega_y \delta q_1 + \omega_z \delta q_4) / 2 \\ (-\omega_x \delta q_1 - \omega_y \delta q_2 - \omega_z \delta q_3) / 2 \end{bmatrix} \quad (7.31)$$

The first step to prove stability is to establish a positive definite Lyapunov function candidate. After some trial and error process the following time invariant Lyapunov function was selected:

$$V(\mathbf{x}) = \frac{1}{2} \omega_x^2 + \frac{1}{2} \omega_y^2 + \frac{1}{2} \omega_z^2 + \delta q_1^2 + \delta q_2^2 + \delta q_3^2 + (\delta q_4 - k_q)^2 \quad (7.32)$$

where k_q is a positive scalar gain. Taking the time derivative of the Lyapunov function yields:

$$\dot{V}(\mathbf{x}, t) = \frac{\partial V}{\partial t} + \nabla V f(t, \mathbf{x}) \quad (7.33)$$

Since the Lyapunov candidate $V(\mathbf{x})$ is time invariant (time doesn't appear explicitly) the following Lyapunov candidate function derivative is obtained:

$$\begin{aligned} \dot{V}(\mathbf{x}, t) &= \nabla V f(t, \mathbf{x}) \\ &= \left[\frac{\partial V(\mathbf{x})}{\partial \omega_x} \quad \frac{\partial V(\mathbf{x})}{\partial \omega_y} \quad \frac{\partial V(\mathbf{x})}{\partial \omega_z} \quad \frac{\partial V(\mathbf{x})}{\partial \delta q_1} \quad \frac{\partial V(\mathbf{x})}{\partial \delta q_2} \quad \frac{\partial V(\mathbf{x})}{\partial \delta q_3} \quad \frac{\partial V(\mathbf{x})}{\partial \delta q_4} \right] f(t, \mathbf{x}) \\ &= [\omega_x, \omega_y, \omega_z, 2\delta q_1, 2\delta q_2, 2\delta q_3, 2(\delta q_4 - k_q)] f(t, \mathbf{x}) \\ &= \omega_x u_{v1}(\mathbf{x}, t) + \omega_y u_{v2}(\mathbf{x}, t) + \omega_z u_{v3}(\mathbf{x}, t) + 2\delta q_1 \left[(\omega_x \delta q_4 - \omega_y \delta q_3 + \omega_z \delta q_2) / 2 \right] + \dots \\ &\quad + 2\delta q_2 \left[(\omega_x \delta q_3 + \omega_y \delta q_4 - \omega_z \delta q_1) / 2 \right] + 2\delta q_3 \left[(-\omega_x \delta q_2 + \omega_y \delta q_1 + \omega_z \delta q_4) / 2 \right] + \dots \\ &\quad + 2(\delta q_4 - k_q) \left[(-\omega_x \delta q_1 - \omega_y \delta q_2 - \omega_z \delta q_3) / 2 \right] \\ &= \omega_x u_{v1}(\mathbf{x}, t) + \omega_y u_{v2}(\mathbf{x}, t) + \omega_z u_{v3}(\mathbf{x}, t) + [\omega_x \delta q_1 \delta q_4 - \omega_y \delta q_1 \delta q_3 + \omega_z \delta q_1 \delta q_2] + \dots \\ &\quad + [\omega_x \delta q_2 \delta q_3 + \omega_y \delta q_2 \delta q_4 - \omega_z \delta q_1 \delta q_2] + [-\omega_x \delta q_2 \delta q_3 + \omega_y \delta q_1 \delta q_3 + \omega_z \delta q_3 \delta q_4] + \dots \\ &\quad + [-\omega_x \delta q_1 \delta q_4 - \omega_y \delta q_2 \delta q_4 - \omega_z \delta q_3 \delta q_4] + [k_q \omega_x \delta q_1 + k_q \omega_y \delta q_2 + k_q \omega_z \delta q_3] \end{aligned} \quad (7.34)$$

Factoring angular rates terms:

$$\begin{aligned}\dot{V}(\mathbf{x}, t) = & \omega_x \left[u_{v1}(\mathbf{x}, t) + \delta q_1 \delta q_4 + \delta q_2 \delta q_3 - \delta q_2 \delta q_3 - \delta q_1 \delta q_4 + k_q \delta q_1 \right] \\ & + \omega_y \left[u_{v2}(\mathbf{x}, t) - \delta q_1 \delta q_3 + \delta q_2 \delta q_4 + \delta q_1 \delta q_3 - \delta q_2 \delta q_4 + k_q \delta q_2 \right] \\ & + \omega_z \left[u_{v3}(\mathbf{x}, t) + \delta q_1 \delta q_2 - \delta q_1 \delta q_2 + \delta q_3 \delta q_4 - \delta q_3 \delta q_4 + k_q \delta q_3 \right]\end{aligned}\quad (7.35)$$

Cancelling terms:

$$\dot{V}(\mathbf{x}, t) = \omega_x \left[u_{v1}(\mathbf{x}, t) + k_q \delta q_1 \right] + \omega_y \left[u_{v2}(\mathbf{x}, t) + k_q \delta q_2 \right] + \omega_z \left[u_{v3}(\mathbf{x}, t) + k_q \delta q_3 \right] \quad (7.36)$$

By inspection, the following control law can be established into the virtual controllers so that the Lyapunov function derivative along trajectories is rendered negative semi-definite.

$$\begin{aligned}u_{v1}(\mathbf{x}, t) = & -c_1(t)\omega_x - k_q \delta q_1 - k_1(t)\delta q_1 \text{sign}(\omega_x \delta q_1) \\ u_{v2}(\mathbf{x}, t) = & -c_2(t)\omega_y - k_q \delta q_2 - k_2(t)\delta q_2 \text{sign}(\omega_y \delta q_2) \\ u_{v3}(\mathbf{x}, t) = & -c_3(t)\omega_z - k_q \delta q_3 - k_3(t)\delta q_3 \text{sign}(\omega_z \delta q_3)\end{aligned}\quad (7.37)$$

Applying these control laws and evaluating them in the Lyapunov function:

$$\begin{aligned}\dot{V}(\mathbf{x}, t) = & -c_1(t)\omega_x^2 - c_2(t)\omega_y^2 - c_3(t)\omega_z^2 - k_1(t)\delta q_1 \omega_x \text{sign}(\delta q_1 \omega_x) - k_2(t)\delta q_2 \omega_y \text{sign}(\delta q_2 \omega_y) - k_3(t)\delta q_3 \omega_z \text{sign}(\delta q_3 \omega_z) \\ = & -c_1(t)\omega_x^2 - c_2(t)\omega_y^2 - c_3(t)\omega_z^2 - k_1(t)|\delta q_1 \omega_x| - k_2(t)|\delta q_2 \omega_y| - k_3(t)|\delta q_3 \omega_z|\end{aligned}\quad (7.38)$$

Since $\delta q_i \omega_i \text{sign}(\delta q_i \omega_i) = |\delta q_i \omega_i| \geq 0$ and $\omega_i^2 \geq 0$, all terms in Eq.(7.38) will be sign definite if and only if:

$$c_1(t), c_2(t), c_3(t), k_1(t), k_2(t), k_3(t) \geq 0 \quad (7.39)$$

If this is true then:

$$\dot{V}(\mathbf{x}, t) \leq 0 \quad (7.40)$$

This result proves that the control law renders the closed loop system globally bounded along trajectories; this result is also known as Lagrange Stability.

To prove angular rate global stability it is required to first find a time invariant function $W(\mathbf{x})$ that holds the following inequality:

$$\dot{V}(\mathbf{x}, t) \leq -W(\mathbf{x}) \leq 0 \quad (7.41)$$

To make this search easier it is possible to assume the existence of scalar functions that lower bound the adaptive gains as follows:

$$\begin{aligned} c_1^* &\leq c_1(t), c_2^* \leq c_2(t), c_3^* \leq c_3(t), \quad \forall t \geq 0 \\ k_1^* &\leq k_1(t), k_2^* \leq k_2(t), k_3^* \leq k_3(t) \end{aligned} \quad (7.42)$$

Lyapunov-Barbalat's Lemma can be used to show that $W(\mathbf{x}) \rightarrow 0$ as $t \rightarrow \infty$ (Slotine & Li, 1991) (Balas & Frost, 2014). To use this theorem, $W(\mathbf{x})$ must be uniformly continuous; this is equivalent to show that the time derivative along trajectories of $\dot{W}(\mathbf{x})$ is bounded. Let the function $W(\mathbf{x})$ be defined as follows:

$$W(\mathbf{x}) = c_1^* \omega_x^2 + c_2^* \omega_y^2 + c_3^* \omega_z^2 \quad (7.43)$$

The time derivative along trajectories of $W(\mathbf{x})$ is:

$$\begin{aligned} \frac{dW(\mathbf{x}(t))}{dt} &= 2c_1^* \dot{\omega}_x + 2c_2^* \dot{\omega}_y + 2c_3^* \dot{\omega}_z \\ &= 2c_1^* u_{v1}(\mathbf{x}, t) + 2c_2^* u_{v2}(\mathbf{x}, t) + 2c_3^* u_{v3}(\mathbf{x}, t) \\ &= 2c_1^* [-c_1(t)\omega_x - k_q \delta q_1 - k_1(t)\delta q_1 \text{sign}(\omega_x \delta q_1)] + \dots \\ &\quad + 2c_2^* [-c_2(t)\omega_y - k_q \delta q_2 - k_2(t)\delta q_2 \text{sign}(\omega_y \delta q_2)] + \dots \\ &\quad + 2c_3^* [-c_3(t)\omega_z - k_q \delta q_3 - k_3(t)\delta q_3 \text{sign}(\omega_z \delta q_3)] \end{aligned} \quad (7.44)$$

All the states of this closed loop system will be bounded after proving Lagrange Stability in Eq.(7.41). Moreover if the adaptive gains $c_1(t), c_2(t), c_3(t), k_1(t), k_2(t), k_3(t)$ are also bounded, then by *Lyapunov-Barbalat's Lemma*:

$$W(\mathbf{x}) = c_1^* \omega_x^2 + c_2^* \omega_y^2 + c_3^* \omega_z^2 \rightarrow 0 \quad \text{as } t \rightarrow \infty \quad (7.45)$$

The only way to satisfy this condition is if $[\omega_x(t), \omega_y(t), \omega_z(t)] \rightarrow 0$ as $t \rightarrow \infty$.

Therefore if the adaptive gains are positive, lower and upper bounded then all the state trajectories will be bounded and the angular rates will go to zero as time goes to infinity.

7.3. Proof of Boundedness of Adaptive Augmentation System

The following time varying adaptive gains are inspired by the immune response of the biological organism (Takahashi & Yamada, 1998).

$$\begin{aligned} k_1(t) &= k_q f[\Delta u_{v1}(t)] \eta_{q1} \\ k_2(t) &= k_q f[\Delta u_{v2}(t)] \eta_{q2} \\ k_3(t) &= k_q f[\Delta u_{v3}(t)] \eta_{q3} \end{aligned} \quad (7.46)$$

$$\begin{aligned} c_1(t) &= c_1 \left\{ 1 + \eta_{\omega_x} f[\Delta u_{v1}(t)] \right\} \\ c_2(t) &= c_2 \left\{ 1 + \eta_{\omega_y} f[\Delta u_{v2}(t)] \right\} \\ c_3(t) &= c_3 \left\{ 1 + \eta_{\omega_z} f[\Delta u_{v3}(t)] \right\} \end{aligned} \quad (7.47)$$

where $\left[\eta_{\omega_x} \quad \eta_{\omega_y} \quad \eta_{\omega_z} \quad \eta_{q1} \quad \eta_{q2} \quad \eta_{q3} \right]$ are scalars and the non-linear adaptive base function $f(\Delta u_{v_i}(t))$ ($i=x,y,z$) is defined as:

$$f(\Delta u_{v_i}(t)) = \left[1 - \frac{\mu_i}{e^{\gamma_i [\Delta u_{v_i}(t)]^2} + e^{-\gamma_i [\Delta u_{v_i}(t)]^2}} \right] \quad (7.48)$$

This is a continuous positive definite function that has the following behavior for different values of γ :

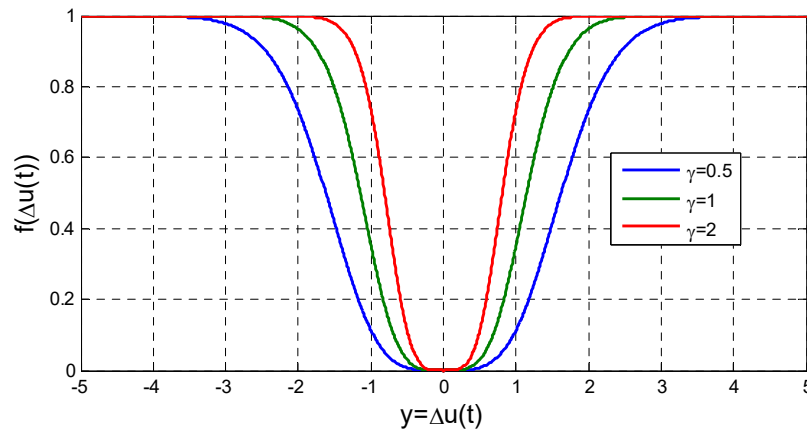


Figure 7.3 Adaptive Base Function.

Additionally, $\Delta u_{v_i}(t)$ will be defined as a time varying signal that incorporates the difference between the actual minus a previous time step virtual control input:

$$\begin{aligned}\Delta u_{v_1}(t) &= u_{v_1}(t) - u_{v_1}(t - \tau) \\ \Delta u_{v_2}(t) &= u_{v_2}(t) - u_{v_2}(t - \tau) \\ \Delta u_{v_3}(t) &= u_{v_3}(t) - u_{v_3}(t - \tau)\end{aligned}\quad (7.49)$$

The adaptive functions defined in Eq.(7.46) and Eq.(7.47) will hold the requirements established in Eq.(7.40) and Eq.(7.43) (positive, lower and upper bounded).

As long as:

$$\begin{aligned}0 < \eta_{q_1} < \infty, \quad 0 < \eta_{\omega_x} < \infty \\ 0 < \eta_{q_2} < \infty, \quad 0 < \eta_{\omega_y} < \infty \\ 0 < \eta_{q_3} < \infty, \quad 0 < \eta_{\omega_z} < \infty\end{aligned}\quad (7.50)$$

After defining the adaptive gains, the final virtual control law will have the form:

$$\begin{aligned}u_{v_1}(\mathbf{x}, t) &= -c_1 \{1 + \eta_{\omega_x} f[\Delta u_{v_1}(t)]\} \omega_x - k_q \{1 + f[\Delta u_{v_1}(t)] \eta_{q_1} \text{sign}(\omega_x \delta q_1)\} \delta q_1 \\ u_{v_2}(\mathbf{x}, t) &= -c_2 \{1 + \eta_{\omega_y} f[\Delta u_{v_2}(t)]\} \omega_y - k_q \{1 + f[\Delta u_{v_2}(t)] \eta_{q_2} \text{sign}(\omega_y \delta q_2)\} \delta q_2 \\ u_{v_3}(\mathbf{x}, t) &= -c_3 \{1 + \eta_{\omega_z} f[\Delta u_{v_3}(t)]\} \omega_z - k_q \{1 + f[\Delta u_{v_3}(t)] \eta_{q_3} \text{sign}(\omega_z \delta q_3)\} \delta q_3\end{aligned}\quad (7.51)$$

which is equivalent to the following more compact result:

$$\begin{aligned}u_{v_1}(\mathbf{x}, t) &= -c_1(t) \omega_x - k_{\delta q_1}(t) \delta q_1 \\ u_{v_2}(\mathbf{x}, t) &= -c_2(t) \omega_y - k_{\delta q_2}(t) \delta q_2 \\ u_{v_3}(\mathbf{x}, t) &= -c_3(t) \omega_z - k_{\delta q_3}(t) \delta q_3\end{aligned}\quad (7.52)$$

where:

$$\begin{aligned}c_1(t) &= \{1 + \eta_{\omega_x} f[\Delta u_{v_1}(t)]\}, \quad k_{\delta q_1}(t) = \{1 + f[\Delta u_{v_1}(t)] \eta_{q_1} \text{sign}(\omega_x \delta q_1)\} \\ c_2(t) &= \{1 + \eta_{\omega_y} f[\Delta u_{v_2}(t)]\}, \quad k_{\delta q_2}(t) = \{1 + f[\Delta u_{v_2}(t)] \eta_{q_2} \text{sign}(\omega_y \delta q_2)\} \\ c_3(t) &= \{1 + \eta_{\omega_z} f[\Delta u_{v_3}(t)]\}, \quad k_{\delta q_3}(t) = \{1 + f[\Delta u_{v_3}(t)] \eta_{q_3} \text{sign}(\omega_z \delta q_3)\}\end{aligned}\quad (7.53)$$

The final control Law can also be conveniently written in vector form as follows:

$$\mathbf{u}_v(\mathbf{x}, t) = -\mathbf{C}(t)\boldsymbol{\omega} - \mathbf{K}_Q(t)\boldsymbol{\delta q}\text{sign}(\boldsymbol{\delta q})\quad (7.54)$$

where:

$$\mathbf{C}(t) = \begin{bmatrix} c_1(t) & 0 & 0 \\ 0 & c_2(t) & 0 \\ 0 & 0 & c_3(t) \end{bmatrix}, \quad \mathbf{K}_Q(t) = \begin{bmatrix} k_{q_1}(t) & 0 & 0 \\ 0 & k_{q_2}(t) & 0 \\ 0 & 0 & k_{q_3}(t) \end{bmatrix} \quad (7.55)$$

It worth noticing that a $\text{sign}(\delta q_4)$ factor was incorporated at the end of the control law to ensure that the fastest path is chosen by the controller to achieve the final desired orientation. Figure 7.4 shows a summarized schematic of the NLDIQ+AIS adaptive

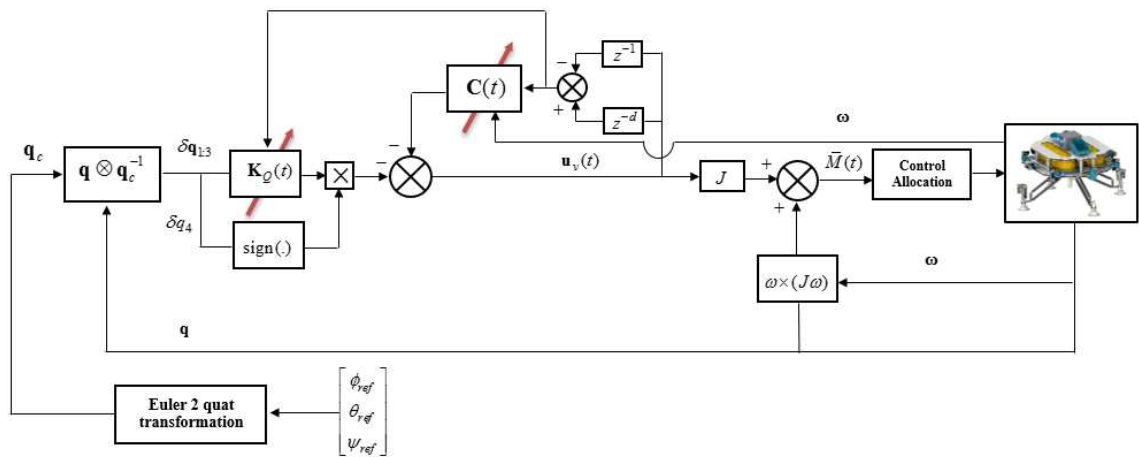


Figure 7.4 Adaptive NLDIQ+AIS Schematic.

8. Implementation of Quaternion Based Adaptation on a Cold Gas Spacecraft

This chapter presents relevant and successful implementation results of the quaternion based immune adaptive control architecture developed in Chapter 7 (NLDI_Q+AIS). The controller was implemented on a concept spacecraft vehicle testbed designed and built at NASA Kennedy Space Center (see Figure 8.1). The concept prototype aims to support the development of novel autonomous prospector space exploration vehicles for in situ resource utilization in environments such as asteroids where gravitational force is minimal.

One of the major goals of this chapter is to demonstrate a preliminary proof of concept of the adaptive configurations presented in Chapter 7. The results obtained intend to evaluate the performance of the developed control laws for missions in which the extreme environment might put the whole system at risk.



Figure 8.1 Gimbaled Asteroid Cold Gas Free Flyer (ACGFF) Prototype.

The spacecraft prototype is mounted on a three degree of freedom gimbaled platform that allows free motion in roll, pitch and yaw axes. The main purpose of this

setup is to demonstrate full attitude control and angular rate regulation in gravity-less environments while tracking pre-defined attitude trajectories and recovering from tumbles or other kind of abnormal conditions that might occur in space.

8.1. Vehicle and Experimental Framework Description

The prototype vehicle is actuated through twelve servo-valve thrusters that regulate the expelling of cold gas from two main reservoir tanks that keep Nitrogen at high pressure. The pressure is regulated by two pressure regulators, Low Pressure 1 (LP1) and Low Pressure 2 (LP2), to achieve a final constant pressure of 130 psi to the solenoids ready for operation. The diagram in Figure 8.2 illustrates the main architecture for the cold gas distribution.

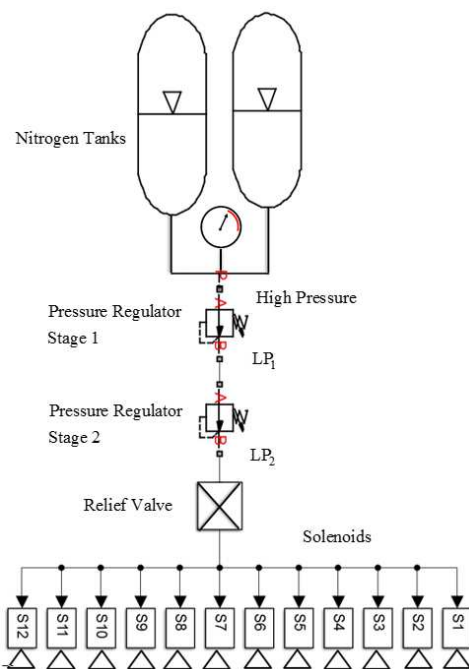


Figure 8.2: Cold Gas System onboard ACGFF.

The opening and closing of the solenoids are regulated by means of Pulse Width Modulated (PWM) signals from the digital IO pins on the onboard computer.

8.1.1. Hardware and Test Setup

The control architectures described in Chapter 7 were first tested in simulation to corroborate proper operation. After confirming that the control laws work properly in simulation, they are deployed into the target flight computer through Simulink-Real time environment. Simulink-Real time is an efficient code generation and prototyping tool from MathWorks. It allows the development of applications directly from Simulink so that they can be tested and run on a dedicated target computer or hardware. The code is generated automatically from Simulink and is compiled into the target machine using a real-time kernel.

The host computer has the capability of downloading the code through a TCP/IP protocol. The flight computer selected for the target is the Athena II from Diamond Systems (see Figure 8.3) that includes serial and analog input modules along with a digital I/O. The digital control signals are used to actuate each of the solenoid valves that regulate the proper amount of gas that each thruster requires for attitude control.



Figure 8.3: Athena II SBC by Diamond Systems®.

The Inertial Measurement Unit (IMU) used in the vehicle is from Microstrain (see Figure 8.4), and is capable of providing accurate measurements of attitude and angular rates (signals required for the controllers). The Microstrain outputs can be read by one of

the serial ports of the Athena II by means of a RS232 communication protocol. Table 8.1 describe some of the main characteristics of the sensors within the Microstrain.

Table 8.1 Characteristics of Microstrain IMU

	Accelerometers	Gyroscopes	Magnetometers
Initial bias	$\pm 0.002 g$	$\pm 0.25^\circ/\text{sec}$	$\pm 0.003 \text{ Gauss}$
Noise density	$80 \mu\text{g}/\sqrt{\text{Hz}}$	$0.03^\circ/\text{sec}/\sqrt{\text{Hz}}$	$100 \mu\text{Gauss}/\sqrt{\text{Hz}}$
Alignment error	$\pm 0.05^\circ$	$\pm 0.05^\circ$	$\pm 0.05^\circ$
Sampling rate	30 kHz	30 kHz	7.5 kHz max



Figure 8.4 GX3-45 IMU by Microstrain

The sensors, actuators, flight computer and additional hardware were tested separately to corroborate full functionality before they were mounted and fully incorporated into the spacecraft prototype shown in Figure 8.1.

The diagram in Figure 8.5 describes the main test framework prepared at Kennedy Space Center at *Swamp Works Laboratories*. The prototype is connected via Wi-Fi to the host computer by a high data rate transmission 5.0 GHz connection. This is achieved by a *Ubiquity Rocket M5* transmission station that is onboard the vehicle. The data transmission rate is crucial for online tuning of the controllers and signal monitoring.

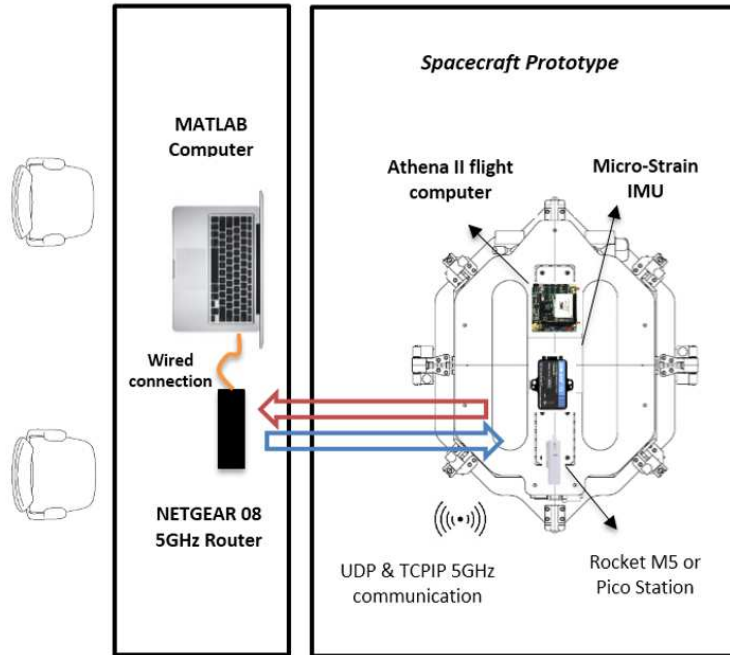


Figure 8.5 Schematic of Test Bed and Hardware Used on the ACGFF Prototype.

8.2. Forces and Moments that Act on ACGFF Spacecraft Prototype

As previously mentioned, there are a total of twelve thrusters in the ACGFF prototype, each of them is capable of supplying a maximum of 1.5 N of force. In order to implement the $NLDI_Q$ and $NLDI_Q+AIS$ controllers, the first step is to develop a set of equations that describe the total forces and moments that act on the prototype due to its specific actuator configuration (see Figure 8.6 - Figure 8.8), the following equation describes the forces and moments in the body frame of the prototype.

$$\sum \vec{M}_b = \begin{bmatrix} L_y(T_1 + T_4) + L_y(T_6 + T_7) - L_y(T_2 + T_3) - L_y(T_5 + T_8) \\ L_x(T_1 + T_2) + L_x(T_7 + T_8) - L_x(T_3 + T_4) - L_x(T_5 + T_6) \\ (T_9 + T_{12})L_{yaw} - (T_{10} + T_{11})L_{yaw} \end{bmatrix} \quad (8.1)$$

$$\sum \vec{F}_b = \begin{bmatrix} T_9 - T_{10} + T_{11} - T_{12} \\ 0 \\ -T_1 - T_2 - T_3 - T_4 + T_5 + T_6 + T_7 + T_8 \end{bmatrix} \quad (8.2)$$

where L_x, L_y, L_{yaw} are the corresponding arm distances from the CG of the vehicle to the actuator locations.

8.3. Control Allocation for ACGFF Spacecraft Prototype

In order to achieve the required moments and forces commanded from the Dynamic Inversion controllers, Eq.(8.1) and Eq.(8.2) must be solved to find the exact thrust required per each actuator. It can be seen that there are a total of six equations for six unknowns, but since there is a total of twelve actuators the system is over actuated. In order to guarantee that a global solution for the forces and moments equations is obtained for all instants of time, a control logic was employed in which each pair of upper and immediately lower thruster is treated as one actuator. If a positive thrust is demanded then the lower thruster is activated; on the other hand if a negative thrust is demanded the upper one is activated. This means that a maximum of six thrusters will be acting at the same time, and therefore there are a total of six equations for six unknowns at every instant of time. The following figures illustrate the actuators that must be turned on for different scenarios.

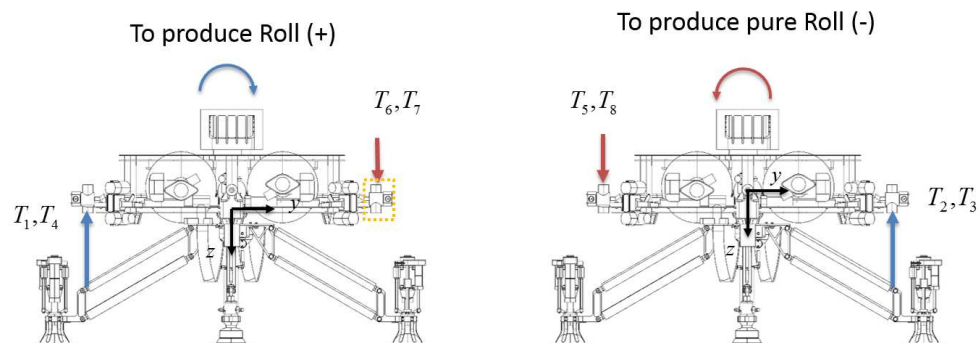


Figure 8.6 Actuators Activated to Produce Roll Motion.

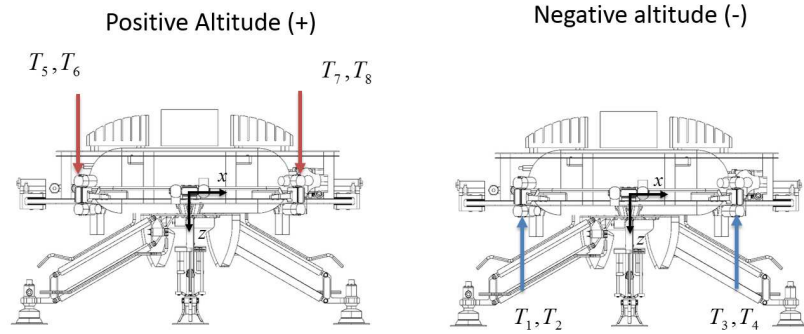


Figure 8.7 Actuators Activated to Produce Altitude Change Motion.

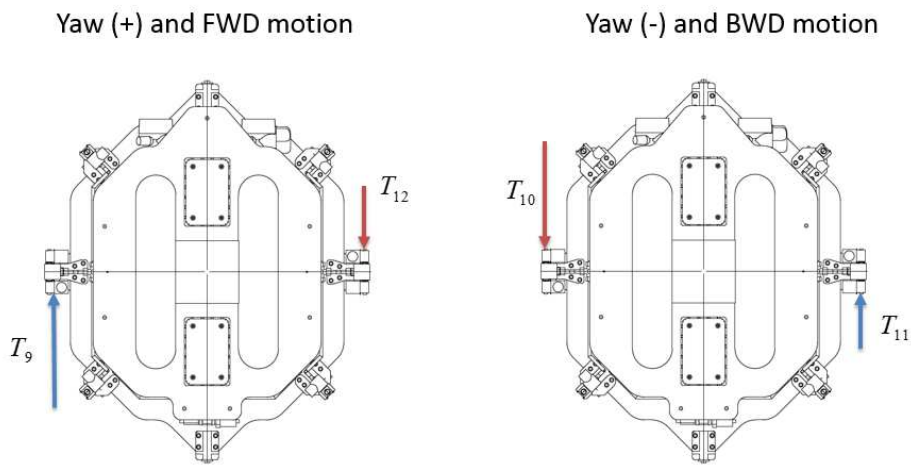


Figure 8.8 Actuators Activated to produce Yaw and Forward Motion.

The exact inverse of required moments to thrust per thruster to achieve any roll or pitching moment can be described by the following set of equations that result from solving Eq. (8.1) and Eq.(8.2):

$$\begin{aligned}
 \frac{F_z}{4} + \frac{M_y}{4L_x} + \frac{M_x}{4L_y} &= \begin{cases} T_5 < 0 \\ T_1 \geq 0 \end{cases} \\
 \frac{F_z}{4} + \frac{M_y}{4L_x} - \frac{M_x}{4L_y} &= \begin{cases} T_6 < 0 \\ T_2 \geq 0 \end{cases} \\
 \frac{F_z}{4} - \frac{M_y}{4L_x} - \frac{M_x}{4L_y} &= \begin{cases} T_7 < 0 \\ T_3 \geq 0 \end{cases} \\
 \frac{F_z}{4} - \frac{M_y}{4L_x} + \frac{M_x}{4L_y} &= \begin{cases} T_8 < 0 \\ T_4 \geq 0 \end{cases}
 \end{aligned} \tag{8.3}$$

To calculate the thrust required per actuator to produce Yaw or Forward/Backward Motion the following equations are obtained:

$$\begin{aligned} \frac{F_x}{2} + \frac{M_z}{2L_z} &= \begin{cases} T_{10} < 0 \\ T_9 \geq 0 \end{cases} \\ \frac{F_x}{2} - \frac{M_z}{2L_z} &= \begin{cases} T_{12} < 0 \\ T_{11} \geq 0 \end{cases} \end{aligned} \quad (8.4)$$

8.4. Definition of Stability and Performance Metrics

In order to have a proper assessment of the different controllers, it is important to establish some qualitative performance criteria that measure how good is the controller in terms of attitude and angular rate tracking error as well as the total activity of the actuators (in this case solenoid valves). This is achieved by recording the time history of different sensors from which tracking error and commanded values can be used for subsequent analysis.

8.4.1. Angular Rate Tracking Error Activity

This performance metric corresponds to the sum of the root mean square of the three angular rate signals.

$$\tilde{\epsilon}_\Omega = \frac{1}{C\Omega} \left(\sqrt{\int_0^T \omega_x^2 dt} + \sqrt{\int_0^T \omega_y^2 dt} + \sqrt{\int_0^T \omega_z^2 dt} \right) \quad (8.5)$$

where $C\Omega$ is a cut-off value used to normalize the resultant index $\tilde{\epsilon}_\Omega$ with respect to the worst case (biggest value) from the set of tests.

8.4.2. Quaternion Tracking Activity

This performance metric corresponds to the root mean square of the total unit quaternion tracking error.

$$\tilde{\delta\mathbf{q}} = \frac{1}{C\Delta\mathbf{Q}} \sqrt{\int_0^T [(1-\delta q_4)^2 + \delta q_1^2 + \delta q_2^2 + \delta q_3^2] dt} \quad (8.6)$$

where $C\Delta\mathbf{Q}$ is a cut-off value used to normalize the resultant index $\tilde{\delta\mathbf{q}}$ with respect to the worst case from the set of tests.

8.4.3. Solenoid Activity

This performance metric corresponds to the total actuation activity of the solenoids. This is calculated as the root mean square of the solenoid PWM commanded signal and is basically the *rms* of the total time the solenoid valves remained open.

$$\tilde{s} = \frac{1}{C\Delta S} \left(\sum_{i=1}^{12} \sqrt{\int_0^T S_i(t) dt} \right) \quad (8.7)$$

where $C\Delta S$ is a cut-off value used to normalize the resultant index \tilde{s} with respect to the worst case from the set of tests.

8.4.4. Global Performance Index

A global performance index can be computed from the previous set of metrics using different weights for each of the indices in Eq.(8.5) - Eq.(8.7) considering relative importance or weight on the study. For that effect, the following expression was used to calculate the global performance index.

$$P_l = 1 - \left[(1/3)\tilde{e}_\omega + (1/3)\tilde{\delta\mathbf{q}} + (1/3)\tilde{s} \right] \quad (8.8)$$

$$\tilde{P}_l = P_l / CP_l$$

where CP_i is a cutoff value to normalize the resultant global performance index with respect to the best nominal case of the set of tests considered. It can be seen that each metric is weighted equally.

8.5. Implementation Results on ACGFF Spacecraft Prototype

This section describes the implementation results of the NLDIQ and NLDIQ+AIS on the ACCG prototype (see Figure 8.1). As mentioned before, the controllers are deployed on the Athena II flight computer through Simulink Real-Time. Different scenarios were tested including induced thruster failures while tracking a predefined trajectory.

8.5.1. Description of Tests Performed

The following scenarios were considered for the tests performed.

Nominal Conditions:

The nominal condition was designed so that the vehicle is able to track the sequence of angles described in Table 8.2. No failures are considered in this scenario.

Table 8.2 Nominal Maneuver

Nominal Maneuver							
Time	0	5	10	15	20	25	30
ϕ (deg)	0	0	0	180	180	180	0
θ (deg)	90	180	180	180	180	180	0
ψ (deg)	0	0	180	180	180	180	0

Abnormal Condition 1:

The first abnormal condition considered consists of a total blockage in thruster 1 five seconds after the test starts. The same maneuver in Table 8.2 was considered for this test.

Abnormal Condition 2:

The second abnormal condition consists of a total blockage of thruster 2 five seconds after the test starts, while the vehicle tracks the same set of angles described in Table 8.2.

8.5.2. Preliminary Implementations of NLDI_Q Cascade Configuration

As previously mentioned, Simulink real-time is the primary tool used to execute the code in the real-time hardware. In order to facilitate the tuning process, a GUI (Guided User Interface) was developed in an extension of Simulink-Real time called Simulink real-time explorer (see Figure 8.9).

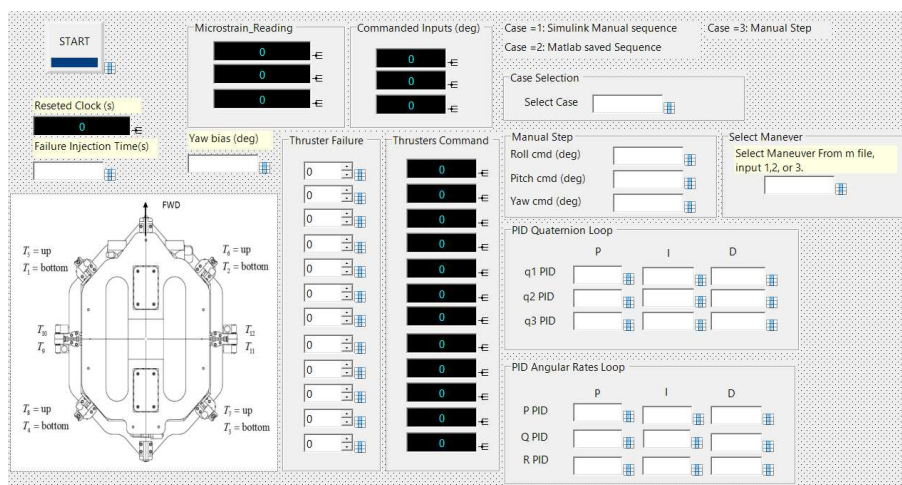


Figure 8.9 GUI Panel for Real Time Code Operation.

This tool enables real-time parameter update within a fast and intuitive interface that is directly linked to the code being executed in the target. Thus, parameters such as control gains can be tuned in real time without requiring to recompile the code. It also supports real-time monitoring of signals (specified by the user) and other capabilities such as manual switches, gauges monitors, among others.

Figure 8.10 and Figure 8.11 show quaternion and angular rate tracking results for tracking a nominal maneuver using the **NLDI_Q** in cascade configuration. It is worth recalling that this initial implementation was for a maneuver different than the one specified in Table 8.2.

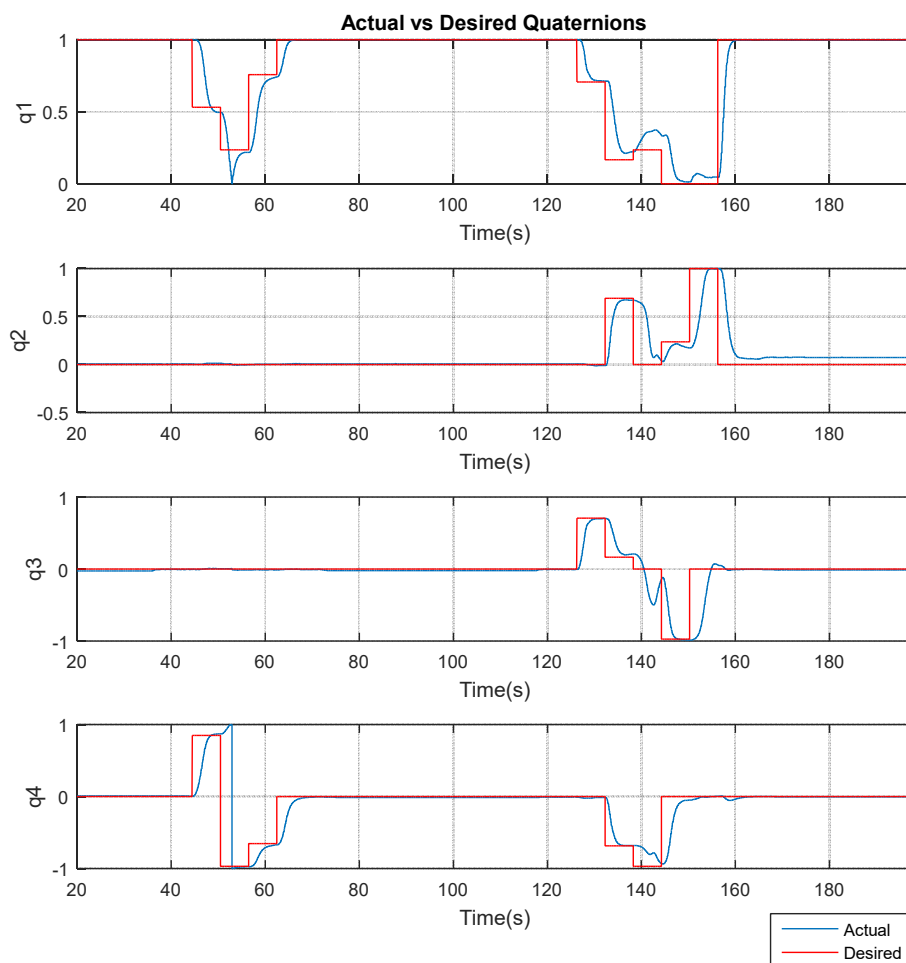


Figure 8.10 Implementation Results for NLDI_Q Cascade for Quaternion Tracking.

From Figure 8.10 it can be seen that the tracking behavior of the four quaternions for a nominal maneuver is quite successful, even though in this configuration no adaptation was engaged and no abnormal condition was imposed into the system.

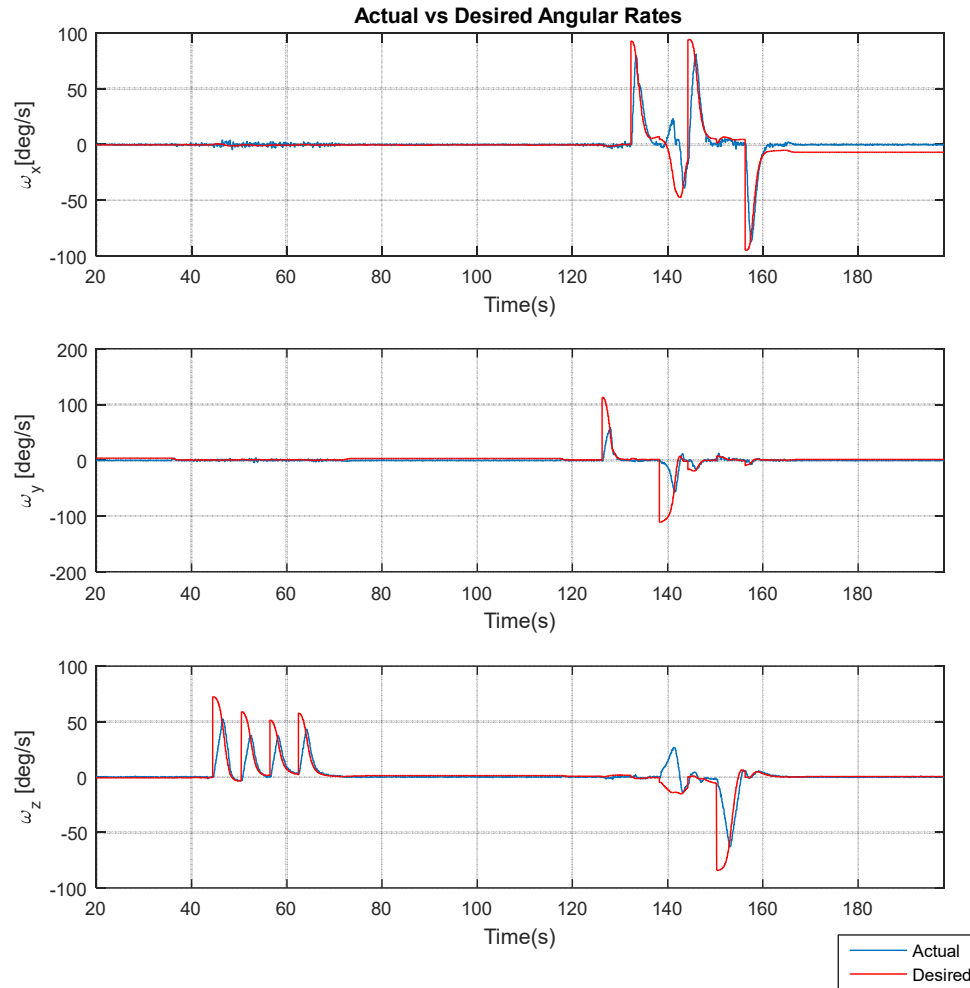


Figure 8.11 Angular Rate Tracking Performance for NLDIQ Cascade.

Figure 8.11 depicts the angular rate response of the inner loop of the NLDIQ cascade control architecture. The tracking performance for ω_{xref} is better than the tracking performance for ω_{yref} . The tracking performance for ω_{zref} is satisfactory although it has a noticeable lag with respect to the desired input.

8.5.3. Implementation Results for Nominal Conditions

Figure 8.12 presents a direct comparison between the $NLDI_Q$ and the $NLDI_Q+AIS$ that incorporates adaptive augmentation. No failure was injected during this test.

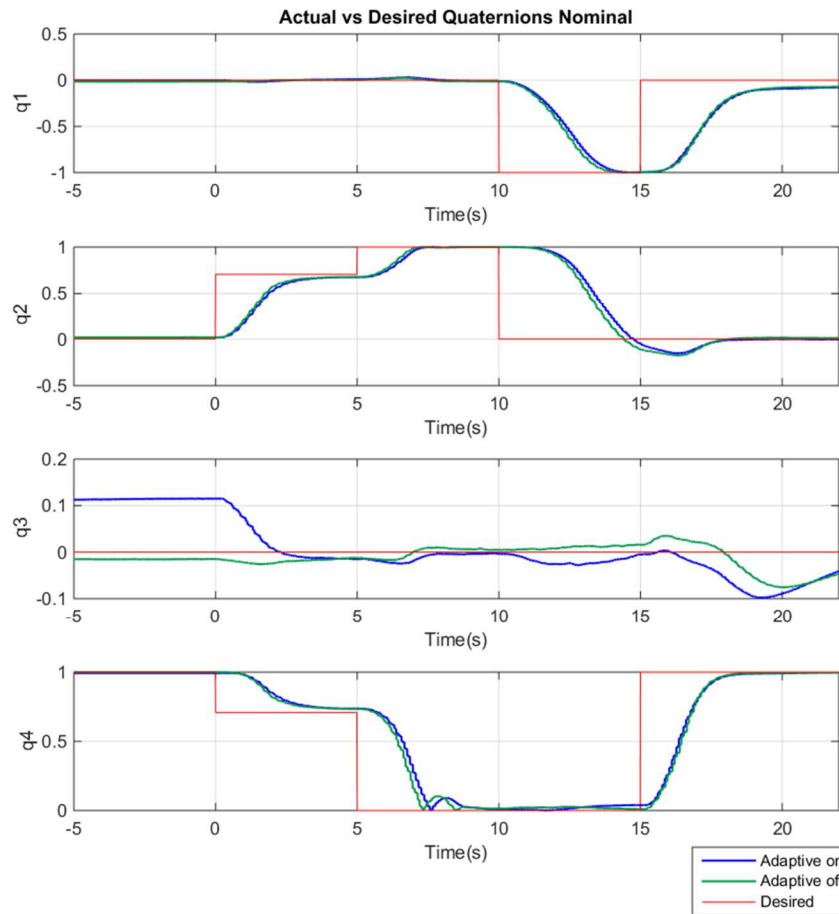


Figure 8.12 Implementation Tracking Performance for Nominal Conditions.

From Figure 8.12 it can be seen that the tracking performance is very similar for both controllers. Figure 8.13 - Figure 8.15 present the performance metrics time history during the maneuver. The results corroborate what is seen in the quaternion tracking behavior since the angular rate, quaternion and solenoid activities are all very similar for both control architectures.

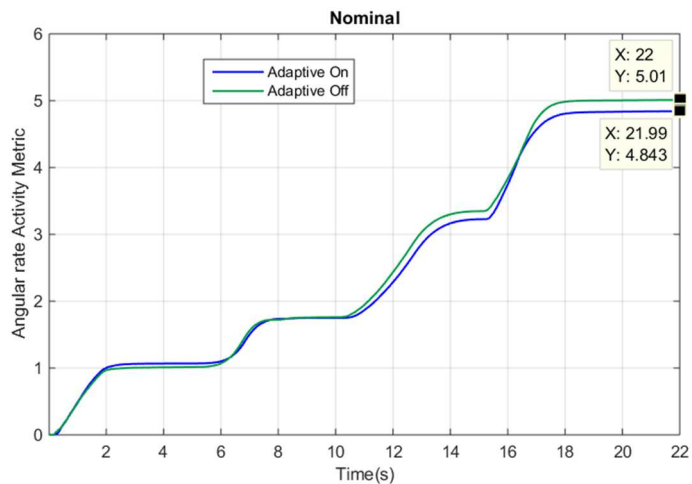


Figure 8.13 Angular Rate Activity Metric at Nominal Conditions.

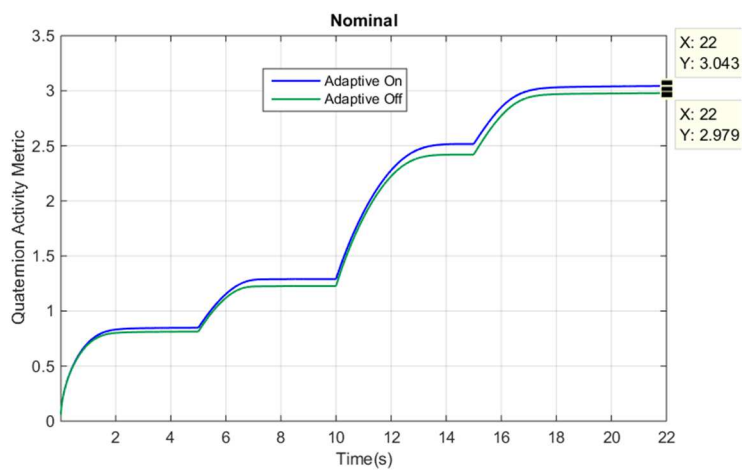


Figure 8.14 Quaternion Activity Metric at Nominal Conditions.

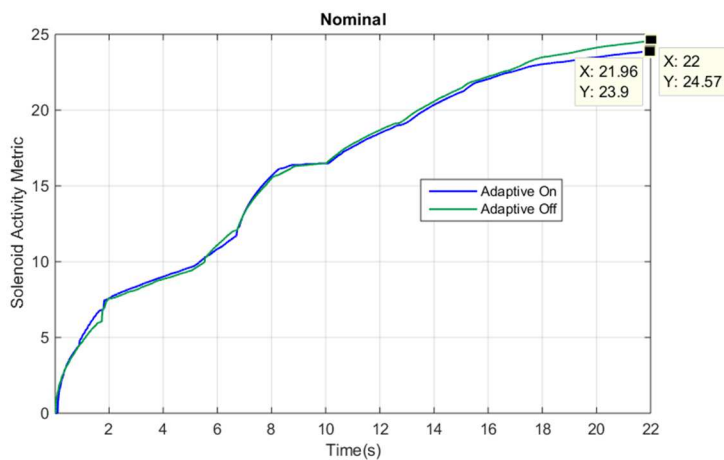


Figure 8.15 Total Solenoid Activity Performance Metric at Nominal Conditions.

8.5.4. Implementation Results for Abnormal Conditions

Figure 8.16 presents a direct comparison between the $NLDI_Q$ and the $NLDI_Q+AIS$ that incorporates adaptive augmentation when thruster two is fully blocked five seconds after the test starts.

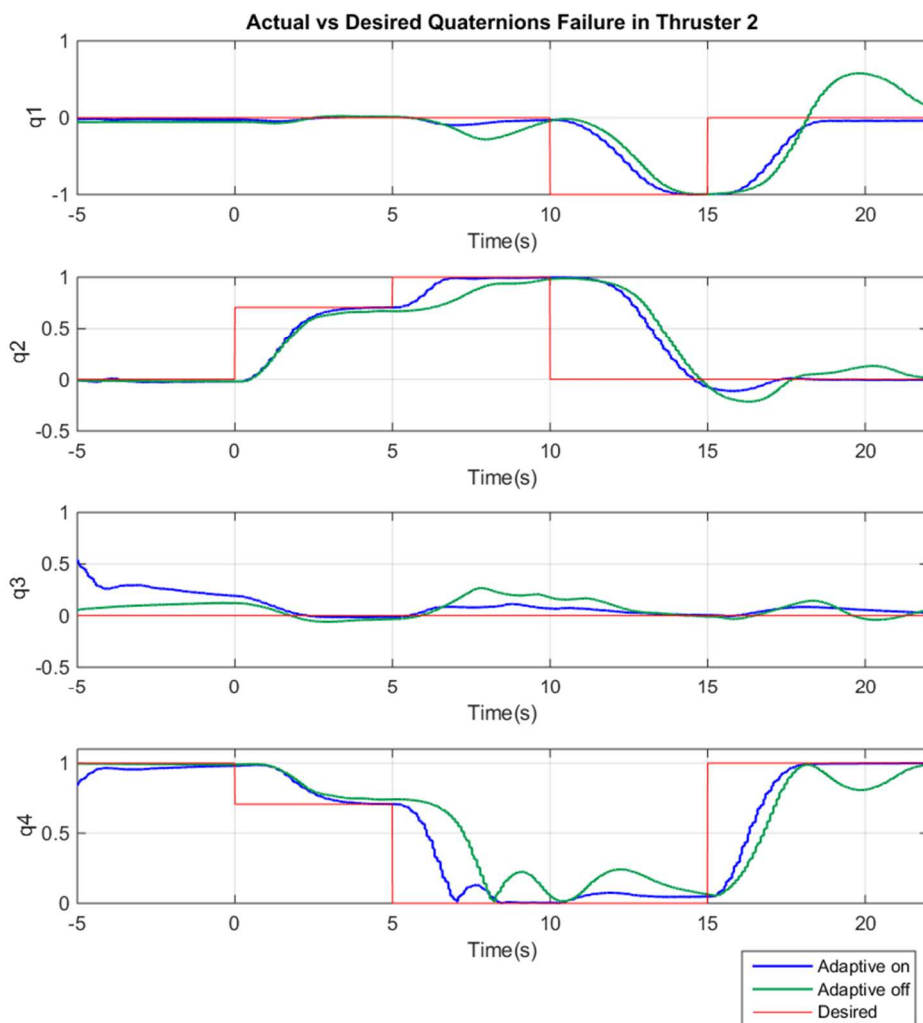


Figure 8.16 Implementation Tracking Performance for Abnormal Conditions.

From Figure 8.16 it is appreciated that the adaptive augmentation helps the system to behave much better in terms of tracking error in the presence of abnormal condition. This can be confirmed if we check the overall quaternion tracking

performance; a noticeable overshoot can be seen for δq_1 in Figure 8.16 for the controller that does not incorporate the adaptive augmentation. Similar results can be appreciated after checking Figure 8.17- Figure 8.19, which correspond to the performance metrics of both controllers at abnormal conditions.

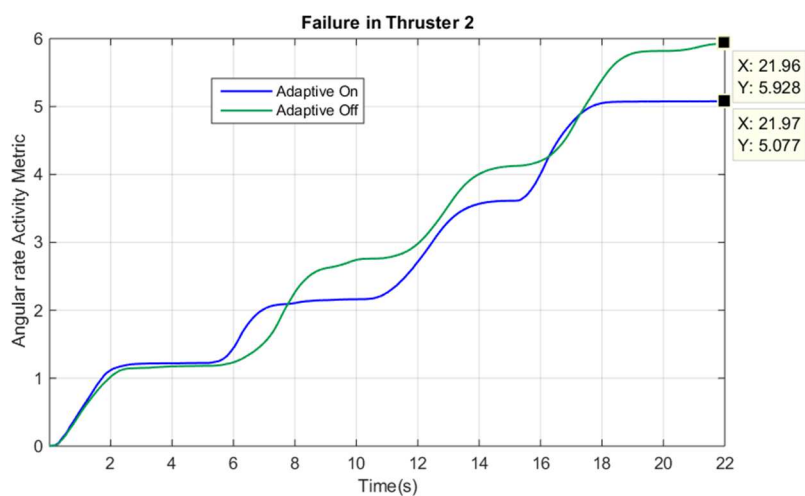


Figure 8.17 Angular Rate Activity Metric for Thruster 2 Failure.

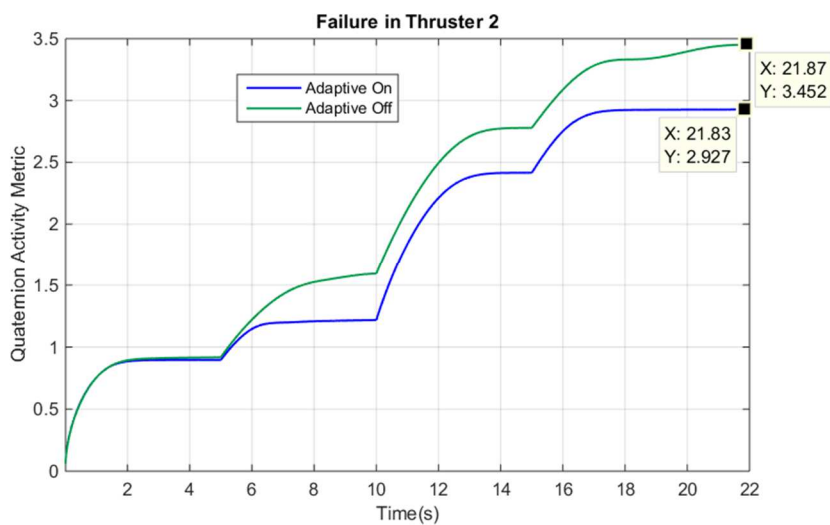


Figure 8.18 Quaternion Activity Metric for Thruster 2 Failure.

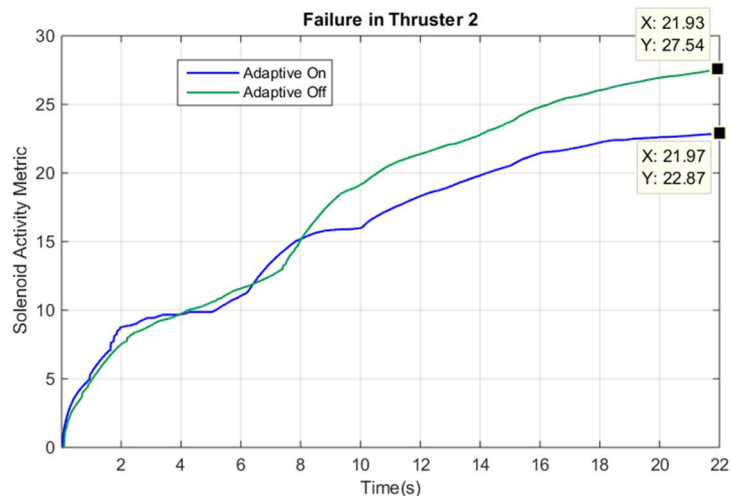


Figure 8.19 Solenoid Activity Metric for Thruster 2 Failure.

From the figures it is possible to conclude that the adaptive augmentation successfully mitigates the failure by reducing the total amount of error while keeping the solenoid activity at low values.

Figure 8.20 and 8.21 present the resultant adaptive gains for this specific test.

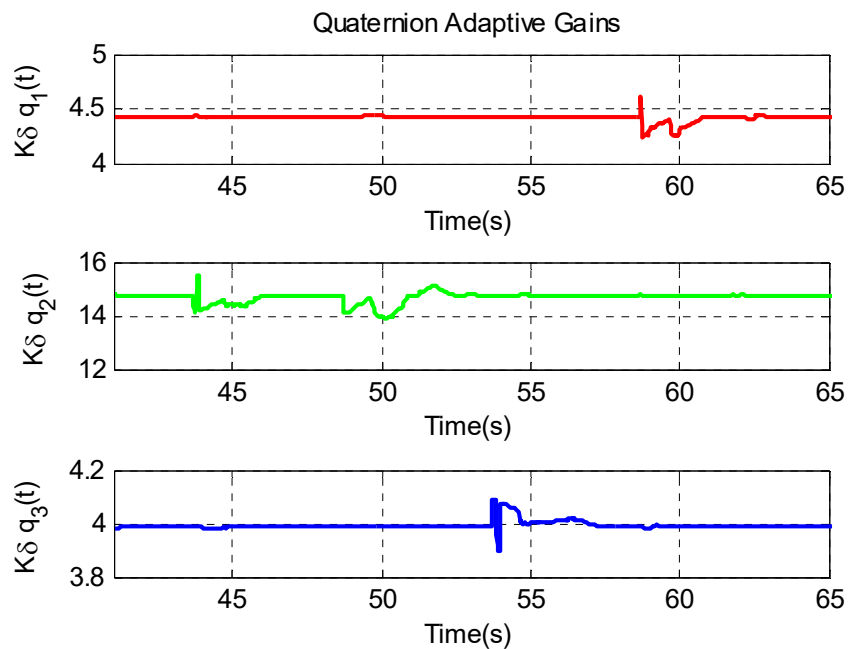


Figure 8.20 Quaternion Adaptive Gains for Thruster 2 Failure.

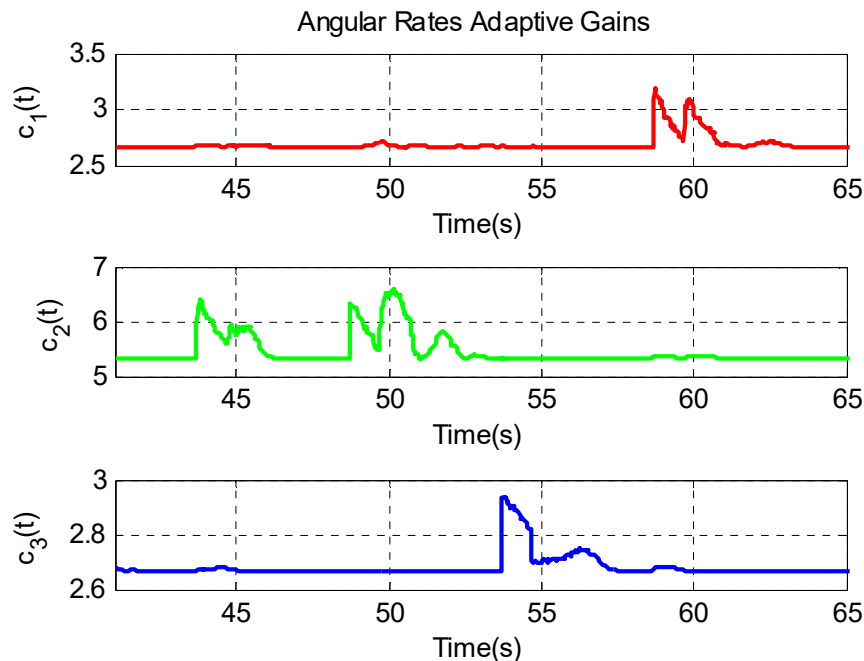


Figure 8.21 Angular Rate Adaptive Gains for Thruster 2 Failure.

8.5.5. Global Performance Controller Assessment

In order to check for consistency in the results 15 tests were performed with at least three tests per condition. Table 8.3 presents the all the cases tested and Table 8.4 shows the average of the global performance index per each case per controller architecture.

Table 8.3 Tests Performed

	NLDI _Q	NLDI _Q +AIS
Nominal	Test 1	Test 7
	Test 2	Test 8
	Test 3	Test 9
T1 Failure	Test 4	Test 10
	Test 5	Test 11
	Test 6	Test 12
T2 Failure	Test 7	Test 13
	Test 8	Test 14
	Test 9	Test 15

Table 8.4 Average Performance Metrics and Performance Index per Case.

	Nominal		T1 Failure		T2 Failure	
	NLDI _Q	NLDI _Q +AIS	NLDI	NLDI _Q +AIS	NLDI _Q	NLDI _Q +AIS
Ang. Rates Activity	5.121	4.839	5.615	5.025	5.558	5.125
Quaternion Activity	2.968	3.034	3.238	2.989	3.022	2.974
Solenoid Activity	24.552	24.182	26.112	23.028	27.923	23.963
P. Index	0.818	0.938	0.182	1.000	0.215	0.869

The same information in Table 8.4 can be shown in Figure 8.22 in which the histogram compares the average global performance index (see Eq.(8.8)) for each case considered.

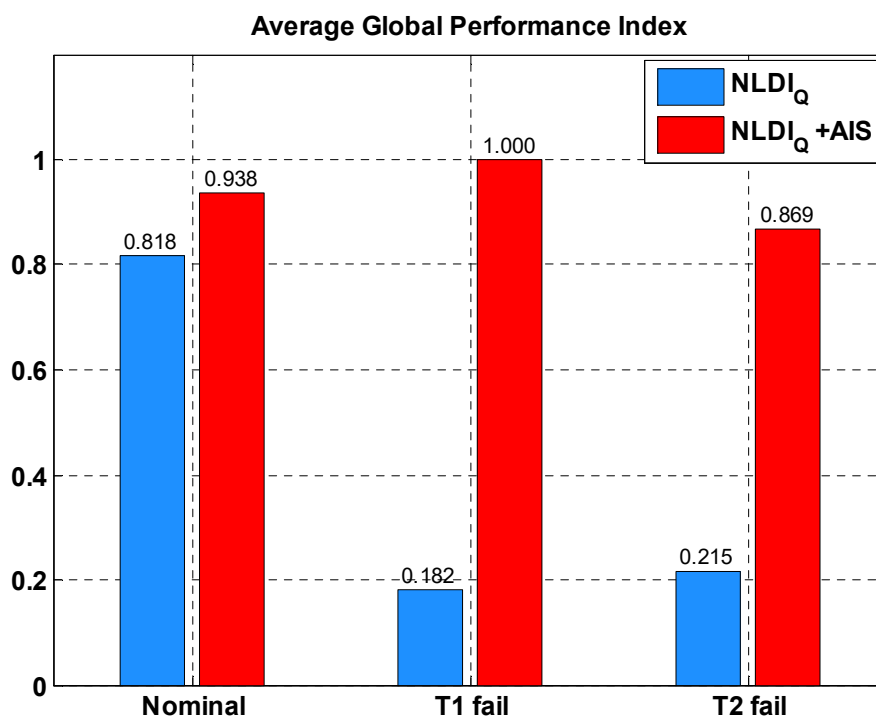


Figure 8.22 Global Performance Index Histogram Comparison.

9. Conclusions and Recommendations

This dissertation covers the development of two main novel adaptive control architectures designed to mitigate failures and abnormal conditions on aerospace systems. Both control laws include adaptive functions that resemble some characteristics present in the response of the immune system of living organisms. The first configuration was designed for aerospace systems that do not attain large angles (i.e. aircraft), and it relies on Euler Angle attitude kinematics. The second adaptive configuration approach relies mainly on quaternion kinematics so that it can be incorporated in aerospace systems that can attain large attitude angles or maneuvers. This chapter covers some key conclusions about these main results and how they could be improved in future work or research.

9.1. Conclusions on Euler Angle Based Adaptive Controller

The main structure of the Euler Angle MRAIS control architecture was developed in Chapter 4 (Sections 4.6 – 4.8). It relies on an exact feedback linearization approach in conjunction with a model reference adaptive controller inspired by a feedback adaptive function that is known to be present in the immune system of living organisms. One of the most relevant contributions of this approach is the development of a solid theoretical framework and proof of absolute stability by means of the Circle Criterion. Based on the author's best knowledge, there has been no prior attempt to link adaptive control theory with the Circle Criterion approach. Furthermore, a proof of robust stability based on the Lyapunov approach is developed in Section 4.8. The results obtained show that the proposed adaptive configuration makes the closed loop system stable even in the presence of bounded and persistent disturbances. The adaptive configuration was successfully tested in simulation, HIL and a preliminary implementation on the MFF

prototype and a quadrotor (see appendix A). From the results presented in Chapter 6 there is a marked improvement on stability and robustness after augmenting the baseline NLDI controller with the bio-inspired adaptive approach. The adaptive augmentation is able to successfully mitigate failures and enhance the overall stability performance of the system under different abnormal conditions. However, for the real case implementations on board the MFF and quadrotor it was noticeable that the system had some more conservative limits in the adaptive power gain before it became unstable. It is important to highlight that for real world implementation, some basic assumptions on the proves developed in Chapter 4 are not applicable. For example, for the stability analysis, the system is assumed to be continuous in time. However, the implementation in hardware requires discretization of the control laws which may behave differently than its continuous counterpart. This can be perceived whenever the sample and execution times are not faster than the fastest dynamic modes of the system. Increasing the adaptive power gain can increase the overall response time of the system; this can increase the frequency of the dynamic modes of the system and as a result the gap between the fastest modes and discretization sample time is reduced. This means that one source of instability could be this discretization process of the control laws.

Another source of instability can be the delay between the control system input and the actuator response. Although this is contemplated in the simulations, it is nearly impossible to consider the actuator dynamics within the proof of stability without loss of generality. A possible future work would be including the effect of actuator time response in the stability proves developed in this dissertation. This might provide more realistic gain bounds to maintain stable operation of the system.

9.2. Quaternion Based Adaptive Controller

As mentioned earlier, the second novel adaptive configuration approach relies mainly on quaternion kinematics and its main goal is to be incorporated in aerospace systems that can attain large attitude angles. A formal proof of boundedness of the quaternion error states and stability of angular rates was presented in Sections 7.2 and 7.3 using the Lyapunov control design technique and Barbalat's Lemma. This adaptive configuration was successfully implemented into the ACGFF (see Chapter 8). The results show that the adaptive augmentation yields improved tracking performance in the presence of thruster failures. It was noticed that the performance of the adaptive controller in failure conditions was actually better than the baseline controller in the nominal case. These results were validated more than once for consistency, which means that the adaptation really improves the overall performance of the system. One interesting thing that was noticed from the results is that the adaptive augmentation reduced the response time without compromising the overall overshoot of the system, something that a linear controller can't achieve.

One possible way to enhance the quaternion based adaptation architecture, that could be an interesting topic for future work, is to include a model reference adaptation similar to the one developed in Chapter 4 instead of the non-model dependence approach adopted for the quaternion configuration. This means that instead of using the change between current and past inputs as the adaptive parameter used by the immune adaptive function, the difference between the actual control input and the input to an idealized model reference plant that runs online could be used. This approach might be a more reliable way to incorporate adaptation since it is well known that MRAC control can

provide enhanced robustness by always comparing the response of the actual system respect to an idealized model response. The main challenge with this new possible approach would be to determine if the proof of stability is not affected by using a model reference plant.

9.3. Additional Remarks

Additional to the recommendations outlined in the previous sections, another possible way to enhance the performance of the closed loop system, augmented by means of bio-inspired adaptive control, would be to include some sort of dynamic feedback within the actual configurations developed in this research. So far the adaptation laws are dynamic and time varying in the sense that they change and accommodate when abnormal conditions occur; this can be clearly seen in most of the results of Chapter 6. However it is worth mentioning that the approach developed in this dissertation does not rely on integrators or differential equations that govern the dynamics of the adaptive gains as opposed to well-known adaptive algorithms. Therefore, incorporating somehow a dynamic feedback, perhaps by using an adaptive integral gain, could be an appropriate way to improve the system performance and robustness to uncertainties.

Another possible way to enhance the MRAIS adaptive controller developed in this dissertation is to allow that the adaptive gains increase and decrease in both directions. So far the adaptive gains can increase or decrease but can never go down from a minimum value, this is due to the original form of the immune system adaptive function and stability restrictions given by Lyapunov's theory. However, based on the resultant adaptive control law for the quaternion case (see Eq.(7.38)) in which two sided adaptation was achieved adding a product of $\text{sign}(\delta q_i \omega_i)$ with the corresponding baseline

immune adaptive gain. This additional modification was required to satisfy Lyapunov stability criteria, therefore if this approach was possible for the quaternion architecture there is a good chance that it is also possible to incorporate an additional modification to the MRAIS controller including a $\text{sign}(\cdot)$ function to allow two sided adaptation.

A. Implementation Results on 3DR-X8 Quadrotor

The MRAIS adaptive architecture was also implemented on a quadrotor while performing an autonomous tracking maneuver with abnormal conditions. The failure consists of a 34% and 37% limited actuation on motors 1 and 4 respectively as shown in Figure A.1.

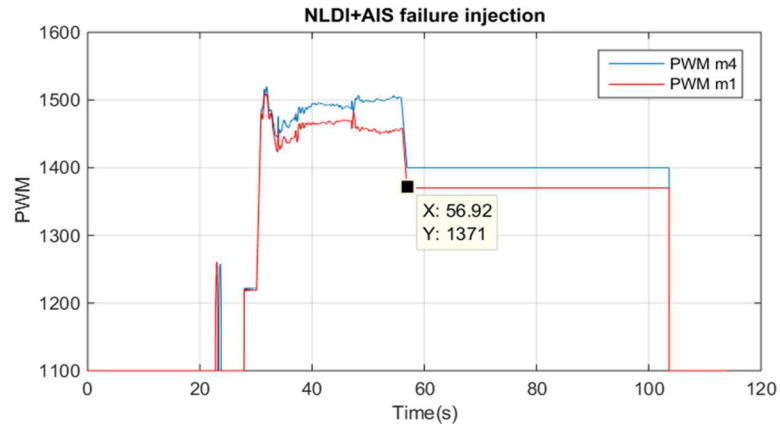


Figure A.1 Failure Injection Profile.

The adaptation control is then compared with respect to the baseline controller; the tracking performance under failure is shown in Figure A.2.

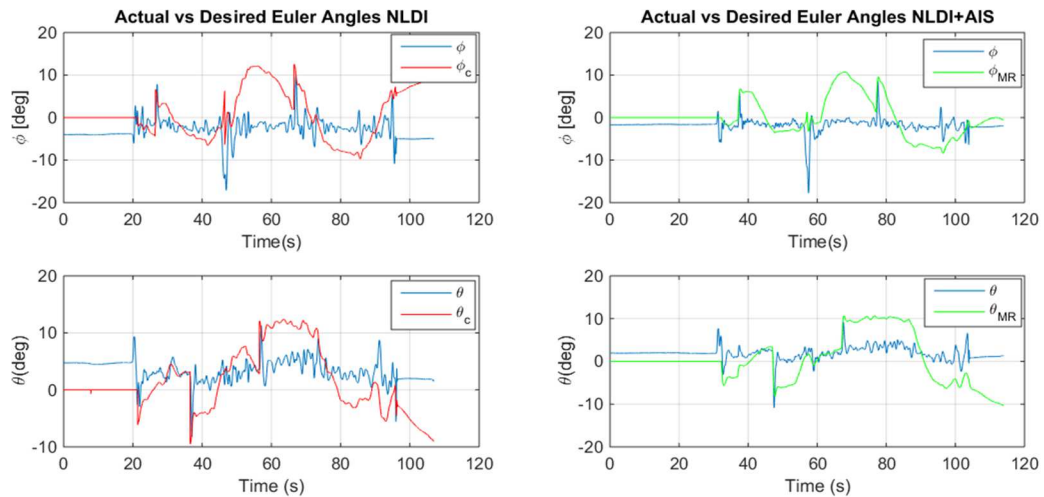


Figure A.2 Euler Angle Tracking Error.

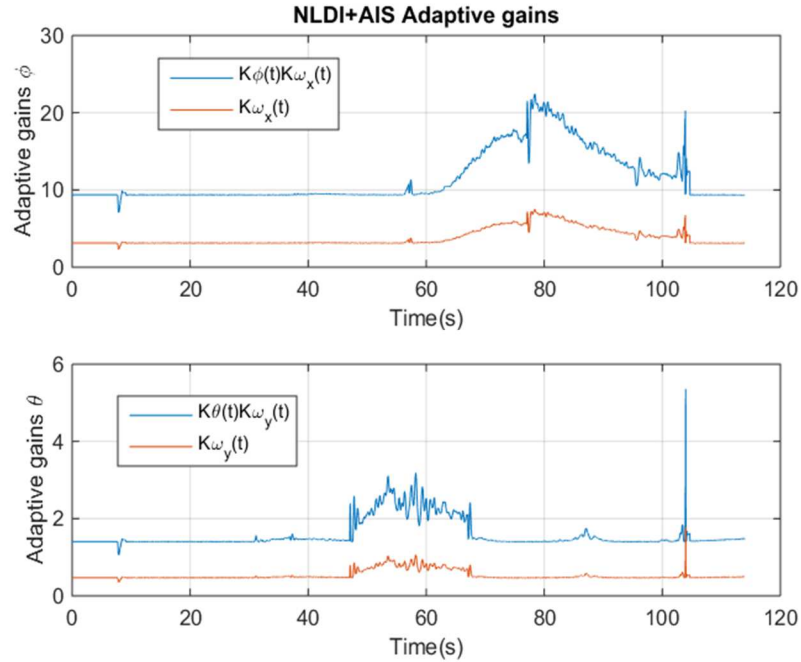


Figure A.3 Roll and Pitch Adaptive Gains.

To analyze the tracking performance of both controllers, the root mean square tracking error was calculated for the roll and pitch channels during the duration of the trajectory. Table A.1 shows the resultant tracking *rms* errors per case.

Table A.1 Tracking Error Activity for X8 Quadrotor Results.

	Failure M1 and M4	
	NLDI	NLDI+MRAIS
$rms e_{\omega_x}$ (deg/s)	22.49	8.92
$rms e_{\omega_y}$ (deg/s)	14.03	6.13
$rms e_{\phi}$ (deg)	7.20	6.79
$rms e_{\theta}$ (deg)	4.47	5.68
Total	48.19	27.51

From Table A.1 it can be seen that the total tracking error of the adaptive augmentation is considerably less than the total tracking error of the NLDI without the augmentation.

10. References

- Acquatella, P., Falkena, W., Van Kampen, E.-J., & Chu, Q. P. (2012). Robust Nonlinear Spacecraft Attitude Control using Incremental Nonlinear Dynamic Inversion. *AIAA Guidance, Navigation, and Control Conference* (pp. 1-20). Delft: American Institute of Aeronautics and Astronautics, Inc.
- Antoniewicz, R. F., Duke, E. L., & Patterson, B. P. (1988). *User's manual for Interactive LINEAR, a Fortran Program to Derive Aircraft Models*. Edwards, California: NASA Technical Paper 2835.
- Balas, M. J., & Fuentes, R. (2004). A Non-Orthogonal Projection Approach to Characterization of Almost Positive Real Systems with an Application to Adaptive Control. *American Control Conference*. IEEE.
- Balas, M. J., & Nelson, J. P. (2011). A New Robustness Theorem for the Adaptive Control of Nonlinear Systems. *Conference on Smart Materials, Adaptive Structures and Intelligent Systems*. Scottsdale: ASME.
- Balas, M., & Frost, S. (2014). Robust Adaptive Model Tracking for Distributed Parameter Control of Linear Infinite-dimensional Systems in Hilbert Space. *IEEE/CAA Journal of Automatica Sinica*, 1-8.
- Bang, H., ShinLee, J., & JuEun, Y. (2004). Nonlinear Attitude Control for a Rigid Spacecraft by Feedback Linearization. *KSME International Journal*, 203-210.
- Belcastro, C. M., & Jacobson, S. R. (2010). Future Integrated Systems Concept for Preventin Aircraft Loss-of-Control Accidetsns. *Proceedings of AIAA Guidance, Navigation and Control Conference*. Toronto, Canada.
- Bell, E. V. (2003, April 6). *Phobos: Doomed Moon of Mars*. Retrieved from Nasa Atronomy Picture of the day: <https://apod.nasa.gov/apod/ap030406.html>
- Benjamini, E. (1992). *Immunology, A Short Course*. New York: Wiley-List Publications.
- Brophy, J. R., Friedman, L., Strange, N. J., Prince, T. A., Landau, D., Jones, T., . . . Manzella, D. (2014). Synergies of Robotic Asteroid Redirection Technologies and Human Space Exploration. *International Astronautical Congress*. Toronto.
- Castro, L. N., & Von Zuben, F. J. (2001). An Artificial Immune Network for Data Analysis. In *Data Mining a Heuristic Approach* (pp. 231-259). Hershey, PA: Idea Group Publising.
- Chen, W., & Wei, H. (2006). Compensatory Controller Based on Artificial Immune System. *Proceedings of the 2006 IEEE International Conference on Mechatronics and Automation*. Luoyang, China.
- Crassidis, J. L., & Markley, L. (2014). *Fundamentals of Spacecraft Attitude Determination and Control*. New York: Springer.
- Dasgupta, D. (1999). *Artificial Immune Systems and Their Applications*. New York: Springer.
- Davis, J., Perhinschi, M. G., & Moncayo, H. (2009). Evolutionary Algorithm for

- Artificial Immune System-Based Failure-Detector Generation and Optimization. *Proceedings of the AIAA Guidance, Navigation and Control Conference*. Chicago, Illinois. doi:10.2514/1.46126
- Edwards, C., Lombaerts, T., & Smaili, H. (2010). *Fault Tolerant Flight Control: A Benchmark Challenge*. Berlin Heidelberg: Springer Verlag.
- Falkena, W., van Oort, E. R., & Chu, Q. P. (2001). Towards Certifiable Advanced Flying Control Systems, A Sensor Based Backstepping Approach. *AIAA Guidance Navigation and Control*. Portland, Oregon.
- Frederick, R. (1999). *Martian Lava Tubes*. Retrieved from Oregon Moonbase Images: <http://www.oregon15.org/mist/15ltms99.html>
- Fuentes, R. J., & Balas, M. J. (2000). Disturbance Accomodation for a Class of Tracking Control Systems. *Guidance, Navigation and Control*. Denver: AIAA .
- Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Boston: Addison-Wesley Longman Publishing Co.
- Gutnikov, S., & Melnikov, Y. (2003). A Simple Non-Linear Model of Immune Response. *Chaos, Solitons & Fractals*, 16(1), 125-132. doi:10.1016/S0960-0779(02)00232-1
- Harvey, S. A. (2008). Spacecraft Attitude Control using Direct Model Reference Adaptive Control, Ph.D. Dissertation. Laramie: University of Wyoming.
- Huang Jinying, A., Ma Bo, B., & Wang Haojing, C. (2009). Design of Vehicle Speed Controller Based on Immune Feedback . *FUZZ-IEEE*. Korea.
- Ioannou, P. A., & Sun, J. (1995). *Robust Adaptive Control*. Los Angeles CA: Prentice-Hall.
- Ireland, M. L., Vargas, A., & Anderson, D. (2015). A Comparison of Closed-Loop Performance of Multirotor Configurations Using Non-Linear Dynamic Inversion Control. *Aerospace*, 325.
- Ito, D., Georgie, J., Valasek, J., & Ward, D. (2001). *Re-Entry Vehicle Flight Controls Design Guidelines: Dynamic Inversion, Final Technical Report, NAG9-1085, Flight Simulation Laboratory, Texas Engineering Experiment Station*,. Texas A&M University.
- Jacklin, S. A. (2008). Closing the Certification Gaps in Adaptive Flight Control Software. *AIAA Guidance Navigation and Control Conference*. Honolulu, Hawaii.
- Jie, W., & Jiong, S. (2009). The design of the Superheated-Steam Temperature Immune Control System and its BIBO Stability Analysis. *Cybernetics and Systems: An International Journal*, 40(4), 310-325.
- Johansen, T. A., & Fossen, T. I. (2013). Control Allocation - A survey. *Automatica*, 49(5), 1087-1103. doi:10.1016/j.automatica.2013.01.035
- Kaufman, H., Barkana, I., & Sobel, K. (1997). *Direct Adaptive Control Algorithms Theory and Applications*. New York: Springer-Verlag.
- Khalil, H. K. (1996). *Nonlinear Systems*. New Jersey: Prentice Hall.

- Kim, P. S., Levy, D., & Lee, P. P. (2009). Modeling and Simulation of the Immune System as a Self Regulating Network. In *Methods in Enzymology Vol 467* (pp. 79-109). Elsevier Inc. doi:10.1016/S0076-6879(09)67004-X
- Krstic, M., Kanellakopoulos, I., & Kokotovic, P. (1995). *Nonlinear and Adaptive Control Design*. New York: John Wiley & Sons.
- Kumar, K. K. (2003). Artificial Immune System Approaches for Aerospace Applications. *AIAA-2003-0457, Proceedings of the 41st Aerospace Sciences Meeting & Exhibit*. Reno, Nevada.
- Liao, X., & Yu, P. (2008). *Absolute Stability of Nonlinear Control Systems (Mathematical Modelling: Theory and Applications)*. Springer.
- Lyons, B., Moncayo, H., Noriega, A., Moguel, I., & Perhinschi, M. G. (2013). Hardware in the Loop Simulation of an Extended Non-linear Dynamic Inversion Augmented with an Immunity-based Adaptive Control System. *Proceedings of the AIAA Modeling and Simulation Technologies (MST) Conference*. Boston, MA.
- Meyer, A., & Van Kampen, H. (2002). X-Plane OnLine Instruction Manual 8th edition. Columbia, South Carolina.
- Mitra, P., & Venayagamoorthy, G. K. (2008). Artificial Immune System Based DSTATCOM control for an Electric Ship Power System. *IEEE Electronics Specialists Conference*. Rhodes.
- Mo, H. (2008). Handbook of Research on Artificial Immune Systems and Natural Computing. Medical Information Science.
- Mo, H., Fu, D., & Xu, L. (2014). Research of a Kind of Improved Immune Controller Based Immune Network. *International Journal of Intelligent Computing and Cybernetics*, 3(2), 310-333.
- Moguel, I., Moncayo, H., Perez, A., Perhinschi, M., Al Azzawi, D., & Togayev, A. (2014). Bio-Inspired Approach for Aircraft Health Assessment and Flight Envelope Estimation. *ASME DSCC* (p. 10). San Antonio Texas: ASME.
- Moncayo, H., & Perhinschi, M. G. (2011). *Aircraft Fault tolerance: A Biologically Inspired Immune Framework for Sub-System Failures*. Saarbruecken, Germany: VDM Verlag Dr. Müller GmbH & Co. KG, VDM Publishing House ltd.
- Moncayo, H., & Perhinschi, M. G. (2012). UAV adaptive control laws using non-linear dynamic inversion augmented with an Immunity-based Mechanism. *Proceedings of the AIAA Guidance, Navigation and Control Conference*. Minneapolis, Minnesota.
- Moncayo, H., Krishnamoorthy, K., Wilburn, B., Wilburn, J., Perhinschi, M. G., & Lyons, B. (2013). Performance Analysis of Fault Tolerant UAV Baseline Control Laws with L1 Adaptive Augmentation. *Journal of Modeling, Simulation, Identification and Control*, 1(4), 137-163.
- Moncayo, H., Perhinschi, M. G., Wilburn, B., & Wilburn, J. (2012). UAV Adaptive Control Laws using Non-linear Dynamic Inversion Augmented with an Immunity Based Mechanism. *Proceedings of the AIAA Guidance, Navigation and Control*

- Conference*. Mineapolis, Minesota: Proceedings of the AIAA Guidance, Navigation and Control Conference.
- Moncayo, H., Perhinschi, M., & Davis, J. (2011). Artificial Immune System Based Aircraft Failure Evaluation over Extended Flight Envelope. *Journal of Guidance, Control and Dynamics*, 34(4), 989-1001.
- Moncayo, H., Perhinschi, M., Wilburn, B., Davis, J., & Ondrej, K. (2012). Extended Nonlinear Dynamic Inversion Control Laws for Unmanned Air Vehicles . *AIAA Guidance, Navigation and Control Conference*. Minneapolis.
- Napolitano, M. R. (2012). *Aircraft Dynamics From Modeling to Simulation*. Hoboken, NJ : Wiley.
- Narendra, K. S., & Annaswamy, A. M. (2005). *Stable Adaptive Systems*. New Jersey: Prentice Hall.
- Nelson, J. P., & Balas, M. J. (2012). Model Reference Adaptive Control of Spacecraft for a PNP Satellite with Unknown Time Varying Input/Output Delays. *IEEE*.
- Nguyen, N., & Boskovic, J. D. (2008). Bounded Linear Stability Margin Analysis of Nonlinear Hybrid Adaptive Control. *Proceedings of American Control Conference*. Seattle, Washington, USA.
- Passino, K. M., & Yurkovich, S. (1998). *Fuzzy Control*. Menlo Park: Addison-Wesley Longman.
- Perelson, A. S. (1997). Immunology for Physicists. *Review of Modern Physics*, 69(4), 1219-1266.
- Perez, A. E., Moncayo, H., & Prazenica, R. J. (2016). Control Laws Development for a Free-Flying Unmanned Robotic System to Support Interplanetary Bodies Prospecting and Characterization Missions. *AIAA Guidance, Navigation, and Control Conference* (pp. 1-23). San Diego, CA: American Institute of Aeronautics and Astronautics.
- Perez, A. E., Moncayo, H., Moguel, I., Perhischi, M. G., Al Azzawi, D., & Togayev, A. (2014). Development of Immunity Based Adaptive Control Laws for Aircraft Fault Tolerance. *ASME Dynamic Systems and Control Conference*. San Antonio Texas.
- Perez, A. E., Moncayo, H., Perhinschi, M., Al Azzawi, D., & Togayev, A. (2015). A Bio-Inspired Adaptive Control Compensation System for an Aircraft Outside Bounds of Nominal Design. *ASME Journal of Dynamic Systems, Meaurement, and Control*, 137.
- Perez, A., Moncayo, H., Prazenica, R., Kern, Z., Zacny, C., Mueller, R., . . . DuPuis, M. (2016). Free-Flying Robotic System for Interplanetary Prospecting and In Situ Resource Utilization. *ASCE Earth and Space Conference*. Orlando.
- Perhinschi, M. G., & Beamer, F. (July, September, 2012). Flight Simulation Environment for Undergraduate Education in Aircraft Health Management. *Computers in Education Journal*, XXII(3), 50-62.

- Perhinschi, M. G., Moncayo, H., & Al Azzawi, D. (2013). Development of Immunity-Based Framework for Aircraft Abnormal Conditions Detection, Identification, Evaluation and Accomodation. *Proceedings of the AIAA Guidance, Navigation and Control (GNC) Conference*. Boston, MA.
- Perhinschi, M. G., Moncayo, H., Wilburn, B., Wilburn, J., Karas, O., & Bartlett, A. (2014). Neurally Augmented Immunity Based Detection and Identification of Aircraft Sub-System Failures. *The Aeronautical Journal*, 118(1205), 775-796.
- Perhinschi, M. G., Napollitano, M. R., Campa, G., & Fravolini, M. L. (2004). A simulation Environment for Testing and Research of Neurally Augmented Fault Tolerant Control Laws Based on Non-Linear Dynamic Inversion. *Proceedings of the AIAA Modeling and Simulation Technologies Conference*. Providence, Rhode Island.
- Perhinschi, M. G., Napollitano, M., & Campa, G. (2008, December). A Simulation Environment for Design and Testing of Aircraft Adaptive Fault-Tolerant Control System. *Aircraft Engineering and Aerospace Technology: An International Journal*, 80(6), 620-632.
- Perhinschi, M., Napolitano, M. R., Campa, G., & Fravolini, M. L. (2003). Integration of Fault Tolerant System for Sensor and Actuator Failures Within the WVU NASA F-15 Simulator. *Proceedings of the AIAA Guidance, Navigation and Control Conference*. Austin, TX: AIAA.
- Siceloff, S. (2015, July 30). *Extreme Access Flyer to Take Planetary Exploration Airborne*. Retrieved from NASA's Kennedy Space Center: <https://www.nasa.gov/feature/extreme-access-flyer-to-take-planetary-exploration-airborne>
- Sidi, M. J. (1997). *Spacecraft Dynamics and Control A Practical Engineering Approach*. New York: Cambridge University Press.
- Sieberling, S., Mulde, J., & Chu, Q. (2010, December). Robust Flight Control Using Incremental Nonlinear Dynamic Inversion and Angular Acceleration Prediction. *Journal of Guidance Control and Dynamics*, 33, 1732-1742.
- Slotine, J.-J., & Li, W. (1991). *Applied Non-Linear Control*. New Jersey: Prentice Hall, New Jersey.
- Snell, A. (1998, August). Decoupling Control Design with Applications to Flight. *Journal of Guidance Control and Dynamics*, 21(4), 647-655.
- Song, H., Fang, B., & Wang, P. (2009). Research and Applications of Immune PID Adaptive Controller in Anti-skid Braking System for Aircraft. *Proceedings of the Information Engineering and Computer Science, ICIECS International Conference*. Wuhan, China.
- Sun, Y., & Xu, C. (2010). Nonlinear PID Controller Based on Fuzzy Immune Method and Its Simulation in Super heater Steam Temperature. *Proceedings of the 2010 IEEE International Conference on Mechatronics and Automation*, (pp. 799 - 803). Xi'an, China.

- Sundararajan, N., Sartchandran, P., & Li, Y. (2002). *Fully Tuned Radial Basis Functions Neural Networks for Flight Control*. Boston: Springer.
- Takahashi, K., & Yamada, T. (1998). Application of an Immune Feedback Mechanism to Control Systems. *JSME International Journal*, 41(Series C), 184-191.
- Taylor, L. W., & Adkins, E. J. (1965). Adaptive Control and the X-15. *Princeton University Conference on Aircraft Flying Qualities*. New Jersey: Princeton.
- Wang, L., & Zhang, Z. :. (2014). Adaptive Dynamic Inversion Controller for Quadrotor Aerial Robot. *Journal of Control Engineering and Technology (JCET)*, 4, 147-152.
- Wang, L., He, Y., Zhang, Z., & He, C. (2013). Trajectory Tracking of Quadrotor Aerial Robot Using Improved Dynamic Inversion Method. *Intelligent Control and Automation*, 4(4), 343-348. doi:10.4236/ica.2013.44040
- Wie, B. (2015). Space Vehicle Guidance, Control and Astrodynamics. *AIAA Educational Series*.
- Wie, B., & Barba, P. M. (1985). Quaternion Feedback for Spacecraft Large Angle Maneuvers. *Journal of Guidance, Control, and Dynamics*, 360-365.
- Wilburn, B., & Perhinschi, M. G. (2014). A Modified Genetic Algorithm for UAV Trajectory Tracking Control Laws Optimization. *International Journal of Intelligent Unmanned Systems*, 2(2), 58-90.
- Yang, Y. (2012). Spacecraft Attitude Determination and Control: Quaterion based Method. *Annaul Reviews in Control*, 198-219.
- Yechout, T. R. (2003). Introduction to aircraft flight mechanics: performance, static stability, dynamic stability and classical feedback control. Reston: AIAA education series.
- Yu, L., Cai, Z., Jiang, Z., & Hu, Q. (2007). An Advanced Fuzzy Immune PID-type Tracking Controller of a Nonholonomic Mobile Robot. *Proceedings of the IEEE International Conference on Automation and Logistics*. Jinan, China.
- Yuan, R., Guoliang, F., Yi, J., & Yu, W. (2009). Robust Attitude Controller for Unmanned Aerial Vehicle Using Dynamic Inversion and Extended State Observer. *Second International Conference on Intelligent Computation Technology and Automation*, (pp. 850-853). Beijing, China.
- Zhao, G., Shen, Y., & Zhang, L. (2013). Research on CNC MACHINE Tool Fuzzy Immune PID Position Controller. *Research Journal of Applied Science, Engineering and Technology, Maxwell Scientific Organization*, 212, 209.