Performance and Robustness Assessment for a Robust Port-Hamiltonian Flight Controller

Samuel B. Widman, 2nd Lt, USAF

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Aerospace Engineering

Craig A. Woolsey, Chair
Mark L. Psiaki
Andrea L'Afflitto

20 November 2025
Blacksburg, Virginia

Keywords: Flight Control, Robustness, Simulation, Port-Hamiltonian, Feedback Linearization

Performance and Robustness Assessment for a Robust Port-Hamiltonian Flight Controller

Samuel B. Widman, 2nd Lt, USAF

(ABSTRACT)

Those conducting research in the field of nonlinear control of unmanned air vehicles are constantly searching to improve robustness and performance of flight control systems through new control laws. A control law can be designed to provide robustness guarantees due to the structure of the aircraft dynamics. This work presents an implementation of a novel control law which simultaneously transforms the nonlinear fixed-wing aircraft dynamics into a port-Hamiltonian structure using feedback linearization, from which input-to-state stability guarantees follow. This novel control law is compared to two industry standard methods, linear quadratic regulator and nonlinear dynamic inversion, which provide a baseline for comparing robustness and performance. To replicate flight, measurement noise, model mismatch, wind, and discretization with time delay were implemented in a collection of simulation studies to understand which disturbances the novel control law was most sensitive to. Due to the non-additive nature and magnitude of the applied disturbances, the novel control law was most sensitive to combinations of disturbances of wind, discretization with time delay, model mismatch, and measurement noise in order of greatest to least sensitivity. The novel control law performed as expected, and much better than both competitors, when the disturbances applied did not include wind. This result was due to a particular interaction between the wind disturbance and the construction of the novel control law that was not present with other disturbances. Future work includes flight testing and extending the robustness guarantees to non-feedback linearized systems.

Performance and Robustness Assessment for a Robust Port-Hamiltonian Flight Controller

Samuel B. Widman, 2nd Lt, USAF

(GENERAL AUDIENCE ABSTRACT)

A control law is a mathematical rule used to cause a physical system to follow a user-defined reference input. Many control laws exist for different systems, and each control law has inherent properties that make it better or worse for a given application. Two metrics used to determine if a control law is effective are tracking performance and robustness. Tracking performance is a measure of how closely a control law can follow a user-defined reference input, while robustness is a measure of how much a system can be disturbed from a nominal condition without significant degradation of performance. This work presents the implementation and analysis of a new control law applied to an unmanned aerial vehicle that is designed to provide a high level of tracking performance and robustness guarantees that ensure tracking performance degrades gracefully under the effect of disturbances. The new control law is compared to two other industry standard control laws in terms of both performance and robustness onboard a computer-simulated aircraft. A variety of realistic disturbances are applied to the aircraft, and the results demonstrate that the new control law is robust to certain disturbances and not to others.

Dedication

To Julia,
who is always there when I need her,
and who always brightens my day.

Acknowledgments

There are a lot of people who got me to where I am today, and to whom I am forever grateful.

I'd like to thank Dr. Craig Woolsey for his support and guidance during this short program. He provided a plan, contingencies, and advice on how to navigate the challenges that arose. I am also very appreciative of Dr. Mark Psiaki and Dr. Andrea L'Afflitto, who both teach excellent graduate courses and provided useful feedback during this program.

The members of the Nonlinear Systems Lab as a whole were crucial to helping me get to this point. Ian Willebeek-LeMair took me in from the beginning and guided me through the end, while filling in gaps in my knowledge the whole time. He was my primary collaborator during this program. Dr. Jeremy Hopwood and Dennis Marquis provided expertise in areas that I severely lacked, especially in regards to flight testing. Without their help, I would not have been as successful. Patrick Corrigan, Garrett Asper, Mitchell Farah, Abhishek Jeyaprakas, Vishal Gautam, and all the other folks provided great company and discussions, constant instruction in proper use of hardware and software, and an excellent work environment. Liam Carey and Stephen Savvides spent many hours with me working on coursework for difficult classes.

My parents are the best role models I could have asked for. They provided everything that I needed growing up, included showing what having character, integrity, reliability, and so much more was all about, while also showing me how to be interested in everything, how to serve to others, and how to give your best at everything you do. My siblings, both older, helped guide me through life in ways they probably don't know. They paved the path and always support their little brother.

All of my mentors and teachers have helped me on both personal and professional levels. I have learned so much from them, more than just the material they taught me - how I can be a better person, friend, peer, husband, and so much more.

Lastly, my amazing wife Julia supported me through all the ups and downs of this program. I am very grateful for her encouragement, her kindness, and for always taking care of me when I didn't have time to. I love you.

Contents

List of Figures viii

List of Tables x

1 Introduction 1

2 Background 3
2.1 Math Preliminaries . . . 3
2.2 Flight Mechanics . . . 6
2.3 Robustness . . . 13
2.3.1 Disturbances . . . 13
2.3.2 Robustness for Linear Systems . . . 15
2.3.3 Robustness for Nonlinear Systems . . . 18

3 Control Methods 20
3.1 Linear Quadratic Regulator . . . 21
3.1.1 Overview . . . 21
3.1.2 Guarantees . . . 21
3.1.3 Derivation . . . 22
3.1.4 Implementation . . . 23
3.2 Nonlinear Dynamic Inversion . . . 27
3.2.1 Overview . . . 27
3.2.2 Guarantees . . . 28
3.2.3 Derivation . . . 29
3.2.4 Implementation . . . 31
3.3 Robust Port-Hamiltonian Control . . . 41
3.3.1 Overview . . . 41
3.3.2 Guarantees . . . 42
3.3.3 Derivation . . . 42
3.3.4 Implementation . . . 45

4 Method of Analysis 52
4.1 Control Law Comparison . . . 52
4.2 Simulation Set Up . . . 54
4.3 Disturbance Implementation . . . 56
4.3.1 Measurement Noise . . . 56
4.3.2 Model Mismatch . . . 57

4.3.3 Wind . . . 59
4.3.4 Constant-hold Control Discretization with Time Delay . . . 61

5 Results and Discussion **64**
5.1 Combined Disturbances . . . 65
5.2 Selected Disturbance Combinations . . . 68
5.2.1 Group 1: Least Significant Disturbances . . . 68
5.2.2 Group 2: Intermediately Significant Disturbances . . . 72
5.2.3 Group 3: Most Significant Disturbances . . . 75
5.3 Summary . . . 80

6 Conclusions **84**

Bibliography **86**

Appendices **92**

Appendix A Simulation Guidance **93**
A.1 Simulation Parameters . . . 93
A.2 Disturbance Parameters . . . 96
A.3 Control Law Implementation . . . 96
A.3.1 Linear Quadratic Regulator . . . 96
A.3.2 Nonlinear Dynamic Inversion . . . 98
A.3.3 Robust Port-Hamiltonian Control . . . 101

Appendix B L_2 Norm Results **102**

List of Figures

2.1 MTD research aircraft 12
2.2 Example Bode plot, used with permission from Dr. Craig Woolsey 16

3.1 Example of LQR control law implementation (a) Linear output tracking without disturbances, (b) Nonlinear output tracking without disturbances, and (c) Nonlinear output tracking with exogenous disturbances. 25
3.2 Example of exogenous disturbances on LQR: Applied exogenous disturbance (left), and experienced disturbance (right). 26
3.3 Example of NDI implementation (a) Linear output tracking without disturbances, (b) Nonlinear output tracking without disturbances, and (c) Nonlinear output tracking with exogenous and model mismatch disturbances. . . 39
3.4 Example of exogenous disturbances and model mismatch on NDI: Applied exogenous disturbance (left), and experienced disturbance (right). 40
3.5 Example of RPH implementation (a) Linear output tracking without disturbances, (b) Nonlinear output tracking without disturbances, and (c) Nonlinear output tracking with exogenous and model mismatch disturbances. . . 50
3.6 Example of exogenous disturbances and model mismatch on RPH: Applied exogenous disturbance (left), and experienced disturbance (right). 51

4.1 Example of ideal output tracking. 53
4.2 Signal flow diagram 55
4.3 Example of noise histogram from measurement data. 57
4.4 (a) Example of model mismatch error histogram, and (b) Example of model mismatch error rate histogram. 57
4.5 (a) Example of applied model mismatch for C_X from flight data, (b) Example of applied model mismatch for C_Z from flight data, (c) Example of applied model mismatch for C_X by choice, and (d) Example of applied model mismatch for C_Z by choice. 58
4.6 Examples of applied wind. 62
4.7 Example of Control Discretization for LQR: (a) Continuous Control, (b) Control sampled at 5 Hz, and (c) Control sampled at 5 Hz with up to 200 ms delay. 63

5.1 L_∞ norm for all disturbances. 65
5.2 L_1 norm for all disturbances. 67
5.3 Failure for all disturbances. 67
5.4 L_∞ norm results for (a) discretization with time delay only, (b) measurement noise only, and (c) measurement noise and discretization with time delay. . . 69

5.5 L_1 norm results for (a) discretization with time delay only, (b) measurement noise only, and (c) measurement noise and discretization with time delay. . . 70
5.6 Failure results for (a) discretization with time delay only, (b) measurement noise only, and (c) measurement noise and discretization with time delay. . . 71
5.7 L_∞ norm results for (a) model mismatch only, (b) model mismatch and discretization with time delay, (c) measurement noise and model mismatch, and (d) measurement noise, model mismatch, and discretization with time delay. 72
5.8 L_1 norm results for (a) model mismatch only, (b) model mismatch and discretization with time delay, (c) measurement noise and model mismatch, and (d) measurement noise, model mismatch, and discretization with time delay. 73
5.9 Failure results for (a) model mismatch only, (b) model mismatch and discretization with time delay, (c) measurement noise and model mismatch, and (d) measurement noise, model mismatch, and discretization with time delay. 74
5.10 L_∞ norm results for (a) wind only, (b) wind and discretization with time delay (c) model mismatch and wind, (d) model mismatch, wind, and discretization with time delay, (e) measurement noise and wind, (f) measurement noise, wind, and discretization with time delay, (g) measurement noise, model mismatch and wind, and (h) all combined disturbances. 77
5.11 L_1 norm results for (a) wind only, (b) wind and discretization with time delay (c) model mismatch and wind, (d) model mismatch, wind, and discretization with time delay, (e) measurement noise and wind, (f) measurement noise, wind, and discretization with time delay, (g) measurement noise, model mismatch and wind, and (h) all combined disturbances. 78
5.12 Failure results for (a) wind only, (b) wind and discretization with time delay (c) model mismatch and wind, (d) model mismatch, wind, and discretization with time delay, (e) measurement noise and wind, (f) measurement noise, wind, and discretization with time delay, (g) measurement noise, model mismatch and wind, and (h) all combined disturbances. 79
5.13 Time history of $(\boldsymbol{x} - \boldsymbol{x}_{\text{ref}})$ for $v_0 = 5$ m/s 82
5.14 Initial transient in longitudinal velocity increment, $u - u_{\text{ref}}$, for various initial lateral velocities, v_0. 83

B.1 L_2 norm results for (a) discretization with time delay only, (b) measurement noise only, and (c) measurement noise and discretization with time delay. . . 103
B.2 L_2 norm results for (a) model mismatch only, (b) model mismatch and discretization with time delay, (c) measurement noise and model mismatch, and (d) measurement noise, model mismatch, and discretization with time delay. 103
B.3 L_2 norm results for (a) wind only, (b) wind and discretization with time delay (c) model mismatch and wind, (d) model mismatch, wind, and discretization with time delay, (e) measurement noise and wind, (f) measurement noise, wind, and discretization with time delay, (g) measurement noise, model mismatch and wind, and (h) all combined disturbances. 104

List of Tables

A.1 Initial random state ranges 93
A.2 MTD mass and geometric properties 94
A.3 Aerodynamic model parameters 95
A.4 Measurement noise standard deviation. 95
A.5 Model mismatch standard deviation of the error and error rates. 95
A.6 LQR control law trim conditions 97

List of Abbreviations

ISS Input-to-State Stable/Stability

LQR[1] Linear Quadratic Regulator

MTD My Twin Dream aircraft

NDI[1] Nonlinear Dynamic Inversion

NSL Nonlinear Systems Lab

RPH[1] Robust Port-Hamiltonian

SLUF Straight, Level, Unaccelerated Flight condition

VT Virginia Tech

[1]This may refer to both the control method and the resulting control law as implied by context

Chapter 1

Introduction

To leverage the agility of advanced unmanned aircraft, there is a need for control design methods that provide robust, effective performance over a large domain. The motion of a fixed-wing aircraft is defined by coupled, nonlinear, ordinary differential equations. Using the fixed-wing aircraft's nonlinear dynamic structure to design a control law can improve the system's performance and robustness compared to other control design strategies. This is called model-based control, where the control law is designed around a model of the aircraft to cause the aircraft to perform as desired.

Model-based control incorporates explicit knowledge of a system's dynamics, which may allow better closed-loop performance than non-model-based control such as Proportional-Integral-Derivative control. A sampling of model-based control methods include linear optimal control [2, 25], backstepping and feedback linearization [18, 23, 24], sliding mode control [46], and adaptive control [3, 28], although a more complete sampling of various control methods used for flight control can be found in [54]. A general resource for control methodology is [5].

As noted by the large number of approaches, no one control design method is superior for all control tasks in all scenarios. For example, linear control techniques are excellent at tracking a linear, user-defined input where the system state deviations remain small in magnitude, but performance is severely degraded when either the input or the state deviation becomes large in magnitude.

A further consideration for flight control is the robustness of each control law. By its nature, flight introduces many disturbances to the aircraft which stress the control law, sometimes beyond its capability to maintain satisfactory flight behavior. Disturbances that usually occur during flight include wind, measurement noise, and model uncertainty. Further, the act of implementing a control law digitally introduces discretization to continuous dynamics; most control laws are designed for continuous systems. Since each of these disturbances cannot be easily accounted for in control design, a general notion of robustness is desired. Some control techniques such as linear quadratic regulator and H_∞ control have inherent robustness, while other control techniques such as feedback linearization have no robustness guarantees and therefore may be sensitive to disturbances.

This work presents a novel control law that leverages the structure of fixed-wing aircraft flight dynamics to impose a port-Hamiltonian structure on the closed-loop system of an in-

tegrator cascade system constructed from feedback linearization, which provides sufficient conditions for input-to-state stability (ISS) guarantees. The novel control law is the robust port-Hamiltonian (RPH) control law derived by Willebeek-LeMair in [50], and this work will analyze RPH by comparing its performance against two other, industry standard control laws. The first comparator is the linear quadratic regulator (LQR) control law, which is a linear control law with strong robustness guarantees. The second competitor is nonlinear dynamic inversion (NDI) control law, which is a nonlinear control law that identically transforms a nonlinear system into a linear system, although there are no inherent robustness guarantees. All three of the control laws are static state feedback control laws to ensure a proper comparison.

The effectiveness of a proposed control law must be validated through extensive flight testing before it can be adopted in practice. In order to mitigate risk during such a flight plan, a thorough simulation study is typically completed before the first flight of any new technology to reveal any potential issues or unforeseen results. This thesis describes the implementation and results of such a simulation study. RPH will be compared to LQR and NDI both in robustness to the above mentioned disturbances commonly present in flight, and in performance for the control task of output tracking. Each simulation study will test the impact of different combinations of disturbances, which will be analyzed in the the results and discussion.

In this work, both the controller design methods and resulting control laws for LQR, NDI, and RPH are heavily referred to. Especially in Chapters 3-5, referring to the LQR, NDI, or RPH control laws in full was detrimental to the discussion. Therefore, for the purposes of clear and condensed writing, those acronyms will be used to refer to the controller design methods, as well as the control laws as implied by context.

The remainder of this work is organized as follows. Chapter 2 provides a background on foundational knowledge for this work. Chapter 3 provides specific information on robustness, derivation, and implementation of each control law. Chapter 4 describes the specific methods that were used to realistically simulate a flight for each control law. Chapter 5 provides results and analysis of the completed simulation study. Chapter 6 provides an overview of this work in its entirety. Appendix A provides specific parameters and settings used in the simulation study presented. Appendix B provides additional simulation study results that were not included in the main discussion.

Chapter 2

Background

2.1 Math Preliminaries

Following is an overview of math concepts and notation used in this thesis.

A scalar is denoted in uppercase or lowercase regular text such as V, while a vector is denoted by a lowercase, bolded letter (e.g., $\boldsymbol{x}$), and a matrix is denoted by an uppercase, bolded letter (e.g., $\boldsymbol{A}$). The first three derivatives with respect to time for a quantity y are denoted $\dot{y}, \ddot{y}, \dddot{y}$. Higher number of time derivatives are noted with $y^{(p)}$, where p is the order of the time derivative. The eigenvalue of a matrix $\boldsymbol{A}$ is denoted by $\lambda(\boldsymbol{A})$, while $\lambda_{\min}(\boldsymbol{A})$ denotes the smallest eigenvalue and $\lambda_{\max}(\boldsymbol{A})$ denotes the largest eigenvalue.

- $\boldsymbol{A}^\top$ denotes the transpose of a vector or matrix $\boldsymbol{A}$.
- $\boldsymbol{A}^{-1}$ denotes the inverse of a square matrix $\boldsymbol{A}$, such that $AA^{-1} = \mathbb{I}$ where $\mathbb{I}$ is the identity matrix with the same dimensions as $\boldsymbol{A}$.
- $\boldsymbol{A}^{-\top}$ denotes the inverse transpose of the matrix A.
- A vector inner product is defined by $\boldsymbol{a}^\top \boldsymbol{b} = \boldsymbol{b}^\top \boldsymbol{a}$, $\boldsymbol{a}^\top \boldsymbol{a} = \|\boldsymbol{a}\|^2$.
- A vector outer product is defined by $\boldsymbol{a}\boldsymbol{b}^\top$.
- A cross product is defined by $\boldsymbol{a} \times \boldsymbol{b} = -\boldsymbol{b} \times \boldsymbol{a}$.
- A positive definite matrix $\boldsymbol{P} \succ 0$ satisfies $\boldsymbol{x}^\top \boldsymbol{P}\boldsymbol{x} > 0$ for all $\boldsymbol{x}$ except $\boldsymbol{x} = \boldsymbol{0}$. A negative definite matrix $\boldsymbol{K} \prec 0$ satisfies $\boldsymbol{x}^\top \boldsymbol{K}\boldsymbol{x} < 0$ for all $\boldsymbol{x}$ except $\boldsymbol{x} = \boldsymbol{0}$.
- A positive semi-definite matrix $\boldsymbol{\mathcal{R}} \succeq 0$ satisfies $\boldsymbol{x}^\top \boldsymbol{\mathcal{R}}\boldsymbol{x} \geq 0$ for all $\boldsymbol{x}$ except $\boldsymbol{x} = \boldsymbol{0}$. A negative semi-definite matrix $\boldsymbol{W} \preceq 0$ satisfies $\boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{x} \leq 0$ for all $\boldsymbol{x}$ except $\boldsymbol{x} = \boldsymbol{0}$.
- A skew symmetric matrix $\boldsymbol{\mathcal{J}}$ has the property $\boldsymbol{\mathcal{J}} = -\boldsymbol{\mathcal{J}}^\top$.

Further, below are a selection of types of matrices used in this work.

- $\mathbb{I}_{(\cdot)}$ denotes the identity matrix of size shown or implied.
- $\boldsymbol{0}_{(\cdot)\times(\cdot)}$ denotes a matrix of zeros with dimensions as shown or implied.
- $\boldsymbol{e}_j$ stands for a unit vector in $\mathbb{R}^3$, with a 1 in the "j" place and zeros elsewhere. For example, $\boldsymbol{e}_2 = [0,\ 1,\ 0]^\top$.

- $(\hat{\cdot})$ denotes a cross-product equivalent matrix for a vector. For example, given $\boldsymbol{v} = [v_1 \quad v_2 \quad v_3]^\top$, $\hat{\boldsymbol{v}} = \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix}$
- Rotation matrices $\boldsymbol{R}$ are a set of orthonormal matrices in which $\boldsymbol{R}\boldsymbol{R}^\top = \boldsymbol{R}^\top\boldsymbol{R} = \mathbb{I}$. Rotation matrices multiplied together create new rotation matrices.

Linear systems are a subtype of nonlinear systems. Time-invariant systems are a subtype of time-varying systems.

A nonlinear system is most generally presented in the following form

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{f}(t, \boldsymbol{x}, \boldsymbol{u}), \qquad \boldsymbol{x}(t_0) = \boldsymbol{x}_0 \\ \boldsymbol{y}(t) &= \boldsymbol{h}(t, \boldsymbol{x}, \boldsymbol{u}) \end{aligned} \tag{2.1.1}$$

where $\boldsymbol{x}(t) \in \mathbb{R}^n$ is the state at time t, $\boldsymbol{u}(t) \in \mathbb{R}^m$ is the control input, $\boldsymbol{y}(t) \in \mathbb{R}^l$ is the output of particular interest, $\boldsymbol{f} : [0, \infty) \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is a vector field, or a vector of functions, that defines the evolution of the state, and $\boldsymbol{h} : [0, \infty) \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^l$ is a vector field that defines a collection of functions of particular interest.

A linear, time-varying system can be represented in the following state space form [35]

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}(t)\boldsymbol{x}(t) + \boldsymbol{B}(t)\boldsymbol{u}(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}(t)\boldsymbol{x}(t) + \boldsymbol{D}(t)\boldsymbol{u}(t) \end{aligned} \tag{2.1.2}$$

where $\boldsymbol{A}(t) \in \mathbb{R}^{n \times n}$ is the state matrix, $\boldsymbol{B}(\boldsymbol{t}) \in \mathbb{R}^{n \times m}$ is the input-state, or control effectiveness, matrix, $\boldsymbol{C}(t) \in \mathbb{R}^{l \times n}$ is the output matrix, and $\boldsymbol{D}(t) \in \mathbb{R}^{l \times m}$ is the throughput matrix. A linear, time-invariant system is defined in a similar way, except the $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, and $\boldsymbol{D}$ do not vary with time.

Linearizing a nonlinear system using a Taylor series expansion approximates the nonlinear system as a linear system. Given the system (2.1.1), suppose $\boldsymbol{f}$ and $\boldsymbol{h}$ are continuously differentiable in $\boldsymbol{x}$ and $\boldsymbol{u}$, and that $(\boldsymbol{x}^*(t), \boldsymbol{u}^*(t))$ is a solution of (2.1.1), that is

$$\frac{d}{dt}\boldsymbol{x}^* = \boldsymbol{f}(\boldsymbol{x}^*, \boldsymbol{u}^*, t), \qquad \boldsymbol{x}^*(t_0) = \boldsymbol{x}_0 \tag{2.1.3}$$

Let $\boldsymbol{y}^*(t)$ denote the output corresponding to the trajectory $(\boldsymbol{x}^*(t), \boldsymbol{u}^*(t))$. By Taylor series expansion,

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}^*, \boldsymbol{u}^*, t) + \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right|_{(\boldsymbol{x}(t), \boldsymbol{u}^*(t))} (\boldsymbol{x} - \boldsymbol{x}^*) + \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}\right|_{(\boldsymbol{x}(t), \boldsymbol{u}^*(t))} (\boldsymbol{u} - \boldsymbol{u}^*) + \mathrm{Rem}_f(\boldsymbol{x}, \boldsymbol{u}, t), \\ \boldsymbol{y} &= \boldsymbol{h}(\boldsymbol{x}^*, \boldsymbol{u}^*, t) + \left.\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}\right|_{(\boldsymbol{x}(t), \boldsymbol{u}^*(t))} (\boldsymbol{x} - \boldsymbol{x}^*) + \left.\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{u}}\right|_{(\boldsymbol{x}(t), \boldsymbol{u}^*(t))} (\boldsymbol{u} - \boldsymbol{u}^*) + \mathrm{Rem}_h(\boldsymbol{x}, \boldsymbol{u}, t) \end{aligned} \tag{2.1.4}$$

where the higher order terms ($\text{Rem}_f(\boldsymbol{x}, \boldsymbol{u}, t)$ and $\text{Rem}_h(\boldsymbol{x}, \boldsymbol{u}, t)$) of the Taylor series expansion can be neglected when the state and control perturbations are sufficiently small. Defining

$$\tilde{\boldsymbol{x}} = \boldsymbol{x} - \boldsymbol{x}^*, \qquad \tilde{\boldsymbol{u}} = \boldsymbol{u} - \boldsymbol{u}^*, \quad \text{and} \quad \tilde{\boldsymbol{y}} = \boldsymbol{y} - \boldsymbol{y}^* \tag{2.1.5}$$

can cause system (2.1.4) to be rewritten as

$$\begin{aligned} \dot{\tilde{\boldsymbol{x}}}(t) &= \boldsymbol{A}(t)\tilde{\boldsymbol{x}}(t) + \boldsymbol{B}(t)\tilde{\boldsymbol{u}}(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}(t)\tilde{\boldsymbol{x}}(t) + \boldsymbol{D}(t)\tilde{\boldsymbol{u}}(t) \end{aligned} \tag{2.1.6}$$

where $\boldsymbol{A}(t)$ and $\boldsymbol{B}(t)$ are the Jacobians of $\boldsymbol{f}$ with respect to $\boldsymbol{x}$ and $\boldsymbol{u}$ respectively, and $\boldsymbol{C}(t)$ and $\boldsymbol{D}(t)$ are the Jacobians of $\boldsymbol{h}$ with respect to $\boldsymbol{x}$ and $\boldsymbol{u}$ respectively, all evaluated along the nominal trajectory $(\boldsymbol{x}^*(t), \boldsymbol{u}^*(t))$.

To create a rotation matrix based on a rotation about a unit vector $\boldsymbol{e}_j$ by angle θ, the Rodrigues' rotation formula is used

$$e^{\theta \hat{\boldsymbol{e}}_j} = \mathbb{I} + \sin(\theta)\hat{\boldsymbol{e}}_j + (1 - \cos(\theta))\hat{\boldsymbol{e}}_j^2 \tag{2.1.7}$$

Young's inequality is expressed as

$$\boldsymbol{a}^\top \boldsymbol{b} \leq \frac{1}{2}\boldsymbol{a}^\top \boldsymbol{W} \boldsymbol{a} + \frac{1}{2}\boldsymbol{b}^\top \boldsymbol{W}^{-1}\boldsymbol{b}, \qquad \forall \;\; \boldsymbol{W} = \boldsymbol{W}^\top \succ 0 \tag{2.1.8}$$

The Rayleigh-Ritz inequality is

$$\frac{1}{2}\lambda_{\min}(\boldsymbol{P})\,\|\boldsymbol{x}\|^2 \leq \boldsymbol{x}^\top \boldsymbol{P}\boldsymbol{x} \leq \frac{1}{2}\lambda_{\max}(\boldsymbol{P})\,\|\boldsymbol{x}\|^2 \tag{2.1.9}$$

A norm on a vector space maps vectors to the non-negative real numbers. Norms satisfy the following properties:

1. Positive definiteness:
$$\begin{aligned} \|\boldsymbol{v}\| &> 0 \qquad \text{for all} \quad \boldsymbol{v} \neq \boldsymbol{0} \\ \|\boldsymbol{v}\| &= 0 \qquad \text{when} \quad \boldsymbol{v} = \boldsymbol{0} \end{aligned} \tag{2.1.10}$$
2. The scaling property: $\|\alpha \boldsymbol{v}\| = |\alpha|\,\|\boldsymbol{v}\|$
3. The *triangle inequality*: $\|\boldsymbol{v}_1 + \boldsymbol{v}_2\| \leq \|\boldsymbol{v}_1\| + \|\boldsymbol{v}_2\|$

For the vector space $\mathbb{R}^n$, the p-norm is commonly defined as

$$\|\boldsymbol{v}\|_p = \left(\sum_{i=1}^{n} |v_i|^p\right)^{1/p} \tag{2.1.11}$$

for any finite scalar $p \geq 1$.

The p-norm can be used to define a norm on the infinite-dimensional vector space of continuous-valued, n-dimensional functions of time. From [18], the space L_p^m is defined as the set of all piecewise continuous functions $\boldsymbol{u} : [0, \infty) \to \mathbb{R}^m$ such that

$$\|\boldsymbol{u}\|_{L_p} = \left(\int_0^\infty \|\boldsymbol{u}(t)\|^p \, dt \right)^{1/p} < \infty \tag{2.1.12}$$

where $p \in [1, \infty]$ norm in the integrand can be any p-norm.

The L_1 norm is

$$\|\boldsymbol{u}\|_{L_1} = \int_0^\infty \|\boldsymbol{u}(t)\| \, dt < \infty \tag{2.1.13}$$

The L_2 norm is

$$\|\boldsymbol{u}\|_{L_2} = \sqrt{\int_0^\infty \boldsymbol{u}(t)^\top \boldsymbol{u}(t) \quad dt} < \infty \tag{2.1.14}$$

The L_∞ norm is

$$\|\boldsymbol{u}\|_{L_\infty} = \sup_{t \geq 0} \|\boldsymbol{u}(t)\| < \infty \tag{2.1.15}$$

This work will solely use the L_∞ and L_1 norms to compare the control laws based on error metrics.

Lastly, the concept of different classes of functions are provided in [18] to classify useful behavior in functions.

- A continuous function $\alpha : [0, a) \to [0, \infty)$ is said to belong to class $\mathcal{K}$ if it is strictly increasing and $\alpha(0) = 0$.
- A continuous function $\alpha : [0, a) \to [0, \infty)$ is said to belong to class $\mathcal{K}_\infty$ if it is in class $\mathcal{K}$, $a = \infty$, and $\alpha(r) \to \infty$ as $r \to \infty$.
- A continuous function $\beta : [0, a) \times [0, \infty) \to [0, \infty)$ is said to belong to class $\mathcal{KL}$ if, for each fixed s, the mapping $\beta(r, s)$ belongs to class $\mathcal{K}$ with respect to r and, for each fixed r, the mapping $\beta(r, s)$ is decreasing with respect to s and $\beta(r, s) \to 0$ as $s \to \infty$.

2.2 Flight Mechanics

The state of a system is the collection of time-varying quantities that are chosen by the user to describe the system. The dynamics of a system are how the state changes through time; flight mechanics focuses on the dynamics of aircraft. The state of an aircraft in flight typically includes position, velocity, attitude, and angular rates. Specifically, the user wants to know where the aircraft is with respect to the inertial frame (inertial position, $\boldsymbol{r}$), what the current velocities of the aircraft relative to its own frame of reference (body velocities, $\boldsymbol{v}_B$), the attitude of the aircraft with respect to the inertial frame (body-frame attitude, $\boldsymbol{\Theta}$ or

$\boldsymbol{q}$) and the angular rates of the aircraft relative to the current attitude of the aircraft (body-axis angular rates, $\boldsymbol{\omega}$). This section describes the flight mechanics of airplanes, specifically propeller-driven airplanes.

The orientation of the aircraft with respect to inertial space is expressed by a rotation matrix relating a body-fixed reference frame to an inertially-fixed reference frame. The body frame for an aircraft is defined by the $b_1, b_2,$ and b_3 axes, which are typically called the longitudinal, lateral, and vertical axes. The longitudinal, or b_1, axis is defined positively out the front of the aircraft, while the lateral, or b_2, axis is defined positively out the right of the aircraft, and the vertical, or b_3, axis is defined positively out the bottom of the aircraft. The state of an aircraft, stated above, fully describes the six degrees of freedom that an aircraft has, which are translations in the direction of the $b_1, b_2,$ and b_3 axes (forward/back, right/left, and down/up), and rotations about the $b_1, b_2,$ and b_3 axes (roll right/left, pitch up/down, and yaw right/left).

The following is how to find each state in the body frame from the inertial frame.

- Inertial position, $\boldsymbol{r}(t) = [N(t) \quad E(t) \quad D(t)]^\top$, is already in the inertial frame. The standard convention for the positive direction for an aircraft's inertial position is North (N), East (E), and Down (to the Earth's center).
- Body velocity, $\boldsymbol{v}_B$, is found from the inertial velocity, $\boldsymbol{v}_I$, using the body-to-inertial frame rotation matrix. This is based on the chosen attitude coordinates, and will be specified for each attitude coordinate choice. The standard convention for the positive direction for an aircraft's body velocity is forward along the longitudinal axis (u), right along the lateral axis (v), and down along the vertical axis (w).

$$\boldsymbol{v}_B(t) = \begin{bmatrix} u(t) \\ v(t) \\ w(t) \end{bmatrix} = \boldsymbol{R}_{BI}\boldsymbol{v}_I = \boldsymbol{R}_{IB}^\top \boldsymbol{v}_I \tag{2.2.1}$$

- Attitude has many coordinate representations from the inertial frame to the body frame [37]. The two that are used in this work are Euler angles and quaternions. Euler angles, $\boldsymbol{\Theta}$, are the standard attitude variables for aircraft due to their intuitive nature, although at certain attitudes the body-to-inertial rotation matrix becomes singular. The standard convention for the positive direction for an aircraft's attitude through Euler angles is roll right (right wing down, ϕ), pitch up (nose up, θ), and yaw right (nose right, ψ). Quaternions $\boldsymbol{q}$ are a 4-dimensional representation of attitude that does not have singularities, although they are much more difficult to visualize.

$$\boldsymbol{\Theta}(t) = \begin{bmatrix} \phi(t) \\ \theta(t) \\ \psi(t) \end{bmatrix} \qquad \text{and} \qquad \boldsymbol{q}(t) = \begin{bmatrix} q_0(t) \\ \\ \bar{\boldsymbol{q}}(t) \\ \\ \end{bmatrix} = \begin{bmatrix} q_0(t) \\ q_x(t) \\ q_y(t) \\ q_z(t) \end{bmatrix} \tag{2.2.2}$$

The body-to-inertial rotation matrices for a 3-2-1 rotation of the Euler angles, and the scalar-first convention for quaternions, are given using the shorthand $c = \cos()$ and $s = \sin()$ below. Note that $\boldsymbol{R}_{IB,\text{ Euler}}$ matches the definition given above, but this is spelled out for reference.

$$\boldsymbol{R}_{IB,\text{ Euler}} = \begin{bmatrix} c\theta c\psi & -c\phi s\psi + s\phi s\theta c\psi & s\phi s\psi + c\phi s\theta c\psi \\ c\theta s\psi & c\phi c\psi + s\phi s\theta s\psi & -s\phi c\psi + c\phi s\theta s\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix} \tag{2.2.3}$$

$$\boldsymbol{R}_{IB,\text{ quaternion}} = \left(q_0^2 - \|\bar{\boldsymbol{q}}\|^2\right)\mathbb{I}_3 + 2\left(\bar{\boldsymbol{q}}\bar{\boldsymbol{q}}^\top\right) + 2q_0\hat{\bar{\boldsymbol{q}}} \tag{2.2.4}$$

Euler angles and quaternions can be converted between each other using

$$\boldsymbol{\Theta}(t) = \begin{bmatrix} \tan^{-1}\left(\frac{q_0q_x + q_yq_z}{1-2\left(q_x^2+q_y^2\right)}\right) \\ \sin^{-1}\left(2(q_0q_y - q_xq_z)\right) \\ \tan^{-1}\left(\frac{q_0q_z + q_xq_y}{1-2\left(q_y^2+q_z^2\right)}\right) \end{bmatrix} \tag{2.2.5}$$

$$\boldsymbol{q}(t) = \begin{bmatrix} -\sin\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right) + \cos\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) \\ \sin\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) + \cos\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right) \\ -\sin\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right) + \cos\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) \\ \sin\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) + \cos\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right) \end{bmatrix} \tag{2.2.6}$$

Further, since quaternions are a 4 dimensional representation of attitude, the following method of right multiplication is used to combine the rotation of two quaternions into one, as opposed to directly adding the effect of repeated Euler angles

$$\boldsymbol{q}_1 \otimes \boldsymbol{q}_2 = \begin{bmatrix} q_{0,1}q_{0,2} - \bar{\boldsymbol{q}}_1^\top\bar{\boldsymbol{q}}_2 \\ q_{0,1}\bar{\boldsymbol{q}}_2 + q_{0,2}\bar{\boldsymbol{q}}_1 + (\bar{\boldsymbol{q}}_1 \times \bar{\boldsymbol{q}}_2) \end{bmatrix} \tag{2.2.7}$$

- Body-axis angular rates, $\boldsymbol{\omega}$, are different from the rates of the Euler angles $(\dot{\phi}, \dot{\theta}, \dot{\psi})$ (inertial angular rates) due to the rotation from the inertial to body reference frame. The standard convention for the positive direction for an aircraft's body-axis angular rate is roll right (p), pitch up (q), and yaw right (r).

$$\boldsymbol{\omega}(t) = \begin{bmatrix} p(t) \\ q(t) \\ r(t) \end{bmatrix} = \underbrace{\begin{bmatrix} -\sin(\theta) & 0 & 1 \\ \sin(\phi)\cos(\theta) & \cos(\phi) & 0 \\ \cos(\phi)\cos(\theta) & -\sin(\phi) & 0 \end{bmatrix}}_{\boldsymbol{L}^{-1}(\boldsymbol{\Theta})} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{2.2.8}$$

where $\boldsymbol{L}^{-1}(\boldsymbol{\Theta})$ is the rotation matrix from the inertial angular rates to the body angular rates.

The reference frames that are important for an aircraft are the body frame, the inertial frame, the velocity frame, and the wind frame. The $e_1, e_2,$ and e_3 axes are used to notate rotations that are made about the inertial axes based on defined angles to form new reference frames. The angles required to convert between each of the frames is given below, along with their order of rotation about the axes.

- Body to inertial frame: Roll (ϕ), pitch (θ), and yaw (ψ) angles

$$\boldsymbol{R}_{IB} = e^{\psi\hat{\boldsymbol{e}}_3} e^{\theta\hat{\boldsymbol{e}}_2} e^{\phi\hat{\boldsymbol{e}}_1} = \left(e^{-\phi\hat{\boldsymbol{e}}_1} e^{-\theta\hat{\boldsymbol{e}}_2} e^{-\psi\hat{\boldsymbol{e}}_3}\right)^\top = \boldsymbol{R}_{BI}^\top \tag{2.2.9}$$

- Inertial to velocity frame: Course (χ) and flight path (γ) angles

$$\boldsymbol{R}_{VI} = e^{-\gamma\hat{\boldsymbol{e}}_2} e^{-\chi\hat{\boldsymbol{e}}_3} = \left(e^{\chi\hat{\boldsymbol{e}}_3} e^{\gamma\hat{\boldsymbol{e}}_2}\right)^\top = \boldsymbol{R}_{IV}^\top \tag{2.2.10}$$

- Wind to body frame: Angles of attack (α) and sideslip (β)

$$\boldsymbol{R}_{BW} = e^{-\alpha\hat{\boldsymbol{e}}_2} e^{\beta\hat{\boldsymbol{e}}_3} = \left(e^{-\beta\hat{\boldsymbol{e}}_3} e^{\alpha\hat{\boldsymbol{e}}_2}\right)^\top = \boldsymbol{R}_{WB}^\top \tag{2.2.11}$$

- Wind to velocity frame: Bank angle (φ)

$$\boldsymbol{R}_{VW} = e^{\varphi\hat{\boldsymbol{e}}_1} = \left(e^{-\varphi\hat{\boldsymbol{e}}_1}\right)^\top = \boldsymbol{R}_{WV}^\top \tag{2.2.12}$$

The equations that define how an aircraft moves through the air are known as the equations of motion. These equations are found by applying Newton's second law to aircraft, and solving for the six degrees of freedom in which an aircraft can move. The derivation of these equations is well established [53], so only the final equations are listed below.

The longitudinal equations of motion are the forces and moment which remain in the b_1-b_3 plane, which are the forces along the b_1 and b_3 axes (X and Z respectively), as well as the pitching moment $\mathcal{M}$ about the b_2 axis. In this formulation, X, Z, and $\mathcal{M}$ are decomposed into more general aerodynamic and thrust moments. This is the most general case in which includes thrust acting along all six degrees of freedom.

$$\begin{aligned} m(\dot{u} + qw - rv) &= -mg\sin(\theta) + X_A + X_T \\ \dot{q}I_{yy} - pr(I_{zz} - I_{xx}) + (p^2 - r^2)I_{xz} &= \mathcal{M}_A + \mathcal{M}_T \\ m(\dot{w} + pv - qu) &= -mg\cos(\phi)\cos(\theta) + Z_A + Z_T \end{aligned} \tag{2.2.13}$$

where m is mass, g is acceleration due to gravity in the inertial positive down direction, I_{xx}, I_{yy}, and I_{zz} are the roll, pitch, and yaw moments of inertia respectively, and I_{xz} is a product of inertia. It is assumed that the I_{xy} and I_{yz} products of inertia are zero, which is a valid assumption for most aircraft.

The lateral-directional equations of motion are the remaining three forces and moments, which are the force Y along the b_2 axis and the rolling $\mathcal{L}$ and yawing $\mathcal{N}$ moments about

the b_1 and b_3 axes respectively. Similarly to the longitudinal equations, Y, $\mathcal{L}$, and $\mathcal{N}$ are decomposed into aerodynamic and thrust components.

$$\begin{aligned} \dot{p}I_{xx} - qr(I_{zz} - I_{yy}) - (\dot{r} + pq)I_{xz} &= \mathcal{L}_A + \mathcal{L}_T \\ m(\dot{v} + ru - pw) &= mg\sin(\phi)\cos(\theta) + Y_A + Y_T \\ \dot{r}I_{zz} + pq(I_{yy} - I_{yy}) + (qr - \dot{p})I_{xz} &= \mathcal{N}_A + \mathcal{N}_T \end{aligned} \tag{2.2.14}$$

However, a rearranged and condensed version is more useful for simulation and control design. By solving (2.2.13) and (2.2.14) for the time rate change of the body velocities and body-axis angular rates, and adding the equations for the time rate change of position and attitude quaternions, the full set of vehicle dynamics can be found. In this formulation, all aerodynamic and thrust forces are combined into forces along their specific axes, and all aerodynamic and thrust moments are combined into moments about their specific axes. The time rate change of both Euler angles and quaternions are listed, although in practice only one set of equations is used.

$$\begin{aligned} \dot{\boldsymbol{r}} &= \boldsymbol{R}_{IB}\boldsymbol{v}_B \\ \dot{\boldsymbol{v}}_B &= \boldsymbol{v}_B \times \boldsymbol{\omega} + \boldsymbol{R}_{BI}(g\boldsymbol{e}_3) + \frac{\boldsymbol{F}}{m} \\ &\begin{cases} \dot{\boldsymbol{\Theta}} = \boldsymbol{L}(\boldsymbol{\Theta})\boldsymbol{\omega}, \\ \dot{\boldsymbol{q}}_B = \dfrac{1}{2}\begin{bmatrix} \bar{\boldsymbol{q}}^\top \\ q_0\mathbb{I} \\ \hat{\bar{\boldsymbol{q}}} \end{bmatrix}\boldsymbol{\omega} - \ln\left(\|\boldsymbol{q}\|_2\right)\boldsymbol{q} \end{cases} \\ \dot{\boldsymbol{\omega}} &= \boldsymbol{I}^{-1}(\boldsymbol{I}\boldsymbol{\omega} \times \boldsymbol{\omega}) + \boldsymbol{M} \end{aligned} \tag{2.2.15}$$

where

$$\boldsymbol{L}(\boldsymbol{\Theta}) = \begin{bmatrix} 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi)\sec(\theta) & \cos(\phi)\sec(\theta) \end{bmatrix} \tag{2.2.16}$$

is the inverse of the matrix in (2.2.8) and where $\boldsymbol{F}$ is the total force (2.2.17), $\boldsymbol{M}$ is the total moment (2.2.20), and $\boldsymbol{I}$ is the inertia matrix (2.2.22).

The total force $\boldsymbol{F} \in \mathbb{R}^3$ along the b_1, b_2, b_3 axes is defined as

$$\boldsymbol{F} = \begin{bmatrix} X + T \\ Y \\ Z \end{bmatrix} = \frac{1}{2}\rho V^2 S \begin{bmatrix} C_X \\ C_Y \\ C_Z \end{bmatrix} + T\boldsymbol{e}_1 \tag{2.2.17}$$

where ρ is the air density, V is the magnitude of the airspeed, and S is the wind surface area, and with the thrust T for a propeller-driven airplane is defined by

$$T = \frac{1}{2}\rho V^2 S C_t \tag{2.2.18}$$

where

$$C_t = C_{x_{\mathcal{J}_c^2}} \mathcal{J}_c^2 + C_{x_{\mathcal{J}_c}} \mathcal{J}_c$$

$$\mathcal{J}_c = \mathcal{J} - \mathcal{J}_0 \qquad \mathcal{J} = \frac{nD}{V} \qquad \mathcal{J}_0 = \frac{n_{\text{cruise}} D}{V_{\text{cruise}}}$$

and where $\mathcal{J}$ is the inverse advance ratio, $\mathcal{J}_c$ is the centered, inverse advance ratio, $\mathcal{J}_0$ is the centering reference value, and $C_{x_{(\cdot)}}$ are the aerodynamic parameters corresponding to the appropriate $\mathcal{J}_c$ terms. The advance ratio is typically expressed using propeller speed n, in rev/s. In this thesis, Ω denotes propeller speed in rad/s, and n denotes propeller speed in rev/s.

The implementation of the nonlinear dynamic inversion control law (NDI) and the robust port-Hamiltonian control law (RPH) requires the body forces in (2.2.17) to be expressed in the wind frame. Therefore, the relation of body force $\boldsymbol{F}$ to the wind frame is

$$\begin{bmatrix} D \\ S \\ L \end{bmatrix} = \underbrace{\begin{bmatrix} -\cos(\alpha)\cos(\beta) & -\sin(\beta) & -\sin(\alpha)\cos(\beta) \\ \cos(\alpha)\sin(\beta) & -\cos(\beta) & \sin(\alpha)\sin(\beta) \\ \sin(\alpha) & 0 & -\cos(\alpha) \end{bmatrix}}_{\boldsymbol{R}_{WB}} \underbrace{\begin{bmatrix} X+T \\ Y \\ Z \end{bmatrix}}_{\boldsymbol{F}} \tag{2.2.19}$$

The total moment $\boldsymbol{M} \in \mathbb{R}^3$ about the b_1, b_2, b_3 axes is defined as

$$\boldsymbol{M} = \begin{bmatrix} \mathcal{L} \\ \mathcal{M} \\ \mathcal{N} \end{bmatrix} = \frac{1}{2}\rho V^2 S \begin{bmatrix} b & 0 & 0 \\ 0 & \bar{c} & 0 \\ 0 & 0 & b \end{bmatrix} \begin{bmatrix} C_{\mathcal{L}} \\ C_{\mathcal{M}} \\ C_{\mathcal{N}} \end{bmatrix} + \boldsymbol{u_M} \tag{2.2.20}$$

where b is the wing span and $\bar{c}$ is the mean aerodynamic chord length, and with the control moment $\boldsymbol{u_M}$ defined by

$$\boldsymbol{u_M} = \begin{bmatrix} u_{\mathcal{L}} \\ u_{\mathcal{M}} \\ u_{\mathcal{N}} \end{bmatrix} = \frac{1}{2}\rho V^2 S \begin{bmatrix} b & 0 & 0 \\ 0 & \bar{c} & 0 \\ 0 & 0 & b \end{bmatrix} \begin{bmatrix} C_{\mathcal{L}_{\delta_a}} & 0 & C_{\mathcal{L}_{\delta_r}} \\ 0 & C_{\mathcal{M}_{\delta_e}} & 0 \\ C_{\mathcal{N}_{\delta_a}} & 0 & C_{\mathcal{N}_{\delta_r}} \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_e \\ \delta_r \end{bmatrix} \tag{2.2.21}$$

where $u_{\mathcal{L}},\ u_{\mathcal{M}},\ u_{\mathcal{N}}$ are the roll, pitch, and yaw control moments respectively, $\delta_a,\ \delta_e,\ \delta_r$ are the aileron, elevator, and rudder control surface deflections respectively, and $C_{\mathcal{L}_{(\cdot)}},\ C_{\mathcal{M}_{(\cdot)}},\ C_{\mathcal{N}_{(\cdot)}}$ are the aerodynamic coefficients corresponding to each control surface deflection.

Lastly, the inertia matrix $\boldsymbol{I} \in \mathbb{R}^{3\times 3}$ is defined as

$$\boldsymbol{I} = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix} \tag{2.2.22}$$

The aerodynamic coefficients seen in (2.2.17) and (2.2.20) are different for every aircraft, and therefore must be identified in order to properly derive a model-based control law and

develop an accurate simulation. The aircraft used in this work is the My Twin Dream (MTD) aircraft, seen in Figure 2.1 with mass and geometric properties given in Table A.2. The MTD fixed-wing uncrewed air vehicle is an electrically propelled, commercial off-the-shelf airplane modified by researchers in Virginia Tech's Nonlinear Systems Laboratory (NSL). The platform is a robust, low-cost device for testing control and estimation algorithms developed by NSL researchers. The aircraft supports testing of control and estimation algorithms, using an onboard autopilot and co-computer. The MTD is equipped with an air data unit that includes a Kiel probe for Pitot and static pressure, and vanes for angle of attack and flank angle, as well as landing gear.

Figure 2.1: MTD research aircraft

Using system identification methods and techniques described in [12, 13, 19, 41, 42, 43], a new aerodynamic-propulsive model of the MTD was found to be

$$\begin{aligned}
C_X &= C_{X_\alpha}\alpha + C_{X_{\alpha^2}}\alpha^2 + C_{X_{\alpha^3}}\alpha^3 + C_{X_{\beta^2}}\beta^2 + C_{X_{\mathcal{J}_c}}\mathcal{J}_c + C_{X_{\mathcal{J}_c^2}}\mathcal{J}_c^2 + C_{X_0} \\
C_Y &= C_{Y_\beta}\beta + C_{Y_r}\hat{r} + C_{Y_{\delta_r}}\delta_r \\
C_Z &= C_{Z_\alpha}\alpha + C_{Z_{\alpha^3}}\alpha^3 + C_{Z_q}\hat{q} + C_{Z_{\delta_e}}\delta_e + C_{Z_{\mathcal{J}_c}}\mathcal{J}_c + C_{Z_0} \\
C_{\mathcal{L}} &= C_{\mathcal{L}_\beta}\beta + C_{\mathcal{L}_p}\hat{p} + C_{\mathcal{L}_r}\hat{r} + C_{\mathcal{L}_{\delta_a}}\delta_a + C_{\mathcal{L}_{\delta_a{}^3}}\delta_a^3 \\
C_{\mathcal{M}} &= C_{\mathcal{M}_\alpha}\alpha + C_{\mathcal{M}_q}\hat{q} + C_{\mathcal{M}_{\delta_e}}\delta_e \\
C_{\mathcal{N}} &= C_{\mathcal{N}_\beta}\beta + C_{\mathcal{N}_{\beta^3}}\beta^3 + C_{\mathcal{N}_p}\hat{p} + C_{\mathcal{N}_r}\hat{r} + C_{\mathcal{N}_{\delta_a}}\delta_a + C_{\mathcal{N}_{\delta_r}}\delta_r
\end{aligned} \tag{2.2.23}$$

where the aerodynamic parameters can be found in Table A.3, and

$$\hat{p} = \frac{pb}{2V}, \quad \hat{q} = \frac{q\bar{c}}{2V}, \quad \text{and} \quad \hat{r} = \frac{rb}{2V} \tag{2.2.24}$$

denote the dimensionless body-axis angular rates.

2.3 Robustness

Robustness describes how much a control law can resist disturbances before performance substantially declines. These disturbances include state-dependent endogenous disturbances due to modeling error and state-independent exogenous disturbances such as the time-varying external force due to wind. To reduce risk of full scale implementation of a control law, a robustness analysis is performed on the control law. This analysis can be done in many ways, from analytical methods to numerical optimization and calculation to find the bounds a system may face before becoming unstable.

For linear, time invariant systems, [52] provides an in depth analysis on six methods that can be applied to find the stability bounds of aerodynamic parameter variation. For nonlinear systems, [49] provides a path to extend linear matrix inequalities to nonlinear matrix inequalities, while [11] use optimal feedback control laws to account for disturbance other than aerodynamic parameter variation, such as static and dynamic input perturbations and stochastic disturbances.

For this work, the robustness analysis was completed by a simulation study, where each simulation included stochastic disturbances. By generating enough simulations, a computational assessment of the robustness can be determined for control laws. The results produced by the simulation studies in this work provide general analysis about what disturbances individual control laws are not robust to, as well as generate specific performance criteria for simulating flight implementation with realistic disturbances.

Below, Section 2.3.1 describes the four disturbances that were implemented in this work: measurement noise, model mismatch, wind, and control discretization with time delay. Section 2.3.2 discusses two common measures of robustness for linear control laws, which are gain and phase margin. Section 2.3.3 discusses the background for a primary nonlinear robustness measure, input-to-state stability.

2.3.1 Disturbances

The selection of the following disturbances was based on disturbances that an aircraft in flight would likely encounter. Therefore, by implementing these disturbances in a simulation study, the varied impact of each of the disturbances would become clear. This provides insight into the impact of different disturbances on the novel RPH, in order to reduce risk before full scale implementation.

Measurement Noise

Measurement noise is a disturbance that comes from sensor measurements. As the name implies, this disturbance comes from measurement data that vary, sometimes significantly, from the true measurement. There are a variety of factors that cause this disturbance, such as digitization and discretization of the analog, continuous data being measured, electronic

field interference, or simply data processing.

This disturbance is inherent in all sensors, and therefore is a realistic disturbance to incorporate in simulation for flight. Measurement noise could perturb the closed-loop system outside of the system's stability bounds. It is assumed that measurement noise occurs at a much higher frequency than the vehicle motion so that it can be filtered out of measurement data for use in the feedback control law. By comparing the filtered data to unfiltered data, the statistics of the measurement noise can be found. Therefore, sampling from each state measurement noise's distribution will provide realistic noise that can be implemented in simulation.

Model Mismatch

Model mismatch is a disturbance to the vehicle that comes from incomplete modeling. The balancing act of system identification is to get a model that matches the flight data well, and is simple to implement. Therefore, some dynamics of the full aircraft motion from flight will be left out.

This is the primary disturbance addressed by robustness analysis. In [11, 22, 32, 39], aerodynamic model uncertainty is the primary disturbance analyzed. Further, in [52], individual aerodynamic parameters in (2.2.23) are perturbed to find the percent difference, or standard deviation, each parameter can have before causing the system to become unstable. However, in this thesis, the model mismatch is applied by finding the error between the aerodynamic coefficients found by flight data and by the aerodynamic model.

The error between the flight data aerodynamic coefficients and the model-calculated aerodynamic coefficients should be purely Gaussian if the modeling is successful, however most of the time the error is not truly Gaussian. Despite this, the difference from a Gaussian distribution is usually negligible and therefore a Gaussian distribution is assumed. By sampling from the error distribution for each aerodynamic coefficient, the model mismatch can be implemented in the simulation.

Wind

There are many ways to model wind. The basic approach to modeling wind is to incorporate two elements into the wind model: steady wind and turbulence. Steady wind is the average velocity of the wind through time. This value can change direction or magnitude slowly, relative to the aircraft motion, but the steady wind term assumed here to remain constant. The primary time- and/or location-varying component of modeled wind is the turbulence, which causes gusts and random variation within the wind model. Turbulence originates in the boundary layer between a flow and an object, such as the Earth's surface, or between two flows moving in shear.

The most common clear air turbulence models are the von Kármán and Dryden models, which provide power spectral density (PSD) of turbulence in the frequency domain. By

sampling from the PSDs and converting the frequencies into time-domain signals, the effect of turbulence can be implemented in simulation.

Discretization and Time Delay

In developing nonlinear control laws, it is common to consider continuous-time dynamics with continuous control inputs, which means that most robustness and stability guarantees that come from theory are only valid for continuous-time systems. Reference [17] (with companion paper [16]) is an example of control law theory applied discrete-time systems, where a few Lyapunov-based stability theorems. However, the theorems that are provided are for either linear systems or equilibria of nonlinear systems. Discretization of continuous-time systems is unavoidable in practice because every continuous signal will become discretized due to digital implementation for every physically implemented system in simulation or flight. Therefore, any simulation or flight that attempts to implement theory must consider the impact of discretization. An approximation of continuous signals can be made by selecting step sizes that are small enough. Across the literature, this approximation is generally held to be true given the sheer number of control laws tested by simulation and flight, although there is relatively little theory bridging the gap between continuous theory and discrete implementation.

There are two primary methods that can be used to turn a continuous control law into a discrete control law. The first method is to design a control law around the continuous system, and then discretize the control law, while the second is to discretize the system, and then design a discrete control law around that new system [33]. The second approach is almost singularly used for linear systems due to the difficulty of discretizing nonlinear systems. However, an example of the construction of a discretized, nonlinear control law based on feedback linearization is given in [26]. Despite the existence of this control law, this thesis only implements continuous control laws to ensure a complete and fair comparison between the other two control laws.

Another necessary consideration when implementing control laws in flight is time delay of said control laws. Time delay can be caused by differences in transmission rates between the flight computer and ground station, or simply obstructions to the signal path. The impact of time delay further exacerbates the impact of discretization on the control input, since the application of time delay widens the gap between when the control input is calculated and when the control input is applied. Although this occurs in general for the discretization of continuous systems, the states of an aircraft before a time delay could be non-negligibly different from after the time delay, although the control input applied is from before the time delay.

2.3.2 Robustness for Linear Systems

The classical measures of robustness are gain and phase margin, which describe how much parameter variation and time delay a linear, time invariant (LTI) system can tolerate before

becoming unstable. The practical application of gain and phase margin can be seen in [20] and [1], where certain robustness limits are set to ensure safety of manned and unmanned systems designed with linear dynamics.

The primary way to determine gain and phase margin is by analyzing the system in the frequency domain, as seen in [53]. A single-input/single output, closed-loop LTI system can be represented in the Laplace domain as

$$L(s) = \frac{KG(s)}{1 + KG(s)H(s)} \tag{2.3.1}$$

where $G(s)$ is the plant transfer function, $H(s)$ is the feedback transfer function, and K is the input gain.

The closed-loop stability criteria can be determined by rearranging the characteristic equation, and finding that the stability limit for closed-loop LTI systems is

$$KG(s)H(s) = -1 \tag{2.3.2}$$

This means that in order for a system to be closed-loop stable, the open-loop system must have a log magnitude less than 0 dB when the phase is $-180°$.

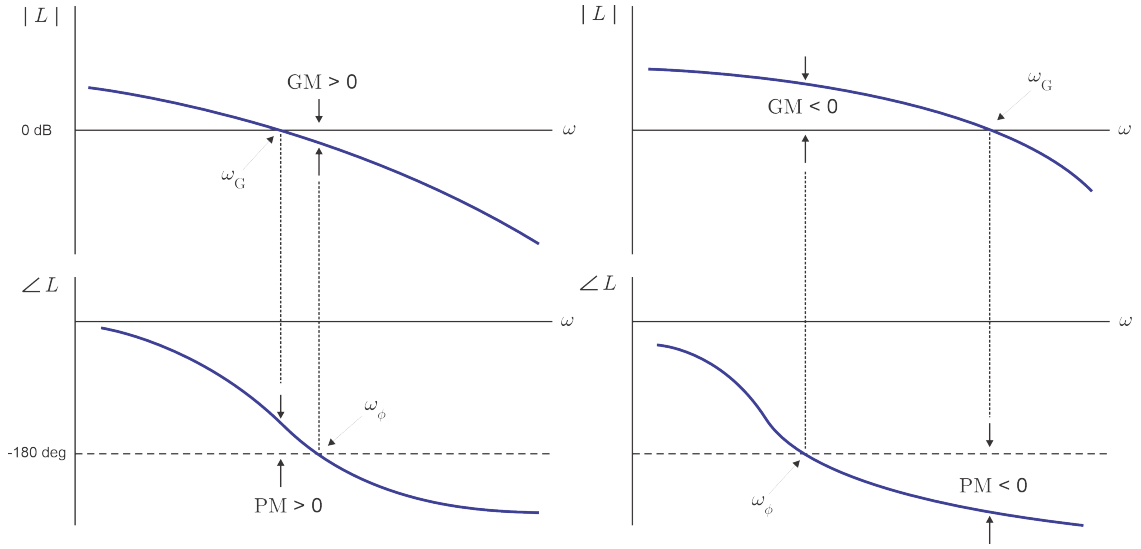

Figure 2.2: Example Bode plot, used with permission from Dr. Craig Woolsey

To find what the frequency response of the system is, $s = i\omega$ is substituted into the closed-loop transfer function (2.3.1), where the real component of L is the gain magnitude in decibels (dB) and the imaginary component is the phase in degrees. By plotting this on a semi-logarithmic scale, that is the abscissa axis plots frequency (ω) logarithmically and ordinate axis plots gain or phase linearly, a Bode plot is formed as seen in Figure 2.2. The

two top plots display the gain margin, or the magnitude of the transfer function, and the two bottom plots display the phase margin, or the phase angle of the transfer function, while the two left plots display positive margins and the two right plots display negative margins. There are four points in each set of plots that are important for robustness analysis. The two primary points occur when the gain magnitude crosses 0 dB, the gain crossover point ω_G, and when the phase crosses $-180°$, the phase crossover point ω_ϕ. The two secondary points occur at the corresponding frequencies on each opposing graph. If the gain and phase crossover point occur at the same frequency, the system has no gain or phase margin, which means any disturbance the system experiences would make the system become unstable. For systems of other than neutral stability, gain margin can be found by identifying the gain present in the system at the frequency at which the phase crossover point occurs. Similarly, phase margin can be found by identifying the phase present in the system at the frequency at which the gain crossover point occurs, and subtracting the phase found from 180°. If the magnitude of the system never crosses 0 dB, the system would have infinite phase margin, and vice versa for infinite gain margin if the phase never crossed $-180°$.

The notion of gain margin is used to define the extent of how much the magnitude of the open-loop system dynamics can be perturbed by while maintaining stability of the closed-loop system. If the system dynamics perturbations were increased by a factor of 2 without changing the feedback or control law, this would cause the gain magnitude in the system to increase by approximately 6 dB.

$$\text{Factor} = 10^{\text{dB}/20} \qquad \text{or} \qquad \text{dB} = 20\log_{10}(\text{Factor}) \tag{2.3.3}$$

Therefore, if the system has a gain margin of 22 dB, the system dynamics perturbations could be increased by a factor of approximately 12.5 before the gain and phase crossover points would share the same frequency, causing instability. Further, if the system dynamics were held constant but an external disturbance was applied with a magnitude of 20 dB, the system would be able to maintain stability since the overall magnitude of the system was less 22 dB.

Similarly, the notion of phase margin is used to define the extent of how much lag the open-loop system dynamics can experience while maintaining stability of the closed-loop system. If the system experienced a delay of 40 milliseconds at a frequency of 2 Hz, then the corresponding impact to the phase of the closed-loop system would be 30° per

$$\phi_L = 360° f(\Delta t) \tag{2.3.4}$$

where f is the frequency and Δt is the time delay. Therefore, if the system has a phase margin of 115°, the system would be able to withstand a significant amount of time delay (up to 0.15 s at 2 Hz) depending on the frequency of the dynamics.

This idea is expanded upon by [38], where an upper and lower margin for gain and phase are introduced. The purpose of this is to generalize the perturbation applied to the system, in case the model of the plant was a factor larger or smaller. For example, if the system

dynamics were made to be much smaller than the control law was designed for, the system could become unstable due to the effect of the control input. Therefore, for gain margin g_L, the limits are found within the noninclusive range $0 < g_L < g_L < g_u < \infty$, while for phase margin ϕ_L the limits are within the symmetric, noninclusive range $-\infty < -\phi_u < \phi_L < \phi_u < \infty$. These limits are the chosen robustness measure for the discussion in Section 3.1.2.

Discussion on another linear robustness measure can also be found in [38], where the concept of disk margin combines the notions of gain and phase margin to include combinations of both that drive the system unstable. For the system previously described, if a 22 dB gain disturbance is applied, the system will remain stable. If a 115° phase disturbance is applied, the system will remain stable. However, before reaching both limits simultaneously, the system will become unstable due to the cross impact. Therefore, the concept of disk margins combines the two notions in a 2-dimensional plot with gain and phase margin on the axes, showing the limits of combined disturbances.

2.3.3 Robustness for Nonlinear Systems

Unlike linear control theory, nonlinear control theory does not have standardized measurements to quantify the robustness of a system. Therefore, the robustness of nonlinear systems is based on providing guarantees of system performance given certain types of inputs and disturbances. Here, robustness is assessed using the notion of input-to-state stability (ISS), the focus of this discussion, which follows closely from [18].

The basis for most nonlinear stability theory is from Lyapunov stability, which identifies functions, when combined with certain inequalities, that provide bounds on the performance or response of a system. The focus of this work will be on boundedness, which is the notion of ensuring the state remains below a certain threshold.

Definition 2.1 (Definition 4.6 in [18])**.** Given a system $\dot{\boldsymbol{x}} = \boldsymbol{f}(t, \boldsymbol{x})$, the solutions for $\boldsymbol{x}(t)$ are said to be

- uniformly bounded if there exists a positive constant c, independent of t_0, and for every $a \in (0, c)$, there is $\beta = \beta(a) > 0$, independent of t_0, such that

$$\|x(t_0)\| \leq a \Rightarrow \|x(t)\| \leq \beta, \quad \forall t \geq t_0 \tag{2.3.5}$$

- uniformly ultimately bounded with ultimate bound b if there exist positive constants b and c, independent of $t_0 \geq 0$, and for every $a \in (0, c)$, there is $T = T(a, b) \geq 0$, independent of t_0, such that

$$\|x(t_0)\| \leq a \Rightarrow \|x(t)\| \leq b, \quad \forall t \geq t_0 + T \tag{2.3.6}$$

The respective properties are global if (2.3.5) and (2.3.6), respectively, hold for arbitrarily large a.

Consider the time-varying nonlinear system

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(t, \boldsymbol{x}, \boldsymbol{u}) \tag{2.3.7}$$

Definition 2.2 (Definition 4.7 in [18])**.** The system (2.3.7) is said to be input-to-state stable if there exists a class $\mathcal{KL}$ function β and a class $\mathcal{K}$ function γ such that for any initial state $\boldsymbol{x}(t_0)$ and any bounded input $\boldsymbol{u}(t)$, the solution $\boldsymbol{x}(t)$ exists for all $t \geq t_0$ and satisfies

$$\|\boldsymbol{x}(t)\| \leq \beta(\|\boldsymbol{x}(t_0)\|, t - t_0) + \gamma\left(\sup_{t_0 \leq \tau \leq t} \|\boldsymbol{u}(\tau)\|\right) \tag{2.3.8}$$

where class $\mathcal{K}$ and $\mathcal{KL}$ functions are defined in Section 2.1. Definition 2.2 defines a local version of ISS with global guarantees, with further discussion in Section 4.9 of [18].

The inequality (2.3.8) guarantees that the entire time history of the state will be uniformly ultimately bounded [18] if the input $\boldsymbol{u}(t)$ is bounded, hence the name input-to-state stability. In other words, ISS boundedness of the state along trajectories on which the disturbance is uniformly bounded. If all trajectories are perturbed by uniformly bounded disturbances, then ISS uniform ultimate boundedness of the state with respect to initial conditions and initial time. Therefore, if a control law is designed with ISS, then the control law will be robust to any bounded disturbance.

However, ISS only applies to bounded, additive disturbances. The class of non-additive disturbances describes all disturbances that do not enter the dynamics additively. For example, in the system $\dot{x} = -2(x + w)$, w is not simply added to the dynamics. On the other hand, if we defined a new disturbance $\tilde{w} = -2w$, then $\tilde{w}$ would enter additively. Non-additive disturbances can have a more significant impact on trajectories, such that bounded, non-additive disturbances can cause the state to grow unbounded.

A sufficient condition for ISS for a system can be found in Theorem 2.3, which applies Lyapunov-like equations to ISS. This theorem was used in the construction of RPH.

Theorem 2.3 (Theorem 2.12 in [31])**.** *Suppose that $\boldsymbol{f}$ is continuous in its arguments, that (2.3.7) is well posed (for any initial condition and input there is a unique maximal solution), and that all uniformly bounded solutions of (2.3.7) are maximal. Let $V : [0, \infty) \times \mathbb{R}^n \to [0, \infty)$ be a continuously differentiable function such that*

$$\alpha_1(\|\boldsymbol{x}\|) \leq V(t, \boldsymbol{x}) \leq \alpha_2(\|\boldsymbol{x}\|) \tag{2.3.9}$$

$$\dot{V}(t, \boldsymbol{x}, \boldsymbol{u}) \leq -\alpha(V(t, \boldsymbol{x})) + \gamma(\|\boldsymbol{u}\|_\infty) \tag{2.3.10}$$

for all $(t, \boldsymbol{x}, \boldsymbol{u}) \in [0, \infty) \times \mathbb{R}^n \times \mathbb{R}^m$, where $\dot{V}(t, \boldsymbol{x}, \boldsymbol{u})$ is the derivative of $V(t, \boldsymbol{x})$ along trajectories, α_1, α_2, and α are class $\mathcal{K}_\infty$ functions, and γ is a class $\mathcal{K}$ function. Then, the system (2.3.7) is input-to-state stable.

Once a control law is constructed, or as part of the process, the sufficient conditions for ISS in Theorem 2.3 are confirmed. Specifically, $V(t, \boldsymbol{x}), \alpha_1, \alpha_2, \alpha$, and γ are defined. If the defined functions satisfy (2.3.9) and (2.3.10), the control law is ISS.

Chapter 3

Control Methods

There are a variety of choices to make when controlling mechanical systems [4]. The first choice is whether to use a model-based approach or not. Examples of non-model based methods are proportional-integral-derivative or reinforcement learning control. Tuning the gains for model-free control laws is often difficult and time consuming.

If the choice is made to use model-based control, the second choice is whether to apply control to the full nonlinear dynamics, or to a linearized version of them. The arguably simpler path is to linearize the set of nonlinear system dynamics, and then control the system using any one of the variety of well-developed linear control methods such as proportional-integral-derivative control, linear quadratic regulator control (LQR), or H_∞ control. However, linear control laws and linearized dynamics typically only provide reasonable results if the perturbations applied to the system are small. Therefore, linear control is typically only used for systems which operate within a small maneuvering envelope, such as straight and level flight for an aircraft. Nonlinear control methods include model reference adaptive control, reinforcement learning, nonlinear dynamic inversion control (NDI), and passivity based control applied to port-Hamiltonian systems.

Another choice to make is between static and dynamic state control laws. Static state control laws are memoryless functions of the state [18]. These control laws are defined by algebraic expressions, rather than differential equations, which reduces the computational complexity of the control law.

Dynamic state feedback control laws are those that require at least one separate state with dynamics to control the system. This additional state may include: adaptive gains such as in the field of adaptive control, where the gains adapt based on the magnitude of error or changing parameters; an integrator state that can be applied to any control law to remove steady-state error; or a state estimator that can be applied to any control law to estimate any states that cannot be measured with sensors. The benefit of having a dynamic state control law is that the gains can change according to prescribed dynamics, accounting for any changes to the system or tuning the control law for better tracking over time. However, dynamic state control laws are computationally more expensive, more complicated to implement, and may be unnecessary depending on the control task.

In this work, two industry standard, static state feedback control laws are compared against a novel control technique presented in [50]. The two comparators are LQR, which is a linear

control law with robustness guarantees, and NDI, which is a nonlinear control law that identically transforms a nonlinear system into a linear system through feedback. The novel technique is a robust port-Hamiltonian control law (RPH), since the construction applies ISS to a port-Hamiltonian system, given a cascaded set of integrators that can be feedback linearized.

The three control laws were designed to perform output tracking on the desired quantities of airspeed $V^\star(t)$, flight path angle $\gamma^\star(t)$, course angle $\chi^\star(t)$, sideslip angle $\beta^\star(t)$. These desired output states for tracking were chosen from the list of options presented in [26].

The remainder of this chapter is devoted to the three control laws implemented in this work. For each control law, a general overview will be provided. Then the robustness guarantees inherent to each control law will be explained, followed by the theoretical derivation. Lastly, the specific implementation of each control law in this work will be discussed.

3.1 Linear Quadratic Regulator

3.1.1 Overview

Linear quadratic regulator control, or LQR, is a method of control applied to linear systems that is designed to minimize the state error and control inputs based on quadratic penalties. The term regulator in LQR comes from the control task of regulating the state to a reference state. LQR is part of the field of linear optimal control which includes extensions on the classical LQR problem, such as linear quadratic gaussian control (which accounts for measurement noise and other stochastic effects), H_2 control (which is a generalization of the linear quadratic gaussian method), and H_∞ control (which is the most robust method of this group of control laws) [7].

The primary benefits of using LQR are the simplicity of design and implementation, and robust performance. The simplicity of implementation is evident given LQR involves a linear control law:

$$\boldsymbol{u} = -\boldsymbol{K}\boldsymbol{x}$$

where $\boldsymbol{K}$ is the gain matrix, and $\boldsymbol{u}$ and $\boldsymbol{x}$ are the perturbations from nominal values of the control and state, respectively. Although the discussion and implementation presented below are for linear time-invariant systems, the theory for LQR extends easily to linear time-varying systems. LQR, and optimal linear control in general, has been studied extensively since the 1960s. More information can be found in [2, 7, 25], among other linear control books. The other benefits of LQR regarding robust performance and gain tuning are discussed below.

3.1.2 Guarantees

This section describes the inherent guarantees that LQR has, which follows closely from Chapter 4 in [7]. Since LQR is a linear control law, the robustness measure will be in terms

of gain and phase margin.

By applying Kalman's inequality, it can be shown that control laws designed by LQR inherently have infinite increasing gain margin ($g_u = \infty$), and a decreasing gain margin of 1/2 ($g_L = 1/2$) in each control input channel [7]. LQR control laws also have a phase margin in each control input channel of $\pm 60°$. This means that LQR control laws can be impacted by both positive and negative phase shift of 60°, due to time delay or other factors, and remain stable if there are no changes to the gain of the system. However, the requirement of these robustness properties is that the state itself is available for feedback. In the event that output, instead of state, feedback is used, all of the robustness guarantees are lost. Here, we assume the full state is available for computing the state feedback control law.

3.1.3 Derivation

For a linear system given by

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \tag{3.1.1}$$

the cost function is given as

$$J = \int_0^\infty \left(\boldsymbol{x}^\top \boldsymbol{Q}\boldsymbol{x} + \boldsymbol{u}^\top \boldsymbol{R}\boldsymbol{u}\right)\, dt \tag{3.1.2}$$

where $\boldsymbol{Q} \in \mathbb{R}^{n\times n}$ is the positive semi-definite state cost weight matrix and $\boldsymbol{R} \in \mathbb{R}^{m\times m}$ is the positive definite input cost weight matrix.

Linear control theory provides a standard control law

$$\boldsymbol{u} = -\boldsymbol{K}\boldsymbol{x} \tag{3.1.3}$$

where $\boldsymbol{K} \in \mathbb{R}^{m\times n}$ is the gain matrix.

In this work, the following linear control law is used,

$$\boldsymbol{u} = -\boldsymbol{K}(\boldsymbol{x} - \boldsymbol{x}_{\text{ref}}) + \boldsymbol{u}_{\text{eq}} \tag{3.1.4}$$

where $\boldsymbol{x}_{\text{ref}}$ is the reference trajectory state and $\boldsymbol{u}_{\text{eq}}$ is the trim control input. This is a valid extension of (3.1.3), since the standard LQR problem regulates the state to zero. However, by driving the quantity $(\boldsymbol{x} - \boldsymbol{x}_{\text{ref}})$ to zero instead, the same control law will drive the vehicle state to track the reference state. In order to not pull the system state too far away from the trim condition about which the aircraft dynamics were linearized, $\boldsymbol{x}_{\text{ref}}$ is required to be small.

In the case of LQR, $\boldsymbol{K}$ is found by

$$\boldsymbol{K} = \boldsymbol{R}^{-1}\boldsymbol{B}^\top \boldsymbol{P} \tag{3.1.5}$$

where $\boldsymbol{P} \in \mathbb{R}^{n \times n}$ is the symmetric, positive semi-definite solution to the algebraic Riccati equation

$$\boldsymbol{A}^\top \boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - \boldsymbol{P}\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^\top \boldsymbol{P} + \boldsymbol{Q} = \boldsymbol{0} \tag{3.1.6}$$

where it is assumed that $(\boldsymbol{A}, \boldsymbol{B})$ is controllable and $(\boldsymbol{A}, \sqrt{\boldsymbol{Q}})$ is observable, where $\sqrt{\boldsymbol{Q}}$ satisfies

$$\boldsymbol{Q} = \sqrt{\boldsymbol{Q}}^\top \sqrt{\boldsymbol{Q}}$$

This provides the minimum cost for the function J above, which penalizes large values of $\boldsymbol{x}$ if $\boldsymbol{Q}$ is large, or $\boldsymbol{u}$ is $\boldsymbol{R}$ is large.

3.1.4 Implementation

LQR for this work was developed by the following process. First, the linear system in state-space form is defined by the state and control inputs. For this work, the state chosen for LQR was

$$\boldsymbol{x}_{LQR} = [\phi,\ \theta,\ \psi,\ u,\ v,\ w,\ p,\ q,\ r]^\top \tag{3.1.7}$$

Inertial position (N, E, D) is omitted because the control task is output tracking, not trajectory tracking. Next, the control input chosen for LQR was

$$\boldsymbol{u} = [\delta_a, \delta_e, \delta_r, \Omega] \tag{3.1.8}$$

where $\delta_{a/e/r}$ are the aileron, elevator, and rudder deflections respectively, and Ω is the propeller speed in rad/s. The control input was chosen because it includes all of the control surfaces available on the MTD, in units that the flight computer expects.

For the simulations implemented in this work, there were three trim conditions that LQR was trimmed about based on the chosen trajectory to find both a trim state and a trim control input: straight, level, unaccelerated flight (SLUF), constant altitude and constant radius turn, and wings-level, unaccelerated climb. These trim states and control inputs can be found in Table A.6. However, only one gain matrix K_{LQR} was used through the entire flight which was based on the straight and level flight condition.

Once the state and control inputs are chosen, the equations of motion of the system are linearized about a nominal trim condition using a numerical Jacobian for both the state and control input to find $\boldsymbol{A}$ and $\boldsymbol{B}$ respectively.

The purpose of the $\boldsymbol{Q}$ matrix is to add weights to the cost function for each state. The higher the weight, the greater an error in the state will be penalized. In order to tune $\boldsymbol{Q}$, the weights should define how much error between the state and reference the user is willing to handle. The values of $\boldsymbol{Q}$ were chosen to keep the Euler angles within about 10° ($\pi/18$ rad), the body velocity within 0.5 m/s, and the angular rates within 30°/s ($\pi/6$ rad/s). Those values were put into the Q matrix by

$$\begin{aligned} \boldsymbol{\kappa} &= [\pi/18 \ \ \pi/18 \ \ \pi/18 \ \ 0.5 \ \ 0.5 \ \ 0.5 \ \ \pi/6 \ \ \pi/6 \ \ \pi/6]^\top \\ \boldsymbol{Q} &= \operatorname{diag}\left(\left[1/(\boldsymbol{\kappa})^2\right]\right) \end{aligned}$$

where the final $\boldsymbol{Q}$ matrix is given in (A.3.2).

Similarly to $\boldsymbol{Q}$, the purpose of $\boldsymbol{R}$ is to add weights to the cost function for each control input. In order to tune $\boldsymbol{R}$, the weights should be similar to the actuator limits for the physical system being controlled that the user wants to impose. The values of $\boldsymbol{R}$ were chosen to penalize actuator deflections of 10° ($\pi/18$ rad) and propeller rotation of 30 rad/s. If some actuators had lower limits than others, the values for $\boldsymbol{R}$ can be tuned to account for that.

$$\boldsymbol{R} = 10 \text{ diag}\left(\begin{bmatrix} 1/(\pi/18 & \pi/18 & \pi/18 & 30)^2 \end{bmatrix}\right)$$

where the final $\boldsymbol{R}$ matrix is given in (A.3.3)

For the gain matrix, there are a variety of methods to solve the algebraic Riccati equation (3.1.6). For this paper, the `lqr(A,B,Q,R)` command in `MATLAB` was used. The final $\boldsymbol{K}_{LQR}$ gain matrix used in this work is given in (A.3.4).

The final part for LQR implementation regarded converting the reference outputs from (A.1.1). Since these outputs were chosen for the use in the feedback linearization-based control laws NDI and RPH, the outputs needed to be converted into the reference usable for LQR.

Given the user-defined desired outputs $V^\star(t)$, $\gamma^\star(t)$, $\chi^\star(t)$, and $\beta^\star(t)$, the following equations were used to compute the necessary reference outputs for LQR to track, denoted $(\cdot)^\star(t)$.

$$\phi^\star(t) = -\tan^{-1}\left(\frac{V}{g}\dot{\psi}_{\text{trim}}\right) \tag{3.1.9}$$

$$\theta^\star(t) = \theta_{\text{trim}} \tag{3.1.10}$$

$$\psi^\star(t) = \chi^\star(t) \tag{3.1.11}$$

$$\begin{bmatrix} u^\star(t) \\ v^\star(t) \\ w^\star(t) \end{bmatrix} = V^\star(t) \underbrace{\left(e^{\psi^\star(t)\hat{\boldsymbol{e}}_3} e^{\theta^\star(t)\hat{\boldsymbol{e}}_2} e^{\phi^\star(t)\hat{\boldsymbol{e}}_1}\right)^\top}_{\boldsymbol{R}_{BI,\text{ ref}}} \underbrace{\left(e^{\chi^\star(t)\hat{\boldsymbol{e}}_3} e^{\gamma^\star(t)\hat{\boldsymbol{e}}_2}\right)}_{\boldsymbol{R}_{IV,\text{ ref}}} \boldsymbol{e}_1 \tag{3.1.12}$$

$$\begin{bmatrix} p^\star(t) \\ q^\star(t) \\ r^\star(t) \end{bmatrix} = \begin{bmatrix} -\sin(\theta^\star(t)) \\ \cos(\theta^\star(t))\sin(\phi^\star(t)) \\ \cos(\theta^\star(t))\cos(\phi^\star(t)) \end{bmatrix} \dot{\psi}_{\text{trim}} \tag{3.1.13}$$

where V is the norm of the aircraft's current body velocity, g is gravity, $\dot{\psi}_{\text{trim}}$ is the trim turn rate listed in Section A.3.1 and θ_{trim} is the trim pitch angle at each selected trim condition.

An example of the implementation for LQR is shown in Figure 3.1. The first example, Figure 3.1(a), demonstrates undisturbed, linear output tracking from an off-nominal condition. Since this is the case that LQR was designed around, the results are stellar in both state output tracking results and very low magnitude control input.

The second example, Figure 3.1(b), shows undisturbed, large deviation output tracking from an off-nominal condition. Although LQR was not designed for large deviation tracking, the resulting outputs are close to the desired state outputs. Small transients can be seen in

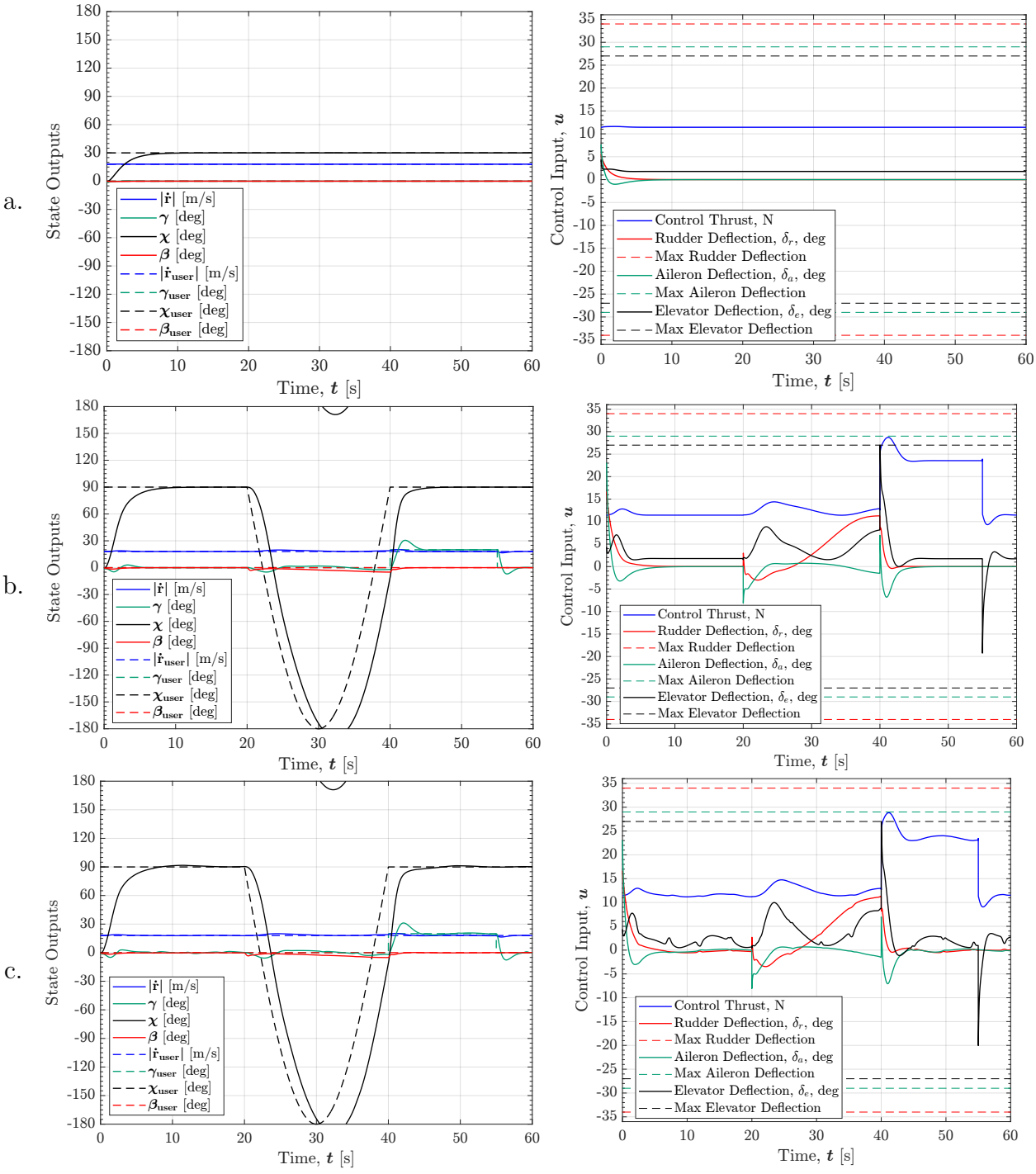

Figure 3.1: Example of LQR control law implementation (a) Linear output tracking without disturbances, (b) Nonlinear output tracking without disturbances, and (c) Nonlinear output tracking with exogenous disturbances.

the state outputs, especially in γ, whenever the nonlinearities grow. However, the control input figure shows large transients in control effort at times 20, 40, and 55 seconds. This is due to the sudden switching of the trim conditions. Although there exists a process to add a transition phase to LQR to smoothly transition between trim conditions, no transition region was implemented in this work. However, even in the large deviation region, LQR has relatively low magnitude control inputs.

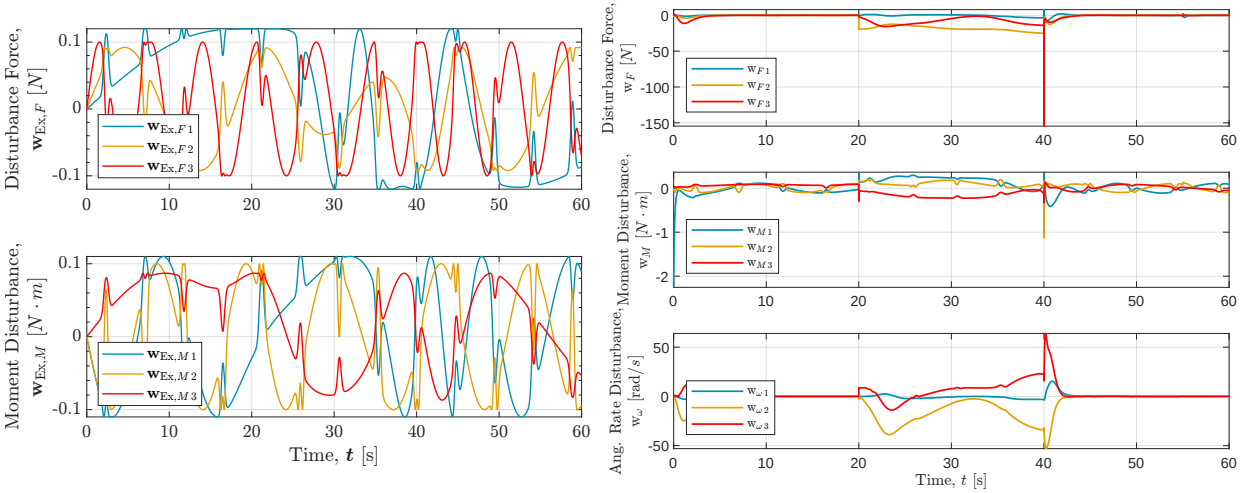

Figure 3.2: Example of exogenous disturbances on LQR: Applied exogenous disturbance (left), and experienced disturbance (right).

Lastly, Figure 3.1(c) demonstrates large deviation output tracking from an off-nominal condition with a generalized applied disturbance. This exogenous applied force and moment disturbance can be seen in Figure 3.2 on the left, whereas the disturbance experienced by LQR, which includes model mismatch due to the linearization of nonlinear dynamics and the exogenous disturbance, is shown on the right. The applied disturbance was chosen to stress the control law, and is defined in (A.2.1). Despite the applied disturbances, the performance of LQR is not significantly different from the undisturbed case in Figure 3.1(b), although there are more small transients in every tracked output, especially during nonlinear maneuvers. The disturbance rejection is most clearly seen in the control inputs, where there are more frequent changes to counteract the disturbance compared to without the disturbance.

3.2 Nonlinear Dynamic Inversion

3.2.1 Overview

Nonlinear dynamic inversion control (NDI), also known as feedback linearization, is a nonlinear control method that uses feedback to identically turn a nonlinear system into a linear system.

In a greatly simplified example, consider a single input system

$$\dot{x} = a(x) + b(x)u \tag{3.2.1}$$

where

$$a(x) = x - x^3 \qquad \text{and} \qquad b(x) = 2 \tag{3.2.2}$$

The method of dynamic inversion would set the control input to be

$$u = \frac{1}{b(x)}\left(-a(x) + v\right) \tag{3.2.3}$$

where v is the linear control term. The ideal control input (3.2.3) would identically transform the nonlinear system (3.2.1) into the linear system

$$\dot{x} = v \tag{3.2.4}$$

using the control input

$$u = \frac{1}{2}\left(-x + x^3 + v\right) \tag{3.2.5}$$

where v can be determined by any linear control method, such as LQR. In (3.2.1), NDI canceled a stabilizing nonlinearity, which may be very wasteful compared to other methods.

However, the primary flaw with NDI is caused by the cancellation of only the known nonlinearities. Changing a from (3.2.2) to include some "unknown" nonlinearities,

$$a = x + x^2 + x^3 \tag{3.2.6}$$

or

$$a = x - \frac{1}{2}x^3 \tag{3.2.7}$$

while applying the same control input (3.2.5) results in different state dynamics after feedback linearization

$$\dot{x} = x^2 + v$$

or

$$\dot{x} = \frac{1}{2}x^3 + v$$

In both (3.2.6) and (3.2.7), the unknown nonlinearity would cause the system to become unstable. In (3.2.6), an unknown and unaccounted for term caused nonlinearities to persist,

while in (3.2.7), a known nonlinearity had an unknown multiplicative factor, which caused the nominal linearizing control law to introduce a destabilizing nonlinearity. NDI only works perfectly when the model is exactly known, but by definition that is not possible as discussed in Section 2.3.1.

The second flaw with feedback linearization is that not all systems can be feedback linearized. The most generally applicable method of feedback linearization is input-output linearization, in which some nonlinear dynamics remain after the designer has constructed a linear mapping from inputs to outputs. These remaining nonlinear dynamics have no visible effect on the linearized portion of the closed-loop system, but the linearizing control law depends on the entire system state, including the state of these unobserved nonlinear dynamics. It is important to verify that the unobservable "zero dynamics" are well-behaved (i.e., asymptotically stable to zero) so that the linearizing control input does not grow so large that it exceeds the actuators' capability.

3.2.2 Guarantees

Unlike LQR-designed control laws, there are no inherent stability or robustness guarantees when a system has been feedback linearized. The primary method of avoiding matrix inversion singularities is by ensuring the flight envelope of the aircraft stays within set limits. However, this is not a realistic method since disturbances can cause the aircraft to enter its singularity condition.

Further, the aircraft model used to calculate NDI is greatly reduced from the model identified by system identification. This has the effect of constantly applying a disturbance to NDI, in that there is known (to the user) model mismatch between the aircraft model and aircraft model used for the construction of NDI. The effect of limiting logic prevents the control inputs, both u and v in (3.2.5) from growing exponentially. Although the LQR outer loop control law likely provides some local robustness benefits, those benefits likely are overstepped immediately by the inherent model mismatch.

There have been methods proposed to improve the robustness of NDI through extensions that transform NDI into dynamic state feedback. A preliminary example is given in [45], where an integrator is added to a feedback linearized control law, resulting in a robustness greater than an unmodified gain-scheduled control law. Another integrator addition is presented in [30], where the robustness of the dynamic inversion outer loop with a proportional-integral inner loop is analyzed with promising results. A more in depth set of extensions was presented in [29], where a robust, and then adaptive, extension was applied to a dynamic inversion foundation, which resulted in asymptotic tracking guarantees through Lyapunov analysis. A separate analysis [21] concluded that including sufficient time scale separation improves the stability of NDI, at the cost of restricting the design freedom.

Another way to mitigate model mismatch disturbances is to change the foundation of the construction to incremental nonlinear dynamic inversion (INDI). This foundation uses sim-

ilar construction to NDI, although the equations of motion are manipulated to different effect and require a greater number of sensor inputs, especially angular acceleration sensors [44]. The primary difference between NDI and Incremental NDI is that NDI computes commanded control surface deflections at every control calculation, whereas INDI computes the increments of the control surface deflection by already noting the influence of the control surfaces previously [40]. This is done by applying NDI to an incremental form of the equations of motion, which greatly reduces the impact that aerodynamic model mismatch has on the control law. This is not done simply by canceling the measured acceleration, but by the structure of controlling changes to the control surfaces instead of absolute deflections. Reference [40] further proposes a practically attainable version of INDI that incorporates a linear filter to predict the state of the aircraft, as opposed to assuming instantaneous knowledge of the state as needed for INDI. Finally, a comprehensive robustness analysis on INDI by [48] is provided, using Lyapunov methods and nonlinear system perturbation theory to prove the state of the closed-loop system is ultimately bounded by a class $\mathcal{K}$ function of the perturbation bounds. The authors of [48] also show how the incremental nature of INDI rejects disturbances more effectively than NDI alone using numerical simulation. Lastly, [6, 14] demonstrates the use of INDI on current US military aircraft.

3.2.3 Derivation

The general derivation of input-output linearization can be found below, which closely follows Section 13.2 in [18], and [36]. Consider a single-input/single-output control-affine nonlinear system

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})u \\ y &= h(\boldsymbol{x}) \end{aligned} \tag{3.2.8}$$

where $\boldsymbol{f}(\boldsymbol{x})$, $\boldsymbol{g}(\boldsymbol{x})$, and $\boldsymbol{h}(\boldsymbol{x})$ are "sufficiently smooth" in a domain $D \in \mathbb{R}^n$, and the mappings $\boldsymbol{f} : D \to \mathbb{R}^n$ and $\boldsymbol{g} : D \to \mathbb{R}^n$ are vector fields on D. The Lie derivative of a function $h(\boldsymbol{x})$ along a vector field $\boldsymbol{f}(\boldsymbol{x})$ is defined as

$$L_f h = \frac{\partial h}{\partial \boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}) \tag{3.2.9}$$

and where iterated Lie derivatives are defined as

$$\begin{aligned} L_f^0 h(\boldsymbol{x}) &= h(\boldsymbol{x}) \\ L_g L_f h(\boldsymbol{x}) &= \frac{\partial L_f h(\boldsymbol{x})}{\partial \boldsymbol{x}} \boldsymbol{g}(\boldsymbol{x}) \\ L_f^2 h(\boldsymbol{x}) = L_f L_f h(\boldsymbol{x}) &= \frac{\partial L_f h(\boldsymbol{x})}{\partial \boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}) \\ L_f^k h(\boldsymbol{x}) = L_f L_f^{k-1} h(\boldsymbol{x}) &= \frac{\partial L_f^{k-1} h(\boldsymbol{x})}{\partial \boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}) \end{aligned} \tag{3.2.10}$$

Therefore, system (3.2.8) has relative degree r in a region D_0 if

$$L_g \psi_i = 0 \qquad \text{for} \qquad i \in \{0, 1, ..., r-1\} \qquad \text{and} \qquad L_g \psi_r = 0 \tag{3.2.11}$$

for all $\boldsymbol{x} \in D_0$ where

$$\psi_1(\boldsymbol{x}) = h(\boldsymbol{x}) \qquad \text{and} \qquad \psi_{i+1}(\boldsymbol{x}) = L_f \psi_i, \qquad i \in \{1, 2, ..., r-1\} \tag{3.2.12}$$

The system (3.2.8) is input-output linearizable in a domain D_0 if it has relative degree $r \leq n$. If the system has relative degree $r = n$, then the system is input-state linearizable.

The above single-input/single-output system (3.2.8) is extended to the square multi-input/multi-output system

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}_1(\boldsymbol{x})u_1 + \boldsymbol{g}_2(\boldsymbol{x})u_2 + \cdots + \boldsymbol{g}_m(\boldsymbol{x})u_m \\ y_1 &= h_1(\boldsymbol{x}) \\ &\vdots \\ y_m &= h_m(\boldsymbol{x}) \end{aligned} \tag{3.2.13}$$

The multi-input/multi-output system (3.2.13) has vector relative degree $(r_1, r_2, ..., r_m)$ in a neighborhood D_0 if for all $\boldsymbol{x} \in D_0$ and for each $j \in \{1, ..., m\}$,

$$L_{g_1} L_f^k h_j = 0 \qquad \text{for} \qquad 0 \leq k \leq r_j - 2 \qquad \text{and} \qquad i \in \{1, ..., m\} \tag{3.2.14}$$

and the $m \times m$ matrix $\boldsymbol{A}(\boldsymbol{x})$

$$\boldsymbol{A}(\boldsymbol{x}) = \begin{bmatrix} L_{g_1} L_f^{r_1 - 1} h_1 & \cdots & L_{g_m} L_f^{r_1 - 1} h_1 \\ \vdots & \ddots & \vdots \\ L_{g_1} L_f^{r_m - 1} h_m & \cdots & L_{g_m} L_f^{r_m - 1} h_m \end{bmatrix} \tag{3.2.15}$$

is nonsingular. If the system has a well-defined vector relative degree, then the system (3.2.13) is input-output linearizable.

If a multi-input/multi-output system is input-output linearizable, then the control input is independent of all except the final derivative of the output mapping. This independence of the control input is known as integrator decoupling, in that each output function derivative is decoupled from the others regarding the control input.

If a system has a well-defined vector relative degree, then the state feedback control law

$$\boldsymbol{u} = -\boldsymbol{A}(\boldsymbol{x})^{-1} \begin{bmatrix} L_f^{r_1} h_1 \\ \vdots \\ L_f^{r_m} h_m \end{bmatrix} + \boldsymbol{A}(\boldsymbol{x})^{-1} \boldsymbol{v} \tag{3.2.16}$$

gives the closed-loop system

$$\begin{pmatrix} y_1^{(r_1)} \\ \vdots \\ y_m^{(r_m)} \end{pmatrix} = \boldsymbol{v} \tag{3.2.17}$$

The last consideration with feedback linearization is in regard to the zero dynamics of a system. It is important to ensure that the feedback linearizing control law (3.2.16) does not depend on components of the system state that fail to converge asymptotically to zero when the output is identically zero. When a system is feedback linearized, some states are unable to be feedback linearized, and therefore become unobservable through the course of the feedback linearization. Zero dynamics are the unobservable dynamics with the output held at zero, and a system is minimum phase if the origin is an asymptotic equilibrium for the zero dynamics. Ensuring that a system is minimum phase guarantees that a control law will perform as expected with no states becoming unbounded.

For the feedback linearization used in this thesis, the zero dynamics include only the inertial position, for which the origin is not an asymptotically stable equilibrium. However, since the control laws are performing inertial velocity output tracking, the position coordinates are ignorable and are not used at all in the control law calculation. Therefore, the zero dynamics can be considered empty, which means that the system is minimum phase.

3.2.4 Implementation

The derivation for NDI in this work came from [27], which is a specific application of input-output linearization. An in depth implementation process is provided below.

The equations of motion for this synthesis are a combined coordinate system of wind and body axes from [8], which defines the state to be

$$\boldsymbol{x}_{NDI_{\text{init}}} = [V, \gamma, \varphi, \chi, \alpha, \beta, p, q, r, T, \dot{T}]^\top \tag{3.2.18}$$

where V is the airspeed, γ, φ, χ are the flight path, bank, and course angles, α and β are the angles of attack and sideslip, p, q, r are the body-axis angular rates, T is the thrust state, and $\dot{T}$ is the thrust rate. The purpose of this double dynamic extension on thrust was to ensure the system was feedback linearizable, since the final control input was chosen to be

$u_{\ddot{T}}$. The state dynamics are

$$\begin{aligned}
\dot{V} &= -\frac{D}{m} - g\sin\gamma \\
\dot{\gamma} &= q_w\cos\varphi - r_w\sin\varphi \\
\dot{\varphi} &= p_w + (q_w\sin\varphi + r_w\cos\varphi)\tan\gamma \\
\dot{\chi} &= (q_w\sin\varphi + r_w\cos\varphi)\sec\gamma \\
\dot{\alpha} &= q - q_w\sec\beta - (p\cos\alpha + r\sin\alpha)\tan\beta \\
\dot{\beta} &= r_w + p\sin\alpha - r\cos\alpha \\
\dot{q} &= \frac{1}{I_{yy}}\left[\mathcal{M} + I_{xz}(r^2 - p^2) + (I_{zz} - I_{xx})rp\right] \\
\begin{bmatrix}\dot{p}\\ \dot{r}\end{bmatrix} &= \begin{bmatrix} I_{xx} & -I_{xz} \\ -I_{xz} & I_{zz}\end{bmatrix}^{-1}\begin{bmatrix} \mathcal{L} + I_{xz}pq + (I_{yy} - I_{zz})qr \\ \mathcal{N} - I_{xz}qr + (I_{xx} - I_{yy})pq\end{bmatrix} \\
\frac{d}{dt}\begin{bmatrix}T\\ \dot{T}\end{bmatrix} &= \begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}\begin{bmatrix}T\\ \dot{T}\end{bmatrix} + \begin{bmatrix}0\\ 1\end{bmatrix}u_{\ddot{T}}
\end{aligned} \tag{3.2.19}$$

where

$$\begin{aligned}
q_w &= \frac{1}{mV}(L - mg\cos\gamma\cos\varphi) \\
r_w &= \frac{1}{mV}(-S + mg\cos\gamma\sin\varphi) \\
p_w &= p\cos\alpha\cos\beta + (q - \dot{\alpha})\sin\beta + r\sin\alpha\cos\beta
\end{aligned} \tag{3.2.20}$$

and p_w, q_w, r_w are the wind-axis angular rates, m is mass, g is gravity, D, S, L are the drag, side-, and lift forces respectively, $\mathcal{L}, \mathcal{M}, \mathcal{N}$ are the roll, pitch, and yaw moments, and $I_{(\cdot)}$ are the respective components of the inertia matrix.

Given the aircraft model in (2.2.23), a few changes are made in order to ensure the system is input-output linearizable.

1. Ignore the dependence on angular rates (p, q, r) and control surface deflections $(\delta_a, \delta_e, \delta_r)$ from the force terms (C_X, C_Y, C_Z)

2. The control input for NDI is chosen to be $\boldsymbol{u} = \begin{bmatrix}\boldsymbol{u_M} & u_{\ddot{T}}\end{bmatrix}^\top$, where the control moments are defined by (2.2.21), and $u_{\ddot{T}}$ which is the double time derivative of the applied thrust.

Therefore, the aircraft aerodynamic model used for the derivation of NDI and RPH was

$$\begin{aligned}
C_X &= C_{X_\alpha}\alpha + C_{X_{\alpha^2}}\alpha^2 + C_{X_{\alpha^3}}\alpha^3 + C_{X_{\beta^2}}\beta^2 + [C_{X_{\mathcal{J}_c}}\mathcal{J}_c + C_{X_{\mathcal{J}_c^2}}\mathcal{J}_c^2] + C_{X_0} \\
C_Y &= C_{Y_\beta}\beta \\
C_Z &= C_{Z_\alpha}\alpha + C_{Z_{\alpha^3}}\alpha^3 + C_{Z_0} \\
C_{\mathcal{L}} &= C_{\mathcal{L}_\beta}\beta + C_{\mathcal{L}_p}\hat{p} + C_{\mathcal{L}_r}\hat{r} + [C_{\mathcal{L}_{\delta_a}}\delta_a] \\
C_{\mathcal{M}} &= C_{\mathcal{M}_\alpha}\alpha + C_{\mathcal{M}_q}\hat{q} + [C_{\mathcal{M}_{\delta_e}}\delta_e] \\
C_{\mathcal{N}} &= C_{\mathcal{N}_\beta}\beta + C_{\mathcal{N}_{\beta^3}}\beta^3 + C_{\mathcal{N}_p}\hat{p} + C_{\mathcal{N}_r}\hat{r} + [C_{\mathcal{N}_{\delta_a}}\delta_a + C_{\mathcal{N}_{\delta_r}}\delta_r]
\end{aligned} \tag{3.2.21}$$

where

$$\hat{p} = \frac{pb}{2V}, \quad \hat{q} = \frac{q\bar{c}}{2V}, \quad \hat{r} = \frac{rb}{2V} \quad \mathcal{J}_c = \frac{nD}{V} - \mathcal{J}_0 \tag{3.2.22}$$

The aerodynamic parameters that are marked by [square brackets] are those terms that are accounted for by the addition of thrust and the control moments in (2.2.18) and (2.2.21). Therefore, those terms are not part of the specific derivation process, but instead are used when calculating the control surface deflection to send to the flight computer.

With the inclusion of the reduced aerodynamic model (3.2.21), the system (3.2.19) is of the form

$$\begin{aligned}
\dot{\boldsymbol{x}} &= \boldsymbol{A}(\boldsymbol{x}) + \boldsymbol{B}(\boldsymbol{x})\boldsymbol{u} \\
\boldsymbol{y} &= \boldsymbol{C}\boldsymbol{x}
\end{aligned} \tag{3.2.23}$$

which is similar to (2.1.2), with the exception that $\boldsymbol{y} \in \mathbb{R}^m$, $\boldsymbol{C} \in \mathbb{R}^{m\times n}$, and $\boldsymbol{D}(\boldsymbol{x}) = \boldsymbol{0}$. $\boldsymbol{C}$ can be found by selecting the desired outputs for the task of output tracking. Since the states chosen for output tracking were $V^\star(t), \gamma^\star(t), \chi^\star(t),$ and $\beta^\star(t)$, the implemented matrix $\boldsymbol{C}$ is given in (A.3.6). It is known that (3.2.19) is of the form (3.2.23) because of the changes made to the aerodynamic model in 3.2.4. If those changes had not been made, then the state dynamics would have been based on (2.2.23), which would not have resulted in the linear analytic form of (3.2.23).

It is convenient to write (3.2.23) in the general form

$$\begin{aligned}
\dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) \\
\boldsymbol{y} &= \boldsymbol{C}\boldsymbol{x}
\end{aligned} \tag{3.2.24}$$

where the matrices $\boldsymbol{A}(\boldsymbol{x})$ and $\boldsymbol{B}(\boldsymbol{x})$ can be recovered using

$$\begin{aligned}
\boldsymbol{A}(\boldsymbol{x}) &= \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{0}) \\
\boldsymbol{B}(\boldsymbol{x}) &= \frac{\partial \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u})}{\partial \boldsymbol{u}}
\end{aligned} \tag{3.2.25}$$

Now that the system (3.2.23) has been defined, the input-output linearization can begin. The linearization process from [27] is implemented slightly differently than explained in Section 3.2.3, but the process is nearly identical.

The primary difference is between the Lie derivative defined in (3.2.9), and the new kth order differentiation operator, $L^k_{\boldsymbol{A}}(\cdot)$ where

$$L^0_{\boldsymbol{A}}(\boldsymbol{x}) = \boldsymbol{x} \qquad \text{and} \qquad L^k_{\boldsymbol{A}}(\boldsymbol{x}) = \left[\frac{\partial L^{k-1}_{\boldsymbol{A}}(\boldsymbol{x})}{\partial \boldsymbol{x}}\right] \boldsymbol{A}(\boldsymbol{x}) \tag{3.2.26}$$

The process begins by taking sequential kth order derivatives of each component of the output, and checking to see if the term multiplying the control input is zero.

$$\boldsymbol{y}_i^{(d_i)} = \boldsymbol{C}_i \boldsymbol{x}^{(d_i)} = \boldsymbol{C}_i \left[\frac{\partial L^{d_i-1}_{\boldsymbol{A}}(\boldsymbol{x})}{\partial \boldsymbol{x}}(\boldsymbol{x})\right] \boldsymbol{A}(\boldsymbol{x}) + \boldsymbol{C}_i \left[\frac{\partial L^{d_i-1}_{\boldsymbol{A}}(\boldsymbol{x})}{\partial \boldsymbol{x}}(\boldsymbol{x})\right] \boldsymbol{B}(\boldsymbol{x})\boldsymbol{u} \tag{3.2.27}$$

where $\boldsymbol{y}_i$ is the ith component of the desired output, $\boldsymbol{C}_i$ is the row of $\boldsymbol{C}$ that corresponds to the desired output, and d is the final derivative that is taken for integrator decoupling.

This is done most easily through Algorithm 1, which produces $\boldsymbol{A}^*_i$ and $\boldsymbol{B}^*_i$ for each desired output, as well as the equations to calculate the derivative of every state, $\boldsymbol{L}_{\text{counter}}$. For example,

$$\begin{aligned} \boldsymbol{L}_1 &= \left[\dot{V}, \dot{\gamma}, \dot{\varphi}, \dot{\chi}, \dot{\alpha}, \dot{\beta}, \dot{p}, \dot{q}, \dot{r}, \dot{T}, \ddot{T}\right]^\top \\ \boldsymbol{L}_2 &= \left[\ddot{V}, \ddot{\gamma}, \ddot{\varphi}, \ddot{\chi}, \ddot{\alpha}, \ddot{\beta}, \ddot{p}, \ddot{q}, \ddot{r}, \ddot{T}, \dddot{T}\right]^\top \end{aligned} \tag{3.2.28}$$

Stacking each ith component of $\boldsymbol{y}$, $\boldsymbol{A}^*$, and $\boldsymbol{B}^*$ results in the final form for the feedback linearization.

$$\boldsymbol{y}^{(d)} = \boldsymbol{A}^*(\boldsymbol{x}) + \boldsymbol{B}^*(\boldsymbol{x})\boldsymbol{u} \tag{3.2.29}$$

where

$$\boldsymbol{A}^*(\boldsymbol{x}) = \begin{bmatrix} \boldsymbol{C}_1 \frac{\partial L^{d_1}_{\boldsymbol{A}}}{\partial \boldsymbol{x}}(\boldsymbol{x}) \\ \boldsymbol{C}_2 \frac{\partial L^{d_2}_{\boldsymbol{A}}}{\partial \boldsymbol{x}}(\boldsymbol{x}) \\ \vdots \\ \boldsymbol{C}_m \frac{\partial L^{d_m}_{\boldsymbol{A}}}{\partial \boldsymbol{x}}(\boldsymbol{x}) \end{bmatrix} \qquad \text{and} \qquad \boldsymbol{B}^*(\boldsymbol{x}) = \begin{bmatrix} \boldsymbol{C}_1 \frac{\partial L^{d_1-1}_{\boldsymbol{A}}}{\partial \boldsymbol{x}}(\boldsymbol{x}) \\ \boldsymbol{C}_2 \frac{\partial L^{d_2-1}_{\boldsymbol{A}}}{\partial \boldsymbol{x}}(\boldsymbol{x}) \\ \vdots \\ \boldsymbol{C}_m \frac{\partial L^{d_m-1}_{\boldsymbol{A}}}{\partial \boldsymbol{x}}(\boldsymbol{x}) \end{bmatrix} \boldsymbol{B}(\boldsymbol{x}) \tag{3.2.30}$$

For the desired outputs $V^\star(t)$, $\gamma^\star(t)$, $\chi^\star(t)$, and $\beta^\star(t)$,

$$\boldsymbol{y}^{(d)} = \begin{bmatrix} V^{\star(3)}(t) \\ \gamma^{\star(3)}(t) \\ \chi^{\star(3)}(t) \\ \beta^{\star(2)}(t) \end{bmatrix} = \begin{bmatrix} \dddot{V}^\star(t) \\ \dddot{\gamma}^\star(t) \\ \dddot{\chi}^\star(t) \\ \ddot{\beta}^\star(t) \end{bmatrix} \tag{3.2.31}$$

In order to input-output linearize the system, the control input is chosen such that

$$\boldsymbol{u} = [\boldsymbol{B}^*]^{-1}(-\boldsymbol{A}^* + \boldsymbol{v}) \tag{3.2.32}$$

resulting in (3.2.29) becoming

$$\boldsymbol{y}^{(d)} = \boldsymbol{v} \tag{3.2.33}$$

The linear control term $\boldsymbol{v}$ is found by transforming the LQR matrix obtained in Section 3.1.4. Since LQR is a linear control law, system (3.2.29) is rewritten for each ith component in the form

$$\begin{bmatrix} \dot{\boldsymbol{y}}_i \\ \ddot{\boldsymbol{y}}_i \\ \dddot{\boldsymbol{y}}_i \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{y}_i \\ \dot{\boldsymbol{y}}_i \\ \ddot{\boldsymbol{y}}_i \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \boldsymbol{v}_i \tag{3.2.34}$$

Overall, the system can be written as

$$\dot{\boldsymbol{x}}_{NDI} = \boldsymbol{A}_c \boldsymbol{x}_{NDI} + \boldsymbol{B}_c \boldsymbol{v}_{NDI} \tag{3.2.35}$$

where

$$\boldsymbol{A}_c = \text{diag}\left(\begin{bmatrix} \boldsymbol{0}_{2\times1} & \mathbb{I}_{2\times2} \\ 0 & \boldsymbol{0}_{1\times2} \end{bmatrix}, \begin{bmatrix} \boldsymbol{0}_{2\times1} & \mathbb{I}_{2\times2} \\ 0 & \boldsymbol{0}_{1\times2} \end{bmatrix}, \begin{bmatrix} \boldsymbol{0}_{2\times1} & \mathbb{I}_{2\times2} \\ 0 & \boldsymbol{0}_{1\times2} \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \right) \tag{3.2.36}$$

and

$$\boldsymbol{B}_c = \text{diag}(\boldsymbol{e}_3, \boldsymbol{e}_3, \boldsymbol{e}_3, \boldsymbol{e}_2) \tag{3.2.37}$$

with the state

$$\boldsymbol{x}_{NDI} = [V,\ \dot{V},\ \ddot{V},\ \gamma,\ \dot{\gamma},\ \ddot{\gamma},\ \chi,\ \dot{\chi},\ \ddot{\chi},\ \beta,\ \dot{\beta}]^\top \tag{3.2.38}$$

The states that included time derivatives were calculated using (3.2.28), which were saved as symbolic functions that could be recalculated.

To solve for $\boldsymbol{K}_{NDI}$ and therefore $\boldsymbol{v}$, the LQR $\boldsymbol{Q}_{LQR}$ and $\boldsymbol{R}_{LQR}$ matrices were transformed to apply the same gains to the new state $\boldsymbol{x}_{NDI}$, using $\texttt{lqr}(A_c, B_c, Q_{NDI}, R_{NDI})$. This is done by choosing $\boldsymbol{Q}_{NDI}$ and $\boldsymbol{R}_{NDI}$ such that

$$\boldsymbol{Q}_{NDI} = \left[\left(\frac{\partial \boldsymbol{x}_{NDI}}{\partial \boldsymbol{x}_{LQR+}} \right)^{-\top} \begin{bmatrix} \boldsymbol{Q}_{LQR} & 0 & 0 \\ 0 & \left(\frac{\partial T}{\partial \Omega}\right)^{-\top} \boldsymbol{R}_{LQR,\Omega} \left(\frac{\partial T}{\partial \Omega}\right)^{-1} & 0 \\ 0 & 0 & 0 \end{bmatrix} \left(\frac{\partial \boldsymbol{x}_{NDI}}{\partial \boldsymbol{x}_{LQR+}} \right)^{-1} \right] \tag{3.2.39}$$

and

$$\boldsymbol{R}_{NDI} = \begin{bmatrix} \left(\frac{\partial \boldsymbol{u}_M}{\partial \boldsymbol{\delta}}\right)^{-\top} \boldsymbol{R}_{LQR,\delta} \left(\frac{\partial \boldsymbol{u}_M}{\partial \boldsymbol{\delta}}\right)^{-1} & 0 \\ 0 & r_{u_{\ddot{T}}} \end{bmatrix} \tag{3.2.40}$$

by evaluating both $\boldsymbol{Q}_{NDI}$ and $\boldsymbol{R}_{NDI}$ at the same trim condition as LQR, given in Table A.6. The specific equations used to transform from $\boldsymbol{Q}_{LQR}$ to $\boldsymbol{Q}_{NDI}$ are given in (A.3.7)-(A.3.11), and the final $\boldsymbol{Q}_{NDI}$, $\boldsymbol{R}_{NDI}$, and $\boldsymbol{K}_{NDI}$ matrices are found in (A.3.12)-(A.3.15).

Lastly, the linear control method $\boldsymbol{v}$ was set to the LQR solution

$$\boldsymbol{v} = \dot{\boldsymbol{x}}^\star_{\text{ref,ff}}(t) - \boldsymbol{K}_{NDI} \left(\boldsymbol{x}_{NDI} - \boldsymbol{x}^\star_{\text{ref}}(t) \right) \tag{3.2.41}$$

where the reference outputs given in (A.1.1) are user-defined and time-varying

$$\boldsymbol{x}^{\star}_{\text{ref}}(t) = [V^{\star}(t),\ 0,\ 0,\ \gamma^{\star}(t),\ 0,\ 0,\ \chi^{\star}(t),\ 0,\ 0,\ \beta^{\star}(t),\ 0]^{\top} \tag{3.2.42}$$

and since dynamics of the reference model are known (the user specified the time history of the desired outputs), a feedforward term $\dot{\boldsymbol{x}}^{\star}_{\text{ref,ff}}(t) = \begin{bmatrix} \dddot{V}^{\star}(t) & \dddot{\gamma}^{\star}(t) & \dddot{\chi}^{\star}(t) & \ddot{\beta}^{\star}(t) \end{bmatrix}^{\top}$ is included to improve tracking performance.

Equations (3.2.32), (3.2.41), and (3.2.42) were implemented in simulation. However, in order to limit the exponential growth of $\boldsymbol{v}$ and $\boldsymbol{u}$, control limiting was implemented. The limit was set notionally at 10^5 for each component of $\boldsymbol{v}$ and $\boldsymbol{u}$, which greatly reduced singularities in the control calculation.

The final implementation step for NDI is the creation of the reference output. For LQR, the user-defined reference outputs were directly tracked because LQR does not require a differentiable reference input. However, since NDI and RPH do require a differentiable reference output, a reference model with dynamics was implemented with desirable performance.

Neither the user-defined desired outputs given in (A.1.1) nor the reference model state were explicitly chosen to avoid commanding the aircraft to enter a state which would cause a matrix inversion singularity. Therefore, the state of the reference model tracking the desired outputs was used to compute the inverted control effectiveness matrix $\boldsymbol{B}^{*}$ from (3.2.32) for the entire time history, and verified that $\boldsymbol{B}^{*}$ never became singular. Further, since the only difference possible in the entire reference output history is caused the small variation in the initial state, because the reference output history did not cause $\boldsymbol{B}^{*}$ to become singular once, $\boldsymbol{B}^{*}$ will likely never become singular due to this reference trajectory.

The reference model dynamics were set to

$$\dot{\boldsymbol{x}}_{\text{ref}} = (\boldsymbol{A}_{\text{ref}} - \boldsymbol{B}_{\text{ref}}\boldsymbol{K}_{\text{ref}})\,(\boldsymbol{x}_{\text{ref}} - \boldsymbol{x}^{\star}_{\text{ref}}(t)) \tag{3.2.43}$$

where $\boldsymbol{x}_{\text{ref}}$ is the reference state, $\boldsymbol{A}_{\text{ref}}$ is the reference state matrix, $\boldsymbol{B}_{\text{ref}}$ is the reference state input matrix, $\boldsymbol{K}_{\text{ref}}$ is the reference gain matrix, and $\boldsymbol{x}^{\star}_{\text{ref}}(t)$ is the desired state, which is the same as (3.2.42). The initial reference state was calculated to lie directly on top of the initial vehicle state for the simulation, except the reference state converted into the $\boldsymbol{x}_{NDI}$ state. This was done to ensure any error metrics come from output tracking itself, and not an initial transient to match an offset reference model.

The form of (3.2.43) was found by applying the control input $\boldsymbol{u}_{\text{ref}} = \boldsymbol{K}_{\text{ref}}\,(\boldsymbol{x}_{\text{ref}} - \boldsymbol{x}^{\star}(t))$ to the state space formulation

$$\dot{\boldsymbol{x}}_{\text{ref}} = \boldsymbol{A}_{\text{ref}}\,(\boldsymbol{x}_{\text{ref}} - \boldsymbol{x}^{\star}_{\text{ref}}(t)) - \boldsymbol{B}_{\text{ref}}\boldsymbol{u}_{\text{ref}} \tag{3.2.44}$$

The matrices $\boldsymbol{A}_{\text{ref}}$ and $\boldsymbol{B}_{\text{ref}}$ become the same matrices (3.2.36) and (3.2.37), respectively, due to the same construction in (3.2.34). However, for $\boldsymbol{K}_{\text{ref}}$, a low pass filter is applied to smooth

out the dynamics of the reference model and reduce the transients in the discontinuous user-defined desired outputs.

The formulation of the low-pass filter is below. Since $V^\star(t)$, $\gamma^\star(t)$, and $\chi^\star(t)$ all require third order response and $\beta^\star(t)$ only requires second order response per (3.2.31), the third order derivation will proceed first, followed by the second order derivation.

The transfer function $H(s)$, for the low-pass filter of a third order system with no zeros is

$$H(s) = \frac{\omega_n^3}{(s+\omega_n)(s^2+2\zeta\omega_n+\omega_n^2)} \tag{3.2.45}$$

where ζ is the damping ratio, and ω_n is the natural frequency of the system. The output function becomes

$$y = H(s)u \tag{3.2.46}$$

By substituting in $H(s)$ and rearranging, (3.2.46) becomes

$$(s+\omega_n)(s^2+2\zeta\omega_n+\omega_n^2)y = \omega_n^3 u \tag{3.2.47}$$

Rearranging further,

$$(s^3+(2\zeta+1)\omega_n s^2+(2\zeta+1)\omega_n^2 s+\omega_n^3)y = \omega_n^3 u \tag{3.2.48}$$

Shifting from the Laplace frame into the time frame, s becomes a derivative, which rearranges (3.2.48) into

$$\dddot{y}+(2\zeta+1)\omega_n\ddot{y}+(2\zeta+1)\omega_n^2\dot{y}+\omega_n^3 y = \omega_n^3 u \tag{3.2.49}$$

Solving for $\dddot{y}$ reveals

$$\dddot{y} = -\omega_n^3(y-u)-(2\zeta+1)\omega_n^2\dot{y}-(2\zeta+1)\omega_n\ddot{y} \tag{3.2.50}$$

which is the form of the control input from (3.2.34).

For the second order derivation, the process follows exactly from above, except the low-pass filter of a second order system is

$$H(s) = \frac{\omega_n^2}{(s^2+2\zeta\omega_n+\omega_n^2)} \tag{3.2.51}$$

with a final form of

$$\ddot{y} = -\omega_n^2(y-u)-2\zeta\omega_n\dot{y} \tag{3.2.52}$$

Therefore, the coefficients of the gain matrix are

$$\boldsymbol{K}_{\text{ref}} = \begin{bmatrix} \omega_n^3 & (1+2\zeta)\omega_n^2 & (1+2\zeta)\omega_n & \mathbf{0}_{1\times 8} & \\ \mathbf{0}_{1\times 3} & \omega_n^3 & (1+2\zeta)\omega_n^2 & (1+2\zeta)\omega_n & \mathbf{0}_{1\times 5} \\ \mathbf{0}_{1\times 6} & \omega_n^3 & (1+2\zeta)\omega_n^2 & (1+2\zeta)\omega_n & \mathbf{0}_{1\times 2} \\ \mathbf{0}_{1\times 9} & \omega_n^2 & 2\zeta\omega_n & & \end{bmatrix} \tag{3.2.53}$$

where the values used for simulation are found in Section A.3.2, and the $\boldsymbol{K}_{\text{ref}}$ used in simulation is found (A.3.17). Note: $\boldsymbol{K}_{\text{ref}}$ has filler zeros to ensure that the multiplication with $\boldsymbol{B}_{\text{ref}}$ lines up correctly.

Lastly, although (3.2.50) and (3.2.52) have terms on the right hand side that are negative, the terms are substituted into the gain matrix $\boldsymbol{K}_{\text{ref}}$ as positive. This is because of the resultant dynamics in (3.2.43), where $\boldsymbol{A}_{\text{ref}}$ is each $\dddot{y}$ or $\ddot{y}$ term, $\boldsymbol{B}_{\text{ref}}$ includes each y, $\dot{y}$, and $\ddot{y}$ term in that order, which causes the above selection of the gain matrix.

An example of the implementation for NDI is shown in Figure 3.3. The first example, Figure 3.3(a), demonstrates undisturbed, linear output tracking from an off-nominal condition. Further, since NDI and RPH were constructed based on a foundation of feedback linearization, the exact model used for feedback linearization described in Section 3.2.4 was implemented as the simulation aerodynamic model for the undisturbed simulation examples. Although NDI is a nonlinear control law, LQR was applied as an outer loop control law on the feedback linearized system with the aim of obtaining a fair comparison with LQR control of the linearized dynamics. To this end, the gain matrix for LQR wrapped around NDI was calculated and transformed based on the state input, so the gain matrix should be applying the same weight to the linear errors. Compared to LQR in Figure 3.1(a), NDI used greater magnitude control inputs but reached the desired state outputs more quickly. This shows that NDI identically transformed the nonlinear system into a linear system, and achieved exact output tracking, just like LQR.

The second example, Figure 3.3(b), shows undisturbed, large deviation output tracking from an off-nominal condition, where the simulated aerodynamic model is the reduced model that the feedback linearization construction was based on. The large deviation output tracking from NDI is as exact to the reference model as for the small deviation output tracking, and the required control inputs have very small magnitudes. Further, unlike the construction of LQR, the lack of switching ensures minimal transients in the control inputs. This figure shows the benefits of implementing NDI on an exact system, in that the results are exact and ideal, and the control input magnitudes are reasonable.

Lastly, Figure 3.3(c) demonstrates large deviation output tracking from an off-nominal condition with a generalized applied disturbance and the full, system-identified aerodynamic model implemented in simulation. This exogenous applied force and moment disturbance can be seen in Figure 3.4 on the left and come from (A.2.1), whereas the disturbance experienced by NDI, which includes model mismatch due to the construction of feedback linearization and the exogenous disturbance, is shown on the right. As can be seen, the disturbance from the model mismatch between the ideal aerodynamic model for construction and the real aerodynamic model for simulation has a much larger magnitude than the general applied disturbance. This experienced disturbance is more than enough to make NDI unstable, as seen in 3.3(c). Although the output tracking is similar to the reference model for V and β, the remaining outputs are not tracked well. This figure demonstrates the lack of robustness in NDI, especially compared to the nonlinear, robust performance of LQR, as well as the

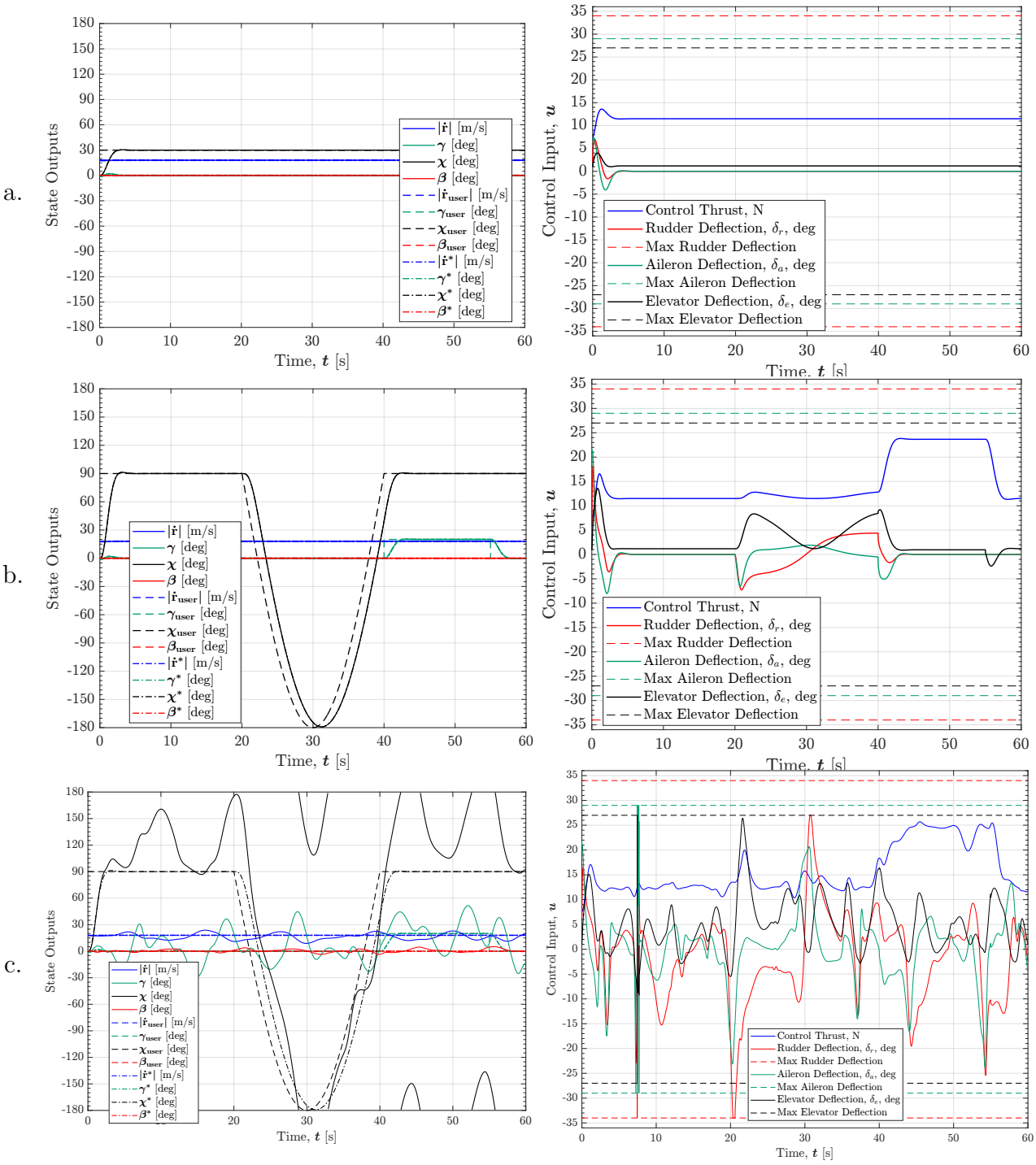

Figure 3.3: Example of NDI implementation (a) Linear output tracking without disturbances, (b) Nonlinear output tracking without disturbances, and (c) Nonlinear output tracking with exogenous and model mismatch disturbances.

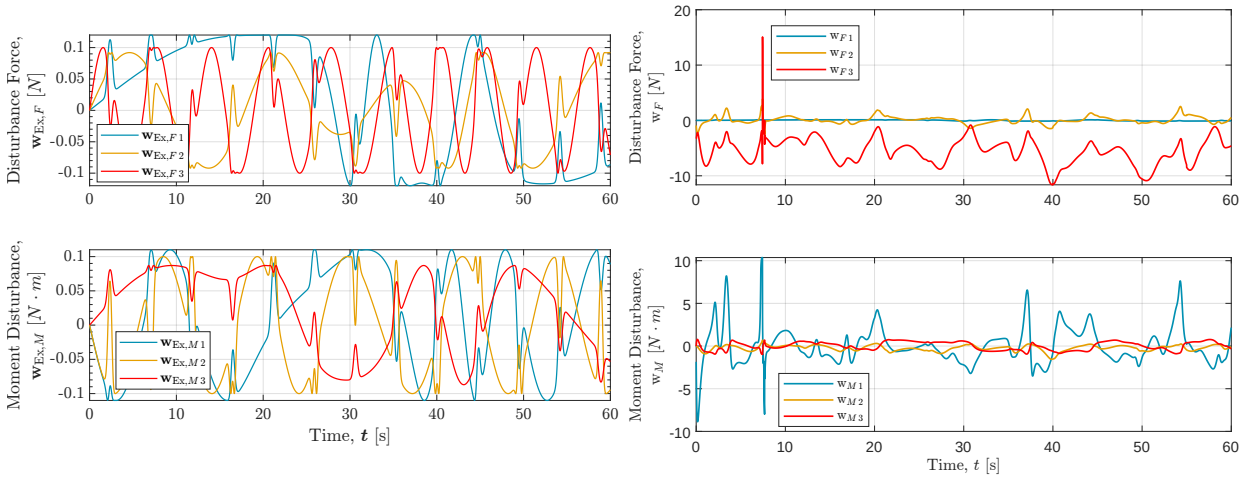

Figure 3.4: Example of exogenous disturbances and model mismatch on NDI: Applied exogenous disturbance (left), and experienced disturbance (right).

challenges of creating a robust control law to wrap around the feedback linearization based on a reduced aerodynamic model.

3.3 Robust Port-Hamiltonian Control

3.3.1 Overview

Hamiltonian mechanics are a way to find the equations of motion of a system by using notions of energy transfer, similar to Lagrangian mechanics. However, classical Hamiltonian mechanics do not account for energy dissipation terms, limiting the applicability of the produced equations of motion since real systems lose energy to drag, friction, and other non-conservative forces. A modification to the classical Hamiltonian mechanics presented in [47] adds energy "ports" to the systems, which provides avenues for the energy to leave and enter the system. Hence, port-Hamiltonian mechanics are a restructuring of the equations of motion into a new form based on energy transfer and dissipation.

The input-state-output form of a port-Hamiltonian system [5] is

$$\begin{aligned} \dot{\boldsymbol{x}} &= \left(\boldsymbol{\mathcal{J}}(\boldsymbol{x}) - \boldsymbol{\mathcal{R}}\boldsymbol{x})\right) \nabla \boldsymbol{\mathcal{H}}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\boldsymbol{u} \\ \boldsymbol{y} &= \boldsymbol{g}(\boldsymbol{x})^\top \nabla \boldsymbol{\mathcal{H}}(\boldsymbol{x}) \end{aligned} \tag{3.3.1}$$

where $\boldsymbol{x}(t) \in \mathbb{R}^n$ is the state, $\boldsymbol{u}(t) \in \mathbb{R}^m$ is the input, $\boldsymbol{y}(t) \in \mathbb{R}^m$ is the passive output of the system, $\boldsymbol{\mathcal{J}}(\boldsymbol{x}) = -\boldsymbol{\mathcal{J}}(\boldsymbol{x})^\top$ is the skew symmetric structure matrix, $\boldsymbol{\mathcal{R}}(\boldsymbol{x}) = \boldsymbol{\mathcal{R}}(\boldsymbol{x})^\top \succcurlyeq 0$ is the symmetric positive semi-definite dissipation matrix, $\boldsymbol{\mathcal{H}}(\boldsymbol{x})$ is the scalar-valued Hamiltonian that often is the system's total energy, the ∇ operator denotes a column-wise derivative, and $\boldsymbol{g}(\boldsymbol{x})$ is the input matrix.

The physical interpretation of (3.3.1) relates to how the energy flows in the system. The $\boldsymbol{\mathcal{J}}(\boldsymbol{x})$ term is from Hamiltonian mechanics, which ensures the energy flows between different states. The $-\boldsymbol{\mathcal{R}}(\boldsymbol{x})$ term is the dissipation term that comes from the "ports", allowing energy to flow out of the system. Lastly, the $\boldsymbol{g}(\boldsymbol{x})\boldsymbol{u}$ term is the input term that allows energy to flow into the system, through control, disturbances, or other inputs. Transforming other forms of the equations of motion into the port-Hamiltonian structure reveals intuitive relations between components that otherwise would be difficult to understand.

The port-Hamiltonian formalism is broadly applicable in engineering, and has been applied to many aerospace systems. Notably, the port-Hamiltonian form was first applied to a fixed-wing aircraft in [9] to construct a passivity-based control law, and it was recently applied in [51] to construct an input-to-output stabilizing control law.

The construction of the robust port-Hamiltonian control law (RPH) in this work was completed by Willebeek-LeMair in [50], which uses a similar process to the construction of the control law in [51]. That is, a change of variables was proposed to convert a closed-loop aircraft system into the structure of a port-Hamiltonian system, which provides a more interpretable control law result. The robust description of the control law came from constructing the system using ISS techniques, and then ensuring that the sufficient conditions were met after the analysis was completed.

3.3.2 Guarantees

As described in Section 2.3.3, the guarantees associated with ISS is robustness to uniformly bounded inputs, which in this work are disturbances.

3.3.3 Derivation

The purpose of the following derivation is to turn a cascaded nonlinear system into a port-Hamiltonian structure with ISS.

Therefore, start with a cascaded nonlinear system

$$\begin{aligned}
\dot{\boldsymbol{x}}_1 &= [\mathbb{I}_{n_1} \quad \mathbf{0}_{n_1\times(n_2-n_1)}]\boldsymbol{x}_2 + \mathbf{g}_1(\boldsymbol{x}_1)\mathbf{w} \\
\dot{\boldsymbol{x}}_2 &= [\mathbb{I}_{n_2} \quad \mathbf{0}_{n_2\times(n_3-n_2)}]\boldsymbol{x}_3 + \mathbf{g}_2(\boldsymbol{x}_1,\boldsymbol{x}_2)\mathbf{w} \\
&\vdots \\
\dot{\boldsymbol{x}}_{N-1} &= [\mathbb{I}_{n_{N-1}} \quad \mathbf{0}_{n_{N-1}\times(n_N-n_{N-1})}]\boldsymbol{x}_N + \mathbf{g}_{N-1}(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_{N-1})\mathbf{w} \\
\dot{\boldsymbol{x}}_N &= \boldsymbol{u} + \mathbf{g}_N(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_N)\mathbf{w}
\end{aligned} \tag{3.3.2}$$

where $\boldsymbol{x}_j \in \mathbb{R}^{n_j}$, $j \in \{1,\ldots,N\}$, is the state of the jth layer, $\boldsymbol{x} = [\boldsymbol{x}_1^\top,\ \boldsymbol{x}_2^\top,\ \cdots,\ \boldsymbol{x}_N^\top]^\top$ is the state of the system, $\boldsymbol{u}$ is the control input, and $\mathbf{w}$ is the unknown disturbance. Note, each layer must have the same number or more states than the preceding layer, i.e. $n_{j+1} \geq n_j$. If the number of states matches between layers, i.e. $n_{j+1} = n_j$, then the matrix $[\mathbb{I}_{n_j}\mathbf{0}_{n_j\times(n_{j+1}-n_j)}]$ reduces to $\mathbb{I}_{n_j}$. The difficulty in applying feedback linearization to a system of the form (3.3.2) is that the disturbance is applied to every cascaded state, most of which are unmatched with a control input. These unmatched disturbances can directly cause a system to become unstable, especially for systems without robustness guarantees.

The reason this construction approach was taken is to apply an ISS control law to a feedback linearizable system. Feedback linearization, as discussed in Section 3.2.3, results in an integrator decoupled system where the control input $\boldsymbol{u}$ only appears in the final term. The only difference from the feedback linearized system in Section 3.2.3 and (3.3.2) is the presence of the disturbance term. Therefore, this approach was successfully applied to a feedback linearized system since both shared the same integrator cascade structure, and therefore a well-defined relative degree.

The desired port-Hamiltonian structure after the transformation is

$$\frac{d}{dt}\begin{bmatrix}\boldsymbol{\xi}_1\\ \boldsymbol{\xi}_2\\ \boldsymbol{\xi}_3\\ \vdots\\ \boldsymbol{\xi}_N\end{bmatrix} = \begin{bmatrix} -\boldsymbol{\mathcal{R}}_{11}(t,\boldsymbol{\xi}_1) & [\mathbb{I}_{n_1}\ \ \mathbf{0}]\boldsymbol{P}_2^{-1} & \mathbf{0} & \cdots & \mathbf{0}\\ -\boldsymbol{P}_2^{-1}[\mathbb{I}_{n_1}\ \ \mathbf{0}]^\top & -\boldsymbol{\mathcal{R}}_{22}(t,\boldsymbol{\xi}_1,\boldsymbol{\xi}_2) & [\mathbb{I}_{n_2}\ \ \mathbf{0}]\boldsymbol{P}_3^{-1} & \cdots & \mathbf{0}\\ \mathbf{0} & -\boldsymbol{P}_3^{-1}[\mathbb{I}_{n_2}\ \ \mathbf{0}]^\top & -\boldsymbol{\mathcal{R}}_{33}(t,\boldsymbol{\xi}_1,\boldsymbol{\xi}_2,\boldsymbol{\xi}_3) & \cdots & \mathbf{0}\\ \vdots & \vdots & \vdots & \ddots & \vdots\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & -\boldsymbol{\mathcal{R}}_{NN}(t,\boldsymbol{\xi}_1,\ldots,\boldsymbol{\xi}_N)\end{bmatrix}\nabla\boldsymbol{\mathcal{H}}(\boldsymbol{\xi}) + \begin{bmatrix}\boldsymbol{\gamma}_1(t,\boldsymbol{\xi}_1)\\ \boldsymbol{\gamma}_2(t,\boldsymbol{\xi}_1,\boldsymbol{\xi}_2)\\ \boldsymbol{\gamma}_3(t,\boldsymbol{\xi}_1,\boldsymbol{\xi}_2,\boldsymbol{\xi}_3)\\ \vdots\\ \boldsymbol{\gamma}_N(t,\boldsymbol{\xi}_1,\ldots,\boldsymbol{\xi}_N)\end{bmatrix}\mathbf{w} \tag{3.3.3}$$

where $\boldsymbol{\xi}_j$ are closed-loop error states and

$$\boldsymbol{\mathcal{H}}(\boldsymbol{\xi}) = \sum_{j=1}^{N}\frac{1}{2}\boldsymbol{\xi}_j^\top\boldsymbol{P}_j\boldsymbol{\xi}_j \tag{3.3.4}$$

is the closed-loop Hamiltonian.

The structure above looks relatively different from the structure provided for a general port-Hamiltonian structure in (3.3.1). However, by noting the properties of $\boldsymbol{\mathcal{J}}(x)$ (skew symmetric) and $\boldsymbol{\mathcal{R}}(x)$ (positive semi-definite), and substituting in $\mathbf{w}$ for $\boldsymbol{u}$, this form can be reached relatively simply.

The desire of the control law is to be able to input-to-output stabilize $\boldsymbol{\xi}_1 = \boldsymbol{x}_1 - \boldsymbol{x}_1^\star$ with respect to the disturbance $\mathbf{w}$ for some set of states $\boldsymbol{x}_1$ and some user-defined state history $\boldsymbol{x}_1^\star$. The following construction is based off that desire, and relies heavily on being able to match terms between the cascade system (3.3.2) and the port-Hamiltonian structure (3.3.3).

The rate of change of $\boldsymbol{\xi}_1$ along trajectories, also known as the dynamics of $\boldsymbol{\xi}_1$, are

$$\dot{\boldsymbol{\xi}}_1 = \dot{\boldsymbol{x}}_1 - \dot{\boldsymbol{x}}_1^\star = [\mathbb{I}_{n_1}\ \ \mathbf{0}]\boldsymbol{x}_2 + \mathbf{g}_1(\boldsymbol{x}_1)\mathbf{w} - \dot{\boldsymbol{x}}_1^\star \tag{3.3.5}$$

from (3.3.2). Further, from (3.3.3)

$$\dot{\boldsymbol{\xi}}_1 = -\boldsymbol{\mathcal{R}}_{11}(t,\boldsymbol{\xi}_1)\boldsymbol{P}_1\boldsymbol{\xi}_1 + [\mathbb{I}_{n_1}\ \ \mathbf{0}]\boldsymbol{\xi}_2 + \mathbf{g}_1(\boldsymbol{x}_1)\mathbf{w} \tag{3.3.6}$$

By setting these equal to each other, the functions in the port-Hamiltonian structure can be chosen to match the cascade structure.

$$\boldsymbol{\gamma}_1(t,\boldsymbol{\xi}_1) = \mathbf{g}_1(\boldsymbol{x}_1) \tag{3.3.7}$$

and

$$[\mathbb{I}_{n_1}\ \ \mathbf{0}]\boldsymbol{x}_2 - \dot{\boldsymbol{x}}_1^\star = -\boldsymbol{\mathcal{R}}_{11}(t,\boldsymbol{\xi}_1)\boldsymbol{P}_1\boldsymbol{\xi}_1 + [\mathbb{I}_{n_1}\ \ \mathbf{0}]\boldsymbol{\xi}_2 \tag{3.3.8}$$

Defining $\boldsymbol{\xi}_2 = \boldsymbol{x}_2 - \boldsymbol{x}_2^\star$, where $\boldsymbol{x}_2^\star$ is any function of t and $\boldsymbol{x}_1$, rearranges (3.3.8) into

$$[\mathbb{I}_{n_1} \;\; \mathbf{0}]\boldsymbol{x}_2^\star = \dot{\boldsymbol{x}}_1^\star - \boldsymbol{\mathcal{R}}_{11}(t, \boldsymbol{\xi}_1)\boldsymbol{P}_1\boldsymbol{\xi}_1 \tag{3.3.9}$$

The analysis is repeated for $j \in \{2, ..., N-1\}$. The rate of change of $\boldsymbol{\xi}_j = \boldsymbol{x}_j - \boldsymbol{x}_j^\star$ along trajectories of (3.3.2) is

$$\dot{\boldsymbol{\xi}}_j = \dot{\boldsymbol{x}}_j - \dot{\boldsymbol{x}}_j^\star = [\mathbb{I}_{n_j} \;\; \mathbf{0}]\boldsymbol{x}_{j+1} + \mathbf{g}_j(\boldsymbol{x}_1, ..., \boldsymbol{x}_j)\mathbf{w} - \dot{\boldsymbol{x}}_j^\star \tag{3.3.10}$$

which is consistent with (3.3.3) if

$$\boldsymbol{\gamma}_j(t, \boldsymbol{\xi}_1, ..., \boldsymbol{\xi}_j) = \mathbf{g}_j(\boldsymbol{x}_1, ..., \boldsymbol{x}_j) \tag{3.3.11}$$

and

$$[\mathbb{I}_{n_j} \;\; \mathbf{0}]\boldsymbol{x}_{j+1} - \dot{\boldsymbol{x}}_j^\star = -\boldsymbol{P}_j^{-1}[\mathbb{I}_{n_{j-1}} \;\; \mathbf{0}]^\top \boldsymbol{P}_{j-1}\boldsymbol{\xi}_{j-1} - \boldsymbol{\mathcal{R}}_{jj}(t, \boldsymbol{\xi}_1, ..., \boldsymbol{\xi}_j)\boldsymbol{P}_j\boldsymbol{\xi}_j + [\mathbb{I}_{n_j} \;\; \mathbf{0}]\boldsymbol{\xi}_{j+1} \tag{3.3.12}$$

Define $\boldsymbol{\xi}_{j+1} = \boldsymbol{x}_{j+1} - \boldsymbol{x}_{j+1}^\star$, where $\boldsymbol{x}_{j+1}^\star$ is any function of t, and $\boldsymbol{x}_1$ through $\boldsymbol{x}_j$, rearranges (3.3.12) into

$$[\mathbb{I}_{n_j} \;\; \mathbf{0}]\boldsymbol{x}_{j+1}^\star = \dot{\boldsymbol{x}}_j^\star - \boldsymbol{P}_j^{-1}[\mathbb{I}_{n_{j-1}} \;\; \mathbf{0}]^\top \boldsymbol{P}_{j-1}\boldsymbol{\xi}_{j-1} - \boldsymbol{\mathcal{R}}_{jj}(t, \boldsymbol{\xi}_1, ..., \boldsymbol{\xi}_j)\boldsymbol{P}_j\boldsymbol{\xi}_j \tag{3.3.13}$$

Lastly, for $j = N$, the rate of change of $\boldsymbol{\xi}_N = \boldsymbol{x}_N - \boldsymbol{x}_N^\star$ along trajectories of (3.3.2) is

$$\dot{\boldsymbol{\xi}}_N = \dot{\boldsymbol{x}}_N - \dot{\boldsymbol{x}}_N^\star = \boldsymbol{u} + \mathbf{g}_N(\boldsymbol{x}_1, ..., \boldsymbol{x}_N)\mathbf{w} - \dot{\boldsymbol{x}}_N^\star \tag{3.3.14}$$

which is consistent with (3.3.3) if

$$\boldsymbol{\gamma}_N(t, \boldsymbol{\xi}_1, ..., \boldsymbol{\xi}_N) = \mathbf{g}_N(\boldsymbol{x}_1, ..., \boldsymbol{x}_N) \tag{3.3.15}$$

and

$$\boldsymbol{u} = \dot{\boldsymbol{x}}_N^\star - \boldsymbol{P}_N^{-1}[\mathbb{I}_{n_{N-1}} \;\; \mathbf{0}]^\top \boldsymbol{P}_{N-1}\boldsymbol{\xi}_{N-1} - \boldsymbol{\mathcal{R}}_{NN}(t, \boldsymbol{\xi}_1, ..., \boldsymbol{\xi}_N)\boldsymbol{P}_N\boldsymbol{\xi}_N \tag{3.3.16}$$

The control law that converts the cascaded system (3.3.2) into the desired port-Hamiltonian structure (3.3.3) is provided in (3.3.16), and the equations that define the disturbance input matrix γ are (3.3.7), (3.3.11), and (3.3.15).

In order to confirm the sufficient condition for ISS is met, the rate of change of the Hamiltonian function $\boldsymbol{\mathcal{H}}$ along trajectories of the desired port-Hamiltonian system (3.3.3) is taken. The following equations omit arguments for readability and brevity.

$$\dot{\boldsymbol{\mathcal{H}}} = \sum_{j=1}^{N} \frac{1}{2}\boldsymbol{\xi}_j^\top \boldsymbol{P}_j\dot{\boldsymbol{\xi}}_j = \sum_{j=1}^{N} \left(-\boldsymbol{\xi}_j^\top \boldsymbol{P}_j\boldsymbol{\mathcal{R}}_{jj}\boldsymbol{P}_j\boldsymbol{\xi}_j + \boldsymbol{\xi}_j^\top \boldsymbol{P}_j\boldsymbol{\gamma}_j\mathbf{w}\right) \tag{3.3.17}$$

Applying Young's inequality (2.1.8) to the second term of (3.3.17), $\boldsymbol{\xi}_j^\top \boldsymbol{P}_j \boldsymbol{\gamma}_j \mathbf{w}$, results in

$$\begin{aligned}\boldsymbol{\xi}_j^\top \boldsymbol{P}_j \boldsymbol{\gamma}_j \mathbf{w} &= \left(\boldsymbol{\gamma}_j^\top \boldsymbol{P}_j \boldsymbol{\xi}_j\right)^\top \mathbf{w} = \mathbf{w}^\top \left(\boldsymbol{\gamma}_j^\top \boldsymbol{P}_j \boldsymbol{\xi}_j\right) \\ &\leq \frac{1}{2}\mathbf{w}^\top \left(b_j \mathbb{I}\right) \mathbf{w} + \frac{1}{2}\left(\boldsymbol{\gamma}_j^\top \boldsymbol{P}_j \boldsymbol{\xi}_j\right)^\top \left(b_j \mathbb{I}\right)^{-1} \left(\boldsymbol{\gamma}_j^\top \boldsymbol{P}_j \boldsymbol{\xi}_j\right) \\ &\leq \frac{1}{2}\boldsymbol{\xi}_j^\top \boldsymbol{P}_j \left(\frac{1}{b_j}\boldsymbol{\gamma}_j \boldsymbol{\gamma}_j^\top\right) \boldsymbol{P}_j \boldsymbol{\xi}_j + \frac{1}{2} b_j \left\|\mathbf{w}\right\|^2\end{aligned} \tag{3.3.18}$$

which then is substituted into (3.3.17) to find

$$\dot{\boldsymbol{\mathcal{H}}} \leq \sum_{j=1}^{N} \left(-\boldsymbol{\xi}_j^\top \boldsymbol{P}_j \left(\boldsymbol{\mathcal{R}}_{jj} - \frac{1}{2}\frac{1}{b_j}\boldsymbol{\gamma}_j \boldsymbol{\gamma}_j^\top\right) \boldsymbol{P}_j \boldsymbol{\xi}_j + \frac{1}{2} b_j \left\|\mathbf{w}\right\|^2\right) \tag{3.3.19}$$

for any coefficients b_j such that $b_j > 0$ for all j. By defining $b = \sum_{j=1}^{N} b_j$,

$$\dot{\boldsymbol{\mathcal{H}}} \leq \sum_{j=1}^{N} \left(-\boldsymbol{\xi}_j^\top \boldsymbol{P}_j \left(\boldsymbol{\mathcal{R}}_{jj} - \frac{1}{2}\frac{1}{b_j}\boldsymbol{\gamma}_j \boldsymbol{\gamma}_j^\top\right) \boldsymbol{P}_j \boldsymbol{\xi}_j\right) + \frac{1}{2} b \left\|\mathbf{w}\right\|^2 \tag{3.3.20}$$

To ensure the formulation is ISS, choose

$$\boldsymbol{\mathcal{R}}_{jj} = \frac{1}{2} k_j \boldsymbol{P}_j^{-1} + \frac{1}{2}\frac{1}{b_j}\boldsymbol{\gamma}_j \boldsymbol{\gamma}_j^\top \tag{3.3.21}$$

for any $k_j > 0$. Taking $k = \min\{k_j\}_{j=1}^{N}$, substituting (3.3.21) in and noting that the ℓ_∞ norm is the maximum ℓ_2 norm, (3.3.20) becomes

$$\dot{\boldsymbol{\mathcal{H}}} \leq -k\boldsymbol{\mathcal{H}} + \frac{1}{2} b \left\|\mathbf{w}\right\|^2 \leq -k\boldsymbol{\mathcal{H}} + \frac{1}{2} b \left\|\mathbf{w}\right\|_\infty^2 \tag{3.3.22}$$

which is of the same form as (2.3.10) in Theorem 2.3.

Lastly, since $\boldsymbol{\mathcal{H}}$ is quadratic, applying the Rayleigh-Ritz inequality reveals

$$\frac{1}{2}\lambda_{\min}\left(\boldsymbol{P}\right) \left\|\boldsymbol{\xi}\right\|^2 \leq \boldsymbol{\mathcal{H}}(\boldsymbol{\xi}) \leq \frac{1}{2}\lambda_{\max}\left(\boldsymbol{P}\right) \left\|\boldsymbol{\xi}\right\|^2 \tag{3.3.23}$$

where $\boldsymbol{P} = \text{diag}(\boldsymbol{P}_1, ..., \boldsymbol{P}_N)$. Equation (3.3.23) is of the same from of (2.3.9), which fulfills both sufficient conditions of Theorem 2.3. Therefore, using the above construction, (3.3.3) is ISS with respect to disturbance $\mathbf{w}$.

3.3.4 Implementation

The initial implementation of RPH is the same as for NDI.

Applying disturbances to the total forces in the wind frame and total moment changes (2.2.19) and (2.2.20) to

$$\begin{bmatrix} D \\ S \\ L \end{bmatrix} = \begin{bmatrix} -\cos(\alpha)\cos(\beta) & -\sin(\beta) & -\sin(\alpha)\cos(\beta) \\ \cos(\alpha)\sin(\beta) & -\cos(\beta) & \sin(\alpha)\sin(\beta) \\ \sin(\alpha) & 0 & -\cos(\alpha) \end{bmatrix} \begin{bmatrix} X+T \\ Y \\ Z \end{bmatrix} + \mathbf{w}_{\boldsymbol{F}} \tag{3.3.24}$$

and

$$\boldsymbol{M} = \begin{bmatrix} \mathcal{L} \\ \mathcal{M} \\ \mathcal{N} \end{bmatrix} = \frac{1}{2}\rho V^2 S \begin{bmatrix} b & 0 & 0 \\ 0 & \bar{c} & 0 \\ 0 & 0 & b \end{bmatrix} \begin{bmatrix} C_{\mathcal{L}} \\ C_{\mathcal{M}} \\ C_{\mathcal{N}} \end{bmatrix} + \boldsymbol{u}_{\boldsymbol{M}} + \mathbf{w}_{\boldsymbol{M}} \tag{3.3.25}$$

where $\mathbf{w}_{\boldsymbol{F}} = [\mathrm{w}_D \quad \mathrm{w}_S \quad \mathrm{w}_L]^\top$ is the disturbance force and $\mathbf{w}_{\boldsymbol{M}} = [\mathrm{w}_{\mathcal{L}} \quad \mathrm{w}_{\mathcal{M}} \quad \mathrm{w}_{\mathcal{N}}]^\top$ is the disturbance moment. Note the placement of the disturbance: ISS only provides guarantees for additive, bounded disturbances, where the disturbance is added to the forces and moments directly, as opposed to being included through non-additive terms.

With the inclusion of the reduced aerodynamic model (3.2.21) and the disturbance forces and moments (3.3.24) and (3.3.25), the system (3.2.19) is of the form

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{A}(\boldsymbol{x}) + \boldsymbol{B}(\boldsymbol{x})\boldsymbol{u} + \boldsymbol{G}(\boldsymbol{x})\mathbf{w} \\ \boldsymbol{y} &= \boldsymbol{C}\boldsymbol{x} \end{aligned} \tag{3.3.26}$$

where $\mathbf{w} \in \mathbb{R}^w$ is the disturbance vector, $\boldsymbol{G}(\boldsymbol{x}) \in \mathbb{R}^{n\times w}$ is the disturbance matrix, and all other terms are defined the same as for (3.2.23). Similarly for NDI, it is known that (3.2.19) is of the form (3.3.26) due to the reduced aerodynamic model and the additions of disturbance forces and moments.

It is convenient to write (3.3.26) in the general form

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \mathbf{w}) \\ \boldsymbol{y} &= \boldsymbol{C}\boldsymbol{x} \end{aligned} \tag{3.3.27}$$

where the matrices $\boldsymbol{A}(\boldsymbol{x})$, $\boldsymbol{B}(\boldsymbol{x})$, and $\boldsymbol{G}(\boldsymbol{x})$ can be recovered using

$$\begin{aligned} \boldsymbol{A}(\boldsymbol{x}) &= \boldsymbol{f}(\boldsymbol{x}, \mathbf{0}, \mathbf{0}) \\ \boldsymbol{B}(\boldsymbol{x}) &= \frac{\partial \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \mathbf{w})}{\partial \boldsymbol{u}} \\ \boldsymbol{G}(\boldsymbol{x}) &= \frac{\partial \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \mathbf{w})}{\partial \mathbf{w}} \end{aligned} \tag{3.3.28}$$

The following process is similar to that for NDI (3.2.27), with the difference that the kth order derivatives of each component of the output include disturbances.

$$\frac{d}{dt}\left(\left(\frac{d}{dt}\bigg|_{\mathbf{w}=0}\right)^{d_i-1}\right)\boldsymbol{y}_i = \boldsymbol{C}_i\left[\frac{\partial L_{\boldsymbol{A}}^{d_i-1}}{\partial \boldsymbol{x}}\right]\boldsymbol{A}(\boldsymbol{x}) + \boldsymbol{C}_i\left[\frac{\partial L_{\boldsymbol{A}}^{d_i-1}}{\partial \boldsymbol{x}}\right]\boldsymbol{B}(\boldsymbol{x})\boldsymbol{u} + \boldsymbol{C}_i\left[\frac{\partial L_{\boldsymbol{A}}^{d_i-1}}{\partial \boldsymbol{x}}\right]\boldsymbol{G}(\boldsymbol{x})\mathbf{w} \tag{3.3.29}$$

This modification is most easily implemented using Algorithm 2, which produces $\boldsymbol{A}_i^*$, $\boldsymbol{B}_i^*$, and $\boldsymbol{G}_i^*$ for each desired output, as well as the equations to calculate the derivative of every state, just as in (3.2.28), except with different L_{counter} values due to the addition of disturbances in the state dynamics. Therefore, ensure that each L_{counter} saved is noted differently between NDI and RPH.

The result of feedback linearization is a system of the form of (3.3.2) with the difference that $\dot{\boldsymbol{x}}_N$ evolves according to

$$\dot{\boldsymbol{x}}_N = \boldsymbol{A}^*(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N) + \boldsymbol{B}^*(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)\boldsymbol{u} + \mathbf{g}_N^*(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)\mathbf{w} \tag{3.3.30}$$

where $\boldsymbol{A}^*$, $\boldsymbol{B}^*$, and $\boldsymbol{G}^* = \left[\mathbf{g}_1^{*\top}, \mathbf{g}_2^{*\top}, \cdots, \mathbf{g}_N^{*\top}\right]^\top$ are found according to Algorithm 2 and are similar to the NDI formulation. Note that $\boldsymbol{G}^*$ has larger dimensions than both $\boldsymbol{A}^*$ and $\boldsymbol{B}^*$ due to each level of the integrator cascade having a disturbance matrix $\mathbf{g}_j^*$.

The ordering of terms and results produced by this method could have some rows and columns that are zero by the nature of the construction. Changing the state vector (3.2.38) via the permutation (A.3.18) to $[V \;\; \gamma \;\; \chi \;\; \dot{V} \;\; \dot{\gamma} \;\; \dot{\chi} \;\; \beta \;\; \ddot{V} \;\; \ddot{\gamma} \;\; \ddot{\chi} \;\; \dot{\beta}]^\top$ will simplify the computation by reducing the state from 11 quantities to the 4 relevant quantities, $[\ddot{V} \;\; \ddot{\gamma} \;\; \ddot{\chi} \;\; \dot{\beta}]$.

In order to input-output linearize the system, the control input is chosen such that

$$\boldsymbol{u} = [\boldsymbol{B}^*]^{-1}(-\boldsymbol{A}^* + \boldsymbol{v}) \tag{3.3.31}$$

resulting in (3.3.30) becoming

$$\dot{\boldsymbol{x}}_N = \boldsymbol{v} + \mathbf{g}_N^*(\boldsymbol{x})\mathbf{w} \tag{3.3.32}$$

where the desired outputs for RPH are the same as for NDI (3.2.31). The outer loop input $\boldsymbol{v}$ will be constructed to be ISS with respect to the disturbances.

The linear term $\boldsymbol{v}$ is constructed according to (3.3.16), where the following terms are defined

$$\boldsymbol{x}_1 = \begin{bmatrix} V \\ \gamma \\ \chi \end{bmatrix}, \qquad \boldsymbol{x}_2 = \begin{bmatrix} \dot{V} \\ \dot{\gamma} \\ \dot{\chi} \\ \beta \end{bmatrix}, \qquad \text{and} \qquad \boldsymbol{x}_3 = \begin{bmatrix} \ddot{V} \\ \ddot{\gamma} \\ \ddot{\chi} \\ \dot{\beta} \end{bmatrix} \tag{3.3.33}$$

As noted in (3.2.31), the outputs V, γ, and χ are third order, while β is second order. Therefore, in order to have the final derivatives appear in the same output vector, β is offset by one state $\boldsymbol{x}$. This offset is why (3.3.3) has $[\mathbb{I}_{(\cdot)}\mathbf{0}]\boldsymbol{P}_{(\cdot)}^{-1}$ instead of only $\mathbb{I}_{(\cdot)}\boldsymbol{P}_{(\cdot)}^{-1}$ - the $\mathbf{0}$ term enables offset outputs.

As is intuitive, the user-defined desired outputs for $\boldsymbol{x}_1$ are

$$\boldsymbol{x}_1^\star = \begin{bmatrix} V^\star \\ \gamma^\star \\ \chi^\star \end{bmatrix}, \qquad \dot{\boldsymbol{x}}_1^\star = \begin{bmatrix} \dot{V}^\star \\ \dot{\gamma}^\star \\ \dot{\chi}^\star \end{bmatrix}, \qquad \ddot{\boldsymbol{x}}_1^\star = \begin{bmatrix} \ddot{V}^\star \\ \ddot{\gamma}^\star \\ \ddot{\chi}^\star \end{bmatrix}, \qquad \text{and} \qquad \dddot{\boldsymbol{x}}_1^\star = \begin{bmatrix} \dddot{V}^\star \\ \dddot{\gamma}^\star \\ \dddot{\chi}^\star \end{bmatrix} \tag{3.3.34}$$

which were defined in the same way as (3.2.42), with the addition of $\beta^\star$ which is included in $\boldsymbol{x}_2^\star$

The disturbance terms $\boldsymbol{g}_1(\boldsymbol{x}_1)$, $\boldsymbol{g}_2(\boldsymbol{x}_1, \boldsymbol{x}_2)$, and $\boldsymbol{g}_3(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3)$ are defined by partitioning $\boldsymbol{G}(\boldsymbol{x})$ into appropriately sized blocks to match $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, and $\boldsymbol{x}_3$. Therefore, $\boldsymbol{g}_1(\boldsymbol{x}_1)$ was defined as the first three rows of $\boldsymbol{G}(\boldsymbol{x})$, $\boldsymbol{g}_2(\boldsymbol{x}_1, \boldsymbol{x}_2)$ was defined as the next four rows, and $\boldsymbol{g}_3(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3)$ was defined as the last four rows.

The final initialization step is for the tuning parameters. The parameters are designed in three categories: error damping parameters, interconnection parameters, and gain parameters. The error damping parameters k_1, k_2, k_3 are designed to tune how much damping the error tracking has, where large $k_{(\cdot)}$ values increase error damping and decreases overshoots. The interconnection parameters p_1, p_2, p_3 are designed to tune how much energy is traded between different outputs from $\boldsymbol{\mathcal{J}}$, where large $p_{(\cdot)}$ values increase the energy tradeoff, or interconnection of the system. The gain parameters μ_1, μ_2, μ_3 are designed to tune the sensitivity of the control law to error, where large $\mu_{(\cdot)}$ values increase the aggressiveness of the control law. Changing each parameter individually will change the control law performance, although the ratio of the interconnection parameters also is important for control law performance. The implemented parameter values can be found in (A.3.19).

The following matches the process described in the RPH Derivation Section (3.3.3), but the specific matrices are formed to provide an implementable solution. This is done sequentially as in the derivation.

1. Define the first disturbance matching condition $\boldsymbol{\gamma}_1$ (3.3.7)

$$\boldsymbol{\gamma}_1 = \boldsymbol{g}_1$$

2. Define the first interconnection $\boldsymbol{P}_1$ and damping matrix terms $\boldsymbol{\mathcal{R}}_{11}$

$$\begin{aligned}\boldsymbol{P}_1 &= p_1 \mathbb{I}_3 \\ \boldsymbol{\mathcal{R}}_{11} &= \frac{k_1}{2}\boldsymbol{P}_1^{-1} + \frac{\mu_1}{2}\text{diag}\left(\left[m^{-2}, \quad (mV)^{-2}, \quad (mV\cos\gamma)^{-2}\right]\right)\end{aligned}$$

3. Define the desired state $\boldsymbol{x}_2^\star$ (3.3.9)

$$\boldsymbol{x}_2^\star = \left[\left(\dot{\boldsymbol{x}}_1^\star - \boldsymbol{\mathcal{R}}_{11}\boldsymbol{P}_1(\boldsymbol{x}_1 - \boldsymbol{x}_1^\star)\right)^\top \quad \beta^\star\right]^\top$$

4. Define the second disturbance matching condition $\boldsymbol{\gamma}_2$

$$\boldsymbol{\gamma}_2 = \boldsymbol{g}_2 - \frac{\partial \boldsymbol{x}_2^\star}{\partial \boldsymbol{x}_1}\boldsymbol{g}_1$$

5. Define the second interconnection $\boldsymbol{P}_2$ and damping matrix terms $\boldsymbol{\mathcal{R}}_{22}$

$$\begin{aligned}\boldsymbol{P}_2 &= p_2 \mathbb{I}_4 \\ \boldsymbol{\mathcal{R}}_{22} &= \frac{k_2}{2}\boldsymbol{P}_2^{-1} + \frac{\mu_2}{2}\left(\boldsymbol{\gamma}_2\boldsymbol{\gamma}_2^\top\right)\end{aligned}$$

6. Define the desired state $\boldsymbol{x}_3^\star$

$$\begin{aligned}\boldsymbol{x}_3^\star = &\frac{\partial \boldsymbol{x}_2^\star}{\partial \boldsymbol{x}_1^\star}\dot{\boldsymbol{x}}_1^\star + \frac{\partial \boldsymbol{x}_2^\star}{\partial \dot{\boldsymbol{x}}_1^\star}\ddot{\boldsymbol{x}}_1^\star + \frac{\partial \boldsymbol{x}_2^\star}{\partial \beta^\star}\dot{\beta}^\star + \frac{\partial \boldsymbol{x}_2^\star}{\partial \boldsymbol{x}_1}[\mathbb{I}_3 \quad \boldsymbol{0}_{3\times 1}]\boldsymbol{x}_2 \\ &- \boldsymbol{P}_2^{-1}[\mathbb{I}_3 \quad \boldsymbol{0}_{3\times 1}]\boldsymbol{P}_1(\boldsymbol{x}_1 - \boldsymbol{x}_1^\star) - \boldsymbol{\mathcal{R}}_{22}\boldsymbol{P}_2(\boldsymbol{x}_2 - \boldsymbol{x}_2^\star)\end{aligned}$$

7. Define the third disturbance matching condition $\boldsymbol{\gamma}_3$

$$\boldsymbol{\gamma}_3 = \boldsymbol{g}_3 - \frac{\partial \boldsymbol{x}_3^\star}{\partial \boldsymbol{x}_1}\boldsymbol{g}_1 - \frac{\partial \boldsymbol{x}_3^\star}{\partial \boldsymbol{x}_2}\boldsymbol{g}_2$$

8. Define the second interconnection $\boldsymbol{P}_2$ and damping matrix terms $\boldsymbol{\mathcal{R}}_{22}$

$$\begin{aligned}\boldsymbol{P}_2 &= p_2\mathbb{I}_4 \\ \boldsymbol{\mathcal{R}}_{22} &= \frac{k_2}{2}\boldsymbol{P}_2^{-1} + \frac{\mu_2}{2}\left(\boldsymbol{\gamma}_2\boldsymbol{\gamma}_2^\top\right)\end{aligned}$$

9. Define the third interconnection $\boldsymbol{P}_3$ and damping matrix terms $\boldsymbol{\mathcal{R}}_{33}$

$$\begin{aligned}\boldsymbol{P}_3 &= p_3\mathbb{I}_4 \\ \boldsymbol{\mathcal{R}}_{33} &= \frac{k_3}{2}\boldsymbol{P}_3^{-1} + \frac{\mu_3}{2}\left(\boldsymbol{\gamma}_3\boldsymbol{\gamma}_3^\top\right)\end{aligned}$$

10. Finally, the control law $\boldsymbol{v}$ is

$$\begin{aligned}\boldsymbol{v} = &\frac{\partial \boldsymbol{x}_3^\star}{\partial \boldsymbol{x}_1^\star}\dot{\boldsymbol{x}}_1^\star + \frac{\partial \boldsymbol{x}_3^\star}{\partial \dot{\boldsymbol{x}}_1^\star}\ddot{\boldsymbol{x}}_1^\star + \frac{\partial \boldsymbol{x}_3^\star}{\partial \ddot{\boldsymbol{x}}_1^\star}\dddot{\boldsymbol{x}}_1^\star + \frac{\partial \boldsymbol{x}_3^\star}{\partial \beta^\star}\dot{\beta}^\star + \frac{\partial \boldsymbol{x}_3^\star}{\partial \dot{\beta}^\star}\ddot{\beta}^\star \\ &+ \frac{\partial \boldsymbol{x}_3^\star}{\partial \boldsymbol{x}_1}[\mathbb{I}_3 \quad \boldsymbol{0}_{3\times 1}]\boldsymbol{x}_2 + \frac{\partial \boldsymbol{x}_3^\star}{\partial \boldsymbol{x}_2^\star}\boldsymbol{x}_3 - \boldsymbol{P}_3^{-1}\boldsymbol{P}_2(\boldsymbol{x}_2 - \boldsymbol{x}_2^\star) - \boldsymbol{\mathcal{R}}_{33}\boldsymbol{P}_3(\boldsymbol{x}_3 - \boldsymbol{x}_3^\star)\end{aligned} \tag{3.3.35}$$

Equations (3.3.31) and (3.3.35) were implemented in simulation. Similarly to NDI, control limiting was implemented to limit the magnitude of each calculated control input from exponentially increasing. The value was similarly set at 10^5, which greatly reduced singularities in the control calculation.

The creation of the reference model followed the same process described in Section 3.2.4, starting with (3.2.43).

An example of the implementation for RPH is shown in Figure 3.5. The first example, Figure 3.5(a), demonstrates undisturbed, linear output tracking from an off-nominal condition. Further, since NDI and RPH were constructed based on a foundation of feedback linearization, the exact model used for feedback linearization described in Section 3.2.4 was implemented as the simulation aerodynamic model for the undisturbed simulation examples.

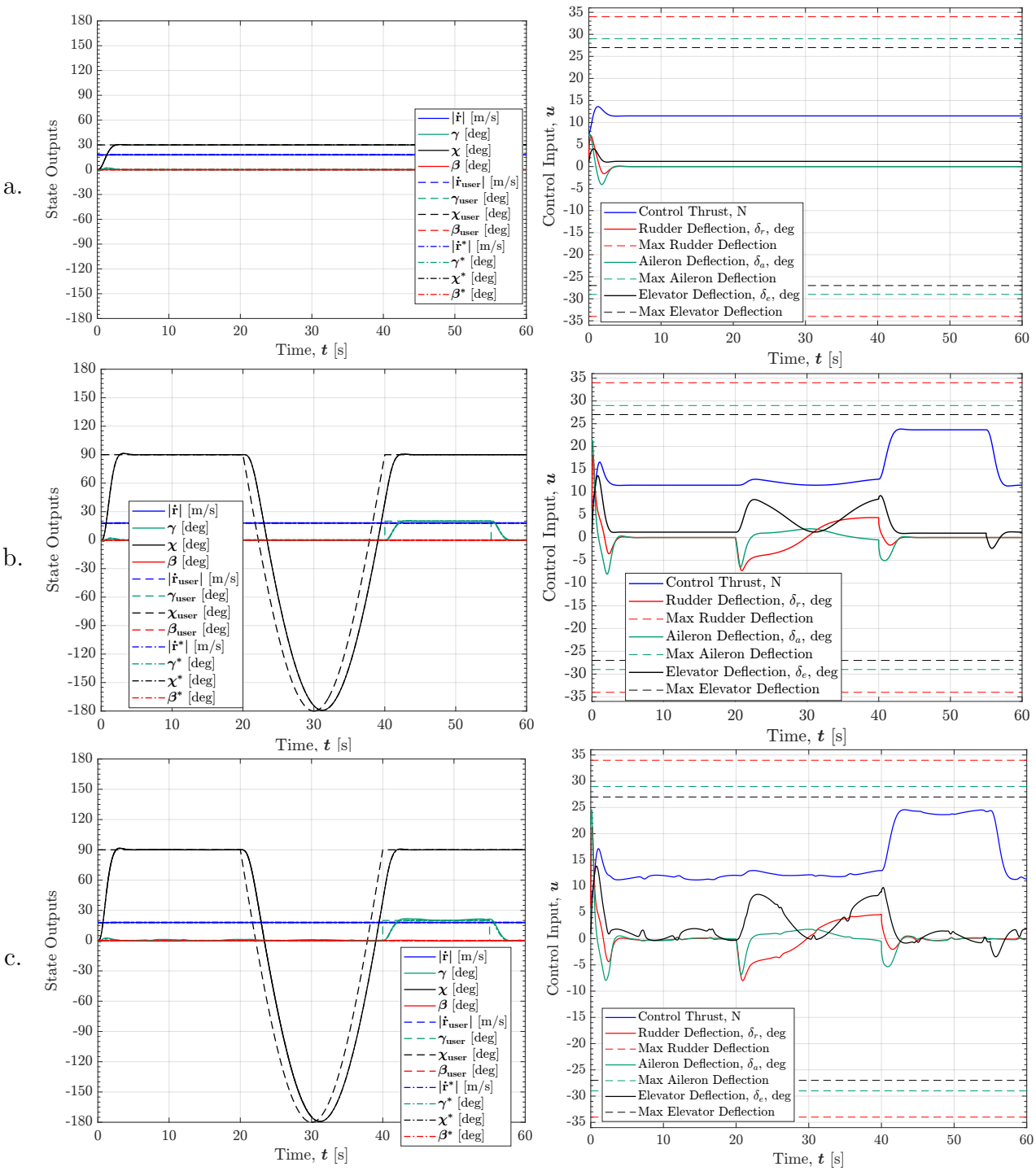

Figure 3.5: Example of RPH implementation (a) Linear output tracking without disturbances, (b) Nonlinear output tracking without disturbances, and (c) Nonlinear output tracking with exogenous and model mismatch disturbances.

Although RPH is a nonlinear control law, it was applied as an outer loop control law on the feedback linearized system.

The second example, Figure 3.5(b), shows undisturbed, large deviation output tracking from an off-nominal condition, where the simulated aerodynamic model is the reduced model that the feedback linearization construction was based on. The identical performance of RPH and NDI from Figure 3.3 overall show that the feedback linearization was successful, and RPH was able to perform equivalently to the LQR outer loop implemented on NDI.

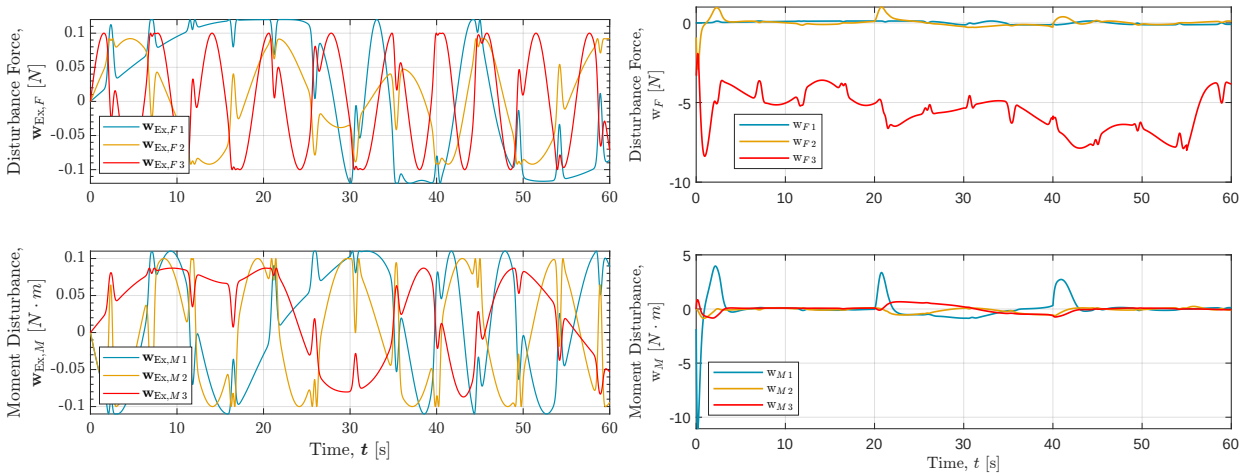

Figure 3.6: Example of exogenous disturbances and model mismatch on RPH: Applied exogenous disturbance (left), and experienced disturbance (right).

Lastly, Figure 3.5(c) demonstrates large deviation output tracking from an off-nominal condition with a generalized applied disturbance and the full, system-identified aerodynamic model implemented in simulation. This exogenous applied force and moment disturbance can be seen in Figure 3.6 on the left and come from (A.2.1), whereas the disturbance experienced by RPH, which includes model mismatch due to the construction of feedback linearization and the exogenous disturbance, is shown on the right. Recall from the construction of RPH that the disturbances were added to the forces (3.3.24) and moments (3.3.25), and therefore bounded, additive disturbances are within ISS. The generalized disturbance in Figure 3.6 is additive, and therefore it is expected that RPH will be robust to the applied disturbances. Compared to Figure 3.4, the experienced disturbance is significantly less for RPH than for NDI, which is likely due to RPH maintaining a stable flight condition which did not excite more "unknown" dynamics to the feedback linearized control laws. However, the greater difference between NDI and RPH is seen in Figure 3.5(c), where RPH performs almost exactly the same with and without disturbances, except for a few, small transients. The control inputs for RPH appear similar to Figure 3.5(b), which shows that RPH actively countered every disturbance applied, while maintaining relatively low magnitude control inputs. This shows that, even on a feedback linearized system, when an additive, uniformly bounded disturbance is applied to RPH, there will be minimal, if any, change to the output tracking.

Chapter 4

Method of Analysis

The method of comparison used in this thesis is simulation studies, where a large number of different simulations are run with varied parameters. This allows a full sampling of the possible simulations variations, compared to doing one simulation at a time. However, due to the large amount of data produced, only a few metrics are analyzed from each simulation. For this work, the metrics were L_1 and L_∞ norms and the failure rate.

In order to replicate flight conditions as closely as possible, the following four exogenous disturbances were added: measurement noise, model mismatch, wind, and control sampling discretization with time delay. These four were chosen due to the realistic and important impacts that each disturbance applies.

4.1 Control Law Comparison

The three control design methods being compared provide static state feedback control laws for time-invariant systems. LQR design produces a linear control law, which provides robustness guarantees for the closed-loop LTI system in the form of minimum values for the gain and phase margin. NDI is a nonlinear control law that transforms a nonlinear system into a linear system through feedback, but without providing any inherent robustness guarantees. RPH is a nonlinear control law applied to the feedback linearized system which ensures robustness to uniformly bounded disturbances.

For the fixed-wing aircraft flight control application considered here, the metric that performance and robustness will be based on is the inertial velocity vector error. The inertial velocity vector was the chosen metric because the primary objective when designing the control laws was to control inertial velocity. Each control law will track a user defined time history of desired outputs $V^\star(t)$, $\gamma^\star(t)$, $\chi^\star(t)$, and $\beta^\star(t)$ that is the same across all simulations in this work. Figure 4.1 shows the time history of desired outputs via a visualization of the aircraft position history.

The inertial velocity vector error is calculated by finding the difference between the vehicle's actual inertial velocity vector, and the reference inertial velocity vector. Three metrics per simulation that are be used to define and compare robust performance between the control laws: the L_1 norm, the L_∞ norm, and the failure rate. The L_1 norm, indicates the average error across a single simulation run, which is useful for overall tracking performance.

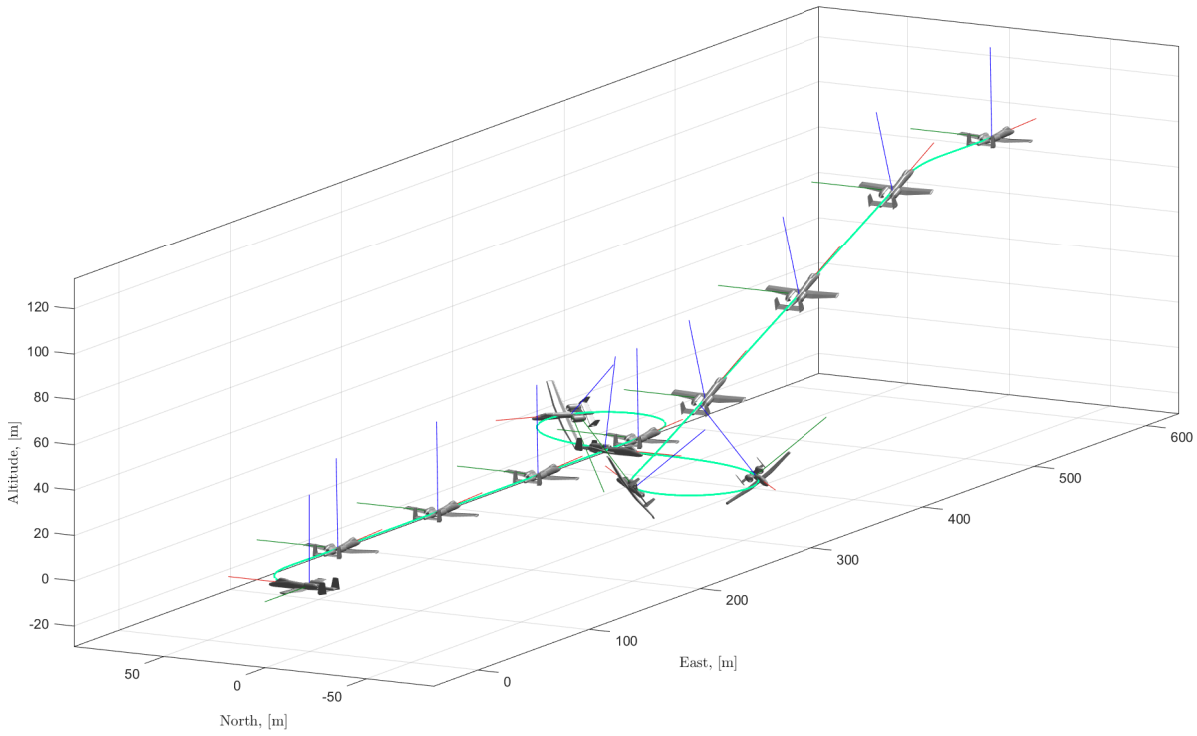

Figure 4.1: Example of ideal output tracking.

The L_∞ norm is computed to determine the largest ℓ_2 norm per simulation, that is, the largest instantaneous velocity tracking error in a given simulation. These two error metrics provide the majority of the performance data that can be gathered from the simulations, and therefore no other L_p norms are included. Lastly, the failure rate would display the number of times a control law failed in a simulation study.

The feedback transformation that converts a nonlinear control system into a linear system is not guaranteed to hold globally. This thesis considers robustness to uncertain inputs in general, rather than model parameter uncertainty specifically. The closed-loop system's sensitivity to these disturbances can result in large state excursions for which the control effectiveness matrix may become singular. For this reason, control failures provide some measure of the non-robustness of a control law. If the transformation becomes undefined as the system state evolves, then the control method will fail. The failure condition for the simulation was if the reciprocal of the condition number of the inverted matrix became very small, or below 10^{-16}. If this happened, the simulation for the specific control law was stopped and a "control failure" outcome was recorded for the simulation rather than the two velocity error measures. This approach provides an assessment of robustness, since a more robust control law should fail less often. Separately, since LQR does not use matrix inversion, the determination for an LQR control law failure was if the resultant error norm was exceedingly large, or above 10^{30}. Only two LQR control law failures occurred during this work (over 20,000 individual simulations). These failures occurred due to the initial

state lying outside the region of attraction of the LQR controller, which only stabilizes the system locally. No other LQR control law error norms across the entire simulation testing of this work exceeded an order of magnitude above the LQR mean for each norm for each case.

The expected results of the simulation are that RPH will have the lowest error in all metrics, including having a negligible or nonexistent failure rate due to the nonlinear nature and ISS of RPH. However, if the applied, non-additive disturbances have a magnitude, or combined effect that is too large, then RPH may have both higher error and higher failure rate. The challenge of the chosen disturbances for RPH is that all of the disturbances, individually, are non-additively bounded. The next best control law is expected to be LQR due to the robustness guarantees. LQR is expected to have significant error due to the nonlinear behavior of the aircraft, although four applied disturbances are not expected to impact the performance of LQR significantly. Lastly, NDI is expected to do the worst due to the lack of robustness. Although the outer loop LQR may provide some robustness, the inherent model mismatch from the construction of the feedback linearized system is likely going to compound with the applied disturbances to become greater than the LQR outer loop is guaranteed to be robust to, and cause the inversion matrix to become singular often.

4.2 Simulation Set Up

A simulation study was run for each combination of the four applied disturbances (measurement noise, model mismatch, wind, and discretization with time delay), resulting in 15 simulation studies. Each simulation study was run identically to the others, including pre-initialized parameters, with the only difference in the studies being which disturbances were applied. In order to save time, the simulations were run using parallel computing (the `parfor` function in `MATLAB`) which uses more cores in the computer to run different computations in parallel.

Within each simulation study, each simulation that was ran calculated the time history of the aircraft dynamics based on each of the three control laws, calculated individually. For example, in a simulation, the 60 second simulation using LQR was run, then the 60 second simulation with NDI, and then RPH. This allowed each control law to control a vehicle which experienced the same disturbances.

The first parameter defined in a simulation was the initial condition that was uniformly randomized within certain bounds; the randomized initial state ranges can be found in Table A.1. Then, the full time history of all four exogenous disturbances that would be applied to the aircraft were precalculated to ensure each control law simulation received the same disturbance at the same time. Although some of the applied disturbances are typically applied based on the current aircraft state at each time step (such as wind turbulence), it was more important to the author to ensure a consistent comparison between the control laws. Then the state dynamics for each simulation control law was integrated through the full

time history, and the results were saved as three values (L_1 and L_∞ norms of the inertial velocity tracking error and failure rate) per simulation. This integration was accomplished by a Runge-Kutta 4$^{\text{th}}$ order integration scheme (RK4) with a fixed step size. During testing, the results from RK4 were compared to an adaptive step sizing version of RK4, the `MATLAB` function `ode45()`. The results were very similar (<1% difference) except the `ode45()` function would greatly reduce the step size, resulting in individual simulation runs that took over 100 times longer than the fixed-step size RK4 solver for certain configurations.

LQR produces actuator deflections and propeller speed. NDI and RPH produce control moments which were converted into control actuator deflections using the inverse of equation (2.2.21), which is (A.3.9). Thrust is a state of NDI and RPH due to the double dynamic extension needed to enable feedback linearization. Thrust was converted into propeller speed by solving for n from (2.2.18) using the quadratic formula. Control saturation was applied to each control actuator deflection, where the saturation limits are found in Section A.1. Once the saturation was applied, the actuator deflections and propeller speed were passed into the aerodynamic model to calculate the aerodynamic forces and moments. Finally, the angles ψ and χ, for both vehicle output and desired output, were wrapped to $[-\pi, \pi)$ and the angles β and γ, for both vehicle output and desired output, were wrapped to $[-\pi/2, \pi/2)$.

The simulations include four sources of uncertainty: measurement noise, model mismatch, wind, and control discretization with time delay. The signal flow diagram in Figure 4.2 illustrates how each disturbance impacts the system. Measurement noise was applied to the control law by adding a sample from a purely Gaussian distribution to each state. Model mismatch was applied the same way to the aerodynamic model coefficients, except by sampling from a random walk instead of pure noise. Wind was applied to the aircraft body velocities using the von Kármán turbulence model before calculating the aerodynamic model coefficients. Control discretization with time delay was applied to the integration scheme to use the correct control input based on the effects of discretization and time delay.

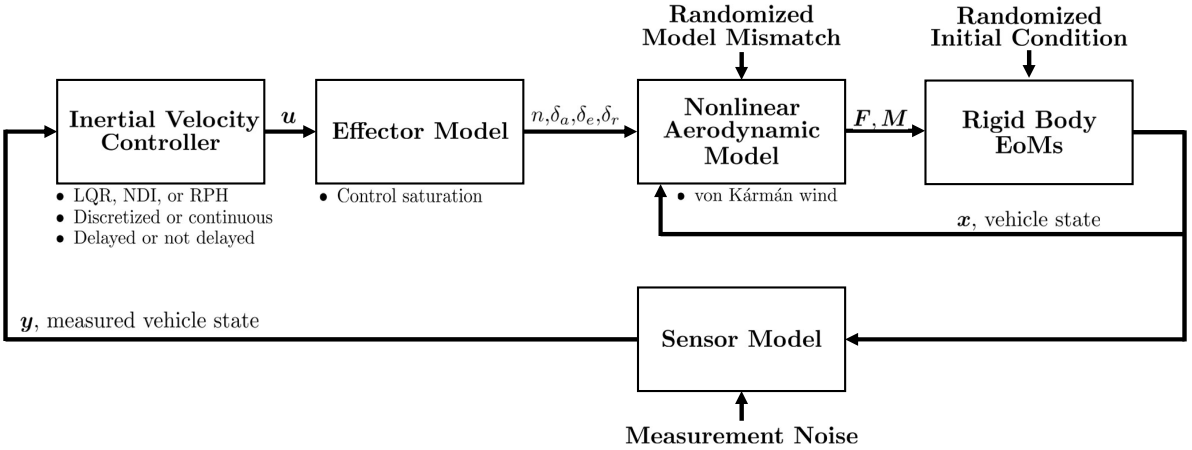

Figure 4.2: Signal flow diagram

In this work, every simulation study had at least 1000 simulation runs. In order to ensure this was a high enough simulation count per simulation study to effectively characterize the effects of uncertainty, six simulations studies were chosen to be re-simulated using the same parameters, but different disturbance time histories, and compared against the previously calculated studies. For each of the six re-runs, the differences in results was not visually evident in the histograms, and each of the calculated means and failure rates were each within 1%. Therefore, 1000 simulations per study was decided to be enough. However, on the primary result presented, the simulation study was run with 10,000 simulations to ensure an even higher confidence result. The only difference between 1000 and 10,000 simulations per study is the "smoothness" of the histogram. Since the analysis can easily be conducted on the 1000-simulation-per-study results, the order of magnitude increase in data collection was not pursued for other simulation studies.

4.3 Disturbance Implementation

This section describes the implementation of the disturbances presented in Section 2.3.1.

4.3.1 Measurement Noise

The measurement noise model used was designed based on flight data. Before any flight data are used for processing, all measured data are passed through a low-pass filter with a cutoff frequency of 10 Hz to remove the high frequency noise. The unfiltered signal less the filtered signal was taken to be the measurement noise distribution. The statistics of that distribution are then recorded for simulation, where a sample is taken from that error distribution and added to the state before the control law calculates the specific control input. The process of adding noise to every state is trivial except for the attitude, described in quaternions. The process for adding noise to quaternions is to use right quaternion multiplication (2.2.7).

Figure 4.3 shows an example of the roll rate measurement error histogram used to gather the statistics in Table A.4.

In the simulation, the applied noise was sampled from a truncated distribution of $\pm 2\sigma$. This means that any value from the Gaussian distribution larger than $\pm 2\sigma$ was resampled. This sampling method was chosen because realistically, measurement noise is bounded because sensors inherently have limitations. Further, ISS guarantees only hold if disturbances are bounded, so the measurement noise was bounded to ensure relevance to formal discussion. Therefore, the author made an arbitrary choice to bound it $\pm 2\sigma$. Bounding the measurement noise in this way is not formally motivated and a future effort should use empirically-motivated bounds.

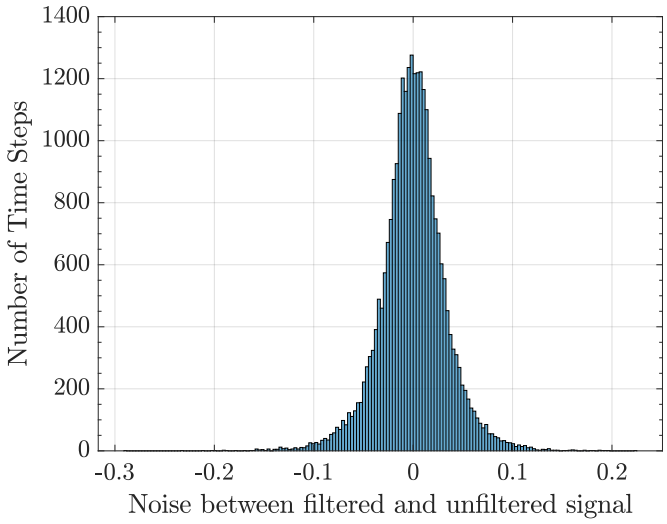

Figure 4.3: Example of noise histogram from measurement data.

4.3.2 Model Mismatch

Model mismatch was calculated based on flight data taken during the system identification of the MTD-4. In order to find the model mismatch, the flight data were analyzed using Equations 8-11 of [42] to find the true force and moment coefficients. Then, the model found from system identification was used to find the force and moment coefficients by substituting in the values from flight. The model coefficients less the flight data coefficients was taken to be the model mismatch error distribution, and the gradient of the model mismatch error across the measurement history was taken to be the model mismatch error rate distribution.

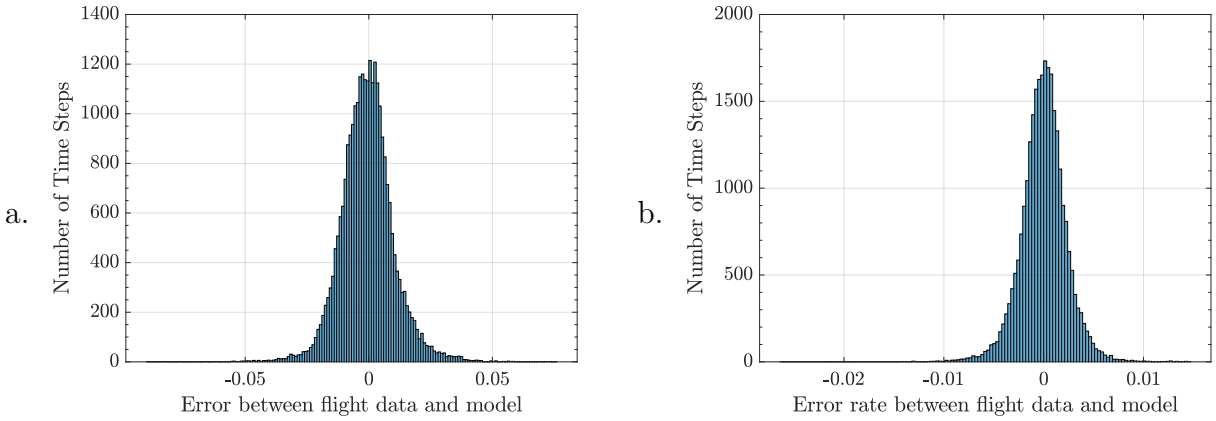

Figure 4.4: (a) Example of model mismatch error histogram, and (b) Example of model mismatch error rate histogram.

Mean values were removed from these differences between the measured and modeled force

and moment coefficients, leaving zero-mean distributions from which the model mismatch terms were sampled in simulation. Examples for the pitching moment coefficient are shown in Figure 4.4. The implementation of model mismatch also included rate limiting on the random variations for the aerodynamic coefficients, so that the stochastic process would be a random walk instead of pure white noise.

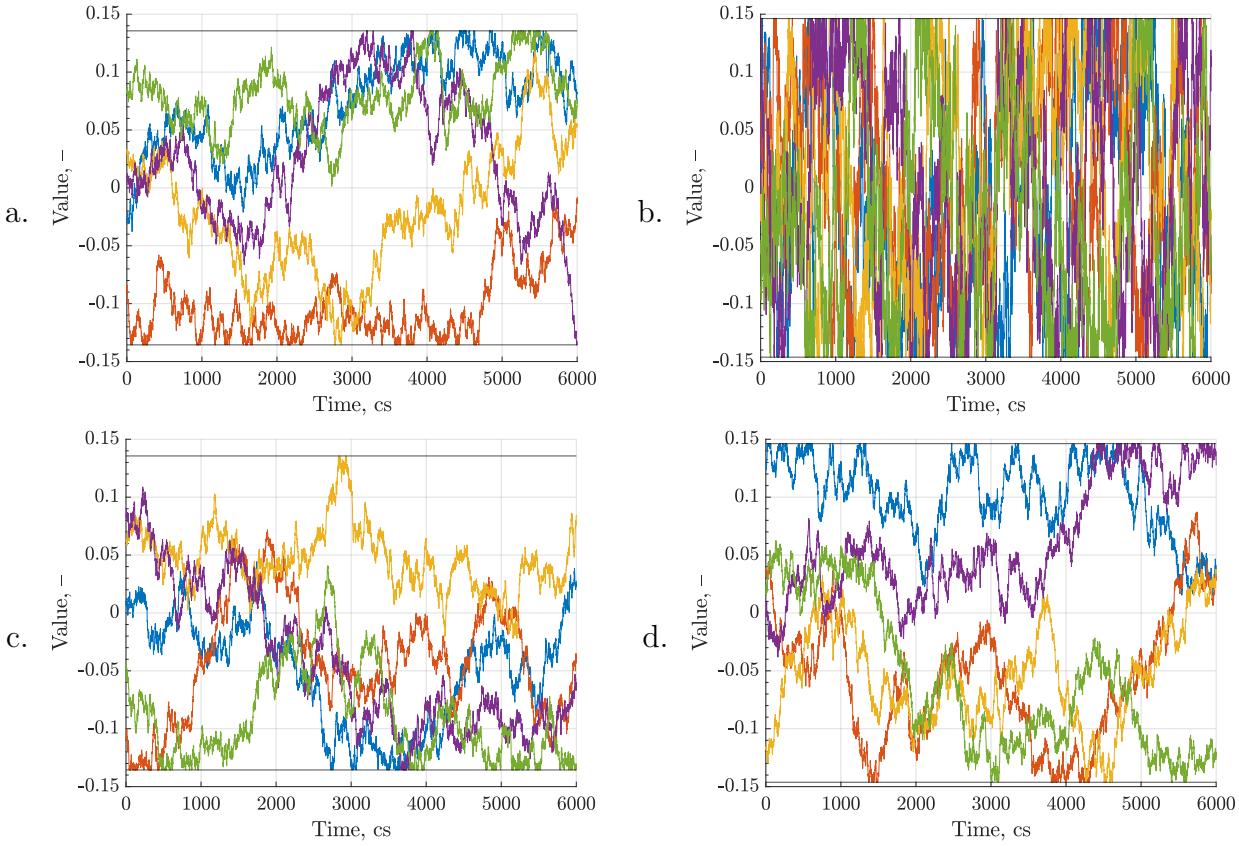

Figure 4.5: (a) Example of applied model mismatch for C_X from flight data, (b) Example of applied model mismatch for C_Z from flight data, (c) Example of applied model mismatch for C_X by choice, and (d) Example of applied model mismatch for C_Z by choice.

Figure 4.5 includes plots showing 5 simulations worth of model mismatch for the C_X and C_Z coefficients. The horizontal lines reflected about the x-axis are the bounds for the range in error variation. The purpose of these graphs is to show the difference in trends due to differences in error standard deviation to error rate standard deviation ratios. These ratios were found by taking the standard deviation of the error σ_{error}, compared to the standard deviations of the error rate σ_{rate}. For C_X, the ratio was approximately 30 to 1, but for every other coefficient, the ratio was approximately 5 to 1 (demonstrated by C_Z). Seen in Figure 4.5(b), if the ratio is too small, then the model mismatch acts like white noise. Since C_X has a ratio of about 30 to 1, the walk from C_X "looks reasonable", in that the model mismatch progresses somewhat smoothly through the entire bounded range, without

jumping too rapidly between the bounds as if white noise or staying too constant as if a bias. Therefore, given the ratio found from flight data for C_X, the error rate standard deviation for all six coefficients were set to exactly 1/30th of their respective error standard deviation. Table A.5 shows the standard deviations of the error from flight data, error rate from flight data, and the precisely chosen error rate.

In simulation, the error standard deviation and the error rate standard deviation were sampled using a resampled Gaussian to be between $\pm\ 2\sigma_{\text{error}}$ or to be within two standard deviations, and $\pm\ 4\sigma_{\text{rate}}$ or to be within four standard deviations, respectively. The error standard deviation of $\pm 2\sigma_{\text{error}}$ was a hard limit, even if the random walk of the error rate was toward greater standard deviation. The purpose of this limiting and resampling was to ensure the model mismatch remained bounded during the duration of the simulation, which meant limiting the maximum and minimum values of mismatch. The limiting to ensure boundedness is reasonable, since, physically, we do not expect aerodynamic coefficients to become unbounded during flight. This was also an important consideration for RPH, since the guarantees only hold for disturbances that are uniformly bounded.

The initial model mismatch was sampled between $\pm 2\sigma_{\text{error}}$ for each aerodynamic coefficient, and then the rest of the time history came from a random walk. The following equations shows the resampling process, where $C_{(\cdot)}$ is the model mismatch error and $\Delta C_{(\cdot)}$ is the model mismatch error walk. This is done for each of the six aerodynamic coefficients.

$$
\begin{aligned}
& C_{(\cdot)}(1) \in\ [-2\sigma_{\text{error}},\ \ 2\sigma_{\text{error}}] \\
\text{for}\ \ k = & [1, t_{\text{final}} - 1] \\
& \Delta C_{(\cdot)}(k+1) \in\ [-4\sigma_{\text{rate}},\ \ 4\sigma_{\text{rate}}] \\
& C_{(\cdot)}(k+1) =\ C_{(\cdot)}(k) + \Delta C_{(\cdot)}(k+1) \\
& \text{Enforce}\ C_{(\cdot)}(k+1) \in\ [-2\sigma_{\text{error}},\ \ 2\sigma_{\text{error}}] \\
& \text{end for}
\end{aligned}
\tag{4.3.1}
$$

4.3.3 Wind

The simulated wind was produced in two parts, steady wind and turbulence, and was parameterized to be qualitatively similar to the wind shown in [10]. This comparison was made because the flight vehicles are the same, and therefore would provide a good estimate of realistic wind conditions. The steady wind was applied by sampling from a Gaussian distribution.

For the turbulence implementation, the von Kármán turbulence model and turbulence parameters described in this section was applied from [1]. The von Kármán model specifies

the power spectral density (PSD) of the turbulence velocities in the body frame using

$$\begin{aligned}\phi_{u_g}(\Omega) &= \sigma_u^2 \frac{2L_u}{\pi} \frac{1}{\left[1+(1.339L_u\Omega)^2\right]^{5/6}} \\ \phi_{v_g}(\Omega) &= \sigma_v^2 \frac{2L_v}{\pi} \frac{1+\frac{8}{3}(2.678L_v\Omega)^2}{\left[1+(2.678L_v\Omega)^2\right]^{11/6}} \\ \phi_{w_g}(\Omega) &= \sigma_w^2 \frac{2L_w}{\pi} \frac{1+\frac{8}{3}(2.678L_w\Omega)^2}{\left[1+(2.678L_w\Omega)^2\right]^{11/6}}\end{aligned} \tag{4.3.2}$$

where Ω is the spatial frequency in rad/m, $\sigma_{(.)}$ is the turbulence severity, and $L_{(.)}$ is the turbulence scale length in ft.

For this simulation, the relation for spatial frequency

$$\Omega = \frac{\omega}{U_{\text{ref}}} \tag{4.3.3}$$

was used to substitute the circular frequency ω in order to correctly apply the fast Fourier transform. This is because the turbulence field produced using Ω is a frozen turbulence field, so therefore in order to convert to a circular frequency, Ω is divided by the aircraft velocity U_{ref} [15].

In order to find this circular frequency ω the frequency resolution was found first where

$$d_\omega = \frac{2\pi}{n_t d_t} \tag{4.3.4}$$

where n_t is the number of time steps in the simulation and d_t is the size of the time step. Then, the circular frequency history ω is found by

$$\omega = d_\omega(0 : n_t - 1) \tag{4.3.5}$$

where $(0 : n_t - 1)$ denotes a sequence of time indices.

The values for turbulence sensitivity and scale lengths were calculated using the below equations which pertain to low altitude flight, or below 1000 feet, in [1]. The turbulence sensitivity, which can be though of as the turbulence intensity, was described by

$$\begin{aligned}\sigma_w &= 0.1u_{20} \\ \sigma_u = \sigma_v &= \frac{\sigma_w}{(0.177+0.000823h)^{0.4}}\end{aligned} \tag{4.3.6}$$

and

$$\begin{aligned}L_u &= \frac{h}{(0.177+0.000823h)^{1.2}} \\ L_v &= \frac{1}{2}L_u \\ L_w &= \frac{1}{2}h\end{aligned} \tag{4.3.7}$$

where u_{20} is the mean wind velocity at 20 feet above the ground in knots, and h is the altitude, in feet. The values of all wind-related parameters implemented in simulation can be found in Section A.1.

In order to move a frequency distribution into the time-domain, a complex and discrete turbulence sample needs to be created with the magnitude related to the power spectral densities. This was done through

$$\boldsymbol{G} = \sqrt{\frac{1}{2}\boldsymbol{\Phi} d_\omega}\ (\cos(\boldsymbol{\eta}) + i\sin(\boldsymbol{\eta})) \tag{4.3.8}$$

where $\boldsymbol{G} = [g_u \quad g_v \quad g_w]^\top$ is the discrete Fourier spectrum for each axis of the wind, $\boldsymbol{\Phi} = [\phi_{u_g} \quad \phi_{v_g} \quad \phi_{w_g}]^\top$ is the vector of von Kármán PSDs, and $\boldsymbol{\eta} \in [-\pi, \pi]$ is the randomly generated phase for the turbulence signal. The phase choice is not significant when generating the turbulence model because there is no pre-defined value for the phase of turbulence signals (the phase could be different for each resulting turbulence signal), so the phase is uniformly randomly sampled between $(-\pi, \pi)$.

The input to inverse fast Fourier transforms is a frequency distribution that has frequencies mirrored about the Nyquist frequency, and includes a bias term. This was done in simulation by setting the second half of the generated $\boldsymbol{G}$ frequency history equal to the conjugate of the reverse order of the first half of $\boldsymbol{G}$. This can arbitrarily be done because this is in the frequency domain, and up until this point every number has been randomly generated.

Then, the inverse fast Fourier transform is applied to the discrete turbulence frequency history, and the output is scaled by the size of the number of time steps. Lastly, the steady wind was added to the time history of the turbulence model. This history was then sampled from at every time step in the simulation as a pure addition to the vehicle's body velocity to create the relative body velocity of the aircraft, which in simulation only impacted the aerodynamic model.

It is important to note that the turbulence model from [1] is in imperial units. Accordingly, wind was computed in imperial units and converted to metric units, before application to the model.

Some examples of the wind model are shown in Figure 4.6, which are qualitatively similar to the wind shown in [10].

4.3.4 Constant-hold Control Discretization with Time Delay

The discretization implemented in this simulation was continuous dynamics, constant hold control in order to best replicate real flight conditions. This was done by simulating the dynamics, reference, and all quantities other than control at the rate of simulation, but Sampling the control input at the slower control rate and holding the input constant for the duration of the corresponding control interval. In order to have a nominal value for

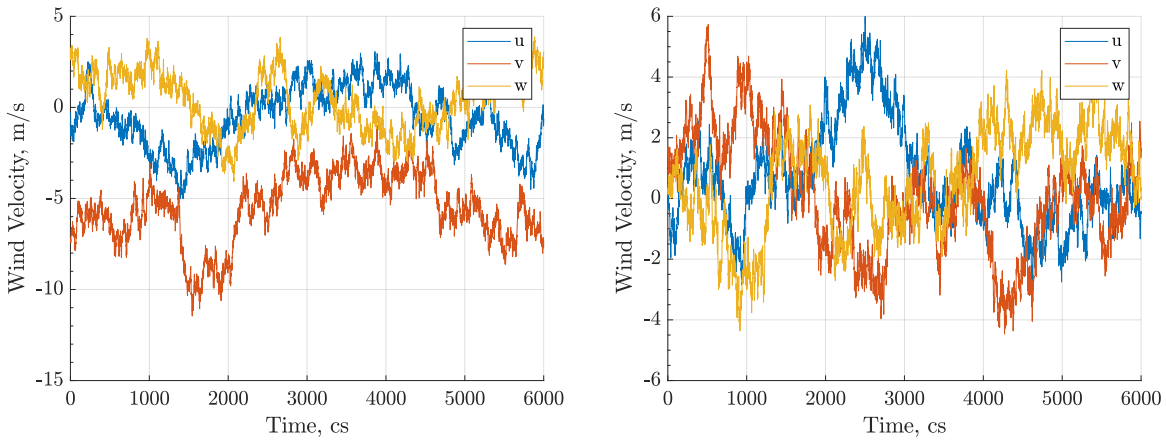

Figure 4.6: Examples of applied wind.

the simulation, a control sampling rate of 25 Hz was chosen whereas the dynamics were integrated at 100 Hz. The value of 25 Hz was chosen since it is a nominal value from flight testing.

A time delay was implemented as well to examine the effect of this phenomenon on closed-loop performance. For this simulation, a time delay was implemented that was uniformly sampled between 0 and 40 milliseconds. This delay was then rounded to the nearest hundredth of a second due to implementation in the 100 Hz simulation. The maximum delay was chosen so that the longest period that would occur without a control update was 2 cycles or 80 milliseconds. This time delay was implemented in simulation by creating a time history of values sampled from the uniform distribution. Each step in this time history was then applied in the numerical integration section of the dynamics to each integration time step, delaying the applied control input by the step amount.

Figure 4.7 gives an example of the effect of discrete-time implementation of a continuous-time control law, with and without time delay. The left column of images in Figure 4.7 is the full control input time history, while the right column shows a close up view of a section of the control input time history to make the differences easier to see. Each case presented was implemented on LQR, as noted by the transients due to switching the trim conditions.

The continuous time implementation shown in Figure 4.7(a) is an example of the control input being updated at the same rate as the aircraft dynamics (100 Hz), which is assumed to be fast enough that the simulation results should match a true continuous time implementation.

Discretization without time delay is shown in Figure 4.7(b) with a sampling rate of 5 Hz for visualization purposes. The impact of pure discretization changes the control inputs from smooth to stair-step functions, with the addition of occasional transients as the delay in the control update causes a correction by the next control update. However, some of the transients caused by trim switching in LQR are reduced by the discretization.

Lastly, Figure 4.7(c) shows the implementation of discretization at 5 Hz with up to a 200 milliseconds delay in control update, which is equivalent to an entire control update delay. It can be seen that the impact of adding time delay to discretization was larger than the impact of adding discretization to the continuous system.

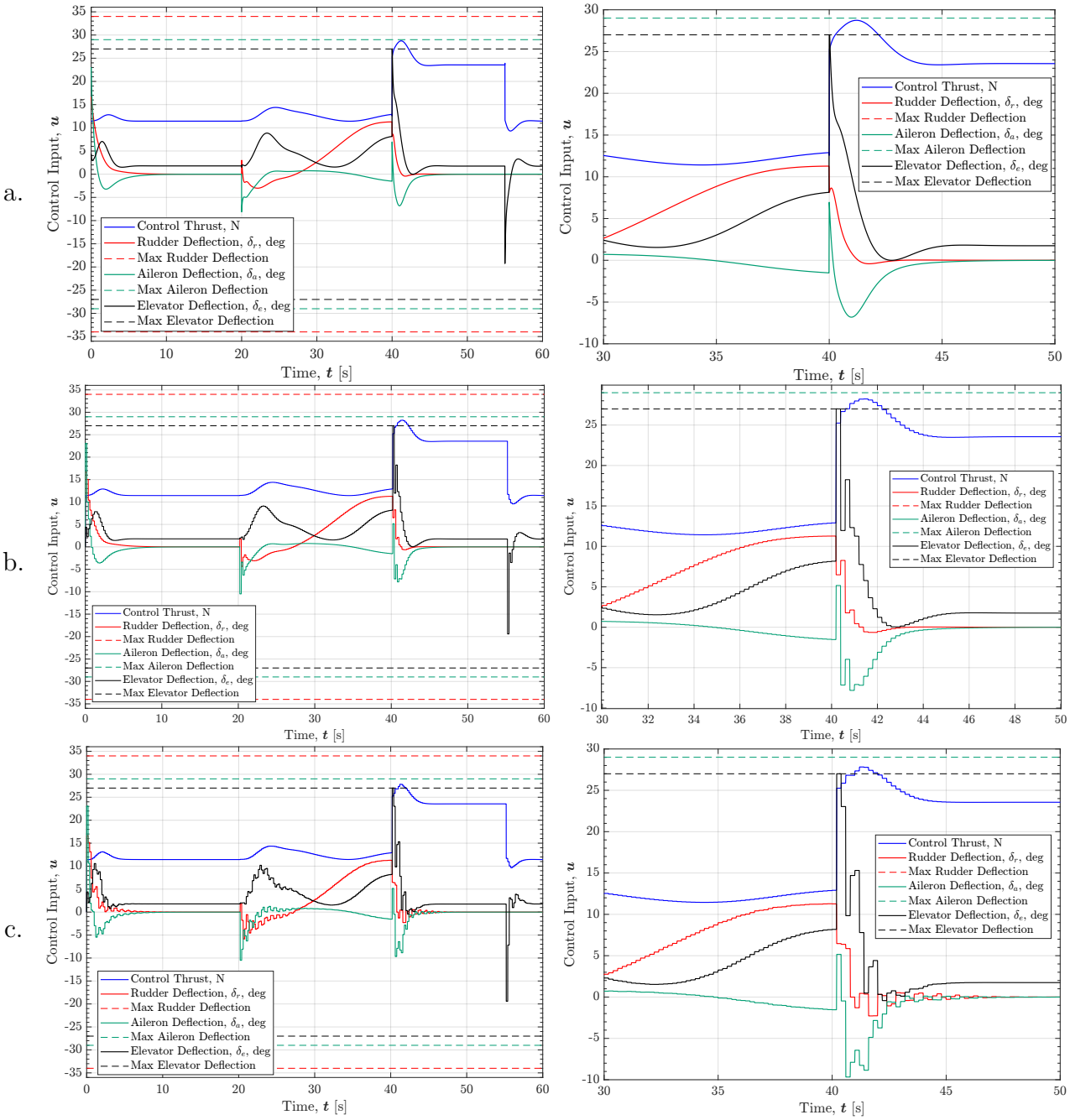

Figure 4.7: Example of Control Discretization for LQR: (a) Continuous Control, (b) Control sampled at 5 Hz, and (c) Control sampled at 5 Hz with up to 200 ms delay.

Chapter 5

Results and Discussion

In this chapter, the results of the simulation studies are presented in a sequence of histograms illustrating the distribution of inertial velocity error. These histograms are analyzed and the relative performance of the studied control laws is discussed. As discussed in Section 4.2, each of the three control laws from Chapter 3 were implemented in simulation studies, from which the comparison metrics (L_1 and L_∞ norms and failure rate) were found for each simulation. The control task was to perform output tracking while experiencing the disturbances described in Section 4.3.

There are a few features to focus on when analyzing error histograms. The first is to understand how the spread and center of each histogram relates to the robustness of each control law. Tighter spreads indicate that the control law enforced more consistent behavior despite the presence of disturbances. Conversely, wider spreads indicate that the control law performed poorly for at least some cases of disturbances. However, the spread of the distribution does not fully describe robustness. If a control law has a very small histogram spread, but at a very high error, then the control law is consistently failing, likely in the same way, as opposed to the control law consistently rejecting all disturbances but still having very high error [34]. Therefore, a control law can be robust but have a wide spread, as long as the histogram has low error performance as well. Therefore, the notion of "repeatability" is introduced, which is when a control law has low variation within the chosen error metric, such that each simulation had very similar error norm results across the set of simulations. A histogram for a control law that has high repeatability would have a low spread and a high peak, regardless of where the center is. The addition of requiring a high peak is the primary difference between a distribution having a small spread and a distribution having high repeatability. Therefore, a control law with high repeatability and low error is considered robust to the applied disturbances.

Another consideration when analyzing error histograms is to consider the histogram tail. In general, error histogram tails point to an occasional, greater failure mode that cannot be compensated for, as opposed to an occasional, smaller error. For this work, the presence of the tails is likely due to the impact of more extreme disturbances that perturb the system towards the limits of the robustness guarantees inherent to each control law. A longer tail shows that a control law is more sensitive to disturbances, compared to a short tail which shows that a control law is not greatly perturbed by the disturbance.

Lastly, the means of each distribution are also reported. These means are primarily used

for comparisons between histograms. Additionally, in the case of multimodal distributions, the mean shows which mode has the greater number of samples as noted by which mode the mean is closer to. The means are also useful for a qualitative analysis, by noting how far from a peak the mean is, compared to how far the mean is from another histogram's mean.

5.1 Combined Disturbances

The most realistic scenario for simulating a physical flight was implemented in this section. All of the previously described disturbances (measurement noise, model mismatch, wind, and discretization with time delay) were applied simultaneously over a 10,000 simulation study.

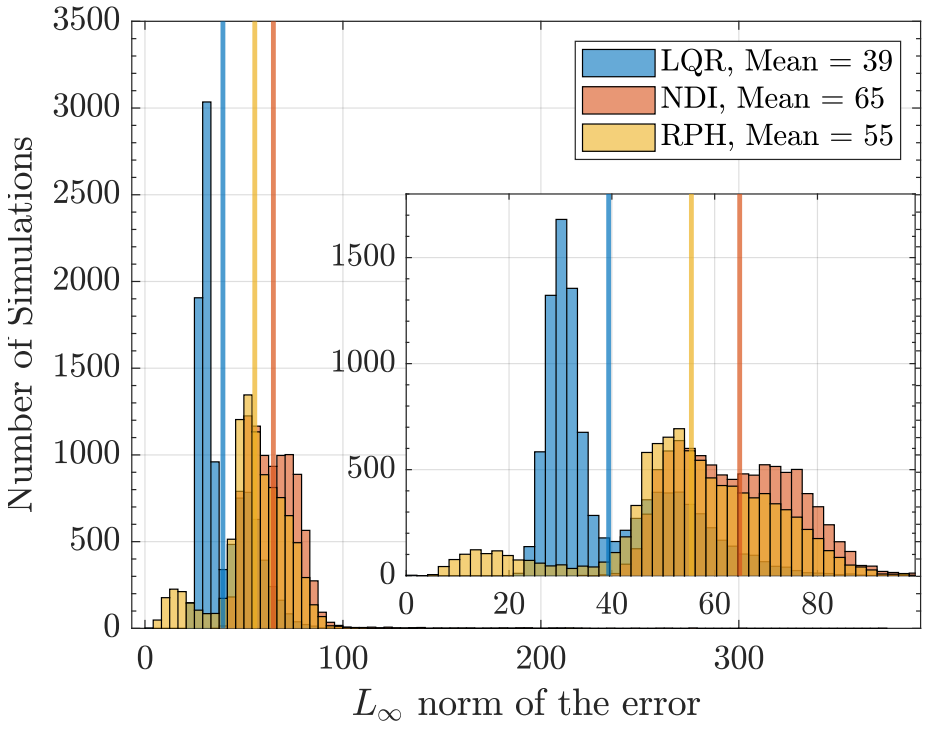

Figure 5.1: L_∞ norm for all disturbances.

Figure 5.1 shows the histogram of the L_∞ norm of the inertial velocity error, which shows the largest error across each simulation; a few standout conclusions can be drawn. First, it can be seen that LQR has the highest repeatability and lowest spread of all three control laws when all disturbances are applied together, even though the error histogram for LQR is bimodal. The bimodal distribution of LQR is likely due to the robustness guarantees and linearity of LQR, since the first peak shows performance within the linear regime, while the second peak shows the degraded performance attained in the nonlinear regime. RPH and NDI perform similarly, in that the majority of the histograms line up with each other. The

similarity in the main shape of the RPH and NDI histograms is likely due to the disturbances greatly surpassing the limits where both control laws remain bounded. It is necessary to note that the L_∞ norm indicates the worst-case excursion of the tracking error over time, as opposed to a time-averaged error. Therefore, the overlapped histograms show that, a majority of the time, RPH and NDI had similar maximum ℓ_2 norms of the error, although RPH tends towards lower norms of the error than NDI.

The second conclusion that can be drawn is in regard to the robust performance of each control law. LQR has the lowest mean error, but the performance of the upper mode overlaps nearly identically with NDI and RPH. This is possibly due to applying too large of disturbances, or more likely, the combination of all disturbances compounded the effect between them. Despite this, the histogram for RPH has a small, lower mode near zero error, which is due to the collection of simulations for which the applied disturbances were bounded and of small magnitude for RPH. This shows that, even when RPH has disturbances applied that are large in magnitude and non-additively bounded, RPH will always perform better than, or at least as well as, NDI. While this is not a high bar, it shows that RPH is more robust, but not necessarily more repeatable, than LQR when both provide outer loop control to a system designed using feedback linearization.

Third, the presence of significant outliers in the data show that NDI can dramatically fail at output tracking. Each histogram in this work was auto-scaled to show every data point. Therefore, even if no data points are easily seen along seemingly blank sections of the figure, outliers are present along the full range. For Figure 5.1, there is at least one outlier at an L_∞ norm error of approximately 400, while smaller outliers from NDI can also be seen at nearly an error of 210, and a few around 160. There are no outliers from LQR or RPH for this figure. The purpose of the inset histogram is to provide a more detailed visualization of the majority of the data.

Complementary information is shown in Figure 5.2, where the L_1 norm indicates the average error across each individual simulation. The most striking data presented shows that the histogram of LQR barely overlaps with the histogram of NDI, but the histogram of RPH overlaps heavily where the histograms of LQR and NDI meet, as opposed to all three control laws overlapping significantly when comparing L_∞ error norm in Figure 5.1. LQR, as with the L_∞ norm comparison, has the best repeatability and performance, except for a small number of simulations in which RPH has better performance. Similarly, NDI, as with the L_∞ norm comparison, has the worst performance, which is significantly separated from the other two control laws. However, the L_1 norm of RPH shows different behavior than the L_∞ norm. The primary analysis regards the histogram shape, since the spread of the RPH L_1 norm histogram is extraordinarily wide. As discussed above, this wide spread alone does not exclude RPH from being a robust control law. To the contrary, the wide spread shows that RPH is sensitive to the applied disturbances, but when the disturbances are within ISS, RPH has the best performance of the control laws. Further, Figure 5.2 shows that, despite commonly having high error, RPH still performs significantly better than NDI even though both were constructed on a feedback linearized system.

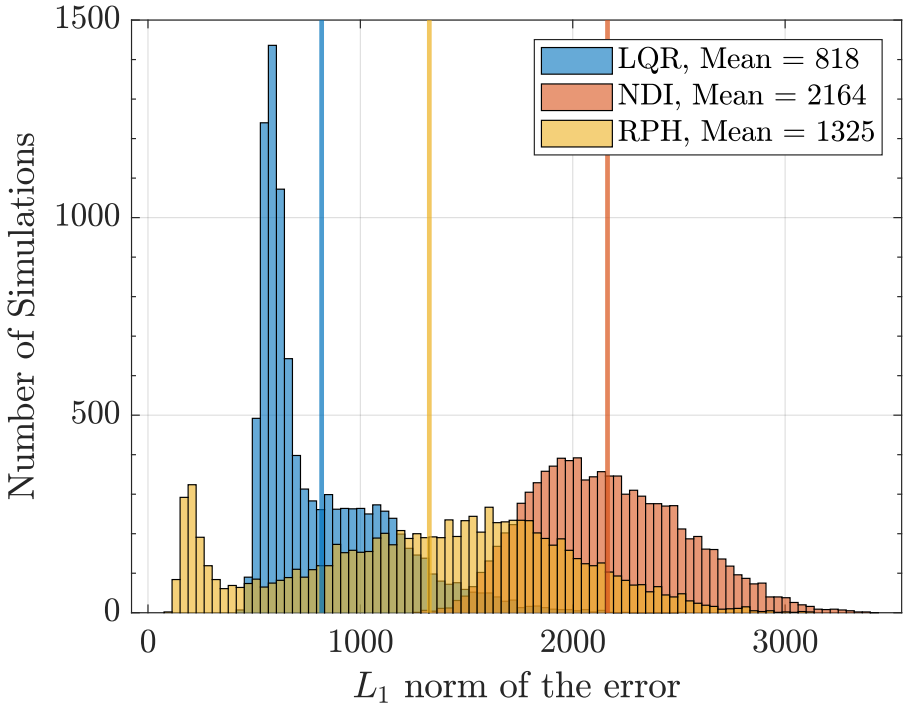

Figure 5.2: L_1 norm for all disturbances.

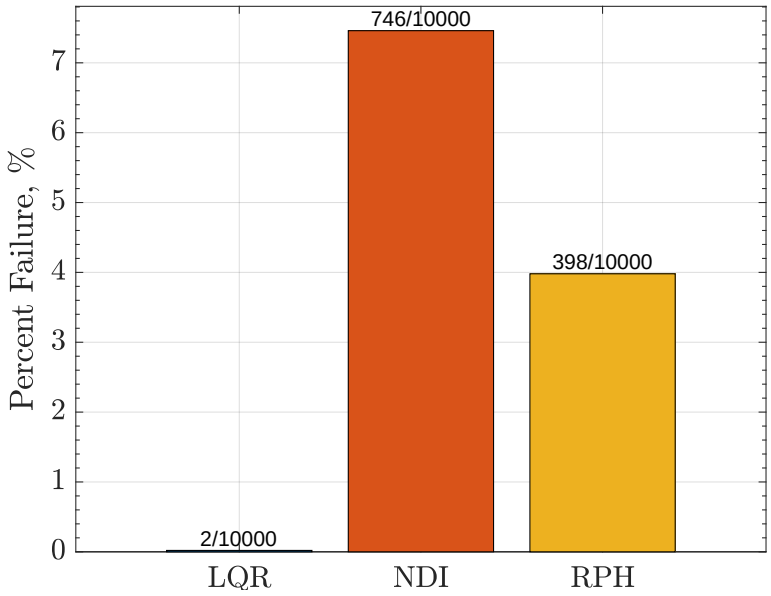

Figure 5.3: Failure for all disturbances.

The control law failure rates in Figure 5.3 complete the analysis of the information presented above. As can be seen, the failure rate for RPH was nearly half of the failure rate for NDI,

from 7.5% to 4%. This aligns with the results above, in which the RPH histogram had high spread but relatively low error compared to NDI. A lower failure rate indicates that the disturbances did not cause the inversion matrix to become singular for RPH as often, whereas the disturbances caused a singular inversion matrix in NDI more frequently. The discussion relating to the LQR failures can be found in Section 4.1.

For the combined disturbance case, the result is clear. The combined impact of all four disturbances stressed each control law more than the control law could account for due to the individual disturbances non-additive nature and magnitude. In general, RPH performed better and failed less than NDI, although RPH performed worse and failed more than LQR. However, based on this presented combined disturbance case, the specific disturbances or combinations thereof that caused the unbounded disturbances cannot be known. Therefore, the following section explores each of the possible combinations to discover which of the disturbance combinations RPH is most sensitive to, and if there are any cases in which RPH clearly outperforms both NDI and LQR.

5.2 Selected Disturbance Combinations

The previous section showed the most realistic scenario that a flight control law might face in flight, but the robustness to individual disturbances was not tested.

The reason for a follow-on section is to find the impact of combined disturbances that maybe have coupled impacts when implemented together, as well as understanding the sensitivity of the various control laws to various types of disturbances. The individual disturbance analysis might suggest modifications that could address particular deficiencies. This individual disturbance testing is still relevant to reducing risk in flight, since some flights may have less severe disturbances present such as a calm flight day or less severe measurement noise due to improved sensors.

The following section presents a selection of all combinations of the four individual disturbances with each other, where similar results are grouped together. The 15 cases, including the previously discussed combined case, were separated into three groups depending on the shape of the histograms for each control law. The discussion for each group includes the figures for each case within the group to enable simpler visual comparison between each case. The case figures were arranged to present an increasing number of disturbances further down the page, and every figure on the left of the group of cases does not include discretization with time delay, while every case on the right does.

5.2.1 Group 1: Least Significant Disturbances

The first group consists of the three cases where all three control laws delivered highly consistent behavior. The cases are

- Discretization with time delay only,

- Measurement noise only, and
- Measurement noise and discretization with time delay.

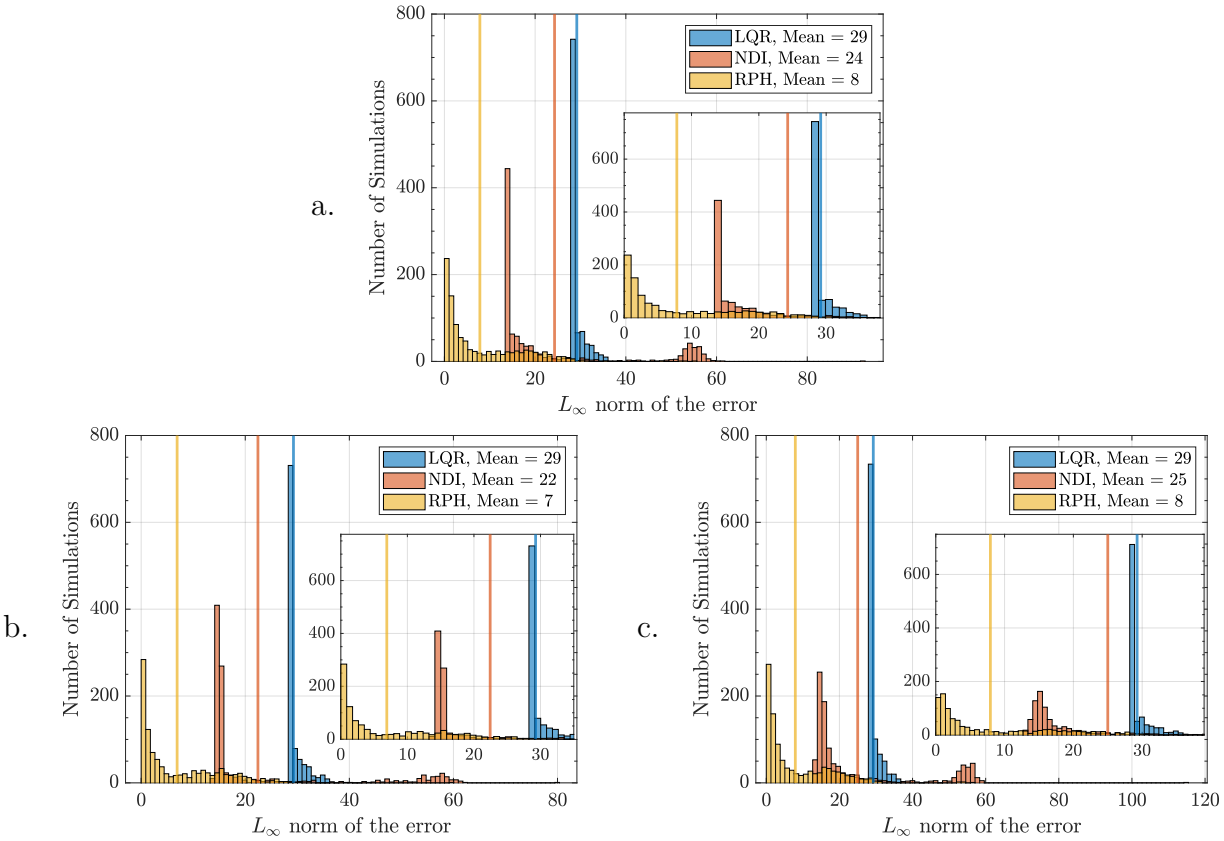

Figure 5.4: L_∞ norm results for (a) discretization with time delay only, (b) measurement noise only, and (c) measurement noise and discretization with time delay.

The histograms shown in Figure 5.4 are unique among the other cases. Only Group 1 cases have a hard lower limit for the L_∞ norm for LQR and NDI, below which there are no cases with lower error. Further, despite the high repeatability at the hard lower limit, each histogram has a thin, but long tail associated with it. LQR performs the most repeatably, as noted specifically by the close proximity of the mean to the highest peak of the control law and by the shortest tail of the three control laws, but also consistently has the highest error. This error is because LQR is tracking a reference input, as opposed to NDI and RPH which track a reference model. The reference input course angle $\chi^\star(t)$ is approximately 90° away from any initial state, as opposed to the reference model which is calculated to start on the initial state. RPH on the other hand has the lowest error, lowest peak, and the longest tail of all the control laws. Lastly, although the lower limit of RPH has a lower repeatability than NDI, NDI has much worse wider spread due to the secondary mode at a higher error.

The high peaks on the RPH histogram reveal that many cases are low error, which means many cases are not greatly impacted by the disturbances. However, any cases that are

not low error cause a long tail, which means that small error and large error impacts are nearly equally likely to occur. Given that measurement noise is a purely Gaussian sampled disturbance and time delay is a purely uniformly sampled distribution, this long tail on RPH reveals the sensitivity to white noise in the system causing at least one significant transient with varying error levels. However, the thickness of the tail demonstrates that RPH is not overly sensitive to measurement noise, and therefore is likely to maintain good performance.

Comparing the three cases reveals further insight. Figure 5.4(c) shows the results of Figure 5.4(b) with the addition of discretization with time delay. Although LQR remains unchanged as noted by the mean value, RPH and NDI are more significantly impacted by the combined disturbance as noted by the drop in the repeatability of both control laws. Despite that, RPH maintains a much lower error and is without outliers or a second mode compared to NDI, although in Figure 5.4(c), RPH is beginning to show signs of a second, higher error, mode that matches the location of the first, lower error, mode of NDI.

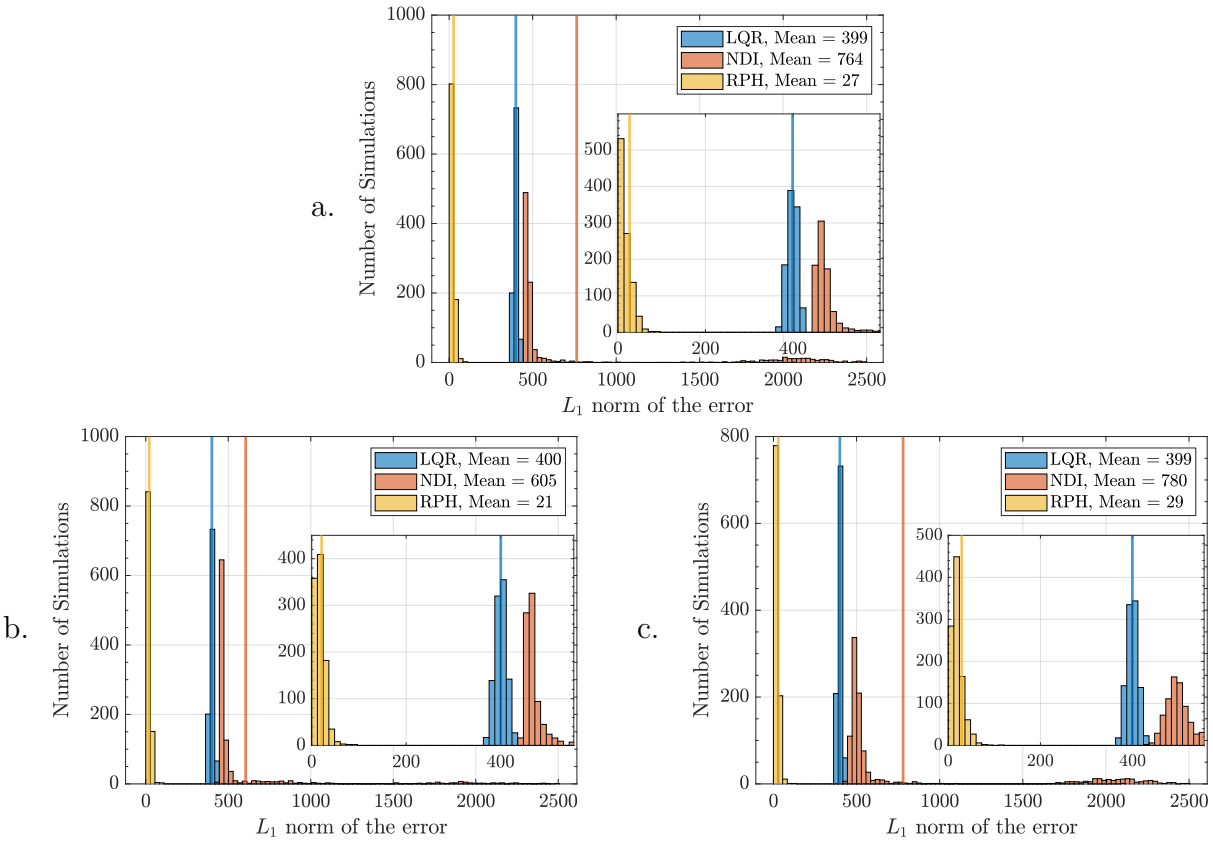

Figure 5.5: L_1 norm results for (a) discretization with time delay only, (b) measurement noise only, and (c) measurement noise and discretization with time delay.

Interestingly, the histograms in Figure 5.5 are significantly different from Figure 5.4. Notably, despite the long tail on the RPH L_∞ histogram, the RPH L_1 histogram shows the highest repeatability, the smallest spread, and the smallest error by a large margin compared to

LQR and NDI. This shows that RPH is robust to the disturbances in this group, which is likely due to the small magnitude of the disturbances.

As noted in Figure 5.4, RPH and NDI are both sensitive to measurement noise and discretization with time delay, especially together. Although the average error per simulation (indicated by the L_1 norm) is much lower for RPH than for LQR and NDI, the greatest error per simulation (from the L_∞ norm) reveals the sensitivity to disturbances. Therefore, if these disturbances had a greater magnitude, the error results would likely have increased as well. This provides an example of confirmation that RPH is able to perform well under non-additive disturbances, as long as the disturbances have a "small enough" magnitude. The sensitivity to these disturbances is easily seen when analyzing the histogram for NDI. Not only is the error much larger and less repeatable for NDI, but the outliers and second mode are nearly two orders of magnitude above the error values for RPH by examining the mean values. This is more easily seen by comparing the mean values to the respective histogram peaks - RPH and LQR match exactly, while NDI has a significant distance between the peak and the mean. Lastly, LQR is insignificantly impacted by these disturbances as noted by the mean values and histograms. The failure rates shown in Figure 5.6 show that

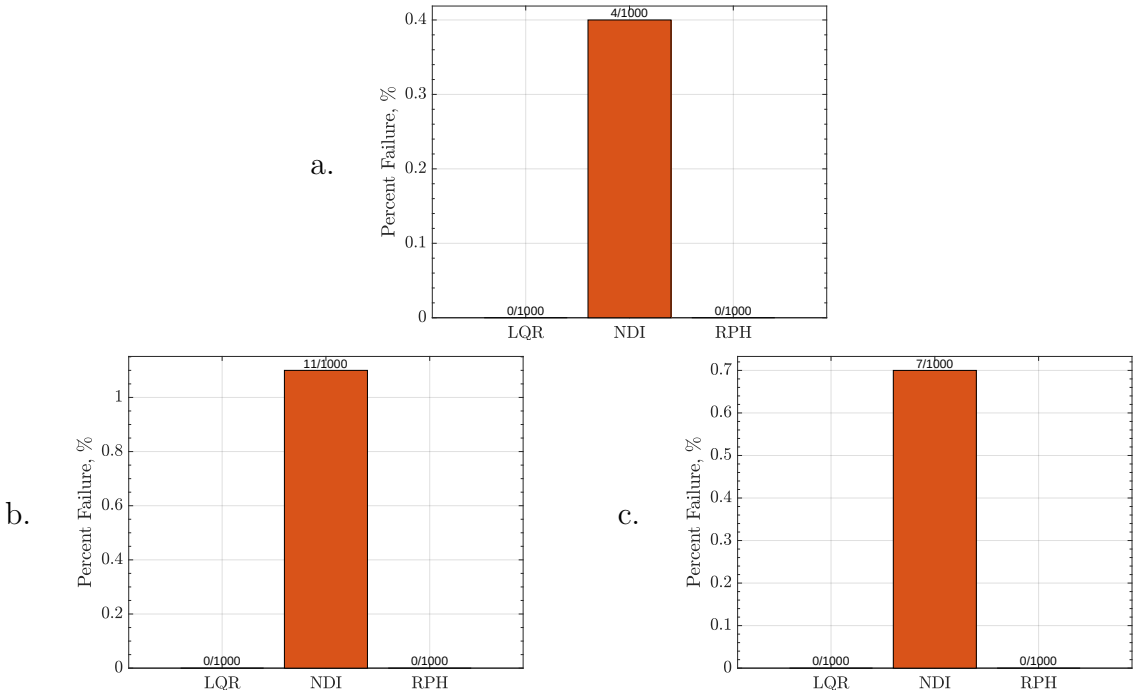

Figure 5.6: Failure results for (a) discretization with time delay only, (b) measurement noise only, and (c) measurement noise and discretization with time delay.

LQR and RPH did not fail from these disturbances, which further emphasizes robustness to these disturbances. This is especially true when comparing RPH to NDI, which had a failure rate of between 0.4% and 1.1%. This rate is not acceptable for flight, although it is acceptable for some applications.

5.2.2 Group 2: Intermediately Significant Disturbances

The second group consists of the four cases that produced greater spread in the L_∞ norm than Section 5.2.1 across all control laws. The cases are

- Model mismatch only,
- Model mismatch and discretization with time delay,
- Measurement noise and model mismatch, and
- Measurement noise, model mismatch, and discretization with time delay.

As listed, the disturbances in this group are all of the model mismatch disturbance combinations that do not include wind. This is because model mismatch is a larger magnitude disturbance than measurement noise, but smaller than wind. It is expected that the impact on RPH will be larger for model mismatch than measurement noise due to the fact that RPH was designed on top of a feedback linearization process.

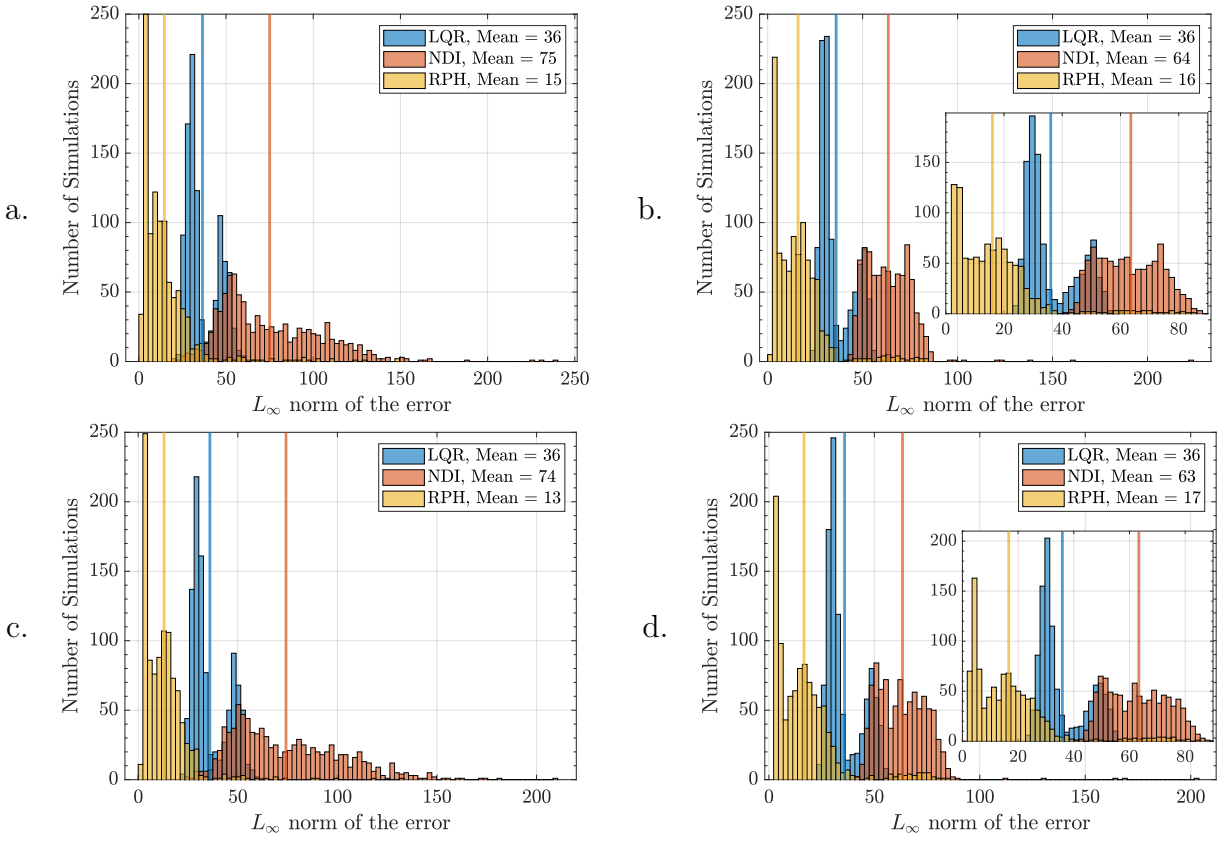

Figure 5.7: L_∞ norm results for (a) model mismatch only, (b) model mismatch and discretization with time delay, (c) measurement noise and model mismatch, and (d) measurement noise, model mismatch, and discretization with time delay.

The results shown in Figure 5.7 have a much higher spread and lower repeatability than every case in Figure 5.4. As noted above and seen by comparing Figure 5.7(a)-(d), the

difference is solely due to model mismatch. Comparing each of the four figures reveals more similarities than differences, again suggesting that measurement noise is not a significant disturbance compared to the other disturbances. However, the impact of discretization with time delay can be seen by comparing Figure 5.7(a) with (b), and (c) with (d). Figures 5.7(a) and (c) show higher repeatability and lower spread for RPH than in (b) and (d), which is the same change that occurred in Figure 5.4(b) to (c). However, NDI received the opposite impact. Figures 5.7(a) and (c) show large spread for NDI, but (b) and (d) display that the spread decreases. The observation that discretization reduces error implies that the poor performance of the NDI controller in the tail is, at least in part, due to the controller overreacting. That is, when discretized, the controller is no longer able to overreact so the performance improves. The same effect is not observed for LQR or RPH, since neither control law is overreacting. Lastly, LQR continues to not be impacted by measurement noise or discretization with time delay, although the addition of model mismatch was significant enough to cause the LQR histogram to become bimodal.

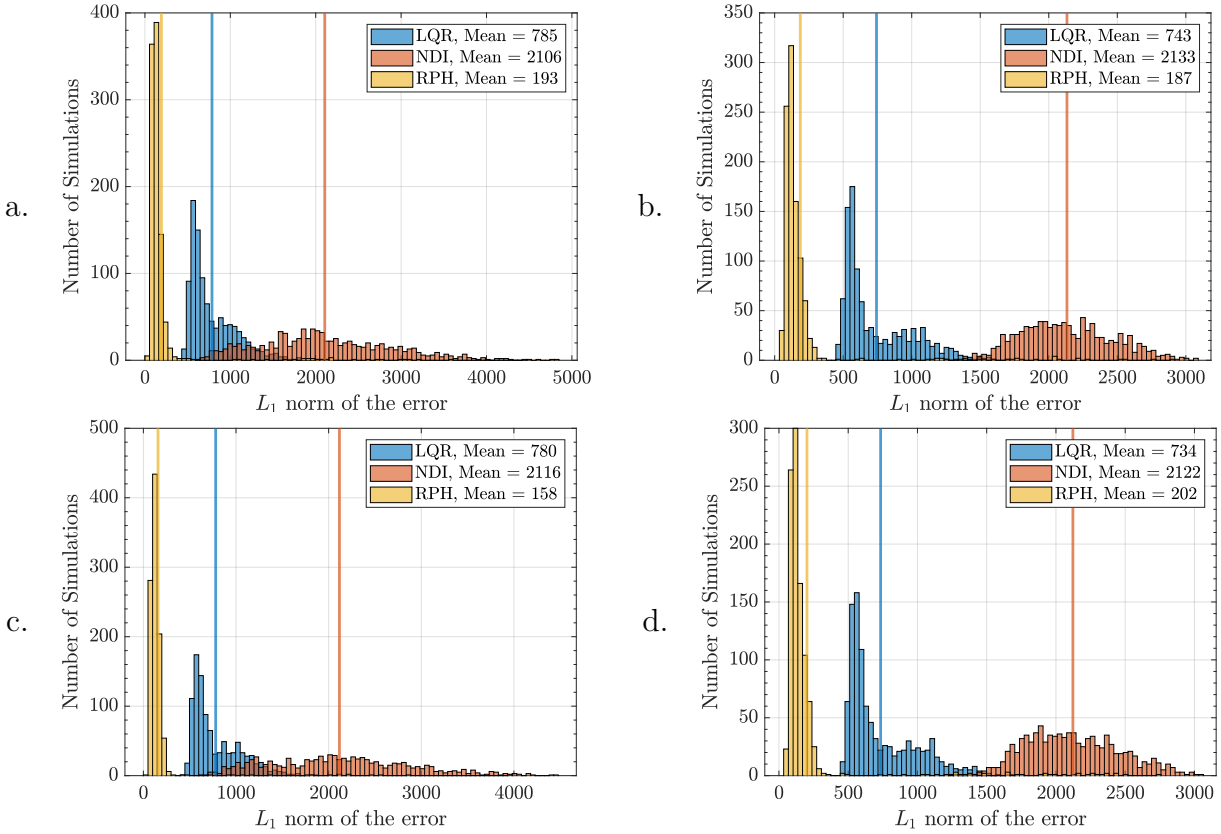

Figure 5.8: L_1 norm results for (a) model mismatch only, (b) model mismatch and discretization with time delay, (c) measurement noise and model mismatch, and (d) measurement noise, model mismatch, and discretization with time delay.

Despite the combinations of disturbances, RPH continues to have the lowest L_∞ error. This means that the impact of model mismatch on the magnitude of the error transients is larger

for LQR and NDI than RPH. These results demonstrate that model mismatch is a more significant disturbance than measurement noise. Comparing Figures 5.4 and 5.7 reveals the impact of model mismatch, which is further compounded by the construction of both NDI and RPH which use an already reduced aerodynamic model.

Figure 5.8 provides a similar result to Figure 5.7, which verifies the discussion above. RPH continues to have the highest repeatability, the smallest spread, and the smallest error compared to LQR and NDI, although by a smaller margin in this group of disturbance cases. Although the figure scales are different, comparing the mean reveals that model mismatch, on average, doubles the mean of LQR, triples the mean of NDI, and increases RPH by an order of magnitude. However, by comparing pure values and not factors, RPH increases the least by an L_1 norm error of over 200.

Focusing on Figure 5.8(d) reveals the most difficult disturbance case without the wind disturbance. Although RPH is noticeably perturbed by the three applied disturbances, the performance is much better than LQR and NDI.

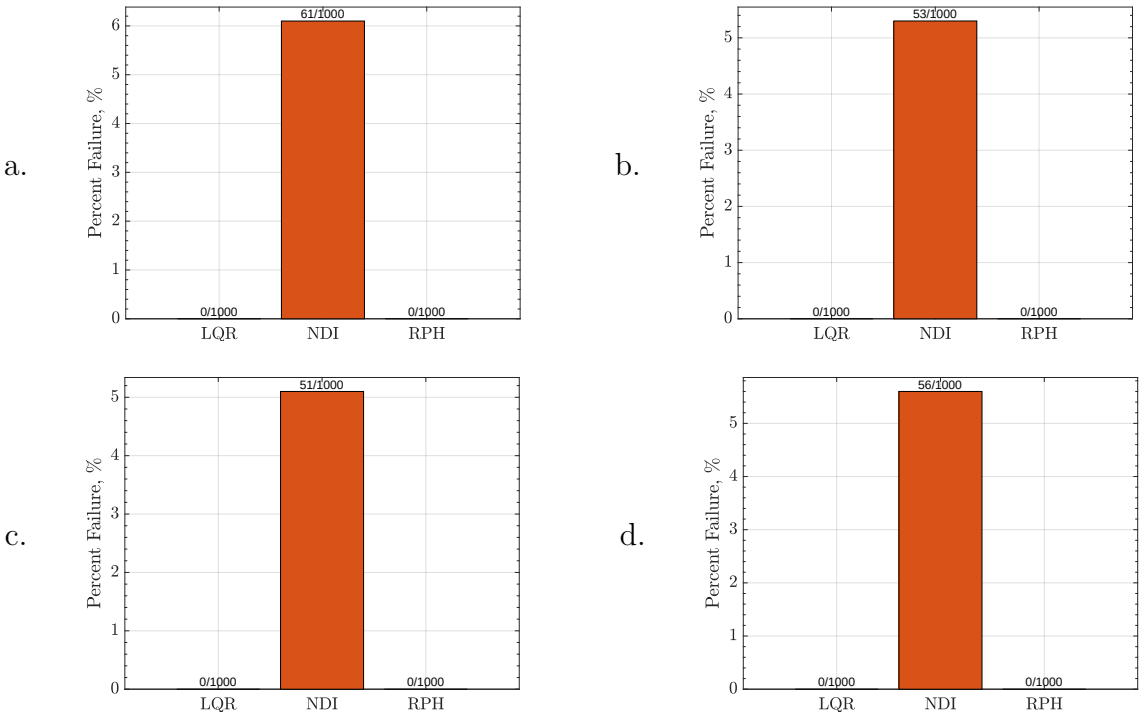

Figure 5.9: Failure results for (a) model mismatch only, (b) model mismatch and discretization with time delay, (c) measurement noise and model mismatch, and (d) measurement noise, model mismatch, and discretization with time delay.

Figure 5.9 completes the discussion by, once again, demonstrating that RPH is more robust than NDI. RPH continues to lack failures, while the addition of model mismatch increases the failure rate for NDI from around 1% to above 5%.

5.2.3 Group 3: Most Significant Disturbances

The third and final group consists of the remaining eight cases that produced the greatest impacts on RPH. Figure 5.10 illustrates the histograms of L_∞ norms for these cases. The cases are

- Wind only,
- Wind and discretization with time delay,
- Model mismatch and wind,
- Model mismatch, wind, and discretization with time delay,
- Measurement noise and wind, and
- Measurement noise, wind, and discretization with time delay,
- Measurement noise, model mismatch, and wind, and
- Measurement noise, model mismatch, wind, and discretization with time delay (all disturbances).

A brief survey of the disturbances applied reveals that every disturbance present incorporates the wind disturbance. The impact posed by wind likely has the largest effect of the disturbances.

Each of the trends identified in previous cases are demonstrated more clearly in Figure 5.10. First, the impact of wind greatly impacts RPH specifically, but LQR and NDI are not significantly changed. This can be seen visually since RPH nearly exactly overlaps the main histogram of NDI with a secondary mode, while usually maintaining a low-error mode. This bimodality is likely caused by the same mechanism that splits LQR in two, in that some simulations were impacted by disturbances that were small in magnitude, while other simulations were impacted by disturbances were large in magnitude and greatly increased the error. This is exacerbated by the inclusion of discretization with time delay in the figures on the right of 5.10 [(b), (d), (f), (h)].

Another trend previously noted was the impact of discretization with time delay decreasing both the spread and error of NDI compared to the corresponding case without the discretization with time delay disturbance. This is likely due to the discretization with time delay limiting the update of the control inputs, which reduces overly aggressive control input changes.

Although Figure 5.11 appears to demonstrate different results than Figure 5.10, the results are fully complementary. The addition of the second, high error, mode in the L_∞ norm error compared to previous simulations results changed the shape of the RPH histogram to have a very wide spread.

Further, it is more clearly seen by comparing figures (d) and (h) that the impact of measurement noise for all of the control laws is minimal. The histogram shapes do not change significantly other than the smoothness of the histogram in (h), which is due to the order of magnitude increase in samples. However, the addition of any other disturbance to any

disturbance combination causes a significant change in the performance of the control law.

Lastly, the impact of discretization with time delay is also clearly visible by comparing the left column of figures to the right column of figures. Despite the wind, the repeatability of RPH is much larger than NDI and the spread present is not very large. However, the combination of wind and discretization with time delay cause the greatest challenge to RPH by severely decreasing the repeatability and increasing the spread.

The first RPH failures are seen in Figure 5.12. Clearly, the changes to failure rate is due to the introduction of wind. Although the failure rate for NDI increased to approximately 8-10% with limited change due to the implementation of discretization with time delay, the failure rate for RPH increased from 0% to 1-1.5% with only wind and 3.4-4.7% with wind and discretization with time delay.

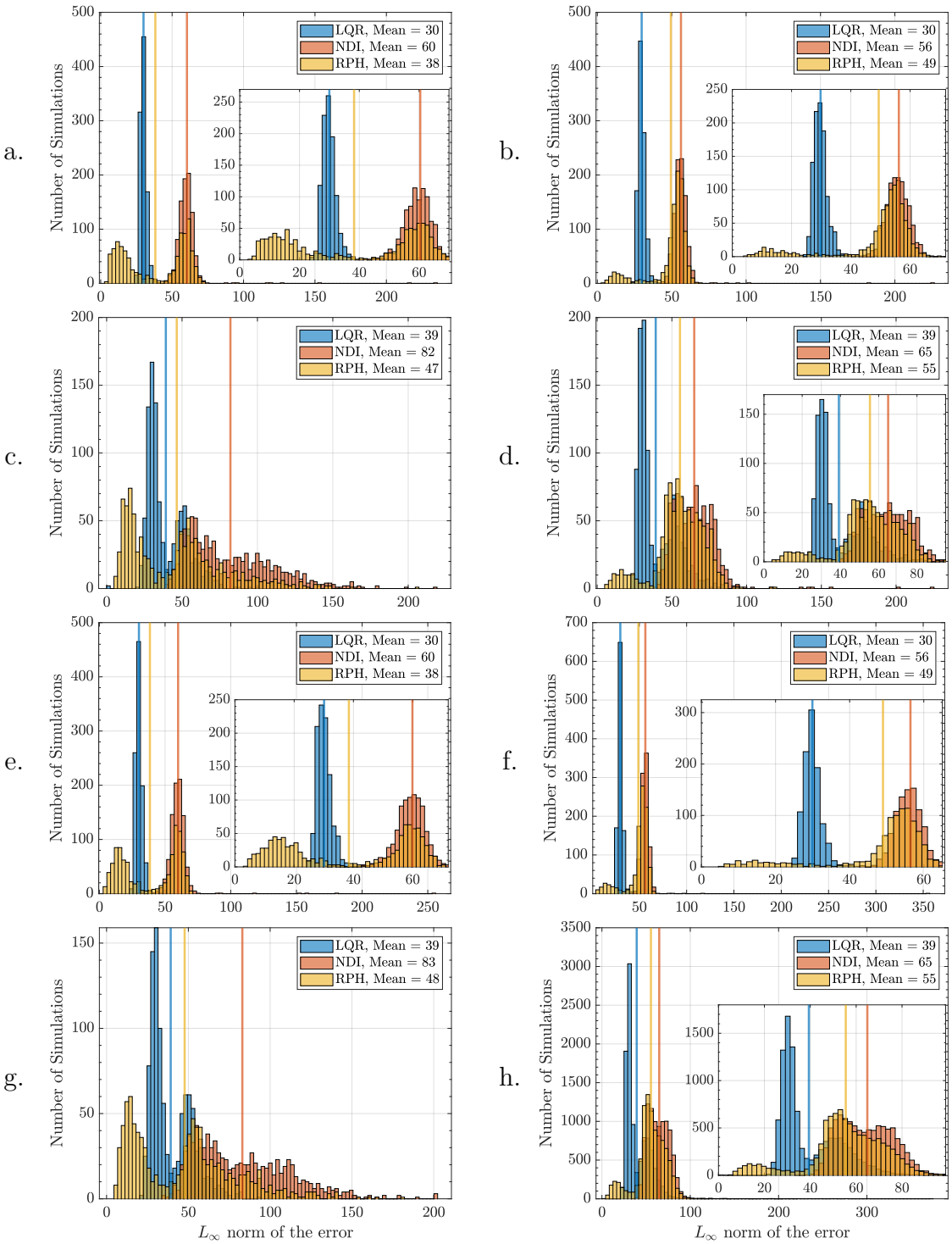

Figure 5.10: L_∞ norm results for (a) wind only, (b) wind and discretization with time delay (c) model mismatch and wind, (d) model mismatch, wind, and discretization with time delay, (e) measurement noise and wind, (f) measurement noise, wind, and discretization with time delay, (g) measurement noise, model mismatch and wind, and (h) all combined disturbances.

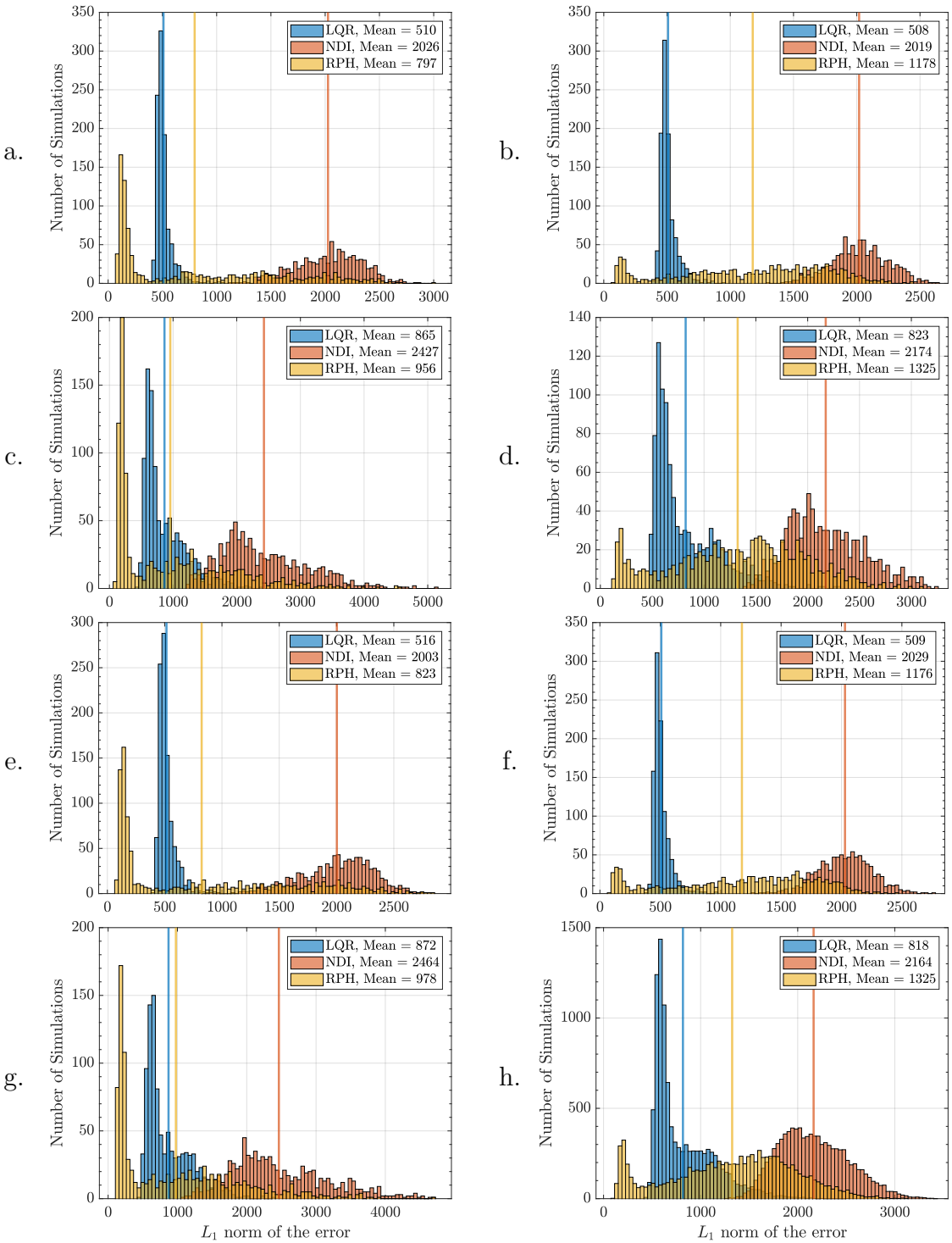

Figure 5.11: L_1 norm results for (a) wind only, (b) wind and discretization with time delay (c) model mismatch and wind, (d) model mismatch, wind, and discretization with time delay, (e) measurement noise and wind, (f) measurement noise, wind, and discretization with time delay, (g) measurement noise, model mismatch and wind, and (h) all combined disturbances.

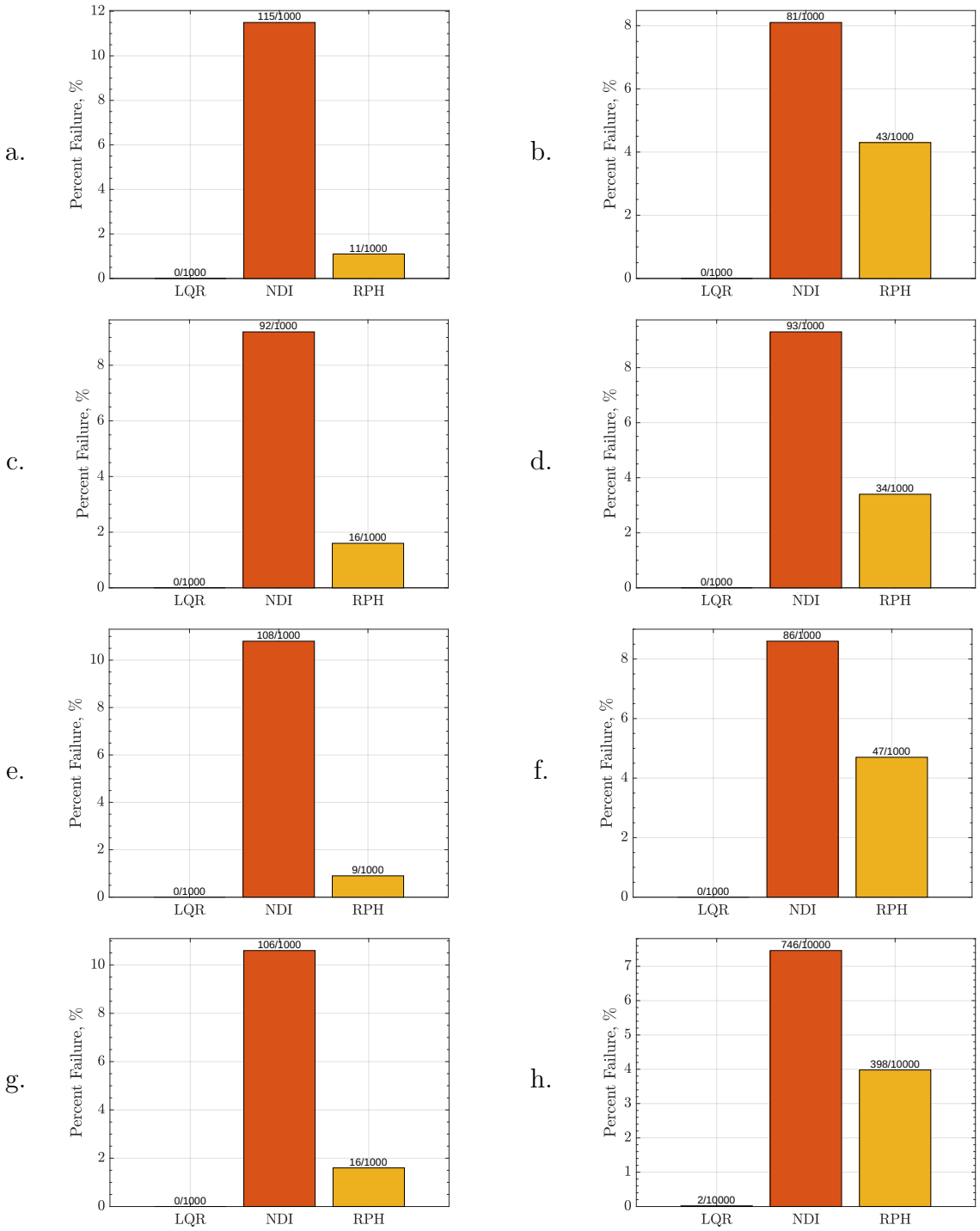

Figure 5.12: Failure results for (a) wind only, (b) wind and discretization with time delay (c) model mismatch and wind, (d) model mismatch, wind, and discretization with time delay, (e) measurement noise and wind, (f) measurement noise, wind, and discretization with time delay, (g) measurement noise, model mismatch and wind, and (h) all combined disturbances.

5.3 Summary

From the Group 1 (Figures 5.4-5.6) shows that RPH is slightly sensitive to measurement noise, as well as individually disturbed control discretization with time delay. However, RPH had lower error and slightly lower failure rate than NDI given the same feedback linearization architecture. Adding model mismatch to each case in Group 1 (Figures 5.4-5.6) led to a wider spread in Group 2 (Figures 5.7-5.9). The performance of RPH was degraded in both error and repeatability, but the performance was still superior than of LQR and NDI.

Replacing the control discretization with time delay disturbance in Group 2 (Figures 5.7-5.9) with the wind disturbance in Group 3 (Figures 5.10-5.12) greatly increased the spread of the histograms. RPH continued to outperform NDI across the cases, although the performance benefits of LQR became more evident. RPH became bimodal in some cases, although the second, high error, mode never had greater error than the NDI histogram. Further, the wind disturbance cases were the only disturbance cases which caused a singularity in RPH, and the wind and discretization with time delay combination increased the failure rate from 1-1.5% to 3.4-4.7%.

There are likely a few reasons that RPH is especially sensitive to the combination of wind and control discretization with time delay. First, wind is a non-additive disturbance. As discussed in Section 3.3.2, bounded non-additive disturbances can quickly become equivalent to unbounded, additive disturbances. The magnitude of applied wind, although bounded, likely caused RPH to apply too large of control inputs to counteract the wind at each time step, although the wind was varying greatly during the entire simulation. This is because the control law does not have wind as an input, so the only knowledge of wind comes from the aerodynamic model performing in ways the control law did not know.

Second, the control discretization with time delay made it much more difficult, or nigh impossible, for RPH to apply appropriate control inputs at the appropriate time. This is especially difficult when the output tracking is nonlinear and changing, and the aerodynamics are impacted by both wind and model mismatch. Even a maximum delay of 80 milliseconds, between the 25 Hz sampling rate and 40 milliseconds time delay, could make normally bounded disturbances become unbounded to the control law, causing the control law to become unbounded as well. This is true for each of the applied disturbance combinations; the addition of control discretization with time delay worsens every disturbance case to either increase spread (decrease robustness), increase error (decrease robust performance), or increase the failure rate.

Comparatively, RPH was moderately sensitive to model mismatch (Figures 5.7-5.9). The addition of model mismatch individually and combined with other disturbances showed that RPH could handle the impact of model mismatch, measurement noise, and control discretization with time delay together, with a corresponding decrease in robustness and repeatability but not error (Figures 5.4-5.6). RPH consistently had the lowest error across all cases except

the Group 3 cases, due to wind. The wind had a much larger impact on the aerodynamic model than model mismatch, likely due to the comparative magnitude of both effects on the aerodynamic performance. If the effect of model mismatch were increased by raising the mismatch limit from $\pm 2\sigma$ to $\pm 3\sigma$, the impact of model mismatch would likely have a greater impact than currently, but still not as high as the effect of wind.

The only disturbance that RPH was not overly sensitive to was measurement noise, individually and in combinations. This can be seen in Figure 5.4, where NDI and RPH had similar robustness to measurement noise, except RPH had lower error and failure rates. This can also be seen by comparing Figure 5.10(a) with Figure 5.10(e), Figure 5.10(b) with Figure 5.10(f), Figure 5.10(c) with Figure 5.10(g), and Figure 5.10(b) with Figure 5.10(h). The only addition between those sets of figures is measurement noise and there is hardly any change to either the mean or histogram shape. In Figure 5.4, the lower error mode of NDI had a similar histogram shape and tail to RPH, although RPH had much better performance. Therefore, the foundation of feedback linearization likely caused the sensitivity to measurement noise as opposed to the construction of RPH in particular.

A final factor to consider is the implementation of control logic to limit the exponential growth of the control input being calculated by RPH. To compare this, each discrete time case presented above was simulated with and without control limiting on the $\boldsymbol{u}$ and $\boldsymbol{v}$ control inputs. The result was that adding control limiting brought the rate of failure for RPH down from over 45% to the presented value at under 4% in the combined disturbance case. This reduction in failure rate was for every case presented in simulation, since even the most benign disturbances caused a singular matrix due to unlimited input. This is an important piece of implementation to consider when the control law has a tendency to attempt inversion of singular matrices if the control inputs are allowed to grow to match the applied disturbances.

Therefore, when RPH does not mathematically become singular or is perturbed by disturbances which can no longer be considered bounded, RPH has the best robust performance over LQR and NDI. Until another control law is developed, either by applying the construction of RPH to a non-feedback linearized system or by accounting for disturbances such as wind in the construction of the control law, the best control law for flight depends highly on the allowable risk. LQR has the lowest risk of the three control laws, but if the physical disturbances to the system are small in magnitude for RPH, LQR would have worse flight performance than RPH. But if the environment is very windy or there are other large disturbances present, LQR is the control law to use.

LQR performed surprisingly well compared to the nonlinear control laws. Therefore, the simulations demonstrated above were analyzed to ensure that the perturbations applied to each control law were within the nonlinear regime. If the applied perturbations remained within the linear regime, then the results presented earlier in this chapter would not properly compare the performance of the control laws as the analysis presented. To verify that the perturbations were within the nonlinear regime, a set of individual LQR simulations was run, since the response of the linear control law would show whether the perturbations are

linear or not. For the simulations run for this analysis, the initial state of the aircraft was set to be the exact initial reference state as listed in (A.1.1), with the exception of an initial lateral velocity, v_0, that was varied between simulations. The lateral velocity state, v, was chosen since most aircraft are not actuated in the lateral direction.

Figure 5.13 shows an example time history of $(\boldsymbol{x} - \boldsymbol{x}_{\text{ref}})$ in (3.1.4) for a single simulation where $v_0 = 5$ m/s. The transients seen beginning at 20, 40, and 55 seconds are nonlinear perturbations due to switching between reference trim conditions. Therefore, the remainder of the analysis will focus on the transient due to the initial condition, which is highlighted by the inset figure. Note that the transition between reference trim conditions will not exhibit the linear scaling associated with scaling the initial condition, since the transients due to the initial condition end after approximately 5 seconds.

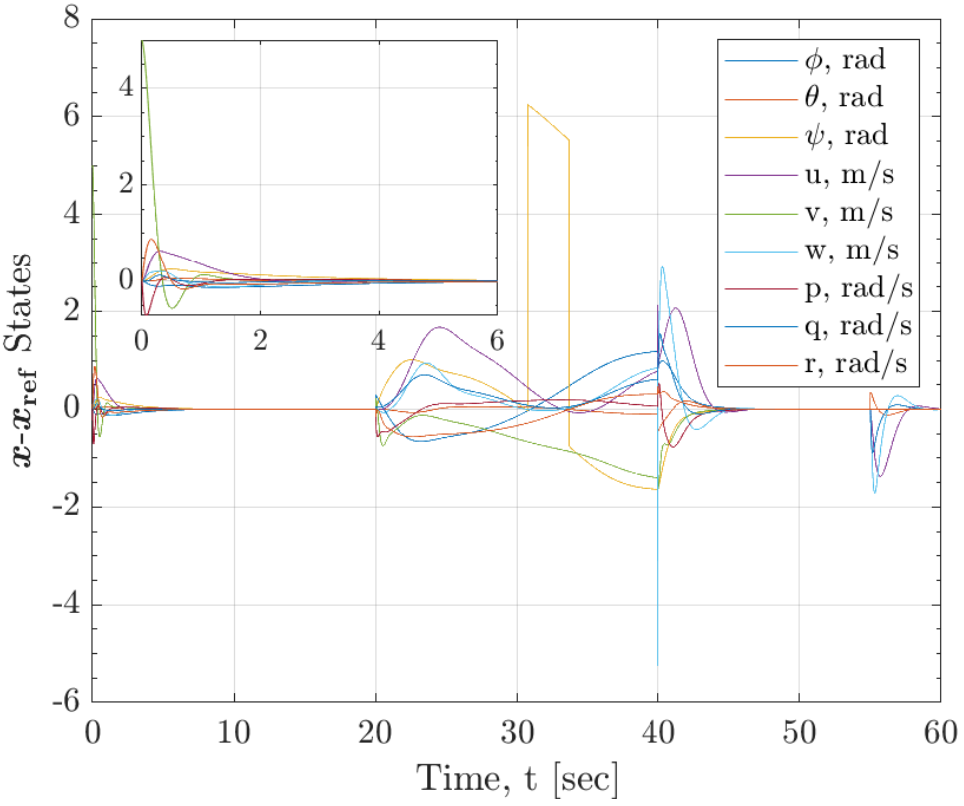

Figure 5.13: Time history of $(\boldsymbol{x} - \boldsymbol{x}_{\text{ref}})$ for $v_0 = 5$ m/s

Figure 5.14 shows the longitudinal velocity response of the aircraft $(u - u_{\text{ref}})$ due to initial lateral velocities $v_0 = 1$, 3, 4, and 5 m/s. For any system with outputs defined by $y = Cx$, any choice of C will demonstrate the linear scaling. Therefore, the quantity $(u - u_{\text{ref}})$ was chosen since it is representative of $(\boldsymbol{x} - \boldsymbol{x}_{\text{ref}})$ and best displayed the nonlinearities in Figure 5.14. The chosen values of v_0 are representative of the random initial state from Table A.1. The top graph of Figure 5.14 shows the direct $(u - u_{\text{ref}})$ response due to each initial lateral velocity, while the bottom graph shows the response normalized by the respective values for v_0.

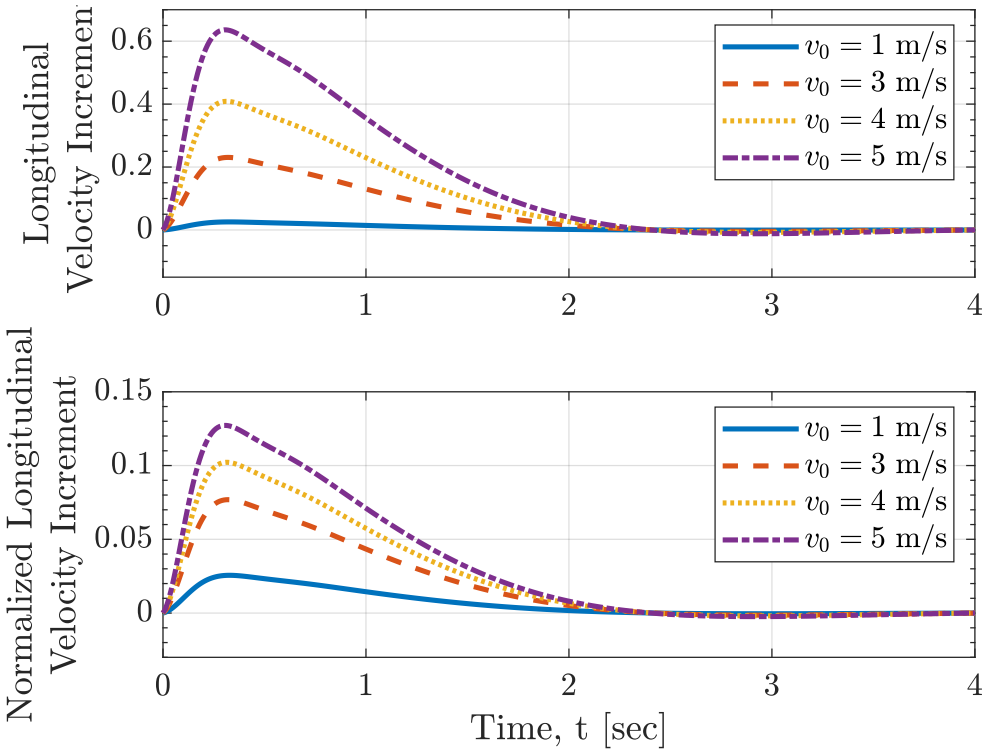

Figure 5.14: Initial transient in longitudinal velocity increment, $u - u_{\text{ref}}$, for various initial lateral velocities, v_0.

If the perturbations applied to the system were linear, then each response on the normalized graph would be identical, i.e. each plotted lines would overlap. However, since the normalized responses are not identical, the perturbations applied are not linear. Given that only the initial vehicle state was changed, the nonlinear perturbations come from the aerodynamic model of the aircraft. This will always impact every control law throughout the entire flight, and therefore the analysis presented here can be applied to the entire analysis presented earlier in the chapter. Therefore, the primary results of this thesis present a valid comparison of the control laws, and the comparatively strong performance of LQR is likely due to deficiencies of NDI and RPH.

During the testing of this work, the L_2 norm was calculated for each simulation in addition to the L_1 and L_∞ norms. However, the amount of additional information provided by analyzing the L_2 norm compared to only the L_1 and L_∞ norms was not enough to justify adding the L_2 norm results to the above section. Therefore, if desired, the figures showing the L_2 norm for each case is provided in Appendix B.

Chapter 6

Conclusions

This work presented the implementation of a novel, nonlinear flight control law, robust port-Hamiltonian control (RPH), with inherent robustness properties as described in [50]. RPH design involved transforming a system from a cascaded integral structure into a port-Hamiltonian structure, using feedback linearization, in order to obtain sufficient conditions for input-to-state stability (ISS) with respect to disturbance inputs. Those robustness guarantees state that the state will remain uniformly bounded for any uniformly bounded input into the system that is additive to the forces and moments, which includes the control moment and disturbance. This implementation was proved successful by running individual simulations with a pseudo-randomized additive disturbance, and showing the state remained uniformly bounded with excellent output tracking performance while being perturbed by both the additive disturbance and the "unknown" aerodynamic model mismatch that came from the construction of the feedback linearization control law.

Two comparable alternative control laws, linear quadratic regulator (LQR) and nonlinear dynamic inversion (NDI), were implemented as well to compare the robustness and control law construction methods. LQR is a linear control law that has inherent robustness guarantees, while NDI is a nonlinear control law without inherent robustness guarantees. However, NDI is also constructed using feedback linearization, to which a LQR is applied as a closed-loop control input to control the identically transformed linear system. The implementation was proved successful for LQR when the output tracking was less exact than RPH, but the generalized, additive applied disturbance did not significantly impact the output tracking. Further, the implementation was proved successful for NDI when the output tracking for an ideal, nonlinear output track was as exact as RPH, but when the disturbances were added, NDI was not able to sufficiently perform output tracking anymore.

To test the robustness guarantees and robustness properties of RPH, four stochastic disturbances were implemented in a simulation study to mimic a flight environment. Measurement noise is a disturbance that comes from the aircraft sensors and only impacts the control law calculation. Model mismatch is a disturbance that comes from system identification and finding the best model possible, but that model still is not the same as real life. Wind is a disturbance that is caused by pressure differences in the atmosphere, and therefore any flying object will encounter wind after enough time in the air. Lastly, the implementation of a continuous system dynamics and continuous control law on hardware will automatically require that the continuous dynamics become discretized. Further, a time delay due to

physical communications of the flight computer can cause the system dynamics to propagate before the calculated control input is applied, further delaying the appropriate control input from the discretized calculation. The concern regarding ISS for these disturbances is that each disturbance is non-additive to the forces or moments. Therefore, even a non-additive disturbance that is uniformly bounded could cause an unbounded response from the system.

All four disturbances were implemented in a 10000 simulation study, where the L_1 and L_∞ signal norms were calculated for each simulation, as well as the failure rate of each control law per simulation. The distribution of inertial velocity tracking error using RPH became bimodal in both the L_∞ and L_1 norms due to the disturbances violating the condition for ISS the majority of the time, although a few simulations of disturbance remained within ISS and had low error. The L_∞ performance of RPH nearly exactly matched the performance of NDI, except RPH was never worse than NDI in any individual simulation. RPH had very wide spread in the L_1 norm due to the disturbances, which proved much better than NDI which had a lower spread at a higher error. NDI had a failure rate of approximately 7% while RPH had a failure rate of nearly half at 4%. However, LQR consistently maintained moderate error and high repeatability (tall peak, small spread) without any failures.

In order to ascertain which applied disturbances were the primary source of error for RPH, all 15 combinations of the four disturbances were analyzed by the same metrics using a 1000 simulation study for each case. The analysis showed that RPH is most sensitive to wind, discretization with time delay, model mismatch, and measurement noise in that order. This aligns with the analysis of the disturbances, since the disturbance magnitude increases from measurement noise to model mismatch to wind, and discretization with time delay changes the time the control input is applied and therefore compounds the effects of any other disturbances.

Extensions of this work include flight testing of the three control laws presented here, deriving another version of RPH which is more robust, and extending this work to compare dynamic state feedback control laws. The flight testing of RPH presented in this work would first require a thorough risk analysis based on the simulation study presented here to ensure the flight test can be performed safely. Deriving a version of RPH which explicitly includes wind would theoretically greatly increase the performance and robustness of RPH. However, the method to do so is fraught with issues. The primary issue is that, unlike disturbances added to the force and moment terms, there is no way to make a system affine with respect to the wind. The construction of RPH is expecting the disturbances to apply a force or moment to the aircraft, which is not applicable for wind since wind can instantaneously change the angle of attack. A more reasonable version of RPH would be to apply a RPH to a system that is not feedback linearized, removing the chance of singular matrices. Conversely, another option would be to apply a disturbance observer to RPH that would estimate the wind disturbance and apply it as additive forces and moments which RPH is designed to handle. Lastly, the two other control laws can be extended to dynamic state feedback using a dynamic extension for each, such as linear quadratic gaussian extending from LQR or a robust or adaptive extension to NDI.

Bibliography

[1] FLYING QUALITIES OF PILOTED AIRCRAFT. Department Of Defense Handbook MIL-HDBK-1797, 1997. (cit. on pp. 16, 59, 60, 61, and 96.)

[2] Brian D. O. Anderson and John B. Moore. *Linear Optimal Control*. Prentice-Hall Network Series. Prentice-Hall, Englewood Cliffs, N.J, 1971. ISBN 978-0-13-536870-1. (cit. on pp. 1 and 21.)

[3] Anuradha M. Annaswamy. Adaptive Control and Intersections with Reinforcement Learning. *Annual Review of Control, Robotics, and Autonomous Systems*, 6(1):65–93, May 2023. ISSN 2573-5144, 2573-5144. doi: 10.1146/annurev-control-062922-090153. (cit. on p. 1.)

[4] Karl Johan Astrom and Richard M Murray. *Feedback Systems - An Introduction for Scientists and Engineers*. 2020. (cit. on p. 20.)

[5] John Baillieul and Tariq Samad, editors. *Encyclopedia of Systems and Control*. Springer International Publishing, Cham, 2021. ISBN 978-3-030-44183-8 978-3-030-44184-5. doi: 10.1007/978-3-030-44184-5. (cit. on pp. 1 and 41.)

[6] Dan Canin, Jeffrey McConnell, and Paul James. F-35 High Angle of Attack Flight Control Development and Flight Test Results. In *AIAA Aviation 2019 Forum*, Dallas, Texas, June 2019. American Institute of Aeronautics and Astronautics. ISBN 978-1-62410-589-0. doi: 10.2514/6.2019-3227. (cit. on p. 29.)

[7] Peter Dorato, Chaouki T. Abdallah, and Vito Cerone. *Linear-Quadratic Control: An Introduction*. Krieger, Malabar, Florida, repr. ed. with corr edition, 2000. ISBN 978-1-57524-156-2. (cit. on pp. 21 and 22.)

[8] Bernard Etkin and T. Teichmann. Dynamics of Flight: Stability and Control, volume 12. 1996. (cit. on p. 31.)

[9] Jean-Michel Fahmi and Craig A. Woolsey. Port-Hamiltonian Flight Control of a Fixed-Wing Aircraft. *IEEE Transactions on Control Systems Technology*, 30(1):408–415, January 2022. ISSN 1063-6536, 1558-0865, 2374-0159. doi: 10.1109/TCST.2021.3059928. (cit. on p. 41.)

[10] Kenneth C. Gahan, Jeremy W. Hopwood, and Craig A. Woolsey. Model-Based Wind Estimation Using H∞ Filtering with Flight-Test Results. *Journal of Guidance, Control, and Dynamics*, 48(3):630–637, March 2025. ISSN 0731-5090, 1533-3884. doi: 10.2514/1.G007735. (cit. on pp. 59, 61, and 96.)

[11] S.Torkel Glad. Robustness of Nonlinear State Feedback—A Survey. *Automatica*, 23(4): 425–435, July 1987. ISSN 00051098. doi: 10.1016/0005-1098(87)90072-0. (cit. on pp. 13 and 14.)

[12] James L Gresham. *Aerodynamic Modeling in Nonlinear Regions, Including Stall Spins, for Fixed-Wing Unmanned Aircraft from Experimental Flight Data*. PhD thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA, March 2022. (cit. on p. 12.)

[13] James L. Gresham, Benjamin M. Simmons, Jeremy W. Hopwood, and Craig A. Woolsey. Spin Aerodynamic Modeling for a Fixed-Wing Aircraft Using Flight Data. *Journal of Aircraft*, 61(1):128–139, January 2024. ISSN 0021-8669, 1533-3868. doi: 10.2514/1.C036835. (cit. on p. 12.)

[14] Jeffrey J. Harris and James Stanford. F-35 Flight Control Law Design, Development and Verification. In *2018 Aviation Technology, Integration, and Operations Conference*, Atlanta, Georgia, June 2018. American Institute of Aeronautics and Astronautics. ISBN 978-1-62410-556-2. doi: 10.2514/6.2018-3516. (cit. on p. 29.)

[15] The MathWorks Inc. Von Karman Wind Turbulence Model (Continuous), 2025. URL https://www.mathworks.com/help/aeroblks/vonkarmanwindturbulencemodelcontinuous.html. (cit. on p. 60.)

[16] R. E. Kalman and J. E. Bertram. Control System Analysis and Design Via the "Second Method" of Lyapunov: I—Continuous-Time Systems. *Journal of Basic Engineering*, 82 (2):371–393, June 1960. ISSN 0021-9223. doi: 10.1115/1.3662604. (cit. on p. 15.)

[17] R. E. Kalman and J. E. Bertram. Control System Analysis and Design Via the "Second Method" of Lyapunov: II—Discrete-Time Systems. *Journal of Basic Engineering*, 82 (2):394–400, June 1960. ISSN 0021-9223. doi: 10.1115/1.3662605. (cit. on p. 15.)

[18] Hassan K. Khalil. *Nonlinear Systems. Hauptbd.* Prentice Hall, Upper Saddle River, NJ, 3. ed edition, 2002. ISBN 978-0-13-067389-3. (cit. on pp. 1, 6, 18, 19, 20, and 29.)

[19] Vladislav Klein and Eugene A. Morelli. *Aircraft System Identification: Theory and Practice*. AIAA Education Series. American Institute of Aeronautics and Astronautics, Reston, VA, 2006. ISBN 978-1-56347-832-1. (cit. on p. 12.)

[20] David H. Klyde, Philip C. Schulze, David Mitchell, and Natalia Alexandrov. Development of a Process to Define Unmanned Aircraft Systems Handling Qualities. In *2018 AIAA Atmospheric Flight Mechanics Conference*, Kissimmee, Florida, January 2018. American Institute of Aeronautics and Astronautics. ISBN 978-1-62410-525-8. doi: 10.2514/6.2018-0299. (cit. on p. 16.)

[21] Johan Knöös, John Robinson, and Fredrik Berefelt. Nonlinear Dynamic Inversion and Block Backstepping: A Comparison. In *AIAA Guidance, Navigation, and Control*

Conference, Minneapolis, Minnesota, August 2012. American Institute of Aeronautics and Astronautics. ISBN 978-1-60086-938-9. doi: 10.2514/6.2012-4888. (cit. on p. 28.)

[22] G. Kreisselmeier and B. Anderson. Robust Model Reference Adaptive Control. *IEEE Transactions on Automatic Control*, 31(2):127–133, February 1986. ISSN 0018-9286. doi: 10.1109/TAC.1986.1104217. (cit. on p. 14.)

[23] Miroslav Krstic and Andrey Smyshlyaev. *Boundary Control of PDEs: A Course on Backstepping Designs*. Number 16 in Advances in Design and Control. Society for Industrial and Applied Mathematics, Philadelphia, Pa, 2008. ISBN 978-0-89871-650-4 978-0-89871-860-7. doi: 10.1137/1.9780898718607. (cit. on p. 1.)

[24] Miroslav Krstic, Ioannis Kanellakopoulos, and Petar V. Kokotović. *Nonlinear and Adaptive Control Design*. Adaptive and Learning Systems for Signal Processing, Communications, and Control. Wiley, New York, 1995. ISBN 978-0-471-12732-1. (cit. on p. 1.)

[25] Huibert Kwakernaak and Raphael Sivan. *Linear Optimal Control Systems*. Wiley-Interscience, New York, 1972. ISBN 978-0-471-51110-6. (cit. on pp. 1 and 21.)

[26] Stephen H. Lane and Robert F. Stengel. Nonlinear Inverse Dynamics Control Laws - A Sampled Data Approach. In *American Control Conference*, IEEE, January 1987. doi: 10.23919/ACC.1987.4789501. (cit. on pp. 15 and 21.)

[27] Stephen H. Lane and Robert F. Stengel. Flight Control Design Using Non-linear Inverse Dynamics. *Automatica*, 24(4):471–483, July 1988. ISSN 00051098. doi: 10.1016/0005-1098(88)90092-1. (cit. on pp. 31 and 33.)

[28] Eugene Lavretsky and Kevin A. Wise. *Robust and Adaptive Control: With Aerospace Applications*. Advanced Textbooks in Control and Signal Processing. Springer International Publishing, Cham, 2024. ISBN 978-3-031-38313-7 978-3-031-38314-4. doi: 10.1007/978-3-031-38314-4. (cit. on p. 1.)

[29] W. MacKunis, P. M. Patre, M. K. Kaiser, and W. E. Dixon. Asymptotic Tracking for Aircraft via Robust and Adaptive Dynamic Inversion Methods. *IEEE Transactions on Control Systems Technology*, 18(6):1448–1456, November 2010. ISSN 1063-6536, 1558-0865. doi: 10.1109/TCST.2009.2039572. (cit. on p. 28.)

[30] Michael McFarland and Shaheen Hoque. Robustness of a Sonlinear Missile Autopilot Designed Using Dynamic Inversion. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Dever,CO,U.S.A., August 2000. American Institute of Aeronautics and Astronautics. ISBN 978-1-62410-301-8. doi: 10.2514/6.2000-3970. (cit. on p. 28.)

[31] Andrii Mironchenko. *Input-to-State Stability: Theory and Applications*. Communications and Control Engineering. Springer International Publishing, Cham, 2023. ISBN 978-3-031-14673-2 978-3-031-14674-9. doi: 10.1007/978-3-031-14674-9. (cit. on p. 19.)

[32] Blaise Morton and Robert McAfoos. A Mu-Test for Robustness Analysis of a Real-Parameter Variation Problem. In *1985 American Control Conference*, 1985. (cit. on p. 14.)

[33] Dragan Nesić and Andrew R. Teel. Sampled-Data Control of Nonlinear Systems: An Overview of Recent Results. In S.O. Reza Moheimani, editor, *Perspectives in Robust Control*, pages 221–239. Springer London, London, 2001. ISBN 978-1-85233-452-9. doi: 10.1007/BFb0110623. (cit. on p. 15.)

[34] Carl Edward Rasmussen and Zoubin Ghahramani. Occam's Razor. In *Advances in Neural Information Processing Systems 13: Proceedings of the 2000 Conference*, volume 13. MIT Press, 2000. (cit. on p. 64.)

[35] Wilson J. Rugh. *Linear System Theory*. Prentice Hall Information and System Sciences Series. Prentice Hall, Upper Saddle River, NJ, 2. ed edition, 1996. ISBN 978-0-13-441205-4. (cit. on p. 4.)

[36] Shankar Sastry. *Nonlinear Systems*, volume 10 of *Interdisciplinary Applied Mathematics*. Springer New York, New York, NY, 1999. ISBN 978-1-4419-3132-0 978-1-4757-3108-8. doi: 10.1007/978-1-4757-3108-8. (cit. on p. 29.)

[37] Hanspeter Schaub and John L. Junkins. *Analytical Mechanics of Space Systems*. AIAA Education Series. American Institute of Aeronautics and Astronautics, Inc(AIAA), Reston, VA, fourth edition edition, 2018. ISBN 978-1-62410-521-0. (cit. on p. 7.)

[38] Peter Seiler, Andrew Packard, and Pascal Gahinet. An Introduction to Disk Margins. *IEEE Control Systems*, 40(5):78–95, October 2020. ISSN 1066-033X, 1941-000X. doi: 10.1109/MCS.2020.3005277. (cit. on pp. 17 and 18.)

[39] Athanasios Sideris and Ricardo S. Sanchez Pena. Robustness Margin Calculation with Dynamic and Real Parametric Uncertainty. In *1988 American Control Conference*, pages 1201–1206, Atlanta, GA, USA, June 1988. IEEE. doi: 10.23919/ACC.1988.4789903. (cit. on p. 14.)

[40] S. Sieberling, Q. P. Chu, and J. A. Mulder. Robust Flight Control Using Incremental Nonlinear Dynamic Inversion and Angular Acceleration Prediction. *Journal of Guidance, Control, and Dynamics*, 33(6):1732–1742, November 2010. ISSN 0731-5090, 1533-3884. doi: 10.2514/1.49978. (cit. on p. 29.)

[41] Benjamin M Simmons. *Advances in Aero-Propulsive Modeling for Fixed-Wing and eVTOL Aircraft Using Experimental Data*. PhD thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA, April 2023. (cit. on p. 12.)

[42] Benjamin M. Simmons, James L. Gresham, and Craig A. Woolsey. Flight-Test System Identification Techniques and Applications for Small, Low-Cost, Fixed-Wing Aircraft.

Journal of Aircraft, 60(5):1503–1521, September 2023. ISSN 0021-8669, 1533-3868. doi: 10.2514/1.C037260. (cit. on pp. 12 and 57.)

[43] Benjamin M. Simmons, James L. Gresham, and Craig A. Woolsey. Nonlinear Dynamic Modeling for Aircraft with Unknown Mass Properties Using Flight Data. *Journal of Aircraft*, 60(3):968–980, May 2023. ISSN 0021-8669, 1533-3868. doi: 10.2514/1.C037259. (cit. on p. 12.)

[44] P. Smith. A Simplified Approach to Nonlinear Dynamic Inversion Based Flight Control. In *23rd Atmospheric Flight Mechanics Conference*, Boston,MA,U.S.A., August 1998. American Institute of Aeronautics and Astronautics. doi: 10.2514/6.1998-4461. (cit. on p. 29.)

[45] S. Snell. Preliminary Assessment of the Robustness of Dynamic Inversion Based Flight Control Laws. In *Guidance, Navigation and Control Conference*, Hilton Head Island,SC,U.S.A., August 1992. American Institute of Aeronautics and Astronautics. doi: 10.2514/6.1992-4330. (cit. on p. 28.)

[46] Vadim I. Utkin. *Sliding Modes in Control and Optimization*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1992. ISBN 978-3-642-84381-5 978-3-642-84379-2. doi: 10.1007/978-3-642-84379-2. (cit. on p. 1.)

[47] Arjan van der Schaft. Port-Hamiltonian systems: An introductory survey. pages 1339–1365. May 2007. doi: 10.4171/022-3/65. (cit. on p. 41.)

[48] Xuerui Wang, Erik-Jan Van Kampen, Qiping Chu, and Peng Lu. Stability Analysis for Incremental Nonlinear Dynamic Inversion Control. *Journal of Guidance, Control, and Dynamics*, 42(5):1116–1129, May 2019. ISSN 0731-5090, 1533-3884. doi: 10.2514/1.G003791. (cit. on p. 29.)

[49] Wei-Min Lu and J.C. Doyle. Robustness Analysis and Synthesis for Nonlinear Uncertain Systems. *IEEE Transactions on Automatic Control*, 42(12):1654–1662, December 1997. ISSN 00189286. doi: 10.1109/9.650015. (cit. on p. 13.)

[50] Samuel B. Widman, Ian Willebeek-LeMair, and Craig A. Woolsey. Performance and Robustness Assessment for a Port-Hamiltonian Input-to-State Stabilizing Flight Controller. In *AIAA SCITECH 2026 Forum*, Orlando, FL, January 2026. American Institute of Aeronautics and Astronautics. In preparation. (cit. on pp. 2, 20, 41, and 84.)

[51] Ian Willebeek-LeMair, Samuel B. Widman, and Craig A. Woolsey. Input-to-State Stable Energy-Based Position Tracking Control for Atmospheric Flight Vehicles. In *AIAA SCITECH 2025 Forum*, Orlando, FL, January 2025. American Institute of Aeronautics and Astronautics. ISBN 978-1-62410-723-8. doi: 10.2514/6.2025-2594. (cit. on p. 41.)

[52] Kevin A. Wise. Comparison of Six Robustness Tests Evaluating Missile Autopilot Robustness to Uncertain Aerodynamics. *Journal of Guidance, Control, and Dynamics*, 15(4):861–870, July 1992. ISSN 0731-5090, 1533-3884. doi: 10.2514/3.20918. (cit. on pp. 13 and 14.)

[53] Thomas R. Yechout, Steven L. Morris, David E. Bossert, Wayne F. Hallgren, and James K. Hall. *Introduction to Aircraft Flight Mechanics: Performance, Static Stability, Dynamic Stability, Classical Feedback Control, and State-Space Foundations*. AIAA Education Series. AIAA, American Inst. of Aeronautics and Astronautics, Reston, Va, 2. ed edition, 2014. ISBN 978-1-62410-254-7. (cit. on pp. 9 and 16.)

[54] Zongyu Zuo, Cunjia Liu, Qing-Long Han, and Jiawei Song. Unmanned Aerial Vehicles: Control Methods and Future Challenges. *IEEE/CAA Journal of Automatica Sinica*, 9 (4):601–614, April 2022. ISSN 2329-9266, 2329-9274. doi: 10.1109/JAS.2022.105410. (cit. on p. 1.)

Appendices

Appendix A

Simulation Guidance

This chapter is designed to help anyone attempting to recreate this simulation with specific guidance. Listed below are simulation values, procedures, and other useful implementation tips that should help with this process.

A.1 Simulation Parameters

The values listed below are in no particular order, other than what order the values were implemented in the simulation for this work.

- The total simulation time was set to 60 seconds, with a step size of 0.01.
- The maximum control actuator deflections are $\delta_{a,max} = \pm 29°$, $\delta_{e,max} = \pm 27°$, and $\delta_{r,max} = \pm 34°$ for the aileron, elevator, and rudder respectively. A notional maximum thrust of 50 N was applied, but the limit likely did not have an effect on the simulation.
- The cruise revolutions per second for the propeller n_{cruise} was 133 rev/s, the cruise airspeed V_{cruise} was found to be 18 m/s, and the centering reference value for the inverse advance ratio $\mathcal{J}_0 = 1.88$.
- The random initial vehicle position was uniformly sampled between the noted range in Table A.1

Table A.1: Initial random state ranges

(a) Position

Variable	N	E	D
Range	0 m	0 m	0 m

(b) Body Velocities

Variable	u	v	w
Range	$18 \pm 5\frac{\text{m}}{\text{s}}$	$\pm 5\frac{\text{m}}{\text{s}}$	$\pm 5\frac{\text{m}}{\text{s}}$

(c) Euler Angles

Variable	ϕ	θ	ψ
Range	$\pm\frac{\pi}{6}$ rad	$\pm\frac{\pi}{6}$ rad	$\pm\frac{\pi}{6}$ rad

(d) Angular Rates

Variable	p	q	r
Range	$\pm\frac{\pi}{2}\frac{\text{rad}}{\text{s}}$	$\pm\frac{\pi}{2}\frac{\text{rad}}{\text{s}}$	$\pm\frac{\pi}{2}\frac{\text{rad}}{\text{s}}$

with the only change being computing the respective quaternion to remove the chance

of singularities in the dynamics. For NDI and RPH, the initial control state was set to be $[T, \dot{T}] = [7.5,0]$ based on testing of the simulation.

- The user-defined reference input that was passed to every control law specified $V^\star(t)$ - airspeed, $\gamma^\star(t)$ - flight path angle, $\chi^\star(t)$ - course angle, and $\beta^\star(t)$ - sideslip angle per Section 4.2

$$
\begin{aligned}
V^\star(t) &= 18.005 \ \text{[m/s]} && \text{For } 0 < t < 60 \\
\gamma^\star(t) &= \frac{\pi}{6} \ \text{[rad]} = 20^\circ && \text{For } 40 < t < 55 \\
\gamma^\star(t) &= 0 \ \text{[rad]} = 0^\circ && \text{Otherwise} \\
\chi^\star(t) &= \frac{\pi}{2} + \frac{\pi}{2}\left(3\cos\left(\frac{\pi}{20}(t-10)\right)\right) \ \text{[rad]} && \text{For } 20 < t < 40 \\
\chi^\star(t) &= \frac{\pi}{2} \ \text{[rad]} = 90^\circ && \text{Otherwise} \\
\beta^\star(t) &= 0 \ \text{[rad]} = 0^\circ && \text{For } 0 < t < 60
\end{aligned}
\tag{A.1.1}
$$

$V^\star(t)$ was chosen based on the cruise airspeed of the MTD aircraft and $\beta^\star(t)$ was chosen to ensure coordinated flight, but $\gamma^\star(t)$ and $\chi^\star(t)$ were chosen to provide nonlinear output tracking to sufficiently test each control law.

The physical aircraft parameters used are shown in Table A.2.

Table A.2: MTD mass and geometric properties

Property	**Symbol**	**Value (SI)**	**Value (US)**
Mass	m	3.644 kg	0.250 slug
Length overall	LOA	1.75 m	5.75 ft
Mean aerodynamic chord	$\bar{c}$	0.254 m	0.833 ft
Projected wing span	b	1.83 m	6.0 ft
Projected wing area	S	0.465 m^2	1.53 ft^2
Propeller diameter	D	0.254 m	0.833 ft
Roll moment of inertia	I_{xx}	0.2949 kg·m^2	0.217 slug·ft^2
Pitch moment of inertia	I_{yy}	0.1365 kg·m^2	0.101 slug·ft^2
Yaw moment of inertia	I_{zz}	0.4703 kg·m^2	0.347 slug·ft^2
Product of inertia	I_{xz}	0.055 kg·m^2	0.041 slug·ft^2
Products of inertia	I_{xy}, I_{yz}	~ 0 kg·m^2	~ 0 slug·ft^2
Number of propellers	N	2	2

The aerodynamic parameters for the aerodynamic model of the MTD are shown in Table A.3.

Table A.3: Aerodynamic model parameters

(a) Force Parameters

C_{X_α}	$C_{X_{\alpha^2}}$	$C_{X_{\alpha^3}}$	$C_{X_{\beta^2}}$	$C_{X_{J_c}}$	$C_{X_{J_c^2}}$	C_{X_0}
0.2526	2.470	1.235	0.3858	0.01520	0.07419	-0.1226

C_{Y_β}	C_{Y_r}	$C_{Y_{\delta_r}}$
-0.4900	0.2529	-0.09060

C_{Z_α}	$C_{Z_{\alpha^3}}$	C_{Z_q}	$C_{Z_{\delta_e}}$	$C_{Z_{J_c}}$	C_{Z_0}
-4.817	11.12	-16.05	0.3583	-0.04413	-0.2450

(b) Moment Parameters

$C_{\mathcal{L}_\beta}$	$C_{\mathcal{L}_p}$	$C_{\mathcal{L}_r}$	$C_{\mathcal{L}_{\delta_a}}$	$C_{\mathcal{L}_{\delta_a^3}}$
-0.05439	-0.5072	0.06422	0.1687	-0.2037

$C_{\mathcal{M}_\alpha}$	$C_{\mathcal{M}_q}$	$C_{\mathcal{M}_{\delta_e}}$
-0.1599	-5.044	0.2297

$C_{\mathcal{N}_\beta}$	$C_{\mathcal{N}_{\beta^3}}$	$C_{\mathcal{N}_p}$	$C_{\mathcal{N}_r}$	$C_{\mathcal{N}_{\delta_a}}$	$C_{\mathcal{N}_{\delta_r}}$
0.07088	0.2097	0.05824	-0.1606	-0.04410	0.0437

Table A.4: Measurement noise standard deviation.

State	ϕ	θ	ψ	u	v	w	p	q	r
State $\boldsymbol{\sigma}$	7.04×10^{-4}	4.62×10^{-4}	4.56×10^{-4}	0.0025	0.1112	0.0810	0.0329	0.0384	0.0207

Table A.5: Model mismatch standard deviation of the error and error rates.

	Error σ	**Error Rate σ, Flt Data**	**Error Rate σ, $\frac{1}{30}^{\text{th}}$**
C_X	0.0678	0.0023	0.0023
C_Y	0.0155	0.0024	0.0005
C_Z	0.0731	0.0143	0.0024
$C_\mathcal{L}$	0.0072	0.0015	0.0002
$C_\mathcal{M}$	0.0108	0.0023	0.0004
$C_\mathcal{N}$	0.0036	0.0007	0.0001

A.2 Disturbance Parameters

Wind: Steady wind sampled from a Gaussian distribution with a standard deviation of 1.5, which has a 3 σ value of approximately 5 m/s. This choice was made because the figures in [10] appeared to have a steady wind of up to 5 m/s on a self-described "gusty" day. The other wind parameters are: the freestream velocity of the aircraft $U_{ref} = 35$ kts, which can be used to scale the turbulence; the aircraft height $h = 300$ ft, a normal height to fly aircraft at VT; and the wind velocity at 20 feet $u_{20} = 10$ based on "light" turbulence from [1], although the MTD is a much smaller aircraft than the MIL-HDBK was providing guidelines for.

Discretization: The sampling rate was set to 25 Hz, and the maximum time delay was set to [1/(sampling rate*time step) + 1]. The inclusion of "+1" allowed the random values to be rounded down to the next integer for use in simulation.

General Disturbance: The equations used to generate the applied general disturbance seen in Figures 3.2(a), 3.4(a), and 3.6(a) are seen in (A.2.1). The functions were generated to be pseudo-random, while also having trends of disturbance to stress the control laws the general disturbances were applied to.

$$
\begin{aligned}
w_{F_X} &= 1.2\sin\left(\frac{t}{12} + 8\sin^{20}\left(\frac{t}{24}\right) + \sin^{40}\left(\frac{2t}{3}\right) - \sin^{10}\left(\frac{t}{18}\right) + \sin\left(\frac{t}{120}\right)\right)\\
w_{F_Y} &= 0.92\sin\left(\frac{t}{3} - 4\sin^{20}\left(\frac{t}{24}\right) + \sin^{40}\left(\frac{2t}{3}\right) + \sin^{10}\left(\frac{t}{18}\right) + \sin\left(\frac{t}{120}\right)\right)\\
w_{F_Z} &= \sin\left(t - 2\sin^{20}\left(\frac{t}{24}\right) + \sin^{40}\left(\frac{2t}{3}\right) - \sin^{10}\left(\frac{t}{18}\right) + \sin\left(\frac{t}{120}\right)\right)\\
w_{M_\mathcal{L}} &= 1.1\sin\left(-\frac{t}{2} + 5\sin^{20}\left(\frac{t}{24}\right) - \sin^{40}\left(\frac{2t}{3}\right) + 3\sin^{10}\left(\frac{t}{18}\right) - \sin\left(\frac{t}{120}\right)\right)\\
w_{M_\mathcal{M}} &= \sin\left(-\frac{4t}{7} + \tfrac{1}{2}\sin^{20}\left(\frac{t}{24}\right) + 2\sin^{40}\left(\frac{2t}{3}\right) - \sin^{10}\left(\frac{t}{18}\right) + 2\sin\left(\frac{t}{120}\right)\right)\\
w_{M_\mathcal{N}} &= 0.87\sin\left(\frac{t}{5} - 4\sin^{20}\left(\frac{t}{24}\right) + \tfrac{4}{5}\sin^{40}\left(\frac{2t}{3}\right) - 5\sin^{10}\left(\frac{t}{18}\right) - 4\sin\left(\frac{t}{120}\right)\right)
\end{aligned}
\tag{A.2.1}
$$

Each was scaled by a factor of 1/10 in order to not overwhelm any individual control law.

A.3 Control Law Implementation

A.3.1 Linear Quadratic Regulator

The three trim conditions used to implement LQR were defined by: a desired airspeed $V^\star(t)$ of 35 kts, or approximately 18 m/s; a trim turn rate $\dot{\psi}$ of $\pi/20$, which was selected based on the rate of change of desired course angle $\chi^\star(t)$ in (A.1.1); and a climb rate of $V^\star(t)\sin(\pi/6)$,

which was chosen based on the flight path angle $\gamma^\star(t)$ in (A.1.1).

The trim states and control inputs for each trim condition are given in Table A.6

Table A.6: LQR control law trim conditions

Condition	Trim Values
SLUF	$\boldsymbol{x}_{\text{level}} = [\ 0\ \ 0\ \ 0\ \ 0.000\ \ 0.045\ \ 0\ \ 18\ \ 0\ \ 0.806\ \ 0.000\ \ 0.000\ \ 0.000\ ]^\top$ $\boldsymbol{u}_{\text{level}} = [0.000\ \ 0.031\ \ 0.000\ \ 215]^\top$
Turn	$\boldsymbol{x}_{\text{turn}} = [\ 0\ \ 0\ \ 0\ \ 0.282\ \ 0.046\ \ 0\ \ 18\ \ 0\ \ 0.859\ \ -0.007\ \ 0.044\ \ 0.151\ ]^\top$ $\boldsymbol{u}_{\text{turn}} = [-0.004\ \ 0.040\ \ 0.025\ \ 214]^\top$
Climb	$\boldsymbol{x}_{\text{climb}} = [\ 0\ \ 0\ \ 0\ \ 0.000\ \ 0.393\ \ 0\ \ 18\ \ 0\ \ 0.788\ \ 0.000\ \ 0.000\ \ 0.000\ ]^\top$ $\boldsymbol{u}_{\text{climb}} = [0.000\ \ 0.031\ \ 0.000\ \ -34.8]^\top$

Using the SLUF trim condition, the values of $\boldsymbol{A}$,$\boldsymbol{B}$, $\boldsymbol{Q}$, and $\boldsymbol{R}$ were found to be

$$\boldsymbol{A} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0.045 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & -9.80 & 0 & 0.15 & 0 & 0.664 & 0 & -0.806 & 0 \\ 9.80 & 0 & 0 & 0 & -0.69 & 0 & 0.806 & 0 & -17.7 \\ 0 & -0.439 & 0 & -0.758 & 0 & -6.73 & 0 & 15.1 & 0 \\ 0 & 0 & 0 & 0 & -1.5 & 0 & -14.9 & 0 & 1.35 \\ 0 & 0 & 0 & 0.068 & 0 & -1.5250 & 0 & -6.11 & 0 \\ 0 & 0 & 0 & 0 & 1.24 & 0 & -0.679 & 0 & -2.77 \end{bmatrix} \tag{A.3.1}$$

$$\boldsymbol{B} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -0.0109 \\ 0 & 0 & -2.30 & 0 \\ 0 & 9.08 & 0 & -0.003 \\ 95.8 & 0 & 2.99 & 0 \\ 0 & 39.5 & 0 & 0 \\ -4.64 & 0 & 16.0 & 0 \end{bmatrix}$$

$$\boldsymbol{Q}_{LQR} = \text{diag}\left([32.8\ \ 32.8\ \ 32.8\ \ 4\ \ 4\ \ 4\ \ 3.65\ \ 3.65\ \ 3.65]\right) \tag{A.3.2}$$

$$\boldsymbol{R}_{LQR} = \text{diag}\left([328\ \ 328\ \ 328\ \ 0.0111]\right) \tag{A.3.3}$$

Therefore, the final gain matrix $\boldsymbol{K}_{LQR}$ was found to be

$$\boldsymbol{K}_{LQR} = \begin{bmatrix} 0.436 & 0 & 0.256 & 0 & 0.0373 & 0 & 0.0581 & 0 & -0.784 \\ 0 & 1.06 & 0 & -0.111 & 0 & -0.004 & 0 & 0.142 & 0 \\ 0.0168 & 0 & 0.185 & 0 & -0.0376 & 0 & -0.0029 & 0 & 0.192 \\ 0 & 18 & 0 & -4.23 & 0 & -0.667 & 0 & 1.07 & 0 \end{bmatrix} \tag{A.3.4}$$

A.3.2 Nonlinear Dynamic Inversion

The equation used to calculate bank angle φ is given below, since the equation was not easily found by the author. The method of calculating φ is to find the velocity-to-wind frame rotation matrix $\boldsymbol{R}_{VW}$, and then calculate the corresponding rotation matrices to go through the other reference frames [(velocity to inertial) times (inertial to body) times (body to wind)] $\boldsymbol{R}_{VW} = \boldsymbol{R}_{VI}\boldsymbol{R}_{IB}\boldsymbol{R}_{BW}$ where each rotation matrix is defined in (2.2.9)-(2.2.12). The calculation for φ comes out to, using shorthand notation $c = \cos()$ and $s = \sin()$,

$$\varphi = \tan^{-1}\left(\frac{c\alpha(c\chi\sigma_1 + s\chi\sigma_2) - s\alpha c\theta s(\chi - \psi)}{c\alpha(c\gamma c\phi c\theta + c\chi s\gamma\sigma_2 - s\chi s\gamma\sigma_1) - s\alpha(-c\gamma s\theta + c\chi c\psi c\theta s\gamma + s\chi c\theta s\gamma s\psi)}\right) \tag{A.3.5}$$

where

$$\begin{aligned} \sigma_1 &= c\psi s\phi - c\phi s\psi s\theta \\ \sigma_2 &= s\psi s\phi + c\phi c\psi s\theta \end{aligned}$$

The matrix used for the input matrix for the kth order derivative in the derivation of NDI is

$$C = \begin{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} & \mathbf{0}_{5\times 5} \end{bmatrix} \tag{A.3.6}$$

Here are the equations used to transform the LQR $\boldsymbol{K}$ matrix into the state of NDI

$$\boldsymbol{x}_{LQR+} = [\boldsymbol{x}_{LQR}, T, \dot{T}]^\top \tag{A.3.7}$$

$$\frac{\partial T}{\partial \Omega} = 2C_{x_{\mathcal{J}^2}}\left(\frac{D}{V}\right)^2 n + \left(C_{x_{\mathcal{J}}} - 2C_{x_{\mathcal{J}^2}}\mathcal{J}_0\right)\frac{D}{V} \tag{A.3.8}$$

$$\left[\frac{\partial \boldsymbol{u_M}}{\partial \boldsymbol{\delta}}\right]^{-1} = \frac{1}{2}\rho V^2 S \begin{bmatrix} b & 0 & 0 \\ 0 & \bar{c} & 0 \\ 0 & 0 & b \end{bmatrix} \begin{bmatrix} C_{\mathcal{L}_{\delta_a}} & 0 & C_{\mathcal{L}_{\delta_r}} \\ 0 & C_{\mathcal{M}_{\delta_e}} & 0 \\ C_{\mathcal{N}_{\delta_a}} & 0 & C_{\mathcal{N}_{\delta_r}} \end{bmatrix} \tag{A.3.9}$$

$$r_{u_{\ddot{T}}} = 0.01 \tag{A.3.10}$$

Algorithm 1 kth Order Differentiation Operator for NDI

$i \leftarrow 1$
while $i < (m+1)$ **do**
  $L \leftarrow x$
  deriv $\leftarrow C_i \left(\frac{\partial L}{\partial \boldsymbol{x}}\right) B(x)$
  counter $\leftarrow 1$
  while deriv $= \mathbf{0}$ **do**
    $L \leftarrow \left(\frac{\partial L}{\partial \boldsymbol{x}}\right) A(x)$
    deriv $\leftarrow C_i \left(\frac{\partial L}{\partial \boldsymbol{x}}\right) B(x)$
    $L_{\text{counter}} \leftarrow L$
    counter $\leftarrow$ counter $+1$
  end while
  $A_i^* \leftarrow C_i \left(\frac{\partial L}{\partial \boldsymbol{x}}\right) A(x)$
  $B_i^* \leftarrow$ deriv
  $i \leftarrow i+1$
end while

However, due to the large symbolic size, $\frac{\partial \boldsymbol{x}_{NDI}}{\partial \boldsymbol{x}_{LQR+}}$ was solved symbolically by

$$\frac{\partial \boldsymbol{x}_{NDI}}{\partial \boldsymbol{x}_{LQR+}} = \frac{\partial \boldsymbol{x}_{NDI}}{\partial \boldsymbol{x}_{NDI_{\text{init}}}} \frac{\partial \boldsymbol{x}_{NDI_{\text{init}}}}{\partial \boldsymbol{x}_{LQR+}} \tag{A.3.11}$$

The translated $\boldsymbol{Q}_{NDI}$ matrix was found to be

$$\boldsymbol{Q}_{NDI} = \begin{bmatrix} \boldsymbol{Q}_1 & \mathbf{0} \\ \mathbf{0} & \boldsymbol{Q}_2 \end{bmatrix} \tag{A.3.12}$$

where

$$\boldsymbol{Q}_1 = \begin{bmatrix} 1.88 \times 10^3 & 6.13 \times 10^3 & -8.26 \times 10^6 & 6.01 \times 10^4 & 4.79 \times 10^3 & 3.32 \times 10^{-3} \\ 6.13 \times 10^3 & 2.00 \times 10^4 & 1.22 \times 10^{-5} & 1.96 \times 10^5 & 1.56 \times 10^4 & -4.89 \times 10^{-3} \\ -8.26 \times 10^6 & 1.22 \times 10^{-5} & 5.00 \times 10^{-7} & -2.22 \times 10^{-7} & -1.34 \times 10^{-3} & -2.01 \times 10^{-4} \\ 6.01 \times 10^4 & 1.96 \times 10^5 & -2.22 \times 10^{-7} & 1.92 \times 10^6 & 1.53 \times 10^5 & 8.92 \times 10^{-5} \\ 4.79 \times 10^3 & 1.56 \times 10^4 & -1.34 \times 10^{-3} & 1.53 \times 10^5 & 1.22 \times 10^4 & 0.538 \\ 3.32 \times 10^{-3} & -4.89 \times 10^{-3} & -2.01 \times 10^{-4} & 8.92 \times 10^{-5} & 0.538 & 0.0809 \end{bmatrix}$$

$$\boldsymbol{Q}_2 = \begin{bmatrix} 32.8 & 2.27 & 0 & -31.3 & 0 \\ 2.27 & 91.2 & 9.28 & 57.7 & 3.29 \\ 0 & 9.28 & 8.71 & 6.03 & 5.97 \\ -31.3 & 57.7 & 6.03 & 1.37 \times 10^3 & 4.52 \\ 0 & 3.29 & 5.97 & 4.52 & 7.74 \end{bmatrix} \tag{A.3.13}$$

The translated $\boldsymbol{R}_{NDI}$ matrix was found to be

$$\boldsymbol{R}_{NDI} = \begin{bmatrix} 0.816 & 0 & 1.58 & 0 \\ 0 & 11.3 & 0 & 0 \\ 1.58 & 0 & 6.03 & 0 \\ 0 & 0 & 0 & 0.01 \end{bmatrix} \tag{A.3.14}$$

Therefore, the final gain matrix $\boldsymbol{K}_{NDI}$ was found to be

$$\boldsymbol{K}_{NDI} = \begin{bmatrix} \boldsymbol{K}_1 & \boldsymbol{K}_2 \end{bmatrix} \tag{A.3.15}$$

where

$$\begin{aligned} \boldsymbol{K}_1 &= \begin{bmatrix} 67.5 & 225 & 20.7 & 2.14 \times 10^3 & 364 \\ 1.91 & 7.60 & 0.893 & 79.4 & 22.7 \\ -17.1 & -57.3 & -4.57 & -549 & -90.6 \\ -0.0637 & -0.538 & -0.916 & -3.46 & -10.6 \end{bmatrix} \\ \boldsymbol{K}_2 &= \begin{bmatrix} 21.7 & -0.656 & -1.36 & -0.504 & 0.160 & -0.0637 \\ 5.31 & 0.477 & 1.04 & 0.446 & -0.0994 & -0.00145 \\ -4.85 & 2.39 & 5.66 & 3.54 & -0.619 & 0.0272 \\ -1.64 & -6.22 & 20.3 & 6.35 & 370 & 38.9 \end{bmatrix} \end{aligned} \tag{A.3.16}$$

The values used in the low-pass filter for the reference dynamics are

1. $\omega_n = 2$ - natural frequency,
2. $\zeta = \frac{\sqrt{2}}{2}$ - damping ratio for $V^\star(t)$, $\gamma^\star(t)$, and $\chi^\star(t)$, and
3. $\zeta = 1$ - damping ratio for $\beta^\star(t)$

This selection was chosen because: a higher natural frequency ω_n caused the control laws to track the reference model less effectively due to the higher response of the reference model; a damping ratio of about 0.7 is the ideal combination of rise time and overshoot, which was perfectly fine for $V^\star(t)$, $\gamma^\star(t)$, and $\chi^\star(t)$; and a damping ratio of 1 was selected for $\beta^\star(t)$ since no overshoot was desired in sideslip.

The final gain matrix for the reference model was

$$\boldsymbol{K}_{\text{ref}} = \begin{bmatrix} 8 & 9.66 & 4.84 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 8 & 9.66 & 4.84 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 8 & 9.66 & 4.84 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 4 \end{bmatrix} \tag{A.3.17}$$

Lastly, the control limiting factor was chosen to be 10^5 to ensure that each feedback linearization calculated control input, both $\boldsymbol{v}$ and $\boldsymbol{u}$, did not cause the control calculation to become singular.

A.3.3 Robust Port-Hamiltonian Control

The permutation used to turn the state from $[V,\ \dot{V},\ \ddot{V},\ \gamma,\ \dot{\gamma},\ \ddot{\gamma},\ \chi,\ \dot{\chi},\ \ddot{\chi},\ \beta,\ \dot{\beta}]^\top$ to $[V\ \ \gamma\ \ \chi\ \ \dot{V}\ \ \dot{\gamma}\ \ \dot{\chi}\ \ \beta\ \ \ddot{V}\ \ \ddot{\gamma}\ \ \ddot{\chi}\ \ \dot{\beta}]^\top$ for use in RPH is $[1, 4, 7, 2, 5, 8, 10, 3, 6, 9, 11]$, and it was implemented in simulation using `MATLAB`

```
P = [1,4,7,2,5,8,10,3,6,9,11];
I = eye(size(Bstar,2));
Astar = I(P,:)*Astar;
Bstar = I(P,:)*Bstar;
Gstar = I(P,:)*Gstar;
```
(A.3.18)

The control gains implemented in the construction of RPH are

$$\begin{aligned} k_1 = k_2 = k_3 = 10 \\ p_1 = p_2 = p_3 = 1 \\ \mu_1 = \mu_2 = \mu_3 = 0.01 \end{aligned} \tag{A.3.19}$$

Algorithm 2 kth Order Differentiation Operator for RPH

$i \leftarrow 1$
while $i < (m+1)$ **do**
  $L \leftarrow x$
  deriv $\leftarrow C_i \left(\frac{\partial L}{\partial \boldsymbol{x}}\right) B(x)$
  counter $\leftarrow 1$
  while deriv $= \mathbf{0}$ **do**
    $L \leftarrow \left(\frac{\partial L}{\partial \boldsymbol{x}}\right) A(x)$
    deriv $\leftarrow C_i \left(\frac{\partial L}{\partial \boldsymbol{x}}\right) B(x)$
    $L_{\text{counter}} \leftarrow L$
    $G_{\text{counter}} \leftarrow C_i \left(\frac{\partial L}{\partial \boldsymbol{x}}\right) G(x)$
    counter $\leftarrow$ counter $+1$
  end while
  $A_i^* \leftarrow C_i \left(\frac{\partial L}{\partial \boldsymbol{x}}\right) A(x)$
  $B_i^* \leftarrow$ deriv
  $G_i^* \leftarrow \{G_{\text{counter}}\}_{j=1}^{d}$
  $i \leftarrow i + 1$
end while

Lastly, the control limiting factor was chosen to be 10^5 to ensure that each feedback linearization calculated control input, both $\boldsymbol{v}$ and $\boldsymbol{u}$, did not cause the control calculation to become singular.

Appendix B

L_2 Norm Results

Although the results presented in Chapter 5 discussed only the L_1 and L_∞ norm results, the L_2 norm results were also found for this work. These results were not included in the primary results section due to the similarity to the L_1 and L_∞ norms presented, so the inclusion of these results would have only distracted the reader and cluttered the space of results. The inclusion of these results here is to confirm the presented results with another norm calculation, as well as to demonstrate the slight differences provided by analyzing the L_2 norm. These results are presented in the same manner as above to facilitate the comparison between the different L_p norms. Further, the presented results above are linked here for easier comparison if desired.

1. Group 1 results: Figures 5.4 to 5.6
2. Group 2 results: Figures 5.7 to 5.9
3. Group 3 results: Figures 5.10 to 5.12

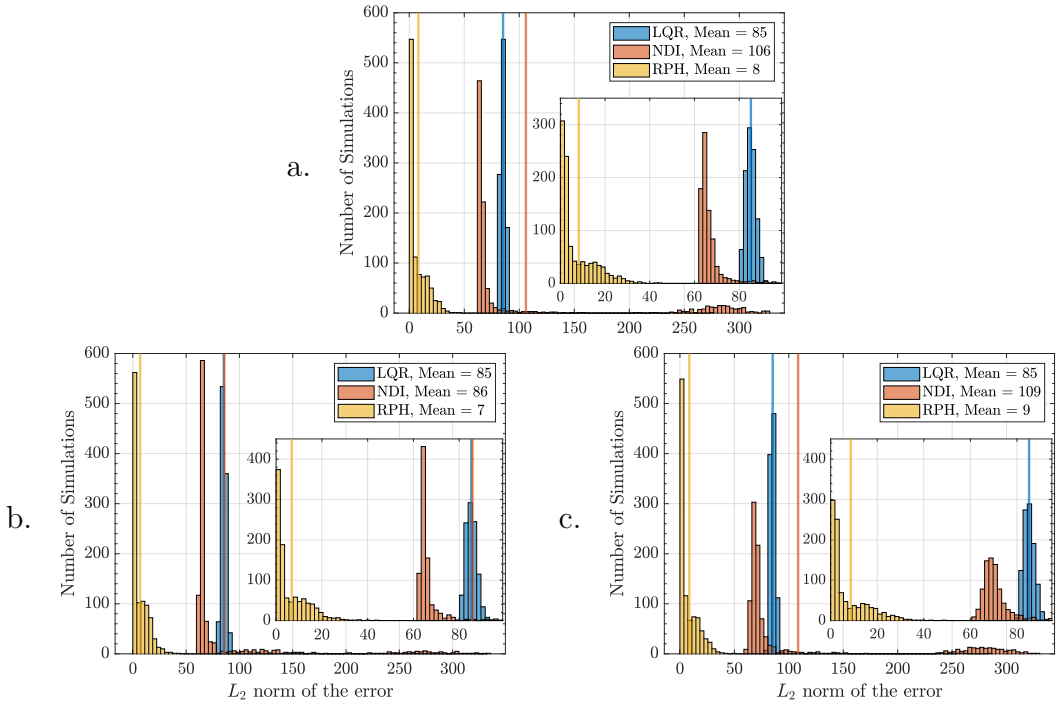

Figure B.1: L_2 norm results for (a) discretization with time delay only, (b) measurement noise only, and (c) measurement noise and discretization with time delay.

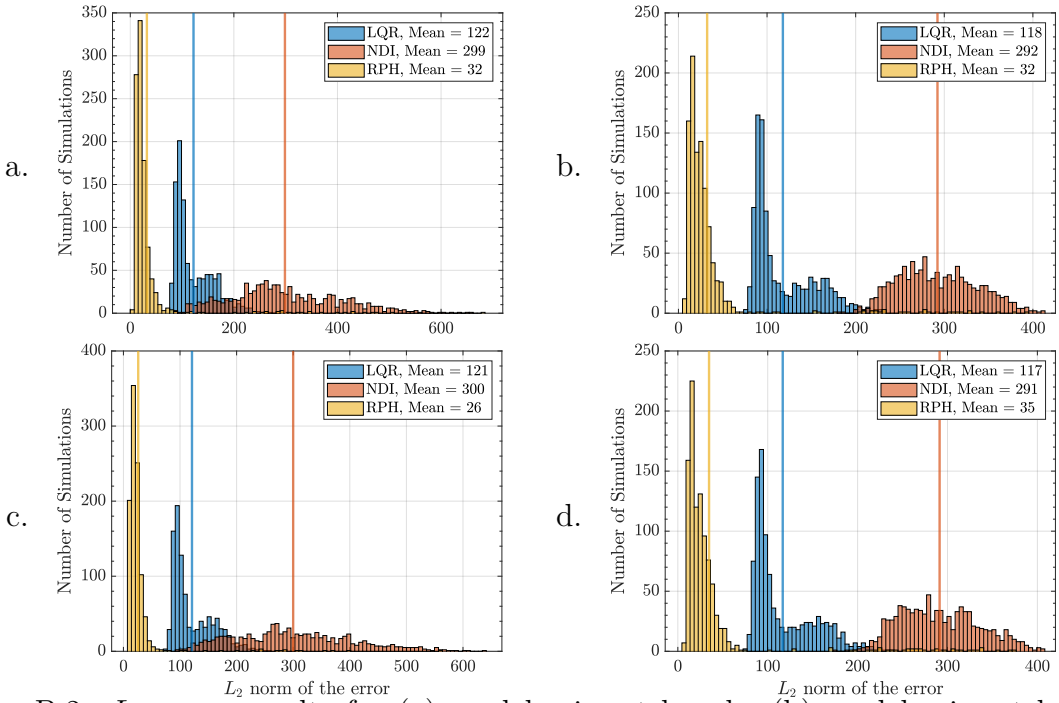

Figure B.2: L_2 norm results for (a) model mismatch only, (b) model mismatch and discretization with time delay, (c) measurement noise and model mismatch, and (d) measurement noise, model mismatch, and discretization with time delay.

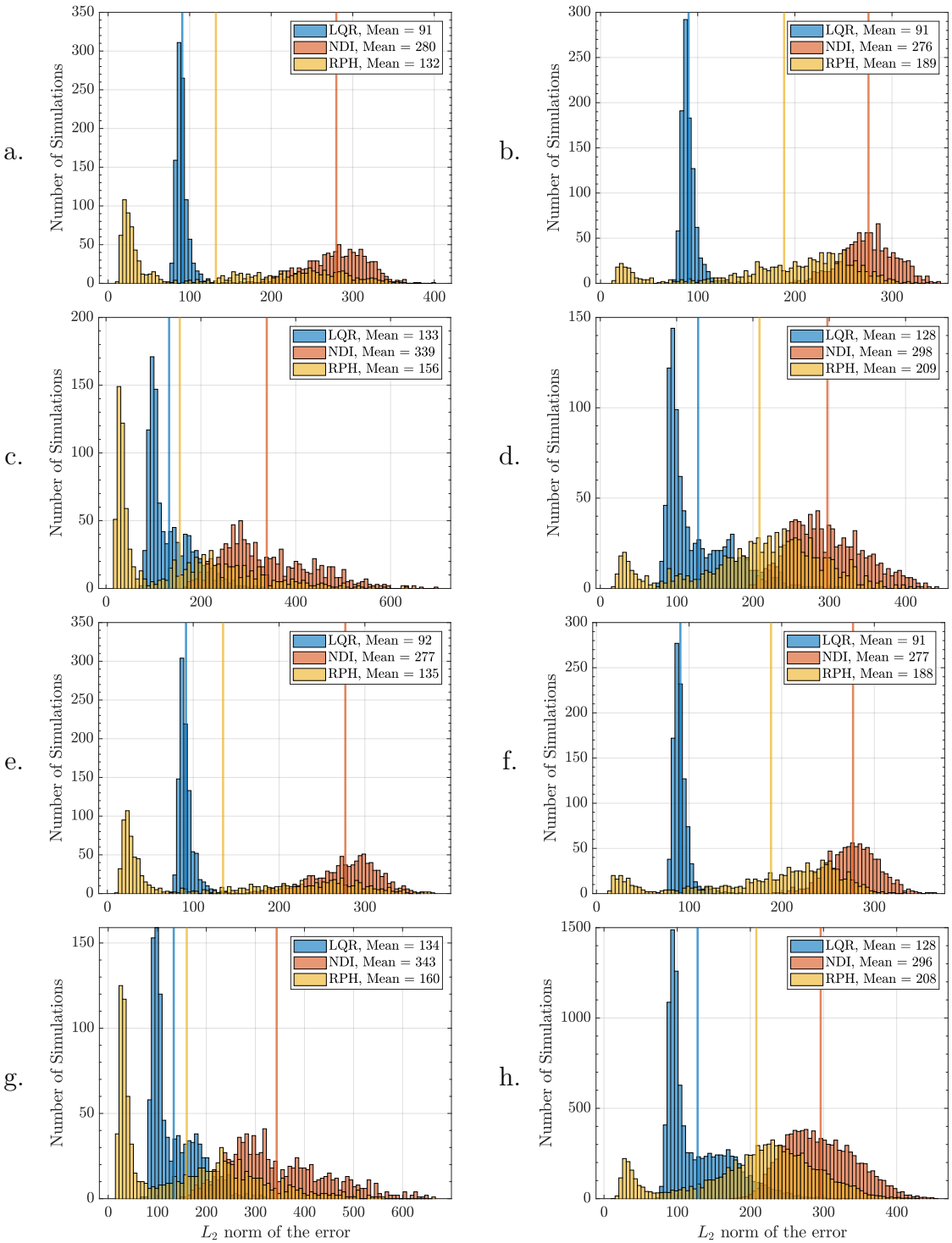

Figure B.3: L_2 norm results for (a) wind only, (b) wind and discretization with time delay (c) model mismatch and wind, (d) model mismatch, wind, and discretization with time delay, (e) measurement noise and wind, (f) measurement noise, wind, and discretization with time delay, (g) measurement noise, model mismatch and wind, and (h) all combined disturbances.