Thrust Vector Control for Flexible Launchers: an H_∞ Open Loop Shaping Approach

João Tiago Diz

TU Delft

Thrust Vector Control for Flexible Launchers: an H_∞ Open Loop Shaping Approach

Thesis report

by

João Tiago Diz

to obtain the degree of Master of Science
at the Delft University of Technology
to be defended publicly on February 25, 2025 at 14:00

Thesis committee:

Chair:	Dr. Ewoud Smeur
Supervisors:	Dr. Eng. Spilios Theodoulis
	Dr. Pedro Simplício
External examiner:	Dr. Jian Guo
Place:	Faculty of Aerospace Engineering, Delft
Project Duration:	February, 2024 - December, 2024
Student number:	5686830

Cover image: artist's impression of a launch vehicle's flight, produced by *Nikonaft* (2012) (modified).

Faculty of Aerospace Engineering · Delft University of Technology

Preface

I am proud to present this thesis report, which is the culmination of (almost) an entire year of hard work. During this work I had the opportunity to work with launch vehicles, something I had always wanted to do, and I was able to learn quite a lot. This would not have been possible without my thesis supervisors, Dr Eng. Spilios Theodoulis and Dr. Pedro Simplício, who not only gave me the opportunity to work on this topic, but also closely monitored my progress and always gave insightful tips. For this, I am very thankful.

It is my belief that this thesis produced interesting results and I hope it will inspire those who follow to build upon its findings. In particular, I trust that the $\mathcal{H}_\infty$ Open Loop Shaping guidelines will prove valuable for future work, as I have made every effort to present them in the simplest and most practical way possible. Additionally, I hope the launch vehicle modeling chapter serves as a useful resource for anyone working in LV control, as such a resource would have been immensely helpful to me during my own research.

Finally, I would like to thank my friends, and most importantly my parents, for the unwavering support throughout this journey. It would not have been possible without you.

Abstract

This thesis investigates the use of $\mathcal{H}_\infty$ Open Loop Shaping (OLS) for designing Thrust Vector Control (TVC) systems during the atmospheric ascent of flexible launch vehicles (LVs). The research has two main goals. The first goal is to assess whether $\mathcal{H}_\infty$ OLS can produce rigid body controllers for launch vehicles with comparable or superior robustness, stability, and performance to controllers designed with $\mathcal{H}_\infty$ Closed Loop Shaping (CLS), while also reducing design complexity and time. The second goal is to explore an integrated $\mathcal{H}_\infty$ OLS approach that tunes both the rigid body controller and the bending filter simultaneously, simplifying the traditional separate design process.

Results demonstrate that $\mathcal{H}_\infty$ OLS and $\mathcal{H}_\infty$ CLS yield controllers with identical performance, robustness, and stability when applied to design controllers with the same order and structure. However, $\mathcal{H}_\infty$ OLS significantly simplifies the design process by avoiding the manipulation of multiple transfer functions, offering higher reproducibility between flight points, and automatically ensuring robustness at both the plant input and output. Although these benefits come at the cost of converting requirements into open loop specifications and setting loop shaping weighting filters to meet the specifications, both challenges have been addressed and simplified in this thesis.

The thesis also extends the rigid body $\mathcal{H}_\infty$ OLS design process into an integrated methodology that simultaneously tunes both the rigid body controller and the bending filter, achieving identical results to the traditional separate design approach while reducing overall design time and complexity.

This research demonstrates that $\mathcal{H}_\infty$ OLS provides a robust framework for simplifying the design of TVC systems for launch vehicles, without compromising performance and robustness. Additionally, the integrated $\mathcal{H}_\infty$ OLS methodology for tuning both the rigid body controller and bending filter effectively streamlines the design process while achieving identical results to the traditional separate approach. These findings validate $\mathcal{H}_\infty$ OLS as a viable alternative to $\mathcal{H}_\infty$ CLS and introduce a more efficient methodology for designing both the rigid body controller and the bending filter for TVC control systems in launch vehicles.

Contents

Nomenclature

List of Abbreviations

Act	Actuation
AMF	Rigid/Flexible Body Analysis Model
AMR	Rigid Body Analysis Model
AoA	Angle of Attack
BF	Bending Filter
BM	Bending Modes
CF	Critical Frequency
CLFT	Closed Loop Transfer Function
CLS	Closed Loop Shaping
CM	Center of Mass
CP	Center of Pressure
CT	Center of Thrust
Dec	Decoupling
DMF	Rigid/Flexible Body Design Model
DMR	Rigid Body Robust Design Model
DoF	Degree of Freedom
DR	Disturbance Rejection
ESA	European Space Agency
FP	Flight Point
GM	Gain Margin
GNC	Guidance, Navigation and Control
HF	High Frequency
HFLMA	High Frequency, Low to Mid Amplitude
HOSMC	Higher-Order Sliding Mode Control
IMU	Inertial Measurement Unit
INDI	Incremental Nonlinear Dynamic Inversion
JAXA	Japan Aerospace Exploration Agency
LF	Low Frequency
LFHA	Low Frequency, High Amplitude
LFT	Linear Fractional Transformation
LL	Lead-Lag
LMI	Linear Matrix Inequalities
LPM	Load Performance Metric
LPV	Linear-Parameter-Varying
LQG	Linear Quadratic Gaussian
LQR	Linear Quadratic Regulator
LR	Load Relief
LTI	Linear-Time-Invariant
LTV	Linear-Time-Varying
LV	Launch Vehicle
MC	Monte Carlo
MIMO	Multiple-Input-Multiple-Output
MM	Multi Model
NASA	National Aeronautics and Space Administration
NCF	Normalized Coprime Factor
NDI	Nonlinear Dynamic Inversion
NMP	Non-Minimum Phase

OLS	Open Loop Shaping
PD	Proportional-Derivative
PI	Proportional-Integrative
PID	Proportional-Integral-Derivative
PM	Phase Margin
PP	Pole Placement
RF	Reference Frame
RHP	Right-Hand-Plane
RQ	Research Question
SIMO	Single-Input-Multiple-Output
SISO	Single-Input-Single-Output
SMART	Specific, Measurable, Achievable, Realistic, Timely
SMC	Sliding Mode Control
SSV	Structured Singular Value
SV	Singular Value
SVD	Singular Value Decomposition
TF	Transfer Function
TVC	Thrust Vector Control
TWD	Tail-Wag-Dog
VEGA	Vettore Europeo di Generazione Avanzata
WC	Worst-case

Launch Vehicle Model Symbols

α	Angle of attack
β	Angle of sideslip
δ_{pitch}	Thrust vector control actuator deflection in the pitch plane
δ_{yaw}	Thrust vector control actuator deflection in the yaw plane
ϵ_i	Attenuation at the center frequency of the ith notch filter
η_i	Width of the ith notch filter
ω	Angular Velocity of Coordinate system S' with regard to Coordinate system S
ω_i	Frequency of the ith bending mode
$\phi/\theta/\psi$	Euler Angles
$\Psi_i'(x)$	Rotational length of the ith bending mode
$\Psi_i(x)$	Translational length of the ith bending mode
ρ	Air density
τ	Time delay
θ	Pitch angle
$\tilde{q}$	Dynamic pressure
$\tilde{q}\alpha$	Load performance metric
$\vec{a}_i$	Acceleration of the ith element of mass with regard to S'
$\vec{F}_i$	Force applied on the ith element of mass
$\vec{F}$	Sum of forces
$\vec{F}_{g/A/T/TWD}$	Gravity/Aerodynamic/Thrust/Tail-Wag-Dog force
$\vec{F}_{x/y/z}$	Components of the Sum of forces
$\vec{M}_{S'x/S'y/S'z}$	Components of $M_{S'}$
$\vec{r}_c$	Position of the center of mass of the LV with regard to the origin of S'
$\vec{r}_i$	Position of the infinitesimal ith element of mass with regard to the origin of S'
$\vec{r}_{S'}$	Position of the origin of S' with regard to S
$\vec{V}_i$	Velocity of the ith Element of Mass with regard to S'
ζ_a	Actuator damping ratio
ζ_i	Damping of the ith bending mode

a_0 Trim velocity

C_n Stability derivative

$H_{0/S'}$ Angular Momentum about the origin of S/S'

I_{hh} Moment of Inertia with regard to axes h and h

$l_N/m_N/I_N$ Length/Mass/MOI of Nozzle

m_o Total Mass

$M_{0/S'}$ Torque about the origin of S/S'

$N_z\alpha/N_y\beta$ Force gradient with respect to α/β

Q_i Generalized force associated with the ith bending mode

q_i Generalized coordinate of the ith bending mode

S_{d0} Trajectory reference frame

S_{d0} Trajectory reference rrame

S_{d1-2} Auxiliary reference frames

S_{d3} Disturbed Body reference frame

S_{ref} Launch vehicle reference surface area

$u/v/w/p/q/r$ Disturbance component of velocity/angular velocity

$U_0/P_0/Q_0/R_0$ Steady-State component of velocity/angular velocity

$U_w/V_w/W_w$ Wind velocity components

v_w Wind Velocity in the pitch plane

V_{rel} Vehicle velocity with respect to the ground

w_a Actuator frequency

$x_{CM}/x_{CP}/x_{CT}/x_{IMU}$ Distance from the lower edge of the LV to a particular point (CM, CP, CT or IMU)

$x_{cm}/y_{cm}/z_{cm}$ Components of $\vec{r}_c$

$R_{S_{d3}S_{d0}}$ Rotation matrix from disturbed body reference frame to the trajectory reference frame

C Aerodynamic force coefficient

D Drag

Nα Normal force gradient with respect to α

P/Q/R Angular Velocity components of $\vec{w}$

S/S' Inertial/Body Coordinate System

T Thrust

U/V/W Velocity components of $\vec{r}_i$

z Lateral drift motion with respect to the reference trajectory frame

Robust Control Symbols

Δ Model Uncertainty

ϵ Inverse of the Robustness Margin

Γ Constraint tightness

γ Robustness Margin

μ Structured Singular Value

$\overline{\sigma}$ Maximum Singular Value

σ Singular Value

$\underline{\sigma}$ Minimum Singular Value

L_I Input open loop transfer matrix

L_O Output open loop transfer matrix

S_I Input Sensitivity Function

S_O Output Sensitivity Function

T_I Input Complementary Sensitivity Function

T_O Output Complementary Sensitivity Function

K Generalized Controller

P Generalized Plant

W1	Pre-Compensator	g	Gravity
W2	Post-Compensator	IMU	Relative to the IMU sensor
Subscripts		N	Relative to the Nozzle
f	Relative to the flexible motion	n	Nominal
A	Aerodynamic	T	Thrust

List of Figures

List of Tables

1

Introduction

Launch Vehicles (LVs) serve as the cornerstone of space exploration, satellite deployment, scientific research, and national security endeavors. Access to space is not only essential for expanding our understanding of the universe but also for driving technological innovation, fostering economic development, and strengthening global cooperation. As the demand for space access continues to rise, driven by the proliferation of satellite constellations, commercial space ventures, and international collaboration, the need for more frequent and reliable launch services becomes increasingly evident.

However, the development of launch vehicles is a particularly difficult task due to different reasons. Firstly, LVs must withstand extreme environments during their journey into space. During atmospheric flight, there are extremely high loads and temperatures applied to the LV, causing intense thermal and mechanical stress. These conditions require robust design of the LV to ensure mission success.

Secondly, the atmosphere presents numerous uncertainties that also need to be taken into account during design. Wind patterns, atmospheric density and other factors are always changing, and will affect the trajectory and loads experienced by the vehicle. This forces the LV to have a guidance, navigation and control system ready for the unknown.

Finally, and to add on top of all the aforementioned challenges, launch vehicles are expensive. This leads to limited flight test opportunities, when compared to aircraft, causing lack of experimental data and the increase of modeling uncertainty. In turn, this causes the design process to become even more critical, as even the slightest mistake can cause the entire LV flight to fail, resulting in a large monetary loss, and, more importantly, possible loss of human life.

Taking all these challenges in mind, launch vehicle manufacturers rely on strategies to attenuate the uncertainty: missions are planned for very particular trajectories, and can only operate within narrow wind and weather conditions. If the day of the launch comes, and conditions are not met, the launch will have to be postponed. This causes the cost of the launch to increase, but most importantly, it limits the opportunities for launching payloads into space.

In this context, the Thrust Vector Control (TVC) system, which is responsible for adjusting the direction of the engine's thrust to control the vehicle's attitude, plays a crucial role during atmospheric ascent flight. As illustrated in Figure 1.1, the TVC system operates using two translational actuators, which in this case are Electro-Mechanical Actuators (EMAs). These devices tilt the engine's movable part, often including the nozzle, to achieve correct thrust pointing as commanded by the Guidance, Navigation, and Control (GNC) system.

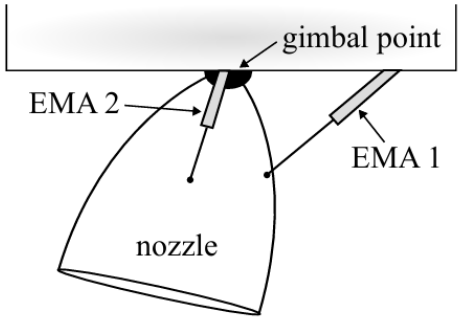

Figure 1.1: Simplified representation of a traditionally used thrust vectoring configuration, retrieved from *Fari* et al. (2024) with the permission of one of the authors.

The TVC system ensures precise control of the launch vehicle by maintaining stability, tracking guidance commands, and countering disturbances from uncertain atmospheric conditions and structural dynamics. These dynamics may include bending motions and, in the case of liquid engines, sloshing effects; however, this work focuses solely on solid-engine launch vehicles, where sloshing does not apply.

While launchers also employ a Roll Control System (RCS) to manage roll axis movement, the focus of this thesis is on the TVC system and its role in ensuring the vehicle follows the intended trajectory while safeguarding structural integrity and minimizing risks.

The challenges faced when designing the TVC control system are many: most launch vehicles are aerodynamically unstable (*Tapia*, 2019), the design needs to take into account several competing requirements (e.g. attitude tracking, load relief, actuation minimization, amongst others - see *Simplício* et al. (2016)), LV experience non-negligible flexible behavior (*Díaz*, 2010) and LVs are highly time varying systems (*Tapia*, 2019).

The traditional approach in designing the TVC control system - which is composed of rigid body controller (attitude controller) and bending filter - has been shown to be successful in tackling these issues (this approach is explained in Section 2.2), but it comes with some drawbacks, namely that it is time consuming and possesses no explicit way to address uncertainty (the control system is designed for nominal conditions and robustness is only considered implicitly through stability margin requirements in dispersed conditions). Improving on these drawbacks could yield several benefits. Firstly, it may result in a decrease in LV design costs, due to the decrease in design time. Secondly, there could be an enhancement in operational reliability and launch opportunities - as seen in *Tapia* (2019), robust control methods, which directly address uncertainty, can result in added performance and robustness of the TVC control system.

Given these insights, it is clear that the design and implementation of the TVC control system for the atmospheric ascent flight of launch vehicles is pivotal for the success and safety of their operations. Motivated by this understanding, an extensive literature review into the latest developments in launch vehicle TVC control for atmospheric ascent flight was performed. This review not only identified existing challenges and advancements but also revealed two significant research gaps. First, no studies have investigated whether $\mathcal{H}_\infty$ Open Loop Shaping (OLS) can be used to design attitude controllers that achieve robustness, performance, and stability equal to or better than those designed with $\mathcal{H}_\infty$ Closed Loop Shaping (CLS), while also reducing design complexity and time. Second, the idea of using $\mathcal{H}_\infty$ Open Loop Shaping for the integrated design of attitude controllers and bending filters, with the goal of reducing design time while maintaining performance, stability and robustness, has not been investigated. By addressing these gaps, this thesis seeks to advance the state of the art in launch vehicle control systems, contributing to safer, more reliable, and efficient atmospheric ascent operations.

1.1. Structure

The structure of the thesis report is as follows.

Chapter 2 discusses the state of the art in launch vehicle TVC control systems, focusing on both the advancements in the field and the key challenges in their design. It covers the historical development of launch vehicles, the limitations of the classical design approach, and the evolution of TVC control methods, culminating in a discussion of current state-of-the-art techniques and their associated challenges.

Chapter 3 outlines the research gap, objectives, and questions, along with the project planning, methodology, and expected outcomes.

Chapter 4 delves into the field of robust control, which serves as the foundation for the design of the control systems in this thesis. The chapter begins with a brief historical overview of the robust control field, followed by a discussion of key concepts such as uncertain system representations, linear fractional transformations, and singular values. Finally, it provides a detailed formulation and explanation of the two main controller synthesis methods utilized in this thesis: $\mathcal{H}_\infty$ Closed Loop Shaping ($\mathcal{H}_\infty$ CLS) and $\mathcal{H}_\infty$ Open Loop Shaping ($\mathcal{H}_\infty$ OLS).

Chapter 5 focuses on the modeling of the launch vehicle, starting with the derivation of the rigid body kinematics. Then, the relevant reference frames and the forces/moments — both rigid and flexible — that are applied to the LV are discussed. Following this, the analysis and design model considering only the rigid body forces are derived and linearized. Afterwards, these models are extended to take into account the flexible body dynamics. The chapter concludes with the presentation of the actuator and delay models, along with the values used in the different linear-time-invariant (LTI) models.

Chapter 6 focuses on the analysis and comparison of two controller design methods: $\mathcal{H}_\infty$ Closed Loop Shaping and $\mathcal{H}_\infty$ Open Loop Shaping. It begins with an analysis of system variability and the definition of the analysis point, followed by the establishment of the requirements for the launch vehicle TVC control system and the definition of the tests used to compare the different controllers that will be produced. The chapter continues with an analysis of the rigid body model, the delay model, and the actuator model, and this is followed by an initial attitude controller design using pole placement. Subsequently, attitude controllers are designed with $\mathcal{H}_\infty$ Closed Loop Shaping and with $\mathcal{H}_\infty$ Open Loop Shaping. A thorough comparison of the three controllers produced with the previous approaches is provided. Next, a design of a higher-order $\mathcal{H}_\infty$ Closed Loop Shaping is done so as to answer questions that arose during the investigation. Posteriorly, a summary of the author's experience with $\mathcal{H}_\infty$ Closed Loop Shaping and $\mathcal{H}_\infty$ Open Loop Shaping as methods for designing the attitude controller for the atmospheric ascent of a launch vehicle is given. The chapter concludes with a discussion of the first research question and an overview of the chapter's key points.

Chapter 7 presents the development of a novel methodology for designing an integrated rigid body controller/bending filter for the launch vehicle using $\mathcal{H}_\infty$ Open Loop Shaping. The chapter outlines the process of designing the rigid body controller and bending filter simultaneously and applies this methodology to the launch vehicle system. The results are then compared with those obtained from the traditional separate design approach. The chapter further explores two additional sections: the first applies the developed methodology to design a Multi Model controller, and the second tests the controller to demonstrate its ability to meet all launch vehicle requirements. The chapter concludes with a discussion addressing the second research question and provides an overview of the chapter's key points.

Chapter 8 presents the overall conclusion and outlines potential future research directions based on the work of this thesis.

2

Literature Review

2.1. History of Launch Vehicles

A launch vehicle, as described in *Encyclopedia Britannica* (2024), is ”a rocket-powered vehicle used to transport a spacecraft beyond Earth’s atmosphere, either into orbit around Earth or to some other destination in outer space”. Their development originated in the post-World War II era, when the onset of the Cold War prompted both the United States and the Soviet Union to prioritize the advancement of intermediate-range and intercontinental ballistic missiles, a pursuit that eventually paved the way for the creation of launch vehicles, as seen in *Wang* et al. (2023). The U.S. utilized captured V-2 rockets (see Figure 2.1) to build their first launch vehicle - Jupiter.

Figure 2.1: V-2 Rocket - *AElfwine* (2004).

Meanwhile, the soviets developed the R-7 based on their previous technology. Both launch vehicles were designed with the purpose of delivering nuclear weapons and not with the goal of accessing space. Following these developments, and, once again moved by the cold war, many LV were developed as a means of showcasing each nations technological prowess. With the advancement of launch vehicles

came parallel progress in the TVC control systems. The following sections will cover the improvements made to the TVC control system designed for the atmospheric ascent. However, before delving into the improvements, the classic approach used to design the TVC control system for the atmospheric ascent will be presented.

2.2. Classical TVC Design

The traditional approach to developing the flight control system for a launch vehicle's atmospheric ascent relies on thrust vector control to manage the yaw and pitch axes across multiple linear design points. The controllers for each design point are interconnected through gain scheduling to fine-tune the control strategy. For each design point, the controller design divides itself into three stages. First, a classic proportional-integrative-derivative (PID) controller, named attitude controller or rigid body controller, is designed using the loop-at-a-time approach while treating the launch vehicle as a rigid body. This is done so as to achieve load relief, drift minimization, tracking of the guidance system's attitude commands and stability requirements, for the yaw and pitch axis (*Tapia*, 2019). Second, a bending filter (BF) is designed so as to attenuate bending modes (BM). Third, the flexible body filter and rigid body controller are tuned together.

The great importance in including the flexible body dynamics lies in the fact that the BM can cause the system to become unstable (*Díaz*, 2010). To tackle the flexible dynamics issue, two strategies can be used - gain stabilization and phase stabilization, as stated in *Greensite* (1967). Gain stabilization aims at attenuating the bending mode so that it does not cause instabilities, while phase stabilization consist of utilizing a filter to shape the phase of the bending mode, so as to guarantee phase margins. The first method is commonly used for stabilization of the higher bending modes, while the second method is generally applied for the stabilization of the first bending mode (*Tapia*, 2019). It is worth noting that in the conventional design of bending filters, only notch and low-pass configurations are typically utilized, as outlined in *Greensite* (1967).

Having designed the controllers for each design point, the next step, is to ”link” them. In the classical approach this is usually done through gain scheduling, which consists of interpolating the linear control laws at intermediate operating conditions. In other words, a plan is developed to adjust the PID gains based on the observed operational parameters (often, time elapsed since launch). This approach enables the utilization of intuitive, linear tools to craft controllers for systems that are inherently non-linear. A thorough theoretical discussion of gain scheduling can be found in *Rugh* (1990).

The final step is to test the scheduled TVC control system to ensure that performance, robustness, and stability meet the required standards across the entire flight envelope. This is done through Monte Carlo and Vertex approach simulations for both time and frequency domains (*Marcos* et al., 2011). Monte Carlo simulations involve randomly varying the parameters used in the flight model and external inputs (e.g., wind) according to their statistical distribution. The number of simulations depends on the desired level of confidence. This method is simple to implement but computationally expensive.

On the other hand, the Vertex approach is a deterministic method that tests extreme cases. While it is also simple and rapid, there is no guarantee of encountering the worst-case scenario, as noted in *Marcos* et al. (2011). Using these methods, time and frequency domain simulations can be performed. Time-domain simulations are relatively straightforward, involving high-fidelity nonlinear simulations across the full mission or relevant phases, as seen in *Marcos* et al. (2011). Frequency-domain simulations, however, are limited to linear-time-invariant (LTI) models, restricting the depth of analysis, especially since

LV flight is highly non-stationary in certain phases (*Marcos* et al., 2011).

2.3. Issues With Classic TVC Design

As already mentioned, classic TVC design has been implemented successfully on different occasions. Nonetheless, there are issues associated with it that are well documented in literature, and that motivated control engineers to try to implement other control strategies. The following discussion aims at addressing the issues related with the design of the rigid body controller and of the flexible body controller (in other words, the bending filter).

Firstly, designing and tuning PID controllers for the rigid body controller is a time-consuming process. For example, the X-33, a planned next-generation, commercially operated reusable launch vehicle, required up to 7,800 values to provide the gains for the PID controller, so as to allow it to handle changing flight conditions, plant uncertainties, and disturbances, as described in *Drake* et al. (2004). One of the reasons for the large number of gains lies in the highly time varying nature of the launch vehicle system.

In addition to the first issue, classic control methods, have no means of explicitly addressing uncertainty during design. Stability robustness is enforced only through gain and phase margins, which means the classical approach guarantees robust control only when vehicle performance and operating conditions remain close to the design points (*Y. Shtessel* et al., 2002). Moreover, classic control methods are tailored for single-input-single-output (SISO) problems, but the LV attitude control problem is normally a multivariable control problem (*Tapia*, 2019).

Taking everything into account, the aim is to achieve a rigid body controller, in a time efficient manner, capable of directly addressing uncertainty during the design phase, ensuring robustness. Additionally, the method used should be tailored for MIMO (multiple-input-multiple-output) systems. The reader should note that utilizing a controller synthesis method capable of addressing the highly time varying nature of the system efficiently, should also result in a reduction in controller design time.

Secondly, there are several challenges related to flexible body control. As discussed in *Noll* et al. (1970), bending filters typically degrade the rigid body stability margins and performance. This issue becomes more pressing when the frequency of the first bending mode is close to the actuator's bandwidth. Additionally, modeling the flexible body behavior is challenging—typically, finite element analysis is required to model these behaviors. Note that the main properties of the bending modes (frequency, damping , etc...) change throughout the flight of the LV (*Tapia*, 2019), which just adds to the problems complexity. Finally, the traditional design approach is quite time-consuming when designing and integrating the bending filter, as stated in *Tapia* (2019). This is mainly because the bending filter is designed independently of the rigid body controller, requiring further retuning of the latter.

The challenges faced in rigid body control have compelled control engineers to explore new control methodologies capable of addressing multivariable problems more efficiently, while also accommodating the considerable uncertainty inherent in launch vehicle systems. Simultaneously, the challenges associated with flexible body behavior have prompted the exploration of new techniques for generating bending filters in a more efficient manner. Their goal is to reduce overall design time for the TVC control system and mitigate stability margins and performance degradation caused by the bending filters (see for instance *Tapia* (2019) and *Jang* et al. (2008)).

2.4. Improvements on Classic TVC Design - Rigid Body

In response to the rigid body controller limitations outlined in the preceding section, control engineers opted to explore novel control methodologies to address these challenges. One of the first advancements was the usage of optimal control. For instance, in *Brito* et al. (2006), the Linear Quadratic (LQ) criterion was used to tune the attitude controller gains for the Brazilian launcher. Similarly, for the Ariane 5 (which can be seen in Figure 2.2), the Linear Quadratic Gaussian (LQG) approach was originally implemented and used to design the rigid body controller, as discussed in *Pignie* (2002). Additionally, other studies have been conducted into the application of optimal control methods to LVs, such as *Blackburn* and *Vaughan* (1971) and *Rynaski* (1966), among many others. These methods and applications appeared to boast great reductions in design time and improvements in performance. They facilitated the expression of the control problem as an optimization problem, while also being tailored for MIMO systems. However, they possessed a major drawback: there was no clear procedure for incorporating robustness requirements in the controller synthesis process, as shown by *Bates* and *Postlethwaite* (2002). Furthermore, *J. C. Doyle* (1978) showed that LQG possessed no guaranteed stability margins against parametric uncertainties and unmodelled dynamics at the input and output of the plant.

In order to address these challenges, a new control theory emerged in the 1970's called robust control. This theory is based on the minimization of the $\mathcal{H}_\infty$ norm. The $\mathcal{H}_\infty$ norm of a system quantifies its maximum singular value, indicating the highest possible steady-state gain the system can achieve for sinusoidal inputs across all frequencies, as seen in *Bates* and *Postlethwaite* (2002). Thus, minimizing the $\mathcal{H}_\infty$ norm entails reducing the energy associated with the most adverse output signal vector. In its essence, robust control techniques are frequency domain techniques: they allow users to manipulate the open or closed loop singular values of the system in order to meet performance and robustness specifications. Additionally,

Figure 2.2: Launch of the Ariane V on 6 August 2019 from Europe's Spaceport in French Guiana - *ESA/CNES/Arianespace* (2019).

they allow for the consideration of system uncertainties, and they are suited for MIMO systems, resulting in the time efficient design of very robust controllers.

2.4.1. $\mathcal{H}_\infty$ Problem

The formal introduction of the $\mathcal{H}_\infty$ minimization problem is attributed to *Zames* (1981). This problem entails the search for a stabilizing controller that effectively fulfills specific constraints in an optimal manner.

The first documented state-space solution to the $\mathcal{H}_\infty$ problem was presented in *J. C. Doyle* (1984). It utilized the work of *Glover* (1984) on Nehari problems to solve the $\mathcal{H}_\infty$ problem. However, this solution required substantial computations. In the following years, a more practical solution was given in *Glover* and *Doyle* (1988). In this work, and in *J. C. Doyle* et al. (1989), Glover and Doyle showed that a sub-optimal solution of the problem can be found by solving two algebraic Riccati equations. The controllers computed were of full order. A few years later, *Gahinet* and *Apkarian* (1994), solved the problem by performing manipulations on linear matrix inequalities, allowing for the synthesis of sub-optimal, full order controllers. The authors also hinted at the possibility of utilizing linear matrix inequalities (LMIs) to produce reduced order controllers, but there was a catch - when structural constraints were enforced on the controller, the minimization problem became non-convex, and so global convergence was no longer guaranteed. At last, *Apkarian* and *Noll* (2006), solved the $\mathcal{H}_\infty$ problem with non smooth optimization techniques, allowing for the design of fixed-order, fixed-structure controllers. Note that up until this last solution, known as structured $\mathcal{H}_\infty$, $\mathcal{H}_\infty$ methods were used to design full order controllers, and then methods of order reduction were applied, to ensure that the designed system was viable. The solution by *Apkarian* and *Noll* (2006) became accessible for engineers between 2006 and 2010. In the context of designing the TVC control system controllers for LV's, this method has been used in different works, see for instance *Simplício* (2019), *Tapia* (2019), *Hernandez* et al. (2021). This solution was also used for other space related missions, for instance, it was used in the Rosetta mission as described in *Apkarian* and *Noll* (2017).

Finally, it is also important to mention that there are other solutions to the $\mathcal{H}_\infty$ problem, these are not mentioned for brevity. However, the interested reader can consult the work of *Schoon* (2024), where some of these solutions are described and implemented.

2.4.2. $\mathcal{H}_\infty$ Methods - Robust Control

The $\mathcal{H}_\infty$ methods have been widely used for the control of launch vehicles. Amongst the different $\mathcal{H}_\infty$ methods, three stand out: $\mathcal{H}_\infty$ Closed Loop Shaping (CLS), $\mu-$synthesis and $\mathcal{H}_\infty$ Open Loop Shaping (OLS). The following discussion will consider the different implementations of each method.

$\mathcal{H}_\infty$ ***CLS*** is a method which focuses on shaping the singular values of different closed loop transfer functions (CLTFs). By utilizing this approach, it becomes possible to synthesize a controller that effectively attenuates disturbances and measurement noise, and that is robust to multiplicative and inverse multiplicative input/output uncertainty and to additive and inverse additive uncertainty. Additionally, it allows for the incorporation of reference model tracking. *Simplício* (2019) used it to increase LV load relief, through the creation of a robust wind observer for the atmospheric part of the flight (both ascent and descent). *Tapia* (2019) worked with the VEGA LV, which is illustrated in Figure 2.3, to show that $\mathcal{H}_\infty$ CLS can be used to recover its legacy controller. This allowed for the demonstration that the same controller can be obtained with a more time-efficient technique. *Tapia* (2019) went on to utilizing $\mathcal{H}_\infty$ CLS to simultaneously compute the rigid body controllers and the bending filters, combining the three design phases mentioned earlier into a single one, while achieving improved performance and robustness. *Hernandez* et al. (2021) applied $\mathcal{H}_\infty$ CLS to design both full order and structured $\mathcal{H}_\infty$ controllers for CALLISTO's aerodynamic

Figure 2.3: VEGA Launch Vehicle - *Carril* (2024).

ascent (CALLISTO is a reusable rocket demonstrator). *Belletti Araque* et al. (2021) applied $\mathcal{H}_\infty$ CLS to design the controller for the atmospheric ascent flight of a LV. In his approach, a genetic optimization algorithm was developed to tune the weight filters that were used to shape the CLTFs, effectively achieving an implementation capable of automatically generating $\mathcal{H}_\infty$ CLS controllers for the LV, regardless of the flight point. *Biertümpfel* et al. (2024) combined the $\mathcal{H}_\infty$ CLS method with a linear time varying (LTV) model. In his implementation, the aim was to synthesize a scheduled controller by formulating the control problem in terms of a time varying model. The approach allowed for the explicit accommodation of the launcher's changing dynamics and evolving control objectives throughout the ascent trajectory, incorporating the latter via time-varying weighting functions that were used to shape the CLTFs. The work was able to achieve promising results. *Navarro-Tapia* et al. (2017) combined $\mathcal{H}_\infty$ CLS with a linear parameter varying (LPV) model. The implementation aimed at synthesizing a scheduled controller by formulating the control problem in terms of a model dependent on time varying parameters. The implementation showed increased robust performance when compared to controllers designed with structured $\mathcal{H}_\infty$. Note that the last two implementations are specifically designed to address the highly time varying nature of the LV system. Other works have been done into applying $\mathcal{H}_\infty$ CLS to LV, such as *Hague* (2000), *George* et al. (2021), etc.

The $\mathcal{H}_\infty$ CLS technique, while powerful, does have certain limitations. For instance, when employing the simpler 2-block formulation—where only two cost functions, such as sensitivity and complementary sensitivity, are minimized—the $\mathcal{H}_\infty$ CLS approach typically ensures robustness only at either the plant input or output (*Bates* & *Postlethwaite*, 2002). The primary advantage of this formulation lies in its simplicity, making controller design more straightforward.

Achieving robustness at both the plant input and output requires adopting a 4-block formulation (where four cost functions are minimized). While this formulation addresses the limitation of the 2-block approach, it comes with the drawback of increased design complexity, due to the designer having to manipulate multiple CLTFs simultaneously.

Another potential drawback of the $\mathcal{H}_\infty$ CLS approach is the challenge of managing conflicting objectives during the shaping process, which can arise based on the specific requirements of the flight control system. This requires the designer to carefully balance and harmonize the shaping of individual CLTFs, often leading to added complexity in the design process as adjustments and iterations are needed to meet all performance requirements and stability constraints.

To overcome these shortcomings, the concept of $\mathcal{H}_\infty$ Open Loop Shaping (OLS) was developed. $\mathcal{H}_\infty$ Loop Shaping is based on the early contributions of Vidyasagar (see for instance *Vidyasagar* and *Kimura* (1986)). Subsequently, it was extensively detailed in *McFarlane* et al. (1988), further refined in *McFarlane* and *Glover* (1988) and *McFarlane* (1989). Finally, in *McFarlane* and *Glover* (1992) a systematic framework and practical application of the method is provided. $\mathcal{H}_\infty$ Open Loop Shaping is a combination of an $\mathcal{H}_\infty$ optimization problem with classic open loop shaping methods. The open loop shaping methods adjust the singular values of the open loop system to achieve a desired shape, meanwhile the $\mathcal{H}_\infty$ optimization aims at providing closed loop stability and a level of robustness at all frequencies. This method ensures the system's robustification against normalized coprime factor uncertainty (NCF), automatically guaranteeing robustness at both the plant input and output, therefore sparing the designer from having to manipulate multiple CLTFs to achieve the same result. Furthermore, by directly shaping the open loop, $\mathcal{H}_\infty$ OLS inherently shapes the different CLTFs, thus preventing the designer from having to reconcile conflicting closed loop specifications.

$\mathcal{H}_\infty$ ***Open Loop Shaping***, to the author's best knowledge, has only been applied to LVs once, in the work of *Xing* and *Bainum* (1996). In this paper, Xing and Bainum began by mathematically formulating the launch vehicle model, where only the pitch plane was considered. Subsequently, they detailed the uncertainties associated with each parameter of the model and provided a description of the actuator model. Next, a description of the launch vehicle attitude control problem is given, highlighting its complexities and distinctive characteristics. It delves into the challenges inherent to this problem and discusses the approach to formulate the controller design problem as an $\mathcal{H}_\infty$ minimization problem. Furthermore, it elucidates the methods employed to address and solve this problem effectively. The next step entailed the $\mathcal{H}_\infty$ Open Loop Shaping procedure description and its application to the LV. Likewise, an LQG and a PID controllers are also designed. The author does not specify how the PID controller was tuned. In the penultimate step, the comparison of the three controllers begins, it is divided into three stages: a comparison on nominal performance, a comparison on robust stability and a comparison on robust performance. In the first comparison, the goal was to test the amplification of disturbances by the nominal system (using the different controllers). It was shown, that $\mathcal{H}_\infty$ Open Loop Shaping was the method with the best nominal results, as it was capable of attenuating disturbances across the entire frequency range tested (the other methods were not). For the second comparison, μ analysis was performed, so as to obtain the stability margin to parameter uncertainty. It was concluded that $\mathcal{H}_\infty$ Open Loop Shaping had the highest robustness against parameter uncertainty, as it could withstand up to 35% deviation from the nominal value in any parameter used in the model, while LQG could only withstand up to 15% and PID 10%. For the third and final comparison, the aim was to test how quickly, and how much, the performance of the system degraded when faced with non-nominal conditions. Once again, it was shown that $\mathcal{H}_\infty$ Open Loop Shaping attained the best results. The final step of this paper entails the time domain simulations of the system utilizing the three developed controllers. The results mostly corroborate the findings of the previous analysis, however, it was shown that all margins were slightly higher than those computed with μ analysis.

In the work of *Xing* and *Bainum* (1996), no comparison has been drawn regarding the $\mathcal{H}_\infty$ CLS approach. Nevertheless, considering the earlier discussion, it would be intriguing to implement both methods to design the attitude controller for the atmospheric ascent phase of a launch vehicle and assess the respective resulting controllers. This comparison would examine whether the $\mathcal{H}_\infty$ Open Loop Shaping method can achieve similar outcomes to the $\mathcal{H}_\infty$ CLS approach for launch vehicle attitude control during atmospheric ascent, while reducing design complexity/time. Furthermore, the added design complexity inherent to the $\mathcal{H}_\infty$ CLS approach could potentially hinder the designer's ability to achieve comparable results, making it

possible for $\mathcal{H}_\infty$ Open Loop Shaping to deliver superior outcomes in practice, while still reducing design complexity/time.

On a different note, the primary limitation of $\mathcal{H}_\infty$ Open Loop Shaping lies in the difficulty of efficiently shaping multiple loops. To address this challenge, *Papageorgiou* (1998) provides design guidelines for the open loop shaping process. In addition to the challenges associated with open loop shaping, $\mathcal{H}_\infty$ Open Loop Shaping does not support reference model following, further limiting its applicability in certain scenarios. However, in *Limebeer* et al. (1993), a technique derived from $\mathcal{H}_\infty$ Open Loop Shaping, called Two Degree of Freedom $\mathcal{H}_\infty$ Open Loop Shaping was introduced. This method allows for the incorporation of reference model tracking, while making sure the system is robust against NCF. From the present literature review, it was concluded that this method has yet to be applied to LV's GNC systems. However, depending on the flight point chosen for the control synthesis, it might happen that no specific demands are placed on reference model tracking and so this technique may not be particularly intriguing.

$\mu-$***synthesis*** focuses on robustifying the system against structured uncertainty, it can allow for the synthesis of controllers that are less conservative than those produced with either $\mathcal{H}_\infty$ Open Loop Shaping or $\mathcal{H}_\infty$ CLS. In the past, it has been applied by *Tony* and *C.* (2015) to the atmospheric ascent of an LV, by *Morita* and *Goto* (2008) to M-V - JAXA's LV. Both applications showed promising results, and the latter even showed successful flight operation, as seen in *Morita* and *Goto* (2008). However, there have been few research studies conducted on this method for launcher control design. Furthermore, its acceptance in the space industry is not yet widespread, as stated by *Tapia* (2019). One of the main reasons being that the controllers resulting of this method are of higher order than those produced by $\mathcal{H}_\infty$ CLS.

2.4.3. Nonlinear Control

Another "family" of control methods that has been widely used to build on tackle the issues associated with classical and optimal control is nonlinear control. Nonlinear control can be divided into two different categories: feedback linearization techniques, such as as ***nonlinear dynamic inversion*** - **NDI**, ***incremental nonlinear dynamic inversion*** - **INDI**; Lyapunov based techniques, such as ***sliding mode control*** - **SMC**, ***higher-order sliding mode control*** - **HOSMC** and ***backstepping***. These techniques have been implemented to design launch vehicle attitude controllers. The following discussion aims at briefly discussing each technique.

Nonlinear dynamic inversion (NDI) is a method aimed at mitigating the effects of nonlinearities within a system, ultimately leading to the creation of a closed loop system that can be described in a linear format. Then, using some control method, a controller can be designed to control the system. Some of the earliest works on dynamic inversion can be found in *Brockett* (1978) or *Hunt* and *Su* (1981). One of the pioneering applications of NDI can be seen in *Meyer* et al. (1984), where NDI was employed for the design of a helicopter autopilot. In *Ansari* and *Bajodah* (2015), an application of NDI to LV attitude control was performed. It was shown that the controller could stabilize the LV, but no analysis was performed on robustness to uncertainty. Additionally, this technique can possess a major drawback - it assumes that the system model is known exactly. For the case of the LV, where uncertainty is one of the greatest problems, this can present as a major issue. Its major benefit lies in the fact that NDI transforms the nonlinear system into one that can be controlled by a single control law - removing the need for gain scheduling.

Incremental nonlinear dynamic inversion (INDI) is a variant of NDI which produces incremental commands, and, additionally, utilizes acceleration feedback (as stated in *Simplício* et al. (2023)). Built upon the principles established by *Bacon* and *Ostroff* (2000) and *Smith* (1998), among others, this method reduces the reliance on precise knowledge of the system's dynamics. The control law developed with

INDI becomes more dependent on measurements, estimations, state derivatives and control input, as demonstrated by *Simplício* et al. (2023). There can be two key downsides with this technique, as discussed in *Simplício* et al. (2023). Firstly, it can have sensitivity to sensor noise and to actuator delay, due to its higher dependency on sensor measurements and control input measurements or estimations. Secondly, it can be challenging to obtain analytical proof of stability (note however, that in the aforementioned work, a way to tackle this problem is proposed). Nonetheless INDI has been successfully applied, in research studies, to the attitude control of LV, see *Simplício* et al. (2023) or *Mooij* (2020). For instance, in the latter work, the efficacy of an INDI controller was assessed alongside a basic PID controller and an adaptive controller. Results indicated that the INDI controller outperformed both counterparts. Moreover, the INDI controller exhibited minimal performance degradation, even when subjected to factors like engine inertia and flexible modes. However, it was noted that the displacement of the IMUs, caused by aeroelastic effects, required careful filtering of certain feedback signals to mitigate potential performance degradation. Finally, it is important to mention that the greatest benefit of INDI is the same as the one for the NDI - it removes the need for gain scheduling.

Sliding mode control is a method that evolved from the work of Emelyanov and Utkin in the 1950s (see *Emelyanov* (1957)and *Kunusch* et al. (2012)). This method works by applying a high frequency discontinuous control signal in order to cause the system to ”slide” across its normal behaviour. The general trend that was identified, is that this technique works very well in systems with uncertain parameters or unmodelled dynamics. However, it has two drawbacks. On one hand, it only allows for the control of output variables whose relative degree is one. On the other hand, high frequency control switching can cause chattering, leading to undesired effects, as explained by *Slotine* and *Li* (1991). Regardless of the downsides, this technique has been employed in the design of LV attitude controllers. In *Y. Shtessel* and *Krupp* (1997), it was possible to achieve robust de-coupled high accuracy tracking of attitude controls.

Higher Order Sliding mode control is a technique that works in a similar principle as SMC, but is no longer constrained by the relative degree limitation or by the chattering effect. This technique has been the area of work of several different researchers, with one of the earliest discussions about it documented in *Levantovsky* (1985) or in *Emelyanov* (1986). The overall trend identified agrees with the previous statements regarding HOSMC. However, a recent study, *Utkin* et al. (2020), showed that chattering may still persist despite previous assumptions of its elimination. This method has been implemented in different research studies, to design the attitude controller of a LV, one example is *Stott* and *Shtessel* (2012), where it was possible to achieve robustness and adequate tracking performance.

Backstepping is a method that operates in two main steps: initially, a controller is designed, based on a Lyapunov function, for a small subsystem; subsequently, the control law is augmented as the subsystem is systematically extended, until the entire system is encompassed. As mentioned in *Kokotović* and *Arcak* (2001), the use of this method as a design tool began with the works of *Sontag* and *Sussmann* (1988), *Byrnes* and *Isidori* (1989), *Tsinias* (1989), amongst some others. In *Vinod* et al. (2012), this approach was implemented to design an attitude controller for the ascent phase of a LV. In the research study, the method was able to achieve desired performance, but no analysis into robustness was performed.

2.4.4. Intelligent Control

Intelligent control methods aim to generate controllers by mimicking biological intelligence (*Passino*, 2001). The only intelligent control method found to be applied for designing an attitude controller for an LV’s atmospheric ascent was reinforcement learning. In *S. Zhou* et al. (2024), the authors employed reinforcement learning to develop an attitude controller for the ascent stage of the Ares-I launch vehicle,

leveraging the Soft Actor-Critic (SAC) algorithm to achieve significantly improved control accuracy compared to traditional methods. Although the authors set out to develop a controller capable of facing uncertainty during the flight, the study did not investigate the impact of model uncertainties on the performance of the designed controller.

2.4.5. Adaptive Control

Adaptive control is a family of control methods distinguished by their ability to adjust their characteristics online, in response to unknown or unmodelled dynamics, nonlinearities, and environmental disturbances (*Orr* & *Zwieten*, 2012). Its advantage lies in the capability it can have in extending flight envelope and adapting against uncertainty. Its key downside lies in the fact that due to the nonlinear features of the adaptive control law, there are currently no formal techniques available that can yield an equivalent analysis to the classical linear stability margins (*Tapia*, 2019). Having given a brief introduction into adaptive control, the following discussion aims at presenting some adaptive control implementations to perform launch vehicle attitude control.

In *Kharisov* et al. (2008), the author implemented the L1 adaptive control strategy to build the launch vehicle TVC control system for the ascent flight. This strategy computes the difference between the output of the plant and the output of the state predictor, and then utilizes this value, together with an adaptation law, to augment the control law. The goal in this research study was to show the implementation of a control law that was capable of handling the flexible body behaviour of the launch vehicle. In the end, the results showed a controller capable of handling flexible behaviour, for the first stage flight, without utilizing notch filters and without re-tuning the controller for different flight points.

In *Orr* and *Zwieten* (2012), for the purpose of LV attitude control, an architecture to perform an adaptive augmentation approach is proposed. After having designed a control law (with the use of a PID), an adaptive control law is designed so as to provide minimal adaptation under nominal conditions, but to extend the flight envelope and increase robustness. Throughout this research, high-fidelity simulations were utilized to assess the proposed architecture, demonstrating that the suggested structure yielded

Figure 2.4: Ares Crew LV - *NASA/MSFC* (2024).

promising results.

In *Tapia* (2019), the adaptive augmentation strategy proposed in *Orr* and *Zwieten* (2012) was utilized, and combined with a joint rigid/flexible controller designed with $\mathcal{H}_\infty$ methods. The findings revealed that the adaptive control law effectively prevented critical failures, such as loss of vehicle, and slightly enhanced performance for extreme off-nominal conditions (beyond those specified in the LV mission).

In *Muse* et al. (2012) an adaptive control system was employed as the TVC control system of NASA's Ares Crew LV (which is illustrated in Figure 2.4). This system utilizes an output feedback neural network adaptive element in combination with a traditional PID controller. In this research study, the approach used was able to maintain performance in nominal conditions, and achieve increased performance in non nominal conditions.

Other additional applications of adaptive control for LV can be found in *Buțu* et al. (2016), *Trotta* et al. (2020), *Wall* et al. (2014), *Costa* et al. (2024), among many others.

2.4.6. Other Works

This subsection aims at discussing research studies into LV attitude control, whose controller synthesis method does not exactly fit into any of the above categories.

In *Muse* and *Calise* (2010), the author combined traditional $\mathcal{H}_\infty$ controllers with an adaptive control law, which was also designed with an $\mathcal{H}_\infty$ minimization process. The goal of this approach, was utilize the adaptive control law, so as to minimize the bounded error between the true system and the ideal system. The research studies showed promising results: for stable cases, the adaptive law increased the performance of the baseline controller; for the unstable cases, the adaptive law was able to recover acceptable tracking for most cases.

In *David* et al. (2008), an approach to derive a self-scheduling controller for the LV attitude control is provided. In this implementation, and in contrary to traditional gain scheduling, the interpolation formula for the controller gain is chosen before synthesizing the controllers for the different flight points. In essence, what happens is that the problem of designing the scheduled controller can be translated into the synthesis of a Multi-model modal controller, with respect to an augmented plant (the augmented plant contains the additional outputs that are utilized in the scheduled control law). Having described the different steps in the approach, the author utilizes it to design the scheduled controller for the atmospheric phase of the launch vehicle, achieving performance objectives for all flight points. The greatest advantage with this method lies in the fact that it can save time during design procedure, in comparison to traditional gain scheduling, as stated by the author.

2.5. Improvements on Classic TVC Design - Flexible Body

This section serves as survey of improvements on the design of bending filters.

In the last years, several strategies have been employed to improve the design of bending filters. *Jang* et al. (2008) utilized numerical constrained optimization to design the bending filter (BF) for Ares-1, with the goal of maximizing stability margins while meeting performance requirements. In the research study, *Jang* et al. (2008) was to achieve stable first and second stage control systems, with minimal stability and performance degradation caused by the BF. *Choi* and *Kim* (2001) employed an adaptive algorithm to identify the actual vibration frequency associated with the bending mode. Subsequently, this determined frequency was integrated into a notch filter, resulting in the stabilization of a sounding rocket that had

previously exhibited instability. *Elmelhi* (2014) utilized a radial basis function neural network, replacing the adaptive algorithm, to determine the BM frequencies and build the BF. In the research study, the stabilization of a LV's flexible modes was achieved.

All these strategies focused on designing the bending filter separately, however, it is also possible to design the bending filter and rigid body controller simultaneously, as demonstrated by *Tapia* (2019). In his work, the bending filter is first parameterized. Subsequently, the rigid body controller and bending filters are designed simultaneously using structured $\mathcal{H}_\infty$ CLS, resulting in reduced design time, whilst achieving improved performance and robustness, when compared to VEGA's original controller. It might be possible to produce a methodology similar to the simultaneous rigid/flexible controller design with $\mathcal{H}_\infty$ CLS, but using $\mathcal{H}_\infty$ OLS. This would be particularly interesting if $\mathcal{H}_\infty$ OLS were shown to achieve the same results as $\mathcal{H}_\infty$ CLS for the rigid body design, while reducing design time/complexity.

2.6. Discussion

Having reviewed several different research studies, it is now important to discuss the obtained information, so as to progress towards the identification of the research gap and questions. This section will contain said discussion.

The classical approach to Launch Vehicle rigid body attitude control faces two main challenges: inefficiency in time usage and the inability to effectively address system uncertainties explicitly. All the reviewed strategies aim, in one way or another, to tackle either one or both of these issues.

Out of all intelligent control methods, reinforcement learning was the only one found to be used for designing the attitude controller of an LV. While the reviewed study demonstrated promising results, it lacked a more thorough analysis, such as one that considered uncertainties in the LV's model.

The methods pertaining nonlinear control have shown very interesting results. Moreover, they are in principle time-efficient methods, as they offer flexibility in addressing the complexities of real-world systems and often require fewer tuning iterations, when compared to the classic approach. For instance, techniques like feedback linearization and sliding mode control can effectively handle nonlinearities, reducing the need for extensive manual tuning. Of the five methods reviewed, SMC and HOSMC stand out, due to their capability to address uncertain systems through direct incorporation of uncertainty. In contrast, NDI can show limitations in this type of system, due its reliance on knowledge of the actual system. INDI fixed the requirement for system knowledge, but this improvement can come as a double edged sword: while it removes the requirement for precise system knowledge, it can also increase sensitivity to sensor noise and actuator delay. Finally, backstepping presents itself as an interesting method due to its systematic nature, which could potentially by used so as to decrease design time. Taking everything into account, SMC, HOSMC, backstepping and INDI could prove as very useful tools in designing attitude controllers for LVs. However, they all share in one common issue: the corresponding non-linear robustness analysis tools are far less developed than linear robustness analysis methods (*Bates* & *Postlethwaite*, 2002).

The methods discussed in the adaptive control subsection also showed very interesting results. The possibility of enlarging the LV flight envelope or of preventing critical failures is very enticing. Adaptive methods can easily be combined with other methods (for instance $\mathcal{H}_\infty$ methods) to prevent critical failures and enhance performance outside the original flight envelope. The primary drawback associated with these approaches stems from the nonlinear characteristics exhibited by the adaptive control law. Currently, there are no formal techniques available that can provide an equivalent analysis to the classical linear stability margins for addressing the nonlinear features present in adaptive control laws (*Tapia*, 2019).

In subsection 2.4.6, works who did not fit into other subsections were discussed. Of the approaches described the last was of particular interest. In this approach, an implementation that is capable of addressing highly time varying systems was described, and implemented for a LV, achieving promising results.

Of the surveyed $\mathcal{H}_\infty$ methods, $\mathcal{H}_\infty$ CLS showed the widest implementation. This method is not only time efficient but allows for the consideration of system uncertainties during the design, making for a very powerful tool. Nonetheless, this method can have some limitations, as previously discussed. $\mathcal{H}_\infty$ Open Loop Shaping goes one step beyond $\mathcal{H}_\infty$ CLS, and addresses the aforementioned limitations through shaping the open loop directly and through robustifying the system against NCF. This enhancement makes $\mathcal{H}_\infty$ Open Loop Shaping an even more powerful tool, leading to the expectation that it can attain similar results to $\mathcal{H}_\infty$ CLS (as both method's shape the singular values of the system), while reducing design time/complexity. In fact, it might even be possible that due to the added complexity of $\mathcal{H}_\infty$ CLS, $\mathcal{H}_\infty$ OLS produces better controllers in practice. It is important to mention that both methods, due to being linear, frequency domain methods, allow for the use of classic control tools (Bode plots, Nyquist plots, Root-locus, etc...), which offer a high level of design transparency and intuition. Moreover, both methods guarantee stability margins (either through direct or indirect means). Finally, the third method surveyed, $\mu-$synthesis, would in principle allow for the synthesis of a less conservative controller (when compared to the other $\mathcal{H}_\infty$ methods discussed), while still allowing for the use of linear control analysis tools. However, the acceptance towards this method is not yet widespread in the aerospace industry (*Tapia*, 2019).

The classical approach to Launch Vehicle flexible body control faces three primary challenges: bending filter design can be very complex, as the designer cannot allow the BF to degrade rigid body controller stability and performance; flexible body behavior is hard to model due to great uncertainty in flexible behavior; bending filter design and integration, utilizing the classic approach, is quite time consuming. The approach taken by *Tapia* (2019), where the bending filter and rigid body controller were designed simultaneously might be of particular interest for this project, as it resulted in a simpler, faster approach to design the TVC control system, effectively tackling the first and third issues. If it were demonstrated that $\mathcal{H}_\infty$ OLS attains the same results as/outperforms $\mathcal{H}_\infty$ CLS, while reducing design time/complexity, it would be worthwhile to develop a new integrated rigid/flexible body controller design strategy based on $\mathcal{H}_\infty$ OLS. This approach would not only reduce overall design time compared to traditional methods but also leverage a technique that is simpler to use than $\mathcal{H}_\infty$ CLS.

2.7. Summary

During this chapter, an evaluation of the state of the art in atmospheric ascent TVC control systems of launch vehicle was performed. Firstly, in section 2.1, a brief history of launch vehicles was given, so as to give context to the launch vehicle thematic. Then, in section 2.2, it was explained how the classic TVC design is performed, so as to establish the launch vehicle control system baseline. The next step was to discuss the issues that are inherent to the classic TVC design (section 2.3). Subsequently, the research studies that have been conducted into fixing the aforementioned issues were discussed. Firstly, in section 2.4, a discussion into the state of art of control techniques for the rigid body was held. Secondly, in section 2.5, the latest improvements in bending filter design were discussed. Following the discussion on the state of the art for LV TVC control systems, comments regarding the learned information were given in section 2.6.

Taking everything into account, it is now possible to identify the research gap and questions. From the discussion, it is clear that there is a knowledge gap when it comes to the comparison of $\mathcal{H}_\infty$ Open Loop Shaping to $\mathcal{H}_\infty$ Closed Loop Shaping, for the particular case of attitude control during the atmospheric ascent phase of a LV. Additionally, the possibility of doing an integrated rigid body controller/bending filter design, with the use of $\mathcal{H}_\infty$ OLS, also seems remarkably interesting. In the next chapter, the research gap will be discussed, and the research questions will be formulated in a SMART (specific, measurable, achievable, realistic, timely) manner.

3

Research Gap, Objectives, Questions and Project Planning

3.1. Research Gap

In the preceding chapter, an extensive review of literature was performed, allowing for the identification of various research opportunities. Note that every idea presented in the following section is related to the **atmospheric ascent** of the launch vehicle.

The first identified research gap arises from the absence of a comparative study between $\mathcal{H}_\infty$ Open Loop Shaping and $\mathcal{H}_\infty$ Closed Loop Shaping, in the design of launch vehicle attitude controllers. As previously discussed, $\mathcal{H}_\infty$ OLS presents itself as a powerful controller design methodology, that addresses the limitations of $\mathcal{H}_\infty$ CLS by directly shaping the open loop and robustifying the system against Normalized Coprime Factor (NCF) uncertainty (recall Subsection 2.4.2).

It is expected that controllers designed using $\mathcal{H}_\infty$ Open Loop Shaping can achieve the same stability, robustness, and performance as those designed with $\mathcal{H}_\infty$ CLS, since both methods shape the system's singular values. However, it is expected that $\mathcal{H}_\infty$ OLS can reduce design time and complexity due to its advantages over $\mathcal{H}_\infty$ CLS. It might even be possible that in practice, controllers produced with $\mathcal{H}_\infty$ OLS outperform controllers produced with $\mathcal{H}_\infty$ CLS due to the latter method inherent design complexity. Therefore, the first objective of this thesis is to evaluate the capability of $\mathcal{H}_\infty$ Open Loop Shaping in designing attitude controllers for launch vehicles. Specifically, the aim is to determine whether this method can produce controllers with equal or superior robustness, stability, and performance compared to those designed using $\mathcal{H}_\infty$ Closed Loop Shaping, while also reducing design time and complexity. Through this comparison, the goal is to assess whether $\mathcal{H}_\infty$ Open Loop Shaping represents a meaningful advancement over the $\mathcal{H}_\infty$ CLS approach.

The second identified research gap focuses on the integrated design of the rigid body controller and the bending filter. Following the approach demonstrated by *Tapia* (2019), which used $\mathcal{H}_\infty$ CLS to simultaneously optimize bending filters and rigid body controllers, this study will explore whether the same can be achieved with $\mathcal{H}_\infty$ OLS. If $\mathcal{H}_\infty$ OLS is shown to produce rigid body controllers that match or outperform those designed with $\mathcal{H}_\infty$ CLS in terms of stability, robustness, and performance, while also reducing design time and complexity, it would be advantageous to develop a new integrated rigid/flexible body controller design strategy using $\mathcal{H}_\infty$ OLS. This strategy could not only shorten overall design time compared to traditional methods but also leverage a technique that is easier to use than $\mathcal{H}_\infty$ CLS.

Therefore, the second objective of this thesis is to develop and evaluate a simultaneous rigid/flexible body controller design methodology using $\mathcal{H}_\infty$ OLS. The goal is to determine whether this approach can reduce design time while maintaining the performance, stability, and robustness requirements of the GNC system, when compared to the traditional separate design approach.

3.2. Research Objectives

In the previous chapter, the research gap was identified, it is now possible to define the research objective and questions. Once more, while not explicitly mentioned, all objectives and research questions pertain to the atmospheric ascent phase of the launch vehicle. The first research objective is as follows:

Research Objective

To evaluate the capability of $\mathcal{H}_\infty$ Open Loop Shaping in producing launch vehicle attitude controllers with equal or superior robustness, stability, and performance compared to those designed using $\mathcal{H}_\infty$ Closed Loop Shaping, while reducing design time and complexity.

The second research objective is as follows:

Research Objective

To investigate the potential of utilizing an integrated rigid body controller and bending filter design strategy based on $\mathcal{H}_\infty$ Open Loop Shaping, with the goal of reducing design time while maintaining robustness, stability, and performance compared to the separate design approach.

With these two research objectives in mind, it is now possible to define the research questions.

3.3. Research Questions

Keeping the research objectives in mind, the first research question can be defined:

Research Question 1

Can $\mathcal{H}_\infty$ Open Loop Shaping be used to generate attitude controllers for launch vehicles that exhibit equal or superior robustness, stability and performance compared to controllers derived using $\mathcal{H}_\infty$ Closed Loop Shaping, while reducing design time/complexity?

This research question can be further subdivided into sub-questions:

Research Question 1.1

How can $\mathcal{H}_\infty$ Open Loop Shaping be applied to design an attitude controller for a launch vehicle? Specifically, what form should the weighting filters take to properly shape the plant?

Research Question 1.2

Can the $\mathcal{H}_\infty$ OLS design methodology reduce overall design time and complexity compared to the $\mathcal{H}_\infty$ CLS design methodology?

Research Question 1.3

Can the attitude controller designed using $\mathcal{H}_\infty$ Open Loop Shaping achieve comparable or superior results in terms of uncertainty handling, disturbance rejection, stability margins, tracking accuracy, drift control, load relief, and actuator usage, compared to the controller produced using $\mathcal{H}_\infty$ Closed Loop Shaping?

Finally, the second research question can be defined:

Research Question 2

Can the integrated attitude controller/bending filter, designed with $\mathcal{H}_\infty$ OLS, achieve similar results to the controller designed with the traditional separate approach, whilst improving design time?

This research question can be further subdivided into sub-questions:

Research Question 2.1

Does the integrated controller demonstrate comparable results in terms of handling uncertainties, disturbance rejection, stability margins, tracking accuracy, drift minimization, load relief, and actuator usage, when compared to the separate design approach?

Research Question 2.2

Can the integrated design methodology reduce the overall design time compared to the traditional separate design process?

3.4. Methodology

In order to address the research questions, a methodology will be outlined. The initial phase of this thesis will involve a comprehensive study into to two key subjects pertaining this project: Robust Control and Launch Vehicle Modeling. This research will culminate into two theoretical chapters.

The chapter on robust control will provide an overview of its history and essential fundamentals. The key concepts to be discussed include $\mathcal{H}_\infty$ Open Loop Shaping and Closed Loop Shaping, along with their foundational principles such as singular values, $\mathcal{H}_\infty$ norm, and linear fractional transformations. Additionally, robustness measures and uncertainty representations will be discussed. The investigation into robust control concepts is expected to continue until the end of the thesis, due to the fact that it is quite difficult to accurately predict the specific knowledge that will be needed throughout the duration of the research.

In the LV model chapter, various aspects of launch vehicle modeling will be examined. This includes discussions on two types of LV models: design and analysis. Additionally, considerations such as reference frames, assumptions used to derive the LV models, forces and moments, actuator models, model parameters and dispersed conditions definition will be addressed. This chapter will contain the derivation of rigid body design and analysis models from scratch. For the rigid/flexible body design and analysis models, only the final set of equations will be presented, due to time constraints. One final remark

regarding the launch vehicle models that will be used is made. Through the use of common assumptions in launch vehicle control (described for instance in *Díaz* (2010) or *Greensite* (1967)), it can be shown that, for the purpose of attitude design/analysis models, yaw and pitch plane motion can be treated as decoupled and identical, and that the longitudinal dynamics can be disregarded (*Bernard*, 2009), without loss of relevance. And so, following previous works (such as *Tapia* (2019) or *Díaz* (2010)), the design and analysis models used for the LV will restrict themselves to the pitch plane. This simplification will enable the incorporation of launch vehicle-specific effects, including engine inertia and flexible body behavior, which would otherwise have to be discarded due to the thesis time constraints.

With the theoretical foundations established, the subsequent objective is to address the first research question. For this purpose, a specific flight point will be selected for analysis, and a set of requirements for the TVC control system will be established. Next, controller design and analysis models, considering only rigid body behavior, will be constructed and implemented in SIMULINK.

Once both models are implemented, an initial analysis of system dynamics will be conducted so as to get a better understanding on the challenges of the dynamical system. The subsequent phase will entail the synthesization of a pole placement attitude controller, to serve as baseline. Subsequently, the $\mathcal{H}_\infty$ CLS controller synthesis will begin.

In this synthesis phase, various combinations of constraints on the CLTFs will be explored to identify the combination that best meets the desired goals and specifications for the LV attitude control system. Once the optimal combination is determined, iterative optimization of the constraint values will be carried out to enhance the controller's robustness, stability, and performance.

The next step will be to design the $\mathcal{H}_\infty$ Open Loop Shaping controller, with the goal of answering Research Question 1.1 and 1.2. This process will involve shaping the open loop singular values to achieve the desired control objectives. An iterative optimization will then be performed on the weighting filters used to shape these singular values, optimizing the controller's robustness, stability, and performance.

Subsequently, time and frequency domain analyses will be conducted, using the aforementioned controller analysis model, to address Research Question 1.3. These analyses will provide insights into the performance, stability, and robustness of the two $\mathcal{H}_\infty$ controllers and also provide a comparison with the pole placement controller. With this analysis it will be possible to finally answer Research Question 1.

Having answered the first research question, the focus will transition to the design of an integrated rigid/flexible body controller. The first step will be to implement controller design and analysis models, which will consider both rigid and flexible behavior. These models will contain a bending filter, with a parametrization that will later be described. This will allow for simultaneous optimization of the bending filter and the rigid body controllers. Once the models are implemented, the separate design method will be used to design a rigid body controller and bending filter, where the rigid body controller will be designed with $\mathcal{H}_\infty$ OLS. Subsequently, the integrated design strategy will be described and implemented to design the integrated rigid/flexible controller.

Afterwards, robustness, stability and performance analysis will be conducted using the analysis model that considers rigid and flexible body behavior. The results of this analysis will allow to answer sub-Research Questions 2.1 and 2.2 and Research Question 2.

One final remark is made: the influence of the navigation system (sensor noise, quantization, errors and delays from navigation filters) will be disregarded to simplify the analysis but can and should be addressed in future work.

3.5. Expected Results

The following section aims at forming and justifying hypothesis regarding the answer of the research questions.

Research Question 1.1

This question cannot be answered prior to actual experimentation with the process of $\mathcal{H}_\infty$ OLS. At most, according to the guidelines given in *Papageorgiou* (1998), it can be expected that integral action is applied in the pre-compensator, whilst lead-lag and low-pass filters are used in the post-compensator.

Research Question 1.2

It is expected that the design process of $\mathcal{H}_\infty$ OLS reduces design time and complexity. The reasoning being that $\mathcal{H}_\infty$ OLS directly shapes the open loop and robustifies the system against NCF uncertainty. The former simultaneously shapes the different closed loop transfer functions, thus preventing the designer from having to reconcile conflicting closed loop specifications, while the latter eliminates the need to manipulate multiple transfer functions to ensure robustness at both the plant input and output.

Research Question 1.3

It is anticipated that the rigid body controller designed using $\mathcal{H}_\infty$ Open Loop Shaping will perform on par with the $\mathcal{H}_\infty$ Closed Loop Shaping controller in terms of handling uncertainty, increasing stability margins, enhancing disturbance rejection, improving tracking accuracy, minimizing drift, reducing load, and minimizing actuator usage. This expectation arises from the fact that both $\mathcal{H}_\infty$ OLS and $\mathcal{H}_\infty$ CLS manipulate the system's singular values, meaning they should theoretically achieve similar results.

It is important to mention that the added complexity of the $\mathcal{H}_\infty$ CLS approach may affect the designer's ability to achieve as effective a controller as with $\mathcal{H}_\infty$ OLS. Meaning that the produced $\mathcal{H}_\infty$ OLS might be superior in the parameters discussed in RQ 1.3. Only through implementation and direct comparison can RQ 1.3 be answered.

Research Question 1

Taking everything into account, it is expected that the controller produced with the $\mathcal{H}_\infty$ Open Loop Shaping method will attain equal robustness, performance and stability as the controller produced $\mathcal{H}_\infty$ CLS method. It is also expected that the $\mathcal{H}_\infty$ OLS approach can reduce design time and complexity. However, as discussed in the expected result for RQ 1.3, it is also possible that $\mathcal{H}_\infty$ OLS proves to be superior. Only through actual implementation will a definitive assessment be possible.

Research Question 2.1

It is expected that the integrated controller will provide comparable results across all the parameters outlined in Research Question 2.1, when compared to the controller designed using the separate approach. The rationale is that if both methods stem from the same technique ($\mathcal{H}_\infty$ OLS) and aim to produce an optimal controller, they should theoretically converge to the same design, yielding equivalent results.

Research Question 2.2

Following the results of *Tapia* (2019), it is expected that the integrated design approach can reduce design time.

Research Question 2

With everything in mind, it is expected that integrated rigid/flexible body controller design strategy will result in reduced design time, whilst maintaining performance, robustness and stability, when compared to the separate design approach.

3.6. Planning

Finally, the methodology's time frame is illustrated through a Gantt chart in Figure 3.1. As it can be seen, all tasks and milestones are scheduled. It is important to mention that there are 7 weeks of holidays

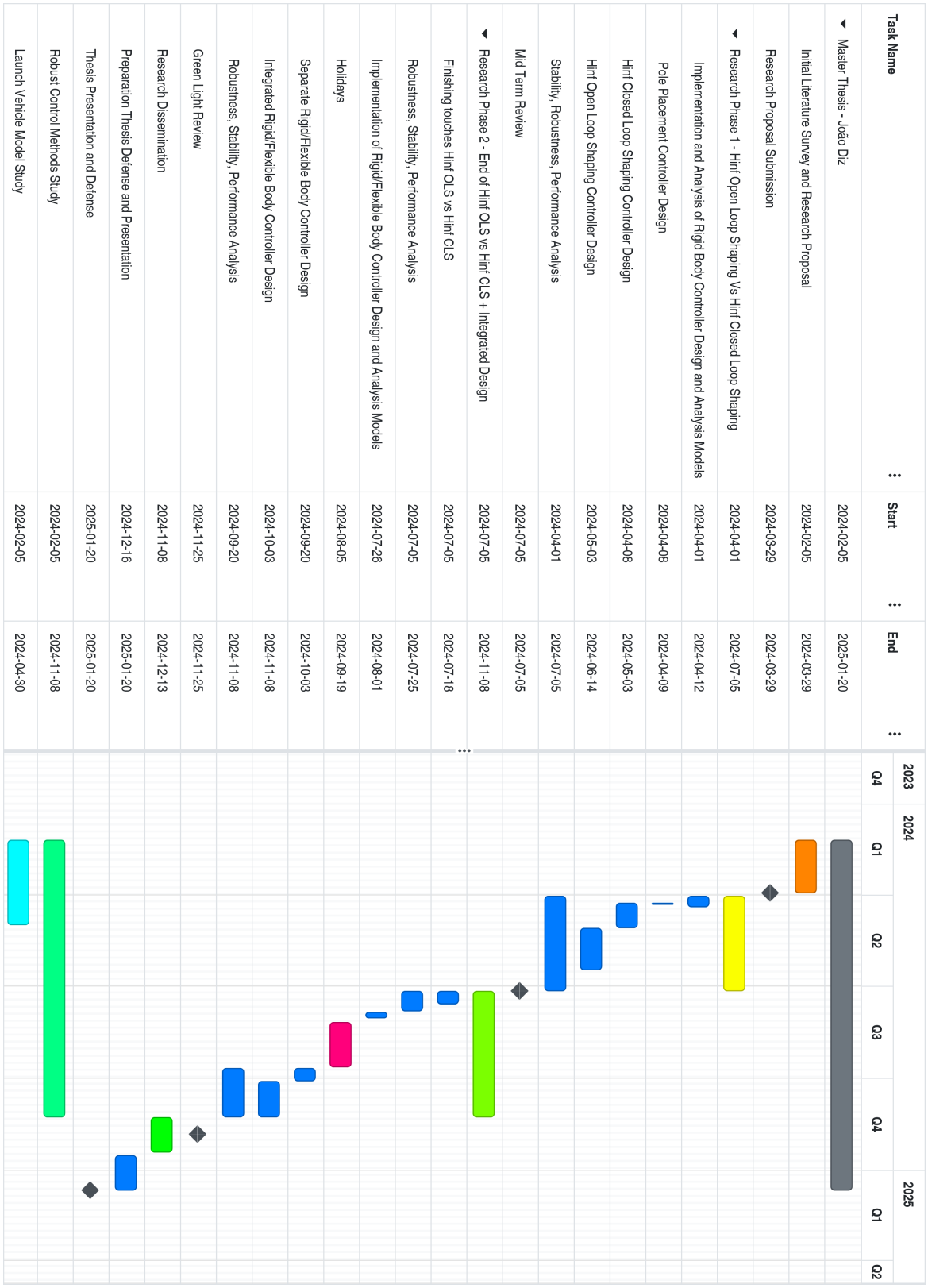

Figure 3.1: Gantt Chart - Planning of the Master Thesis.

scheduled during research phase 2, but are only there to serve as placeholder - it is not known when these will be used, but most likely they will be distributed through the different phases so as to not interfere with work. Additionally, some days were reserved in the beginning of research phase 2 to tackle any issues remaining in the $\mathcal{H}_\infty$ OLS and $\mathcal{H}_\infty$ CLS controller implementations.

4

Fundamentals of Robust Control

The aim of this chapter is to perform a literature review into $\mathcal{H}_\infty$ CLS and $\mathcal{H}_\infty$ OLS, as well as other robust control fundamentals that will be required for this thesis. And so, the structure of the chapter is stated next.

Section 4.1 gives a broad view on the history and context of robust control. Section 4.2 states the general structure used to represent systems in robust control. Section 4.3 briefly describes linear fractional transformations (mathematical method used to compute the general robust control structure). Next, singular values are described in Section 4.4. Subsequently, $\mathcal{H}_\infty$ system norm and the small gain theorem will be discussed in Sections 4.5 and 4.6, respectively. These two concepts are relevant for the formulation of the $\mathcal{H}_\infty$ controller design methods and for the formulation of robustness and performance measures. With all the basic concepts introduced, the discussion will move on to the various ways uncertainty can be represented in Section 4.7, followed by a description of balanced disk margins as a measure for robust stability in Section 4.8. Finally, the two $\mathcal{H}_\infty$ controller design methods that will be used in this thesis, will be discussed in Section 4.9 and 4.10, respectively.

4.1. History and Developments of Robust Control

Respect The Unstable. A sentence attributed to *Stein* (2003), often seen in control papers. Indeed when the unstable was not respected, such as in the cases of the Trident submarine (see the comments by Michael Attans in page 40 of *Sandell* (1979)), or the SAAB JAS-39, unwanted behaviour or accidents occurred. A lack of attention to robustness was identified as being the cause of these issues, and this lead to a change in paradigm, culminating into the birth of robust control theory. The ensuing discussion aims at telling a brief history of the developments of robust control.

The developments in the field of robust control began in the 70s. During this decade, many developments were achieved, for instance, *J. C. Doyle* (1978) introduced the singular value as a solution for the issues associated with traditional stability margins analysis in MIMO systems. At the same time *Wong* and *Athans* (1975) introduced much of the foundation of multivariable stability margins. During that year another publication further motivated the development of robust control: *J. C. Doyle* (1978) showed that there are no guaranteed margins for LQG controllers at input and output of the plant.

Eventually, in *Safonov* (1977) and in *Safonov* (1980), some fundamental robustness concepts were outlined, namely: diagonal uncertainties, topological separation ability and multivariable robustness analysis (bode plots of real eigenvalues, associated with the separation condition, were to be used to analyse robustness). The latter concept was later rewritten in terms of singular values bode plots and became a

metric that is of the utmost importance till today.

The next step in the robust control revolution was the introduction of concepts such as the excess stability margin, k_m (introduced by *Safonov* and *Athans* (1981)), and the structured singular value, μ (introduced by *J. Doyle* et al. (1982)), which paved the way for the μ-synthesis (a robust method to synthesise controllers).

Subsequently, and very importantly, *Zames* (1981) formally introduced the $\mathcal{H}_\infty$ minimization problem, and solved it for a simple SISO system. In this application, Zames utilized $\mathcal{H}_\infty$ minimization to shape a closed loop transfer function of a SISO system. This innovative approach, eventually gained recognition as the $\mathcal{H}_\infty$ mixed sensitivity method. Additionally, Zames's work paved the way for *J. C. Doyle* (1984) and *Glover* (1984), who reached the first state-space solution for the $\mathcal{H}_\infty$ minimization problem. However, this solution was computationally heavy, and so, in the works of *Glover* and *Doyle* (1988), *J. C. Doyle* et al. (1989), and *Glover* et al. (1991), a new solution which only involved solving two Riccati equations was developed. The reached solution could be used to derive full order controllers.

At around the same time as the $\mathcal{H}_\infty$ problem was solved, and based on the early works of Vidyagasar (such as *Vidyasagar* and *Kimura* (1986)), *McFarlane* et al. (1988) detailed a new method called $\mathcal{H}_\infty$ Open Loop Shaping. This method was further refined in *McFarlane* and *Glover* (1988) and in *McFarlane* (1989). Finally, in *McFarlane* and *Glover* (1992) a systematic framework and a practical application of the method were given. This method, which focused on shaping the open loop transfer function instead of the closed loop transfer functions, was shown to solve the issues associated with $\mathcal{H}_\infty$ Closed Loop Shaping (for an example of such issues see *Sefton* (1991)). However, $\mathcal{H}_\infty$ OLS was not capable of enforcing reference model following (which $\mathcal{H}_\infty$ CLS is). Later in *Limebeer* et al. (1993), a method called 2 Degree of Freedom $\mathcal{H}_\infty$ Open Loop Shaping, was introduced, effectively solving $\mathcal{H}_\infty$ Open Loop Shaping's limitation.

In the following year, a new solution to the $\mathcal{H}_\infty$ minimization problem was given in *Gahinet* and *Apkarian* (1994). This new solution, utilized algebraic manipulation of linear matrix inequalities to solve the minimization problem, and the authors hinted at the possibility of using this solution to produce fixed-order controllers.

Many years later, in *Apkarian* and *Noll* (2006), the $\mathcal{H}_\infty$ problem was solved with the use of non-smooth, non-convex optimization, effectively achieving a solution capable of producing fixed-order structured controllers.

Finally, three more milestones are worth mentioning. The first, is the proof of the small gain theorem by *Zames* (1966) and then its reinterpretation by *J. Doyle* et al. (1982), which allowed for the formulation of a performance robustness theorem (discussed later on). The second is the introduction of MATLAB's robust control toolbox by *Chiang* and *Safonov* (1988). The third and final one, is the writing of robust control books, namely *Bates* and *Postlethwaite* (2002) and *Skogestad* and *Postlethwaite* (2005), which allowed for the gathering and synthesization of much of the knowledge regarding robust control. The reader should note that while earlier books on robust control exist, these particular references were chosen because they were extensively utilized in the development of this thesis.

4.2. General Robust Control Design Configuration

The first step in designing a robust controller lies in representing the dynamic system in a way that captures uncertainty and facilitates robust controller design. One way to achieve this is by representing the system as a set of interconnected blocks, which facilitates the incorporation of various configurations arising from changes in the controller's form and location, as well as system uncertainties. A general configuration for

this purpose was first introduced by *J. C. Doyle* (1983), and is illustrated in Figure 4.1.

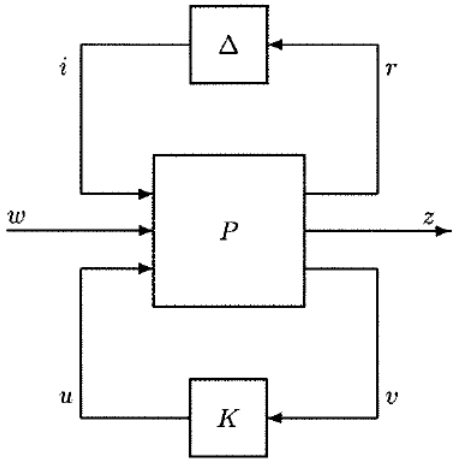

Figure 4.1: General configuration for robust controller design, figure retrieved from *Bates* and *Postlethwaite* (2002).

In this representation, P represents the generalized plant, and is composed of the nominal plant, the disturbance model, the interconnection structure between plant and controller, and weighting functions - in the case it is being used to formulate a design problem. K represents the generalized controller, while Δ represents model uncertainty. w is named the vector of exogenous input - it includes external disturbances, commands inputs and measurement noise. z represents the set of controlled outputs, including tracking errors, system variables, and any other elements necessary to define the specifications of the control problem. Moreover, v represents the vector of sensed outputs (outputs available for feedback), while u represents the controllers output to the plant. Note that if model uncertainty is not considered during the design, the Δ block can be removed.

4.3. Linear Fractional Transformations

Linear Fractional Transformations, hereafter referred as LFT, are useful tools to represent multiple systems that are connected by feedback paths. Consider the system representation shown in Figure 4.2, where the block M contains the matrices P and K (the known parts of the system).

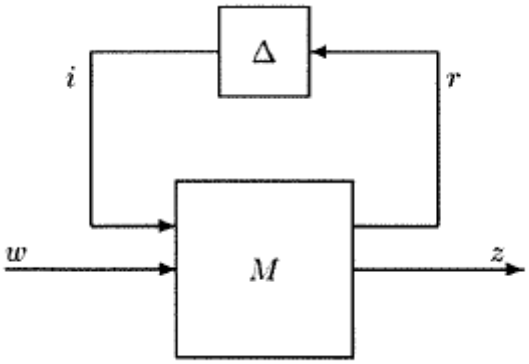

Figure 4.2: General configuration for robust controller analysis, taken from *Bates* and *Postlethwaite* (2002).

Using the algebraic relations:

$$\begin{aligned} z &= M_{11}w + M_{12}u \\ y &= M_{21}w + M_{22}u \\ i &= \Delta r \end{aligned} \tag{4.1}$$

The closed loop transfer matrix, from w to z, can be written as:

$$\begin{aligned} z &= \left\{ M_{22} + M_{21}\Delta \left(I - M_{11}\Delta\right)^{-1} M_{12} \right\} w \\ &= F_u(M, \Delta)w \end{aligned} \tag{4.2}$$

where $F_u(M, \Delta)$ is the upper LFT of the system represented in figure 4.2. This transformation will be used later on to formulate the controller synthesis theorems used in this thesis. It is important to note that the lower LFT can be computed in an analogue manner, and so a detailed explanation on its derivation will not be given.

4.4. Singular Values

Measuring the size of scalar LTI systems is relatively simple when they are SISO (single-input-single-output): the magnitude of the frequency response of a SISO system enables the possibility of knowing whether an input signal will be attenuated or magnified. For MIMO systems (multiple-input-multiple-output), regular bode plots no longer suffice, as the direction of the input signal will also affect the size of the output signal. A solution for this issue is to use singular values (SV). This mathematical tool was first introduced in 20th century (*Stewart*, 1993), but it only gained relevance in the field of control when *J. C. Doyle* (1978) introduced it as a solution for the issues with traditional stability margin analysis in MIMO systems. In essence, singular values allow its user to generalize the concept of ”gain” from a SISO system to a MIMO system.

Before delving more thoroughly into a geometric explanation of the SV concept, it is important to consider how these values can be computed. Consider a constant complex $m \times n$ matrix called G. This matrix can be decomposed into its singular value decomposition:

$$G = U\Sigma V^H \tag{4.3}$$

where U is an $m \times m$ unitary matrix ($U^h = U^{-1}$), whose columns are the output singular vectors. Similarly V is an $n \times n$ unitary matrix, whose columns are the input singular vectors. The vectors of U are orthonormal amongst themselves, and the same can be said the regarding the vectors of V. The $m \times n$ matrix Σ contains a diagonal matrix Σ_1 with the real, non-negative singular values (σ_i) of G arranged in descending order (from $\Sigma_1(1,1)$ to $\Sigma_1(k,k)$):

$$\Sigma = \begin{bmatrix} \Sigma_1 \\ 0 \end{bmatrix}; \quad m > n \tag{4.4}$$

or

$$\Sigma = \begin{bmatrix} \Sigma_1 & 0 \end{bmatrix}; \quad m < n \tag{4.5}$$

or

$$\Sigma = \Sigma_1; \quad m = n \tag{4.6}$$

The singular values are the positive square roots of the eigenvalues of the matrix $G^H G$:

$$\sigma_i(G) = \sqrt{\lambda_i \left(G^H G\right)} \tag{4.7}$$

where G^H is the complex conjugate transpose of G.

One way to look at this decomposition is the following: if G is a nonsingular square matrix, then U, V and Σ can be considered a linear transformation. Σ is a diagonal square matrix, therefore it will only amplify or shrink each entry of the input vector. Meanwhile, V and U are matrices that will either rotate or reflect the input vector. This way, the SVD can be considered a way of decomposing a linear transformation into three operations: rotation or reflection followed by contraction or expansion followed by rotation or reflection.

Another way to interpret the SVD is through a more concrete example. First, consider a unit circle, if a linear transformation G (composed of rotation and expansion) is applied to this circle, an ellipsoid will be formed. Then, the singular values of G will give the lengths of the semi-axes of the ellipsoid. The vectors that form U give the mutually orthogonal directions of the ellipsoid major and minor axis. Meanwhile, the input singular vectors that form V, are mapped into the output singular vectors that form U, with a scaling equal to the corresponding singular value. What is important to note here, is that SVD not only gives the directions of most and least expansion (or contraction), but it also gives the scaling in these directions. In this way, if SVD is applied to a system matrix, it will be possible to predict if a signal is attenuated or magnified.

4.5. $\mathcal{H}_\infty$ System Norm

The next step in measuring the size of an LTI system, whose initial conditions are zero, is to consider the $\mathcal{H}_\infty$ system norm. This norm is defined as the maximum value of the largest singular value of the system's frequency response across all frequencies:

$$\|G(s)\|_\infty = \max_\omega \overline{\sigma}(G(j\omega)) \tag{4.8}$$

Thus minimizing the $\mathcal{H}_\infty$ norm of the aforementioned LTI system corresponds to pressing down the peak singular value of the system's frequency response. This norm also has several time domain interpretations, one example is given next. For an LTI system with input w, output z, and transfer function matrix $G(s)$ it can be shown that:

$$\|G(s)\|_\infty = \sup_{w \neq 0} \frac{\|z(t)\|_2}{\|w(t)\|_2} \tag{4.9}$$

where $||.||_2$ represents the H_2 norm.

Therefore, minimizing the $\mathcal{H}_\infty$ norm of the system, corresponds to minimizing the the energy of the worst-case output signal vector. For other times domain interpretations see pages 165 and 166 of *Skogestad* and *Postlethwaite* (2005).

4.6. Small Gain Theorem

The Small Gain Theorem, proved by *Zames* (1966), serves as basis for several robustness tests and will be referenced numerous times throughout this thesis, it is therefore relevant to state it. Consider the system portrayed in Figure 4.3.

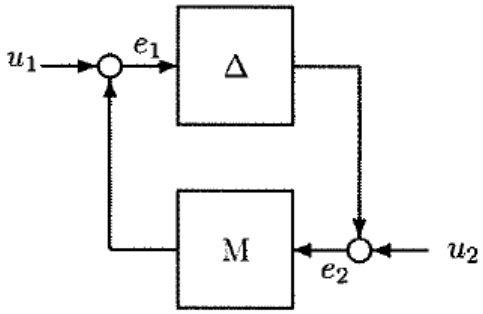

Figure 4.3: Block diagram interconnection structure, retrieved from *Bates* and *Postlethwaite* (2002).

Supposing the system shown in Figure 4.3 has a stable open loop transfer function matrix $L(s) = M\Delta(s)$, with no hidden unstable poles arising from right hand plane pole-zero cancellations, then the closed loop system is table if:

$$\|L(j\omega)\| < 1 \quad \forall\omega \tag{4.10}$$

where $\|L\|$ stands for any induced norm that satisfies $\|AB\| \leq \|A\| \cdot \|B\|$.

4.7. Uncertainty Models

In this section, an explanation about several types of uncertainty models will be given. This is of particular relevance, as both of the design methods described in a later section, provide robustness against different types of unstructured uncertainty. Conversely, structured uncertainty representations are useful for the analysis of the produced controllers.

4.7.1. Unstructured Uncertainty

Unstructured uncertainty models offer general approaches for representing the ”full” complex Δ matrix, making them particularly useful when detailed information about the uncertainty is unavailable. As highlighted by *Bates* and *Postlethwaite* (2002), the only constraint is $||\Delta||_\infty \leq r$, where r is a defined constant.

There are various unstructured uncertainty models discussed in the literature. *Skogestad* and *Postlethwaite* (2005) addresses additive uncertainty, inverse additive uncertainty, input and output multiplicative uncertainty, input and output inverse multiplicative uncertainty, and coprime factor uncertainty. Meanwhile, *Bates* and *Postlethwaite* (2002) and *K. Zhou* and *Doyle* (1998) covers only the non-inverse models. The following discussion describes additive uncertainty, input and output multiplicative uncertainty, and coprime factor uncertainty models. For a mathematical treatment of the inverse models (also called feedback models), refer to pages 313 and 314 of *Skogestad* and *Postlethwaite* (2005).

Additive and Multiplicative Uncertainty

Let $P(s)$ correspond to the transfer matrix of the true plant, and $P_0(s)$ correspond to the nominal plant. Then, the additive uncertainty, input multiplicative uncertainty, and output multiplicative uncertainty models can be mathematically described as follows:

$$\begin{aligned} P(s) &= P_0(s) + \Delta_A(s) \\ P(s) &= P_0(s)\,(I + \Delta_I(s)) \\ P(s) &= (I + \Delta_O(s))\,P_0(s) \end{aligned} \tag{4.11}$$

where, Δ_A represents an additive uncertainty, Δ_i represents an input multiplicative uncertainty, and Δ_o

represents an output multiplicative uncertainty - each Δ is unknown but stable with bounded $\mathcal{H}_\infty$ norm. It is important to note that a weight matrix W can be used to prioritize particular uncertainties in the uncertainty matrix. In order to do so, the Δ block is usually normalized so that $\|\Delta\|_\infty \leq 1$, and Δ is replaced with ΔW.

The next step, is to describe the robustness measures for the aforementioned unstructured uncertainty models. For input multiplicative uncertainty, the robust stability condition can be derived from the small gain theorem, resulting in the condition that the uncertain closed loop system will remain stable only if:

$$\max_\Delta \|M\Delta(j\omega)\|_\infty \leq 1 \tag{4.12}$$

This expression can be manipulated into the following condition (full deduction shown in pages 41 and 42 of *Bates* and *Postlethwaite* (2002)):

$$\text{Robust Stability} \Leftrightarrow \|W_I(j\omega)T_I(j\omega)\|_\infty < 1 \tag{4.13}$$

where $T_I(j\omega)$ is the input complementary sensitivity function, defined as $T_I = L_I\,(I + L_I)^{-1}$. $L_I = KG$ is called input open loop transfer matrix and $W_I(j\omega)$ represents the weight matrix discussed previously.

The same process can be done for the output multiplicative uncertainty and for the additive uncertainty. The condition for robust stability of the output multiplicative uncertainty model is:

$$\text{Robust Stability} \Leftrightarrow \|W_O(j\omega)T_O(j\omega)\|_\infty < 1 \tag{4.14}$$

where T_O is the output complementary sensitivity function and is defined as $T_O = L_O(I+L_O)^{-1}$. $L_O = GK$ is called output open loop transfer matrix and $W_O(jw)$ is a matrix utilized for assigning weights to the uncertainties.

The condition for robust stability of the additive uncertainty model is:

$$\text{Robust Stability} \Leftrightarrow \|W_A(j\omega)K(j\omega)S_O(j\omega)\|_\infty < 1 \tag{4.15}$$

where S_O is the output sensitivity function, and is defined as $S_O = (I + L_O)^{-1}$. Once again, $W_A(jw)$ represents a matrix used to attribute weights to the uncertainties. Finally, it is important to note that in this condition, the controller $K(jw)$ is also present.

Before moving onto the coprime factor uncertainty, it is opportune to introduce one final concept: S_I is the input sensitivity function, and is defined as $S_I = (I + L_I)^{-1}$.

Coprime Factor Uncertainty

Coprime Factor Uncertainty is an important model for $\mathcal{H}_\infty$ Open Loop Shaping, it is therefore crucial to address it. Firstly, the process to derive a left coprime factorization will be given (as the process for the right factorization is analogous).

Consider a system $G(s)$, a left coprime factorization of G is:

$$G(s) = M_l^{-1}(s)N_l(s) \tag{4.16}$$

where $M_l(s)$ and $N_l(s)$ are stable coprime functions.

From a stability point of view, $N_l(s)$ must contain all right-hand-plane (RHP) zeros of G(s), while $M_l(s)$ must contain all RHP poles of G(s) as its RHP zeros. Meanwhile, coprimeness requires that the coprime

functions cannot share common RHP zeros or otherwise they will result in pole-zero cancellations.

Coprime factorization leads to M and N matrices that are not unique. However, an extra condition can be added to the formulation so that the matrices become unique. This leads to the so-called normalized coprime factorization. Let $N_l(s)$ and $M_l(s)$ be a left coprime factorization of $G(s)$. Then, $N_l(s)$ and $M_l(s)$ are a normalized left coprime factorization of $G(s)$ if, and only if:

$$M_l M_l^* + N_l N_r^* = I \tag{4.17}$$

where $M_l^*(s) = M_l^T(-s)$. Note that for $s = jw$, the operator $M^* = M^T(-s)$ is the same as the complex conjugate transpose.

A normalized left coprime factorization, $G(s) = M_l^{-1}(s)N_l(s)$ can then be used to represent an uncertain plant model G_p:

$$G_p = \left\{ (M_l + \Delta_M)^{-1} (N_l + \Delta_N) : \|[\Delta_N \Delta_M]\|_\infty < \epsilon \right\} \tag{4.18}$$

where the Δ matrices are stable unknown transfer functions and represent the uncertainty in the nominal plant G.

A visual representation of G_P is given in Figure 4.4. It is crucial to emphasize that the uncertainty matrices, Δ_N and Δ_M are not normalized to be smaller than unity. The reason for this will become clear in the $\mathcal{H}_\infty$ Open Loop Shaping section.

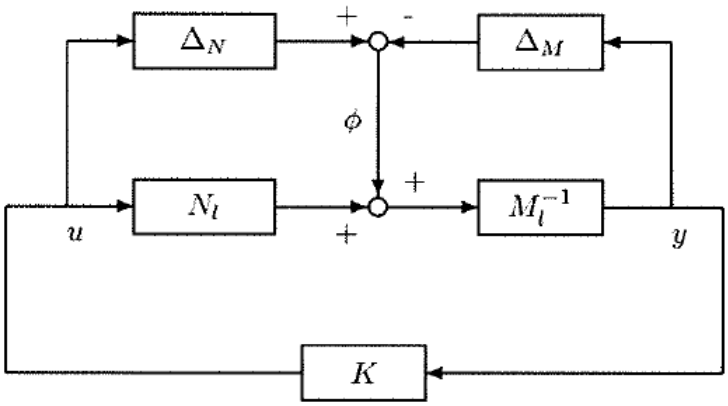

Figure 4.4: Uncertain plant model, retrieved from *Bates* and *Postlethwaite* (2002)

To define the robust stability measure for the coprime factor uncertainty models, the blocks of Figure 4.4 are rearranged into the configuration shown in Figure 4.3. This leads to a Δ and M matrices defined as follows:

$$\Delta = [\Delta_N \ \Delta_M]; \quad M = \begin{bmatrix} K \\ I \end{bmatrix} (I - GK)^{-1} M_l^{-1} \tag{4.19}$$

Then, using the small gain theorem, it is possible to show that:

$$\text{Robust Stability } \forall \, \|[\Delta_N \ \Delta_M]\|_\infty \leq \epsilon \Leftrightarrow \|M\|_\infty < \frac{1}{\epsilon} \tag{4.20}$$

where ϵ is a constant that represents the maximum $\mathcal{H}_\infty$ norm of the uncertainty matrices.

The condition given in eq. 4.20 provides a measure of robust stability, however, it is more convenient

to introduce γ as a robustness margin measure:

$$\gamma := \left\| \begin{bmatrix} K \\ I \end{bmatrix} (I - GK)^{-1} M_l^{-1} \right\|_\infty \tag{4.21}$$

This parameter, which will later emerge during the $\mathcal{H}_\infty$ Open Loop Shaping section, is a measure of how much coprime factor uncertainty the system can withstand. Suppose $\gamma = 2$, then the closed loop system can withstand coprime factor uncertainty with a maximum singular value bound of 0.5.

Advantages and Disadvantages of Unstructured Uncertainty Models

Having already described several types of unstructured uncertainty models, it is important to compare them. This will be done in the following discussion.

While all the models described offer conditions for closed loop stability in the presence of simultaneous gain/phase uncertainty, only coprime factorization permits the representation of situations where parameter variations can turn the open loop system unstable. This stems from the fact that additive, input multiplicative, and output multiplicative models are only capable of utilizing uncertainty descriptions whose transfer function matrices are stable. Furthermore, coprime factorization enables the description of uncertainty in the locations of lightly damped poles, which the other models do not support - both previous statements were demonstrated in *Bates* and *Postlethwaite* (2002).

Finally, a drawback of robustness measures based on unstructured uncertainty is that the resulting stability margins tend to be overly conservative or pessimistic, as compared to those derived from structured uncertainty approaches, as demonstrated by *Skogestad* and *Postlethwaite* (2005).

4.7.2. Structured Uncertainty

Structured uncertainty is born from the need to produce less conservative uncertainty models. It refers to a type of uncertainty for which detailed information is available and that can be represented more accurately than a complete matrix block. This level of detailed information enables the formulation of less conservative controllers, as the system's uncertainties can be precisely characterized and accounted for in the control design and analysis process. An example of possible available information is knowing which system parameters are uncertain, for instance, knowing that only center of gravity and moment of inertia are uncertain.

In this subsection, a method that enables studying structured uncertainty will be approached. The concept in question is the structured singular value (SSV), which only requires the uncertainty block to be diagonal, and allows for less conservative controller design. This is particularly relevant, as according to *J. Doyle* (1982), "any norm-bounded perturbation problem, regardless of structure, can be trivially rewritten as a block-diagonal perturbation problem".

Structured Singular Value

Consider that the uncertainty matrix can be represented by a block diagonal structure:

$$\Delta(j\omega) = \text{diag}\left(\Delta_1(j\omega), \ldots\ldots, \Delta_n(j\omega)\right), \overline{\sigma}\left(\Delta_i(j\omega)\right) \leq k \quad \forall\omega \tag{4.22}$$

With the further assumption that the nominal closed loop system is stable, *J. Doyle* (1982) introduced the following concept:

$$\mu_\Delta\left(M_{11}\right) = \frac{1}{\min\left(k \text{ s.t. } \det\left(I - M_{11}\Delta\right) = 0\right)} \tag{4.23}$$

where M represents the known part of the system (nominal plant and controller), and $M_{1,1}$ represents the transfer function from the output of the Δ block (i) to the input of the Δ block (r) in figure 4.2.

The above equation defines the Structured Singular Value (μ), which serves as a criterion for assessing the stability of a closed loop system in the presence of structured uncertainty. Robust stability under structured uncertainty is ensured if the following condition is satisfied (*Skogestad* & *Postlethwaite*, 2005):

$$\text{Robust stability} \Leftrightarrow \mu\left(M_{11}\right) < 1 \quad \forall\omega \tag{4.24}$$

Additionally, the SSV can be used for many other tests. For example, computing the worst-case gain of a particular transfer function is equivalent to computing the SSV for an appropriate block structure, as seen by the results of *Packard* et al. (2000). While this application of the tool will be utilized in this work, a detailed explanation is beyond the scope of this project. For more information, the reader can refer to *Packard* et al. (2000).

Advantages and Drawbacks

The greatest advantage of the SSV lies in its ability to reduce the conservativeness of the uncertainty model. Moreover, unlike other uncertainty models, it enables the evaluation of robust performance by incorporating an additional fictitious uncertainty block. A detailed explanation of this process can be found on page 343 of *Skogestad* and *Postlethwaite* (2005) and pages 61–63 of *Bates* and *Postlethwaite* (2002). However, it is important to note that in this approach, performance is solely evaluated from a frequency domain standpoint. Thus, it may not be adequate for fully demonstrating the comprehensive functioning of the control system.

The primary drawback associated with the SSV stems from its computational demands. As outlined in *Toker* and *Özbay* (1995), the computational complexity of μ increases severely with the number of parameters involved, posing significant challenges. Given the substantial size of many aerospace systems, computing the SSV becomes nearly impractical. Nonetheless, algorithms to compute tight upper and lower bounds on the SSV have been employed to tackle this issue. They will not be described in this review, as that would go beyond the purpose of this work. However, some descriptions of these algorithms can be found in pages 63-65 of *Bates* and *Postlethwaite* (2002) and pages 334-340 of *Skogestad* and *Postlethwaite* (2005).

4.8. Balanced Disk Margins

Disk margins, discussed in *Seiler* et al. (2020), are a tool to assess system relative stability. They provide a comprehensive measure of robust stability by accounting for simultaneous gain and phase perturbations. This section focuses specifically on describing the balanced disk margins (also referred to as S-T disk margins), a specific type of disk margin which is utilized in this work.

S-T disk margins are termed ”balanced” as the allowable gain/phase increase and decrease are symmetric. They are computed in two steps. First, a complex-valued multiplicative factor, f, is introduced into the open loop L, resulting in a perturbed open loop $L_p = fL$. For balanced disk margins, f is parametrized as follows:

$$f \in D(\alpha) = \left\{ \frac{1+\frac{1}{2}\delta}{1-\frac{1}{2}\delta} : \delta \in \mathbb{C} \text{ with } |\delta| < \alpha \right\}. \tag{4.25}$$

where α can be thought of as the parameter that determines how large the gain and phase variation introduced by f is.

The second step involves determining the largest value of α that maintains closed loop stability. Formally, the balanced disk margin α_{max} is defined as the largest value of α such that the closed loop with fL is well-posed and stable for all complex perturbations $f \in D(\alpha)$.

Once α_{max} is determined, guaranteed gain and phase margins can be computed. These are hereafter referred to as the S-T disk-based gain margins (γ_{min} and γ_{max}) and the S-T disk-based phase margin (ϕ_m). The guaranteed margins provide a lower estimate for the classical gain-only and phase-only margins, and can be computed with the following equations:

$$\gamma_{min} = \frac{2 - \alpha_{max}}{2 + \alpha_{max}} \text{ and } \gamma_{max} = \frac{2 + \alpha_{max}}{2 - \alpha_{max}} \text{ and } \cos\phi_m = \frac{1 + \gamma_{min}\gamma_{max}}{\gamma_{min} + \gamma_{max}} \tag{4.26}$$

4.9. $\mathcal{H}_\infty$ Closed Loop Shaping Controller Design

The aim of this section is to give the reader a comprehensive understanding of the $\mathcal{H}_\infty$ Closed Loop Shaping Controller Design method. This method aims at manipulating the singular values of different closed loop transfer functions so as to achieve the different design goals.

The first application of $\mathcal{H}_\infty$ CLS was done in the work of *Zames* (1981), through a specific formulation called $\mathcal{H}_\infty$ Mixed Sensitivity. In the Mixed Sensitivity formulation, the sensitivity function is shaped alongside with one or more additional closed loop transfer functions (for instance, T_O). However, this approach operates by shaping the singular values of a MIMO system, which can be limiting. This work adopts a broader $\mathcal{H}_\infty$ CLS framework, based on a method named Signal-based $\mathcal{H}_\infty$ (see pages 396-399 of *Skogestad* and *Postlethwaite* (2005)), which offers greater design flexibility by enabling the independent manipulation of individual closed loop transfer functions.

The discussion about this method will be divided into different subsections. Firstly, the $\mathcal{H}_\infty$ problem will be discussed in Subsection 4.9.1. Then, in Subsection 4.9.2, the $\mathcal{H}_\infty$ CLS method will be described. Afterwards, in Subsection 4.9.3, a discussion on the different closed loop transfer functions (CLTFs) and their correspondence to specifications on system performance and robustness will be given. Finally, the advantages and disadvantages of this controller synthesis method will be considered in Subsection 4.9.4.

4.9.1. $\mathcal{H}_\infty$ Control Problem Formulation

Recall the general feedback configuration shown in Figure 4.1, but assume the Δ matrix, which represents the uncertainty block, is null and thus disregarded. With this assumption, the transfer matrix from w to z, can be given by the lower LFT:

$$\begin{aligned} z &= \left\{ P_{11} + P_{12}K\left(I - P_{22}K\right)^{-1} P_{21} \right\} w \\ &= F_l(P, K)w \end{aligned} \tag{4.27}$$

With this in mind, the problem of computing the controller can be formally defined. The standard $\mathcal{H}_\infty$ control problem is the minimization of the $\mathcal{H}_\infty$ norm of $F_l(P, K)$ over all stabilizing controllers (*Bates* & *Postlethwaite*, 2002):

$$\inf_K \left\| F_l(P, K)(j\omega) \right\|_\infty \tag{4.28}$$

The different solutions for this problem can be found in *J. C. Doyle* et al. (1989), *Gahinet* and *Apkarian* (1994), *Apkarian* and *Noll* (2017). In this work, the latter solution will be utilized through the use of MATLAB's systune function. This solution employs a non-smooth, non-convex optimization approach to

design controllers with a fixed structure and order. While this means that the design may be influenced by the choice of the optimization's starting point, the issue can often be alleviated by conducting multiple optimization runs from different initial points. However, this raises concerns about the repeatability of the results. Despite these challenges, this approach has been successfully implemented in various projects, including the ESA Rosetta mission (see *Falcoz* et al. (2015) or *Apkarian* and *Noll* (2017)).

4.9.2. $\mathcal{H}_\infty$ Closed Loop Shaping Method

The $\mathcal{H}_\infty$ CLS design method consists of three steps, and will be discussed for the case when systune is used to apply it. First, the plant and the robustness and performance goals for the control system are defined. Secondly, the transfer functions that can be used to accomplish the aforementioned goals are selected and singular value limitations are established for the particular transfer functions. Thirdly, the controller is computed with the use of systune, where the $\mathcal{H}_\infty$ norm minimization of each goal is performed individually. Then, after analyzing the results, the SV limitations of each function can be re-tuned to achieve better results.

In practice, step two typically cannot be completed in a single iteration and is often integrated with step three as follows. First, limitations are applied to the singular value shape/magnitude of one or two transfer functions, and a controller is computed using systune. Singular value plots of all relevant closed loop transfer functions are then generated. The designer evaluates these plots to identify any issues and, if necessary, adjusts the singular value limitations or adds additional constraints (e.g., limiting the singular value of another transfer function). The controller is then recalculated. This process is repeated until the designer is satisfied with the singular value plots for all transfer functions.

4.9.3. Closed Loop Transfer Function Manipulation

This section examines the effects of manipulating the singular values of various transfer functions. Specifically, the following list gives examples of how adjustments to the so-called 'gang of six' (S_I, S_O, T_I, T_O, S_OG, KS_O) can achieve key objectives such as disturbance rejection, noise attenuation, reference tracking and control signal attenuation. For a complete discussion, the reader is encouraged to consult pages 93-95 of *Bates* and *Postlethwaite* (2002) and pages 371-373 of *Skogestad* and *Postlethwaite* (2005).

- To attain disturbance rejection of output disturbance signals at the plant output, ensure that $\overline{\sigma}(S_O)$ is small[1].
- To attain disturbance rejection of input disturbance signals at the plant input, ensure that $\overline{\sigma}(S_I)$ is small.
- For effective reference tracking, $\overline{\sigma}(T_O)$ and $\underline{\sigma}(T_O)$ should be close to 1 (but this is achieved by ensuring that $\overline{\sigma}(S_O)$ is small).
- To attain noise attenuation at the plant output, ensure that $\overline{\sigma}(T_O)$ is small.
- To prevent large control signals in response to reference inputs, ensure that $\overline{\sigma}(KS_O)$ is small (note that this can only be done at high frequencies, as otherwise it would impact disturbance rejection at the plant input, see pages 13 and 14 of *Papageorgiou* (1998)).

Besides the goals that have already been mentioned, it is also possible to enforce reference model following. This can be done by limiting the singular value of a transfer function from the command input to the error between the system's response and the response of a reference model. Moreover, closed loop

[1] $\overline{\sigma}$ denotes the maximum singular value, while $\underline{\sigma}$ represents the minimum singular value.

specifications can be used to directly translate other performance goals. For the case of the LV, in which load relief is a requirement, a limitation on the $\mathcal{H}_\infty$ norm of the transfer function from wind input to angle of attack can be used to limit the load placed on the LV (this will be discussed later on). Additionally, closed loop specifications can also be used to indirectly impose stability margins, this is done by limiting the peak SV of the system's sensitivity function (see page 37 of *Skogestad* and *Postlethwaite* (2005)).

Finally, it should be noted that depending on the objectives for the control system, conflicting requirements may arise. An example is provided to illustrate this issue. Consider the following hypothetical scenario: there are two requirements for the control system, namely robust stability to output multiplicative uncertainty and attenuation of output disturbance signals, at the plant output. The first requirements can be met by minimizing $\overline{\sigma}(T_O)$, while the second can be met by minimizing $\overline{\sigma}(S_O)$. Remember now that $S_O + T_O = I$. If the frequency range relevant to each goal differs, there may not be a need for a trade-off between requirements, as each closed loop transfer function can be optimized independently within its respective frequency band. However, in the scenario where the relevant frequency range is the same for multiple goals, a trade-off becomes inevitable.

4.9.4. Advantages and Disadvantages

Having discussed how to utilize the $\mathcal{H}_\infty$ CLS controller design method, it is now relevant to discuss its advantages and disadvantages.

The primary strength of the $\mathcal{H}_\infty$ CLS controller design method lies in its ability to directly shape various closed loop transfer functions, providing a clear and transparent design process where performance requirements can often be explicitly defined. Another notable advantage is its capacity to produce controllers robust to both additive and multiplicative uncertainties, as well as its flexibility in incorporating reference model following.

However, challenges arise when conflicting design requirements are present, as the designer must carefully balance and reconcile the shaping of individual closed loop transfer functions (CLTFs). Additionally, ensuring robustness at both the plant input and output requires simultaneous manipulation of multiple CLTFs, which increases design complexity. As will be demonstrated in the next section, the $\mathcal{H}_\infty$ Open Loop Shaping method addresses both of these challenges effectively.

4.10. $\mathcal{H}_\infty$ Open Loop Shaping Controller Design

The $\mathcal{H}_\infty$ Open Loop Shaping design procedure originates from the early contributions of Vidyasagar (see for instance *Vidyasagar* and *Kimura* (1986)). Subsequently, it was extensively detailed in *McFarlane* et al. (1988), further refined in *McFarlane* and *Glover* (1988), *Glover* and *McFarlane* (1989), and in *McFarlane* (1989). Finally, in *McFarlane* and *Glover* (1992) a systematic framework and practical application of the method is provided.

$\mathcal{H}_\infty$ Open Loop Shaping is a combination of an $\mathcal{H}_\infty$ optimization problem with classic open loop shaping methods. The open loop shaping methods aim at giving the open loop singular values the desired shape, while the $\mathcal{H}_\infty$ optimization aims at robustly stabilizing the shaped plant against normalized coprime factor (NCF) uncertainty.

There are several benefits associated with this method, which will be discussed in Subsection 4.10.4. However, the main advantages stem from its ability to directly shape the open loop and robustify the system against NCF uncertainty, both of which address the challenges inherent to the $\mathcal{H}_\infty$ CLS method. Before exploring these benefits in greater detail, the steps required to apply $\mathcal{H}_\infty$ Open Loop Shaping will

be outlined. The first step, explained in Subsection 4.10.1, involves shaping the open loop singular values. The second step, covered in Subsection 4.10.2, is the computation of the NCF robustifying controller. The third and final step, discussed in Subsection 4.10.3, focuses on implementing the controller and the weighting filters. This subsection will also provide guidance on how to integrate and apply each step within the overall design process.

4.10.1. Open Loop Shaping

The first step in the $\mathcal{H}_\infty$ Open Loop Shaping procedure is to shape the singular values of the open loop transfer functions ($L_O = GK$ and $L_I = KG$). This is done so as to achieve achieve the desired performance and robustness goals. The following list gives examples of how the singular values of L_O and L_I can be manipulated to achieve goals such as disturbance rejection, noise attenuation, reference tracking and control signal attenuation. For a more detailed discussion, the reader should refer to pages 111-113 of *Bates* and *Postlethwaite* (2002) or pages 373 and 374 of *Skogestad* and *Postlethwaite* (2005).

- To attenuate output disturbance signals at the plant output, ensure that $\underline{\sigma}(L_O)$ is large.
- To attenuate input disturbance signals at the plant input, ensure that $\underline{\sigma}(L_I)$ is large.
- To achieve good reference tracking, ensure that $\underline{\sigma}(L_O)$ is large.
- To attenuate measurement noise signals at the plant output, ensure that $\overline{\sigma}(L_O)$ is small.
- To avoid large control signals, ensure that $\overline{\sigma}(K)$ is small (this can only be done at high frequencies, as otherwise it would impact disturbance rejection) and that $\overline{\sigma}(G)$ is large (this is independent of the controller).

For the purpose of shaping the open loop transfer functions, two (generally diagonal) compensators are used: $W1$ and $W2$, which correspond to the pre-compensator and post-compensator, respectively. The full assembly, consisting of $W1(s)$, $W2(s)$, and $G(s)$, represents the shaped plant $G_s(s)$, as illustrated in Figure 4.5.

According to the guidelines established in *Papageorgiou* (1998), $W1$ is used to add integral action and reasonable roll-off rates for the open loop singular values, resulting in zero steady-state error while tracking commands, disturbance rejection and output decoupling. Additionally, proportional action can be added to $W1$ to decrease the phase-lag at crossover, resulting in higher stability. The choosing of the proportional gain, and therefore the PI's zero, results directly from the trade-off between bandwidth and robustness. Meanwhile, $W2$ is used for noise rejection (low-pass filter), and usually also contains lead-lag filters to adjust the crossover frequency, and reduce phase lag at crossover.

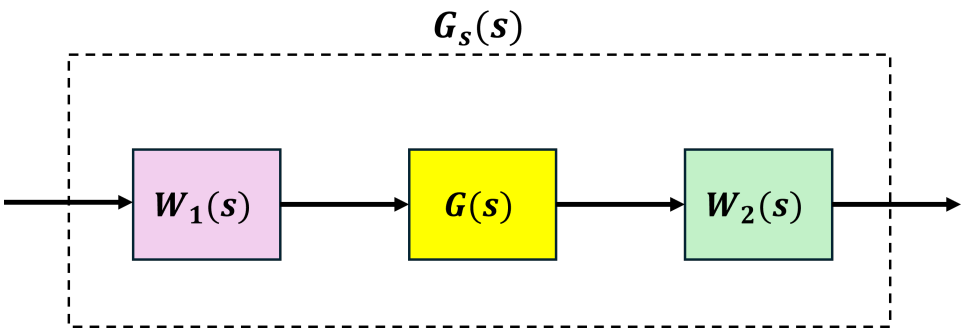

Figure 4.5: Shaped plant.

4.10.2. Robustifying Controller

The second step in $\mathcal{H}_\infty$ Open Loop Shaping is to compute a controller K which robustly stabilizes the shaped plant against normalized coprime factor uncertainty. Consider the shaped plant G_s, with a normalized left coprime factorization $G_s = M^{-1}N$. An uncertain plant model can be written using the NCF specification as:

$$G_p = (M + \Delta_M)^{-1}(N + \Delta_N) \tag{4.29}$$

where the Δ's are stable and proper. The aim of the robust stabilization is to compute a controller which will robustly stabilize the class of perturbed plants defined by:

$$G_p = \left\{ (M + \Delta_M)^{-1} (N + \Delta_N) : \left\| [\Delta_N \Delta_M] \right\|_\infty < \epsilon \right\} \tag{4.30}$$

The problem to be solved is to find the controller that maximizes (or nearly maximizes) the allowable coprime factor uncertainty (ϵ).

In *Glover* and *McFarlane* (1989) this problem is solved, and an analytical solution to compute a sub-optimal controller is found. This solution however, is only capable of generating full-order controllers. In this work, the controller used in the NCF robustification process will consist of four gains (a discussion on this is given later), therefore a full-order solution is not ideal.

In *McFarlane* and *Glover* (1992), it is shown that the process of finding a controller which maximizes ϵ is equivalent to finding a controller which minimizes the $\mathcal{H}_\infty$ norm of a specific four-block transfer function. So, combining that knowledge with systune, it is possible to compute a structured robustifying controller which maximizes the allowable coprime factor uncertainty. Figure 4.6 illustrates the diagram of the transfer function that must be minimized to compute the robustifying controller.

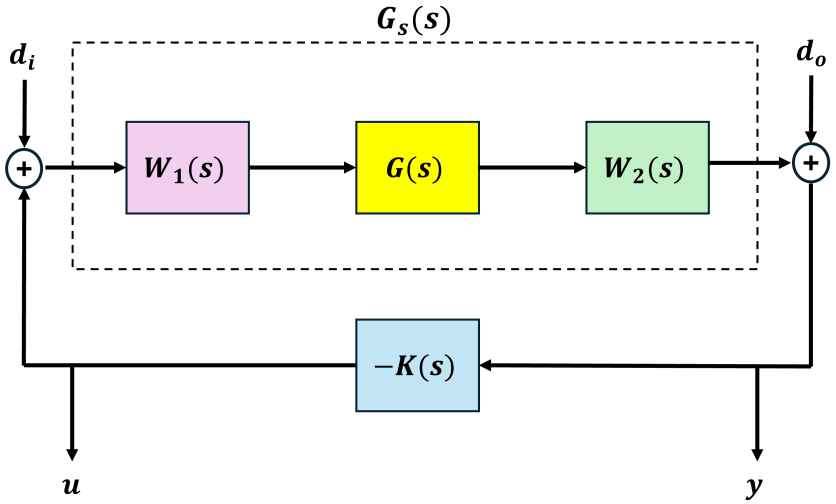

Figure 4.6: NCF robustification scheme.

Finally, one useful mathematical formula is given. The greatest achievable robustness margin (γ_{min}) can be computed analytically through the following formula (*McFarlane* & *Glover*, 1992):

$$\gamma_{min} = \epsilon_{max}^{-1} = \left\{ 1 - \left\| \begin{bmatrix} N & M \end{bmatrix} \right\|_H^2 \right\}^{-\frac{1}{2}} \tag{4.31}$$

where $||.||_H$ is the Hankel norm. The Hankel norm of a system $G(s)$ is defined as (*Skogestad* & *Postlethwaite*, 2005):

$$\|G(s)\|_H = \sqrt{\rho(PQ)} \tag{4.32}$$

where ρ is the spectral radius, P is the controllability Gramian and Q is the observability Gramian (the formulas for P and Q can be found in pages 133 and 136 of *Skogestad* and *Postlethwaite* (2005), respectively).

4.10.3. $\mathcal{H}_\infty$ Open Loop Shaping Design Process and Implementation

The process of applying $\mathcal{H}_\infty$ Open Loop Shaping consists of three steps. The first step involves shaping the open loop singular values, as described in Subsection 4.10.1. However, this alone does not guarantee a robustly stable closed loop system.

The second step involves computing the robustifying controller to ensure closed loop robust stability. This step is divided into two sub-steps. The first sub-step involves using equation 4.31 to compute the maximum attainable stability margin. If $\gamma_{min} > 4$, the designer should revisit step one and adjust the weighting filters. If $\gamma_{min} < 4$, the loop shaping is considered successful, and the designer can proceed to the second sub-step. It is important to note that when $\gamma_{min} < 4$, implementing the robustifying controller will not significantly alter the shape of the open loop singular values (see *McFarlane* and *Glover* (1992)). Thus, robust stability is achieved without significantly degrading the performance characteristics specified by the open loop shaping.

The second sub-step involves designing the robustifying controller. This is accomplished by implementing the setup shown in figure 4.6 in SIMULINK and defining the controller $K(s)$ as a tunable parameter in systune. Then, $K(s)$ is computed by finding the stabilizing controller which minimizes the $\mathcal{H}_\infty$ norm of the transfer function pertaining the setup of figure 4.6.

The final step is to implement the $\mathcal{H}_\infty$ OLS robustifying controller and weighting filters. This can be achieved in various ways, including the regulator implementation, tracker implementation, and observer implementation. For this particular work, only a specific tracker implementation is relevant. The launch vehicle must be capable of tracking references, which makes the regulator implementation unsuitable. Additionally, the observer implementation would result in a controller of full order, which is impractical for this application. The tracker implementation in question is discussed in page 408 of *Skogestad* and *Postlethwaite* (2005), and is illustrated by Figure 4.7.

In the implementation of Figure 4.7, a command pre-filter is used to ensure zero steady-state tracking

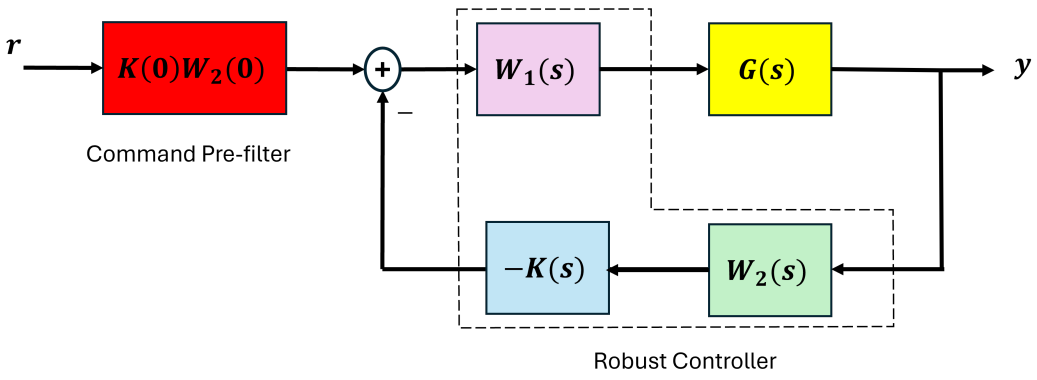

Figure 4.7: $\mathcal{H}_\infty$ OLS tracker implementation.

error, and the robust controller consists of the weighting filters and the NCF robustifying controller. It is important to note that this implementation avoids the direct excitation of the robustifying controller's dynamics by the reference commands, effectively preventing significant overshoot.

4.10.4. Advantages and Disadvantages

The goal of this subsection is to address the different pros and cons of the $\mathcal{H}_\infty$ Open Loop Shaping method (mostly in comparison with $\mathcal{H}_\infty$ CLS method).

Firstly, *Glover* et al. (2000) showed that by robustifying the system against NCF uncertainty, the system will have guaranteed simultaneous symmetric multivariable gain/phase margins at the plant input and output. These can be computed with the following equations (*Bates* & *Postlethwaite*, 2002):

$$\begin{aligned} -20\log_{10}\sqrt{\frac{1+\epsilon}{1-\epsilon}}\ \text{dB} \leq \text{GM} \leq +20\log_{10}\sqrt{\frac{1+\epsilon}{1-\epsilon}}\ \text{dB} \\ -\sin^{-1}\epsilon\ \text{degrees}\ \leq \text{PM} \leq +\sin^{-1}\epsilon\ \text{degrees} \end{aligned} \tag{4.33}$$

These gain and phase variations are allowed simultaneously at each input and output of the plant. For SISO systems, the guaranteed margins are twice as large as those given by the previous equations.

Secondly, there is clear management of conflicting specifications. By shaping the open loop directly, the 'gang of six' (S_I, S_O, T_I, T_O, S_OG, KS_O) is being shaped simultaneously, hence preventing the specification of conflicting requirements.

It is also important to mention that the advantage of having a transparent design is maintained: there is a clear correspondence between open loop shaping and the subsequent robustness and performance results.

One disadvantage of $\mathcal{H}_\infty$ Open Loop Shaping is that it does not allow for reference model tracking. As mentioned earlier, *Limebeer* et al. (1993) introduced the two-degree-of-freedom $\mathcal{H}_\infty$ Open Loop Shaping method, which enables reference model tracking. However, this approach is not particularly relevant for this work because, for the specific flight point selected for controller synthesis (as discussed in Section 6.2), there are typically no requirements for reference model tracking.

Finally, it should be mentioned that the process of selecting the weighting filters used to shape the open loop requires expertise. Some guidelines are given to help on the process, but it will always come down to the designer's capabilities.

5

Launch Vehicle Model

The objective of this chapter is to provide comprehensive information on the launch vehicle modeling. For this purpose, the chapter will begin with the derivation of the kinematic equations pertaining the rigid motion of the launch vehicle (Section 5.1). The next step will be to describe the external forces (Thrust, Gravity, Aerodynamic, Tail-Wag-Dog) that affect the launch vehicle, as well as the flexible behavior of the LV, this will be done in Section 5.2. As the reader might have noticed, cold-gas thrusters and fins are not being considered as possible actuators, due to them not usually being used throughout the atmospheric ascent. In Section 5.3, the rigid design/analysis models will be derived. Meanwhile, the flexible and rigid design/analysis model will be derived in Section 5.4. Section 5.5 will entail the description of the actuator model and delay model. Finally, Section 5.6 contains the listing of the nominal and uncertain values of each parameter of the LTI models.

5.1. Rigid Body Kinematics Derivation

This section outlines the derivation of the kinematic equations pertaining the rigid body motion of the LV, following the process described in *Greensite* (1967) and drawing additional insights from *Zipfel* (2007).

The first step in this derivation is to introduce two reference frames. Let S' be the body reference frame, a right-handed reference frame. This frame consists of three axes, X', Y', and Z', with its origin located at the center of mass of the launch vehicle. At this stage, it is not necessary to specify the orientation of these axes.

Next, let S denote the inertial reference frame, comprising axes X, Y, and Z. For the purposes of this derivation, the explicit definition of the inertial reference frame is not required, as the Earth's rotation can be considered negligible during the launch phase. More importantly, the position and velocity of the launch vehicle relative to an inertial frame are not the primary focus, since the objective of these equations is to support control purposes rather than guidance. Nevertheless, the reference frame can be thought of as an Earth-centered inertial frame, with its axes conveniently oriented with respect to a stellar reference. To illustrate this setup, Figure 5.1 depicts a possible configuration of both reference frames. In this figure, point EC corresponds to Earth's center of mass, while CM represents the LV's center of mass.

Having discussed the reference frames, four vectors are defined in eq. 5.1: $\vec{r}'_S$ - the position of the origin of S' with regard to S, $\vec{v_{S'}}$ - the linear velocity of the origin of S' with regard to the inertial reference frame, $\vec{\omega}$ - the angular velocity of S' and $\vec{r}_i$ - the position of the infinitesimal ith element of mass of the

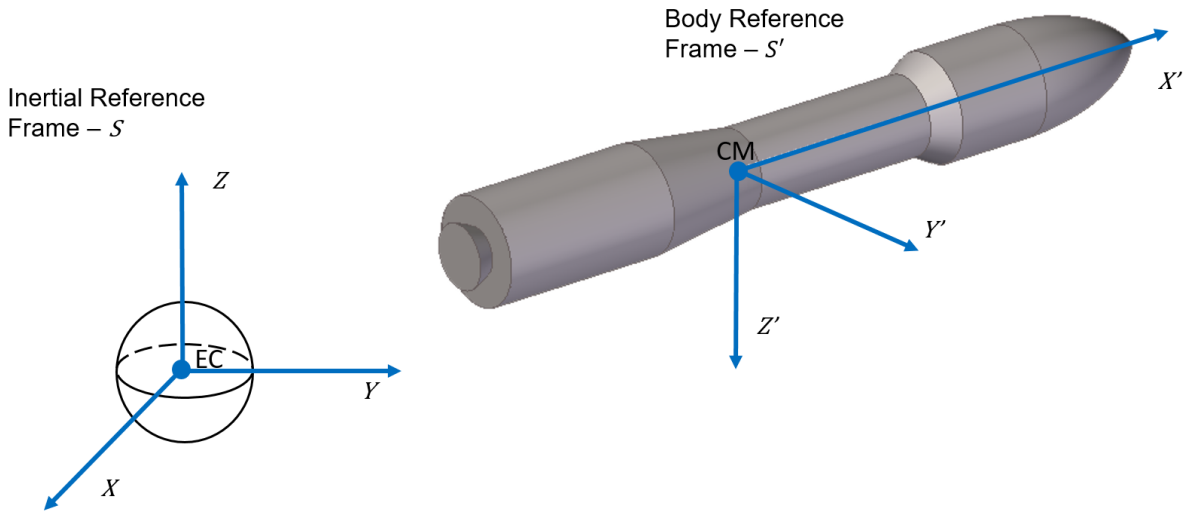

Figure 5.1: Representation of reference frames S and S'.

launch vehicle with regard to the origin of S'.

$$\begin{aligned}
\vec{r}_{S'} &= x'_s\vec{i} + y'_s\vec{j} + z'_s\vec{k} \\
\dot{\vec{r}}_{S'} &= U\vec{i} + V\vec{j} + W\vec{k} \\
\vec{r}_i &= x_i\vec{i} + y_i\vec{j} + z_i\vec{k} \\
\vec{\omega} &= P\vec{i} + Q\vec{j} + R\vec{k}
\end{aligned} \tag{5.1}$$

where $\vec{i}, \vec{j}, \vec{k}$, are a triad vector in the S' reference frame. With this in mind, the velocity of the element of mass can be written:

$$\vec{V}_i = \dot{\vec{r}}_{S'} + \frac{dr_i}{dt} = \dot{\vec{r}}_{S'} + \vec{\omega} \times \vec{r}_i \tag{5.2}$$

And now, the acceleration of the ith element of mass follows logically:

$$\begin{aligned}
\vec{a}_i = \frac{d\vec{V}_i}{dt} &= \frac{d\dot{\vec{r}}_{S'}}{dt} + \vec{\omega} \times \frac{d\vec{r}_i}{dt} + \frac{d\vec{\omega}}{dt} \times \vec{r}_i = \\
&= \dot{U}\vec{i} + \dot{V}\vec{j} + \dot{W}\vec{k} + \vec{\omega} \times \dot{\vec{r}}_{S'} + \vec{\omega} \times (\vec{\omega} \times \vec{r}_i) + (\dot{P}\vec{i} + \dot{Q}\vec{j} + \dot{R}\vec{k}) \times \vec{r}_i = \\
&= \ddot{\vec{r}}_{S'} + \vec{\omega} \times \dot{\vec{r}}_{S'} + \vec{\omega} \times (\vec{\omega} \times \vec{r}_i) + \dot{\vec{\omega}} \times \vec{r}_i
\end{aligned} \tag{5.3}$$

The next step is to integrate the acceleration of the elements of mass and compute the acceleration of the LV, so as to allow for the use of Newton's second law:

$$\begin{aligned}
\vec{F} &= \int \bar{a}_i \, dm \\
&= m_0(\ddot{\vec{r}}_{S'} + \vec{\omega} \times \dot{\vec{r}}_{S'}) + \dot{\vec{w}} \times \int \vec{r}_i \, dm + \vec{\omega} \times (\vec{\omega} \times \int \vec{r}_i \, dm) = \\
&= m_0(\ddot{\vec{r}}_{S'} + \vec{\omega} \times \dot{\vec{r}}_{S'}) + \dot{\vec{w}} \times \vec{r}_c m_0 + \vec{\omega} \times (\vec{\omega} \times \vec{r}_c) m_0 \\
&= m_0(\ddot{\vec{r}}_{S'} + \vec{\omega} \times \dot{\vec{r}}_{S'})
\end{aligned} \tag{5.4}$$

where, m_o corresponds to the vehicle's total mass, and $\vec{r_c}$ corresponds to the position of the center of mass

with regard to S'. Since the center of mass is at the origin of S', all terms involving its position are zero.

Having derived the equation that relates the external forces experienced by the LV with its kinematics, the next step is to do the same but for the torques experienced by the LV. Consider all quantities defined above again. Let m_i and $\vec{r}_i$ be the mass and position of the infinitesimal ith element of mass of a body, with regard to the referential S'. Then, the angular momentum with regard to the origin of the inertial reference frame ($\vec{H}_S$) can be computed as:

$$\begin{aligned}\vec{H}_S &= \int((\vec{r}_{S'} + \vec{r}_i) \times (\frac{d\vec{r}_{S'}}{dt} + \frac{d\vec{r}_i}{dt}))dm = \\ &= \int((\vec{r}_{S'} + \vec{r}_i) \times (\dot{\vec{r}}_{S'} + (\vec{\omega} \times \vec{r}_i)))dm = \\ &= \int(\vec{r}_{S'} \times \dot{\vec{r}}_{S'} + \vec{r}_{S'} \times (\vec{\omega} \times \vec{r}_i) + \vec{r}_i \times \dot{\vec{r}}_{S'} + \vec{r}_i \times (\vec{\omega} \times \vec{r}_i))dm = \\ &= \int(\vec{r}_{S'} \times \dot{\vec{r}}_{S'} + \vec{r}_{S'} \times (\vec{\omega} \times \vec{r}_i) + \vec{r}_i \times \dot{\vec{r}}_{S'})dm + \vec{H}_{S'}\end{aligned} \tag{5.5}$$

where $\vec{H}_{S'}$ is simply equal to $\int(\vec{r}_i \times (\vec{\omega} \times \vec{r}_i)dm)$. The previous equation can be differentiated with regard to time to compute the torque about the origin of the inertial reference frame ($\vec{M}_S$):

$$\begin{aligned}\vec{M}_S = \frac{dH_S}{dt} = \int(&\dot{\vec{r}}_{S'} \times \dot{\vec{r}}_{S'} + \vec{r}_{S'} \times \ddot{\vec{r}}_{S'} + \dot{\vec{r}}_{S'} \times (\vec{\omega} \times \vec{r}_i) + \\ &+ \vec{r}_{S'} \times (\dot{\vec{\omega}} \times \vec{r}_i + \vec{\omega} \times (\vec{\omega} \times \vec{r}_i)) + (\vec{\omega} \times \vec{r}_i) \times \dot{\vec{r}}_{S'} + \vec{r}_i \times \ddot{\vec{r}}_{S'})dm + \frac{d\vec{H}_{S'}}{dt}\end{aligned} \tag{5.6}$$

This expression can be simplified by considering that the cross product of a vector times itself is 0, that the sum of two cross products between the same vectors but with opposite order is 0, and by recalling that S' is centered on the LV's center of mass. Additionally, the equivalences $\dot{\vec{r}}_i = (\vec{\omega} \times \vec{r}_i)$ and $\ddot{\vec{r}}_i = (\dot{\vec{\omega}} \times \vec{r}_i + \vec{\omega} \times (\vec{\omega} \times \vec{r}_i))$ are used to simplify the expression, leading to:

$$\vec{M}_S = \int(\vec{r}_{S'} \times (\ddot{\vec{r}}_{S'} + \ddot{\vec{r}}_i))dm + \frac{d\vec{H}_{S'}}{dt} \tag{5.7}$$

Additionally, the moment of the external forces about the origin of S can be written as:

$$\vec{M}_S = \int((\vec{r}_{S'} + \vec{r}_i) \times \frac{\vec{F}_i}{m_0})dm = \int(\vec{r}_{S'} \times \frac{\vec{F}_i}{m_0})dm + \vec{M}_{S'} = \vec{r}_{S'} \times \int(\frac{\vec{F}_i}{m_0})dm + \vec{M}_{S'} \tag{5.8}$$

where the moment of external forces about S', $\vec{M}_{S'}$, is given by $M_{S'} = \int(\vec{r}_i \times \frac{\vec{F}_i}{m_o})dm$.

Now, through the combination of eq. 5.8 and eq. 5.7, it is possible to compute an expression that relates the external momentum about the origin of S' and the kinematics of the LV:

$$\vec{r}_{S'} \times \int(\frac{\vec{F}_i}{m_0})dm + \vec{M}_{S'} = \int(\vec{r}_{S'} \times (\ddot{\vec{r}}_{S'} + \ddot{\vec{r}}_i))dm + \frac{d\vec{H}_{S'}}{dt} \tag{5.9}$$

By realizing that $\int((\ddot{\vec{r}}_{S'} + \ddot{\vec{r}}_i)dm)$, is simply equal to the sum of the external forces, the previous expression can be simplified, leading to:

$$\vec{M}_{S'} = \frac{d\vec{H}_{S'}}{dt} \tag{5.10}$$

In order to define $\frac{d\vec{H}_{S'}}{dt}$, recall that the angular momentum about S' can be computed with:

$$\vec{H}_{S'} = \int \vec{r}_i \times (\vec{\omega} \times \vec{r}_i) dm \tag{5.11}$$

Using the triple product expansion, $\vec{H}_{S'}$ can be rewritten as:

$$\vec{H}_{S'} = \int (\vec{\omega}\vec{r}_i^2 - \vec{r}_i(\vec{\omega}\,\vec{r}_i)) dm \tag{5.12}$$

Finally, the derivative of $\vec{H}_{S'}$ with regard to time can be written as:

$$\frac{d\vec{H}_{S'}}{dt} = \dot{\vec{H}}_{S'} + \vec{\omega} \times \vec{H}_{S'} \tag{5.13}$$

where $\dot{\vec{H}}_{S'} = \dot{H}_x\vec{i} + \dot{H}_x\vec{j} + \dot{H}_x\vec{k}$.

With this, all equations required to fully relate the torque and the LV kinematics are defined. The next step in this derivation is to transform the equations that relate torque and kinematics (eq. 5.10), and force and kinematics (eq. 5.4) into component form. For equation 5.4 this is equivalent to computing the external products and expanding the vectors into component form:

$$\begin{aligned} \vec{F} =& [m_0(\dot{U} + QW - RV)]\vec{i} \\ &+[m_0(\dot{V} + RU - PW)]\vec{j} \\ &+[m_0(\dot{W} + PV - QU)]\vec{k} \end{aligned} \tag{5.14}$$

For equation 5.10 the process has a few extra steps. First, utilizing equation 5.12, the components of $\vec{H}_{S'}$ are computed. Secondly, they are replaced into eq. 5.13. Finally, eq. 5.13 is replaced into eq. 5.10. By using this process, the following expression can be derived:

$$\begin{aligned} \vec{M}_{S'} =& [I_{xx}\dot{P} - I_{xy}(\dot{Q} - PR) - I_{xz}(\dot{R} + PQ) + I_{yz}(R^2 - Q^2) + (I_{zz} - I_{yy})QR]\vec{i} \\ &+ -I_{xy}(\dot{P} + QR) + I_{yy}\dot{Q} - I_{yz}(\dot{R} - PQ) + I_{xz}(P^2 - R^2) + (I_{xx} - I_{zz})PR]\vec{j} \\ &+ -I_{xz}(\dot{P} - QR) - I_{yz}(\dot{Q} + PR) + I_{zz}\dot{R} + I_{xy}(Q^2 - P^2) + (I_{yy} - I_{xx})PQ]\vec{k} \end{aligned} \tag{5.15}$$

The two previous sets of equations fully describe the motion of the launch vehicle with regard to the body reference frame. Now, the goal is to attain equations that can be used for the design and analysis models. For that purpose, the equations will be linearized around a trajectory reference frame, S_{d0}, to enable the analysis of perturbations relative to the desired trajectory. This frame is centered at the launch vehicle's center of mass, with the X-axis (X_{d0}) aligned tangentially along the ascent trajectory, the Y-axis (Y_{d0}) situated within the local horizontal plane, and the Z-axis (Z_{d0}) oriented to complete a right-handed orthogonal reference frame. Additionally, it is assumed that the axis of S_{d0} coincide with the principal axes of the launch vehicle, allowing the inertia tensor to be assumed diagonal.

To achieve the linearization, the linear and angular velocities are expressed as the sum of a steady-state component and a disturbance component (e.g., $U = U_0 + u$). The steady-state values are assumed to match those expected along the trajectory. Additionally, the wind components are subtracted from the linear velocities, so that the resulting velocity in the equations corresponds to the airspeed. Thus, the

variables are replaced with the following expressions:

$$\begin{aligned} U &= U_0 + u - U_w \\ V &= v - V_w \\ W &= w - W_w \\ P &= P_0 + p \\ Q &= Q_0 + q \\ R &= R_0 + r \end{aligned} \tag{5.16}$$

Next, it is assumed that all disturbed velocities and wind velocities are small, when compared to the U_0. Additionally, the following definitions are introduced:

$$\begin{aligned} \alpha &= \frac{w - W_w}{U_0} \\ \beta &= \frac{v - V_w}{U_0} \end{aligned} \tag{5.17}$$

With this it is possible to substitute equation 5.16 into equations 5.14 and 5.15. The process will be shown for the $\vec{F}_x$ equation, as it is trivial to replicate it for the other equations.

By utilizing equation 5.16, the expression for $\vec{F}_x$ becomes:

$$\begin{aligned} \vec{F}_x &= [m_0(\dot{U}_0 + \dot{u} - \dot{U}_w + (Q_0 + q)(w - W_w) - (R_0 + r)(v - V_w))]\vec{i} = \\ &= [m_0(\dot{u} + Q_0 w - Q_0 W_w + qw - qW_w - R_0 v + R_0 V_w - rv + rV_w)]\vec{i} \end{aligned} \tag{5.18}$$

Subsequently, using the definitions of equation 5.17, as well as eliminating the higher order terms (qw, $-qW_w$, rv and rV_w) and the steady state terms, the expression for $\vec{F}_x$ simplifies to:

$$\vec{F}_x = m_0[\dot{u} + U_0(Q_0\alpha - R_0\beta)] \tag{5.19}$$

Finally, one further simplification can be made by assuming that all steady state values (but U_0) are of the same order of magnitude as the perturbation variables. And so the equation can be further reduced to:

$$\vec{F}_x = m_0[\dot{u}] \tag{5.20}$$

The process demonstrated for $\vec{F}_x$ can be applied to the other components in eq. 5.14 and eq. 5.15. This leads to the following final set of equations:

$$\begin{aligned} \vec{F}_x &= m_0\dot{u} \\ \vec{F}_y &= m_0\dot{v} + U_0 r \\ \vec{F}_z &= m_0\dot{w} - U_0 q \\ \vec{M}_x &= I_{xx}\dot{p} \\ \vec{M}_y &= I_{yy}\dot{q} \\ \vec{M}_z &= I_{zz}\dot{r} \end{aligned} \tag{5.21}$$

The set of equations given by eq. 5.21 describes the perturbed motion of the launch vehicle relative to the trajectory reference frame. Before incorporating the forces and moments into these equations so as to

develop the design and analysis models, a final concept must be introduced: the Euler angles. With it, it will be possible to describe the vehicle's orientation with regard to the trajectory reference frame.

First, a reference frame centered on the launch vehicle's center of mass is introduced: S_{d3}, with axes X_{d3}, Y_{d3}, and Z_{d3}. Reference frame S_{d3} has its axes aligned with the vehicle's body axes in their disturbed condition and it can be obtained from reference frame S_{d0} in the following manner. Firstly, a rotation by an angle ψ around Z_{d0}, in counterclockwise fashion, generates a new reference frame called S_{d1}, with axes X_{d1}, Y_{d1}, and Z_{d1}; secondly, a rotation by an angle θ around Y_{d1}, in counterclockwise fashion, produces a new reference frame called S_{d2}, with axes X_{d2}, Y_{d2}, and Z_{d2}; thirdly, a rotation by an angle ϕ around the axis X_{d2}, in counterclockwise fashion, results in the reference frame S_{d3}. With this, the rotation matrix that transforms S_{d0} to S_{d3} is given by:

$$\begin{aligned} R_{S_{d3}S_{d0}} &= R_{X_{d2}}(\phi) \cdot R_{Y_{d1}}(\theta) \cdot R_{Z_{d0}}(\psi) \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \cdot \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \cdot \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\cos\theta \\ \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\cos\theta \end{bmatrix} \end{aligned} \tag{5.22}$$

where $R_{S_{d3}S_{d0}}$ is composed of the three successive rotations. And where the inverse of the rotation matrix is simply its transverse.

Now, let the angular velocity of the launch vehicle in reference frame S_{d0} be given by $\vec{\omega} = p\vec{i} + q\vec{j} + r\vec{k}$. Alternatively, the angular velocity can also be expressed in terms of the Euler angular rates $\vec{\omega} = \dot{\phi}\vec{i}_{d3} + \dot{\theta}\vec{j}_{d2} + \dot{\psi}\vec{k}_{d1}$. The axes used in this latter expression can be written as a function of the axes of S_{d0}:

$$\begin{aligned} \vec{i}_{d3} &= \cos(\theta)(\cos(\psi)\vec{i} - \sin(\psi)\vec{j}) + sin(\theta)\vec{k} \\ \vec{j}_{d2} &= \sin(\psi)\vec{i} + \cos(\psi)\vec{j} \\ \vec{k}_{d1} &= \vec{k} \end{aligned} \tag{5.23}$$

By replacing the previous set of equations into $\vec{\omega} = \dot{\phi}\vec{i}_{d3} + \dot{\theta}\vec{j}_{d2} + \dot{\psi}\vec{k}_{d1}$, and then comparing it with $\vec{\omega} = p\vec{i} + q\vec{j} + r\vec{k}.$, it is possible to write:

$$\begin{cases} p = \dot{\theta}\sin(\psi) + \dot{\phi}\cos(\theta)\cos(\phi) \\ q = -\dot{\phi}\cos(\theta)\sin(\psi) + \dot{\theta}\cos\psi, \\ r = \dot{\psi} + \sin(\theta)\dot{\phi} \end{cases} \tag{5.24}$$

Then, by assuming that ϕ, ψ and θ are small, eq. 5.22 and 5.24 can be simplified, leading to eq. 5.25 and 5.26, respectively.

$$R_{S_{d3}S_{d0}} = \begin{bmatrix} 1 & \psi & -\theta \\ -\psi & 1 & \phi \\ \theta & -\phi & 1 \end{bmatrix} \tag{5.25}$$

$$
\begin{aligned}
p &= \dot{\phi} \\
q &= \dot{\theta} \\
r &= \dot{\psi}
\end{aligned}
\tag{5.26}
$$

5.2. Reference Frames, Forces and Moments

The next step in this chapter will entail the modeling of the forces and moments applied to the LV, so as to be able to implement them in the design and analysis models. It was identified that there are six types of forces that are usually considered for the LV modeling problem, namely: gravity, thrust, aerodynamic, nozzle inertia (also known as tail-wag-dog - TWD), propellant sloshing and forces induced by flexible behavior. Out of these, sloshing will not be considered because the data used to populate the LTI models in this work is based on an LV whose first three stages use solid propellants, with only the fourth stage employing liquid fuel. As noted in *Tapia* (2019), sloshing can be disregarded during the atmospheric ascent of vehicles that predominantly use solid fuel.

Next, it is important to note that each force will be modeled and expressed with respect to the disturbed body reference frame. Consequently, the X, Y, and Z directions referenced in the following discussion align with the orientations of the axes of the disturbed body reference frame (X_{d3}, Y_{d3}, and Z_{d3}). Moreover, only the perturbation forces will be considered (steady-state values are disregarded), as the final goal is to have a model which describes the perturbed motion of the launch vehicle. One final remark is that, for simplicity, the time dependence of the various time-dependent variables is not shown.

In order to aid understanding, this section will begin by presenting a figure that portrays the rigid body forces applied to the LV, the reference frames, the moment arms, and other important angles/points/distances.

5.2.1. Launch Vehicle Representation

Figure 5.2, portrays the two reference frames described in the previous section (S_{d0} and S_{d3}), and other important quantities. Note that for the sake of simplicity, the image only shows the pitch plane, and the two

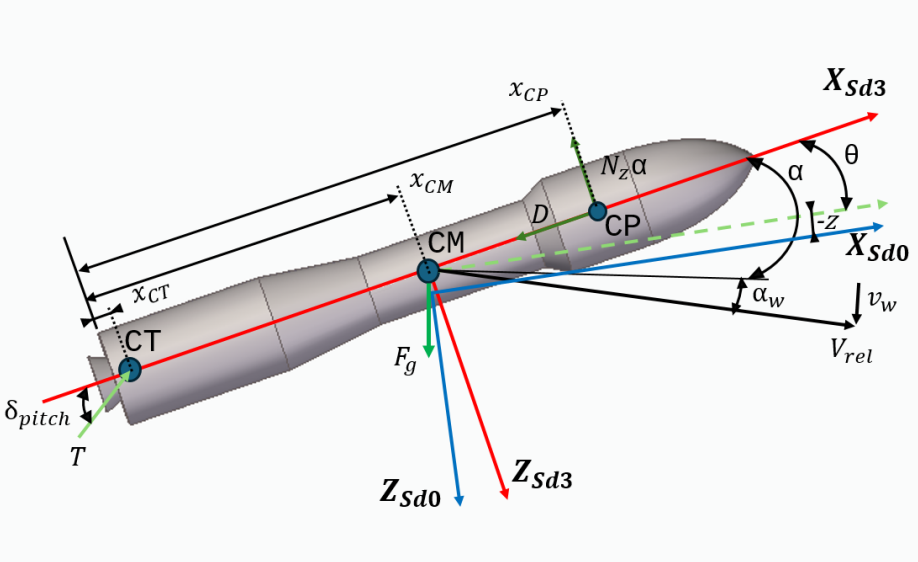

Figure 5.2: LV - Rigid body diagram

frames only differ from a rotation around the Y axis. In this figure, V_{rel} is the vehicle velocity with respect to the ground, the dashed green line represents where the trajectory reference frame's X axis would be if no drift motion had occurred, α is the angle of attack, v_w is the wind velocity, D and $N_z\alpha$ represent the drag force and the force gradient with respect to α, respectively. F_g represents the gravitational force applied to the LV[1], T is the thrust, θ is the pitch angle, δ_{pitch} is the thrust vector control actuator deflection in the pitch plane, $-z$ represents a negative drift motion, CT is the point around which the engine rotates, CM is the center of mass, CP is the center of aerodynamic pressure. Some additional distances are defined: the distance of the point around which the engine rotates to the lower edge of the launch vehicle is given by x_{CT}, the distance of the center of mass to the same reference is given by x_{CM}, the distance of the aerodynamic center of pressure to the same reference is given by x_{CP}.

5.2.2. Forces and Moments

Gravity

For analysis and design models, gravitational forces are usually disregarded due to the gravity turn assumption (see, for instance, page 27 of *Tapia* (2019)). The gravity turn assumption states that the commanded pitch reference is set such that the centripetal acceleration of the launch vehicle equalizes the gravitational force. However, it should be noted that the gravity term will still appear throughout the derivation of the design and analysis models. And so, a representation for this force is needed. Consequently, the components of the gravitational force in the disturbed body reference frame are represented as follows:

$$\begin{aligned} \vec{F}_{gx} &= F_{gx} \\ \vec{F}_{gy} &= 0 \\ \vec{F}_{gz} &= F_{gz} \end{aligned} \tag{5.27}$$

Thrust

The thrust generated by the launch vehicle's engine serves as the propulsive force driving the vehicle during its ascent. As it can be seen in figure 5.2, thrust is applied in point CT, which is at a distance of $(x_{CM} - x_{CT})$ from the center of mass. Due to this distance, a torque will be generated by thrust.

For design and analysis models, thrust's contribution towards the systems dynamics is normally computed by assuming thrust only depends on time (see, for instance, page 20 of *Díaz* (2010)). And so, the mathematical formulation that will be used to model the perturbation thrust for analytical and design models is shown next:

$$\begin{aligned} \vec{F}_{Tx} &= T\cos(\delta_{pitch})\cos(\delta_{yaw}) \\ \vec{F}_{Ty} &= -T\sin(\delta_{yaw})\cos(\delta_{pitch}) \\ \vec{F}_{Tz} &= -T\sin(\delta_{pitch}) \\ \vec{M}_{Tx} &= 0 \\ \vec{M}_{Ty} &= -T(x_{CM} - x_{CT})\sin(\delta_{pitch}) \\ \vec{M}_{Tz} &= T(x_{CM} - x_{CT})\sin(\delta_{yaw})\cos(\delta_{pitch}) \end{aligned} \tag{5.28}$$

where δ_{pitch} and δ_{yaw} represent the actuator angles in the pitch and yaw planes, respectively. Both angles are considered positive in the counterclockwise direction.

[1] Gravity is assumed to act at the center of mass, based on the assumption that the center of gravity coincides with the center of mass.

It is important to note that while the thrust force's component in the X direction is typically considered a steady-state value, it cannot be disregarded. As will be demonstrated later, converting forces from the body frame to the trajectory frame involves multiplying the difference between the thrust and drag forces in the X direction by an Euler angle, resulting in perturbation forces in the direction of the Z_{d0} and Y_{d0} axes of the trajectory reference frame. Therefore, this value must be accounted for.

Aerodynamic Forces

Aerodynamic forces are the forces generated from the interaction of the LV with the atmosphere. They are applied in the center of pressure of the LV, shown in figure 5.2 as CP. For the sake of this thesis, the approach that will be used to model aerodynamic forces is the quasi-steady-state aerodynamic theory (following the work of *Greensite* (1967) and *Díaz* (2010)). The aerodynamic forces and moments can be expressed in the following manner:

$$\begin{aligned}
&\vec{F}_{Ax} = -\tilde{q}S_{ref-x}C_A = D \\
&\vec{F}_{Ay} = \tilde{q}S_{ref-y}C_Y = \tilde{q}S_{ref-y}C_{N_\beta}\beta = N_y\beta \\
&\vec{F}_{Az} = -\tilde{q}S_{ref-z}C_N = -\tilde{q}S_{ref-z}C_{N_\alpha}\alpha = -N_z\alpha \\
&\vec{M}_{Ax} \approx 0 \\
&\vec{M}_{Ay} = \tilde{q}S_{ref-z}C_N(x_{CP} - x_{CM}) = -\tilde{q}S_{ref-z}C_{N_\alpha}\alpha(x_{CM} - x_{CP}) = -N_z\alpha(x_{CM} - x_{CP}) \\
&\vec{M}_{Az} = \tilde{q}S_{ref-y}C_Y(x_{CP} - x_{CM}) = \tilde{q}S_{ref-y}C_{N_\beta}\beta(x_{CM} - x_{CP}) = N_y\beta(x_{CM} - x_{CP})
\end{aligned} \tag{5.29}$$

where C_A, C_N, and C_Y represent the coefficients of aerodynamic force in the X, Z, and Y directions, respectively. S_{ref-i} denotes the reference surface area, $\tilde{q}$ represents the dynamic pressure, C_{N_β} represents the β stability derivative, C_{N_α} represents the α stability derivative, D represents Drag (aerodynamic force in the X direction), while $N_y\beta$ and $N_z\alpha$ represent the force gradient with regard to β and α, respectively.

In order to reach the final expressions it was assumed that the coefficients (C_x) in equation 5.29, could be expanded into a Taylor series about some steady-state condition. With that in mind, each coefficient becomes described by the sum of a constant value plus a set of stability derivatives multiplied by their corresponding variable. Furthermore, following the work of *Tapia* (2019), *Díaz* (2010), *Greensite* (1967), it is assumed that the only stability derivatives that are relevant for the purpose of representing the aerodynamic force perturbation are: C_{N_α} and C_{N_β}. The justification for this approach lies in the fact that the launch vehicle has little to no lifting surfaces, making so that other derivatives are negligible.

Additionally, it should be noted that the force in the X direction is largely free from perturbations (as stated in *Díaz* (2010) and *Greensite* (1967)). However, it must be considered for the same reasons that the thrust in the X direction is taken into account. On the other hand, the torque in X direction is considered to be disregardable for long slender LVs, as stated in *Greensite* (1967).

For a detailed derivation of previous aerodynamic formulas, as well as detailed expressions on how to compute the stability derivative coefficients, the reader should refer to pages 30-35 of *Greensite* (1967).

One final comment is made. Under the assumption that the Earth's atmosphere rotates with the planet without slippage or shearing, the dynamic pressure is given by equation 5.30 (*Simplício*, 2019)

$$\tilde{q} = \frac{1}{2}\rho V_{rel}^2 \tag{5.30}$$

where $V_{rel} = \sqrt{U^2 + V^2 + W^2}$.

Tail-Wag-Dog Forces

The Tail-Wag-Dog forces and moments are caused by the motion of the gimbaled engines. Many times, they are not considered in design and analysis models (see for instance *Hernandez* et al. (2021), *Wie* et al. (2008), *Díaz* (2010), etc..). However, when they are considered, as in *Tapia* (2019), they can be modeled using simple expressions that account for the nozzle's mass, length, and the acceleration of the thrust vector actuator deflection ($\ddot{\delta}_{pitch}$). For this thesis, the mathematical formulation that will be used is the same as the one used in *Tapia* (2019):

$$\begin{aligned}
\vec{F}_{TWD-y} &= -m_N l_N \ddot{\delta}_{yaw} \\
\vec{M}_{TWD-y} &= -\left(m_N l_N (x_{CM} - x_{CT}) + I_{Ny}\right) \ddot{\delta}_{pitch} \\
\vec{F}_{TWD-z} &= -m_N l_N \ddot{\delta}_{pitch} \\
\vec{M}_{TWD-z} &= -\left(m_N l_N (x_{CM} - x_{CT}) + I_{Nz}\right) \ddot{\delta}_{yaw}
\end{aligned} \tag{5.31}$$

where m_N is the mass of the nozzle, l_N is the distance between the nozzle's center of mass and CT, I_{Ny} is the moment of inertia of the nozzle around an axis perpendicular to the pitch plane and that crosses point CT. Meanwhile, I_{Nz} is the moment of inertia of the nozzle around an axis perpendicular to the yaw plane that passes through point CT.

Flexible Body Behavior - Forces and Moments

Finally, it is relevant to discuss the modeling of launch vehicle's flexible motion. It should be noted, that the following discussion limits itself to the pitch plane. The formulas for the yaw plane are completely analogous.

The basic equation pertaining bending modes is the following (*Tapia*, 2019):

$$\ddot{q}_i + 2\zeta_i \omega_i \dot{q}_i + \omega_i^2 q_i = Q_i \tag{5.32}$$

where q_i is the generalized coordinate of the ith bending mode (it can represent displacement, elastic torsion, etc); ω_i is the frequency of the ith bending mode; ζ is the damping of the ith bending mode; Q_i is the generalized force associated with the ith bending mode. The generalized force can be computed with the following expression (*Díaz*, 2010):

$$Q_i = \int_0^L \left(\sum F_z \Psi_i(x) + \sum M_y \Psi_i'(x)\right) dx \tag{5.33}$$

Considering only the first order effects (for simplicity), the ith bending mode is mainly excited by the LV engine, as stated by *Greensite* (1967). And so, the generalized force can be computed by substituting eq. 5.28 and eq. 5.31 (Thrust and TWD) into the generalized force expression:

$$Q_i = \left(-T\delta - m_N l_N \ddot{\delta}\right) \Psi_i(x_{CT}) + I_n \ddot{\delta} \frac{\partial \Psi(x_{CT}))}{\partial x} \tag{5.34}$$

where Ψ_i represents the Translational Length of the ith bending mode; $\Psi_i'(x) = \frac{\partial \Psi_i}{\partial x}$ is the Rotational Length of the ith bending mode.

Using eq. 5.34, the generalized coordinate q_i can be computed at each instant. Subsequently, the

forces and moments induced by the bending modes can be determined using (*Tapia*, 2019):

$$\begin{aligned} F_f &= T \sum_{i=1} \frac{\partial \Psi_i(x_{CT})}{\partial x} q_i \\ M_f &= -T \left((x_{CM} - x_{CT})) \sum_{i=1}^{k} \frac{\partial \Psi_i(x_{CT})}{\partial x} q_i + \sum_{i=1}^{k} \Psi_i(x_{CT}) q_i \right) \end{aligned} \tag{5.35}$$

where F_f is a force applied parallel to the Z axis of the body reference frame.

Flexible Body Behavior - Effects on Sensors

The LV utilizes an Inertial Measurement Unit (IMU) to measure both attitude changes and drift/drift rate. However, the displacement caused by the bending modes will affect the readings of the IMU, possibly causing the system to become unstable, as discussed in *Tapia* (2019). Consequently, the impact of the flexible body motion on the sensor readings must be accounted for. This can be achieved through the following equations (*Tapia*, 2019), where the resulting variables are evaluated at the IMU's location and account for its displacement:

$$\theta_{IMU} = \theta - \sum_i q_i \Psi_i' (x_{IMU}) \tag{5.36}$$

$$\dot{\theta}_{IMU} = \dot{\theta} - \sum_i \dot{q}_i \Psi_i' (x_{IMU}) \tag{5.37}$$

$$z_{IMU} = z - (x_{IMU} - x_{CM})\theta + \sum_{i=1}^{k} q_i \Psi_i(x_{IMU}) \tag{5.38}$$

$$\dot{z}_{IMU} = \dot{z} - (x_{IMU} - x_{CM})\dot{\theta} + \sum_{i=1}^{k} \dot{q}_i \Psi_i(x_{IMU}) \tag{5.39}$$

where x_{IMU} represents the position of the IMU sensor. Note that in the previous set of equations, the terms involving $(x_{IMU} - x_{CM})$ arise due to the IMU's position relative to the center of mass, rather than from the flexible behavior of the system.

5.3. Design and Analysis Models - Rigid Body Only

The design model should be as simple as possible, while still allowing for the design of a controller that performs well with the analysis and simulation models. On the other hand, the model utilized for stability and robustness analysis typically falls, in terms of complexity, between the design and simulation models. It can be used to ascertain the performance of the controller, without modeling every physical effect, allowing for a fast analysis of controllers.

In this section, the analysis model for the rigid body will be derived. Subsequently, two design models will be presented, one simpler, to allow for hand design of a controller, and one more complicated, to be used with robust control methods. It should be noted that both design models are trivially derived from the analysis model, and as such only the analysis model derivation is shown.

5.3.1. Analysis Model - AMR

The derivation of the analysis model (AMR) begins with the computation of the sum of the forces and moments:

$$\begin{aligned}
\sum F_x &= T\cos(\delta_{pitch})\cos(\delta_{yaw}) - D + F_{gx} \\
\sum F_y &= -T\sin(\delta_{yaw})\cos(\delta_{pitch}) - m_N l_N \ddot{\delta}_{yaw} + N_y\beta \\
\sum F_z &= -T\sin(\delta_{pitch}) - m_N l_N \ddot{\delta}_{pitch} - N_z\alpha + F_{gz} \\
\sum M_x &= 0 \\
\sum M_y &= -T(x_{CM} - x_{CT})\sin(\delta_{pitch}) - (m_N l_N(x_{CM} - x_{CT}) + I_N)\ddot{\delta}_{pitch} + N_z\alpha(x_{CP} - x_{CM}) \\
\sum M_z &= T(x_{CM} - x_{CT})\sin(\delta_{yaw})\cos(\delta_{pitch}) + N_y\beta(x_{CP} - x_{CM}) - (m_N l_N(x_{CM} - x_{CT}) + I_N)\ddot{\delta}_{yaw}
\end{aligned} \tag{5.40}$$

These equations can be simplified by assuming that the small angle approach is valid for δ_{pitch} and for δ_{yaw}, leading to:

$$\begin{aligned}
\sum F_x &= T - D + F_{gx} \\
\sum F_y &= -T\delta_{yaw} - m_N l_N \ddot{\delta}_{yaw} + N_y\beta \\
\sum F_z &= -T\delta_{pitch} - m_N l_N \ddot{\delta}_{pitch} - N_z\alpha + F_{gz} \\
\sum M_x &= 0 \\
\sum M_y &= -T(x_{CM} - x_{CT})\delta_{pitch} - (m_N l_N(x_{CM} - x_{CT}) + I_N)\ddot{\delta}_{pitch} + N_z\alpha(x_{CP} - x_{CM}) \\
\sum M_z &= T(x_{CM} - x_{CT})\delta_{yaw} + N_y\beta(x_{CP} - x_{CM}) - (m_N l_N(x_{CM} - x_{CT}) + I_N)\ddot{\delta}_{yaw}
\end{aligned} \tag{5.41}$$

Furthermore, these can be rotated into the reference frame of the trajectory with the use of the inverse of the matrix described in equation 5.25:

$$\begin{aligned}
\sum F_x &= (T - D + F_{gx}) - (-T\delta_{yaw} - m_N l_N \ddot{\delta}_{yaw} + N_y\beta)\psi + (-T\delta_{pitch} - m_N l_N \ddot{\delta}_{pitch} - N_z\alpha + F_{gz})\theta \\
\sum F_y &= (T - D + F_{gx})\psi + (-T\delta_{yaw} - m_N l_N \ddot{\delta}_{yaw} + N_y\beta) - (-T\delta_{pitch} - m_N l_N \ddot{\delta}_{pitch} - N_z\alpha + F_{gz})\phi \\
\sum F_z &= -(T - D + F_{gx})\theta + (-T\delta_{yaw} - m_N l_N \ddot{\delta}_{yaw} + N_y\beta)\phi + (-T\delta_{pitch} - m_N l_N \ddot{\delta}_{pitch} - N_z\alpha + F_{gz}) \\
\sum M_x &= 0 \\
\sum M_y &= (-T(x_{CM} - x_{CP})\delta_{pitch} - (m_N l_N(x_{CM} - x_{CP}) + I_N)\ddot{\delta}_{pitch}) + N_z\alpha(x_{CP} - x_{CM}) \\
&\quad - (T(x_{CM} - x_{CT})\delta_{yaw} + N_y\beta(x_{CP} - x_{CM}) - (m_N l_N(x_{CM} - x_{CT}) + I_N)\ddot{\delta}_{yaw})\phi \\
\sum M_z &= (-T(x_{CM} - x_{CP})\delta_{pitch} - (m_N l_N(x_{CM} - x_{CP}) + I_N)\ddot{\delta}_{pitch})\phi + N_z\alpha(x_{CP} - x_{CM})\phi \\
&\quad + (T(x_{CM} - x_{CT})\delta_{yaw} + N_y\beta(x_{CP} - x_{CM}) - (m_N l_N(x_{CM} - x_{CT}) + I_N)\ddot{\delta}_{yaw})
\end{aligned} \tag{5.42}$$

Next, the assumption of gravity turn is made: it is assumed that the commanded pitch/yaw reference is such that the centripetal acceleration of the LV equalizes the gravitational force. Basically, this means that it is assumed that the thrust and velocity of the launch vehicle are parallel to each other. This assumption allows for the linearization of the equations of motion about the local trajectory reference frame and is quite common amongst LV control literature (see page 246 of *Simplício* (2019), or page 27 of *Tapia* (2019)). Moreover, it was assumed that the contribution of F_{gx} is small and can be disregarded when compared to thrust and drag, due to the low pitch angle of the trajectory. And so, joining equations 5.42 and 5.21, whilst

utilizing all the aforementioned assumptions, leads to the following equations:

$$\begin{aligned}
m_o\dot{u} &= T - D - (-T\delta_{yaw} - m_N l_N \ddot{\delta}_{yaw} + N_y\beta)\psi + (-T\delta_{pitch} - m_N l_N \ddot{\delta}_{pitch} - N_z\alpha + F_{gz})\theta \\
m_o\dot{v} + U_0 r &= (T - D)\psi + (-T\delta_{yaw} - m_N l_N \ddot{\delta}_{yaw} + N_y\beta) - (-T\delta_{pitch} - m_N l_N \ddot{\delta}_{pitch} - N_z\alpha + F_{gz})\phi \\
m_o\dot{w} &= -(T - D)\theta + (-T\delta_{yaw} - m_N l_N \ddot{\delta}_{yaw} + N_y\beta+)\phi + (-T\delta_{pitch} - m_N l_N \ddot{\delta}_{pitch} - N_z\alpha) \\
I_{xx}\ddot{\phi} &= 0 \\
I_{yy}\ddot{\theta} &= (-T(x_{CM} - x_{CP})\delta_{pitch} - (m_N l_N(x_{CM} - x_{CP}) + I_N)\ddot{\delta}_{pitch} + N_z\alpha(x_{CP} - x_{CM})) - \\
&\quad - (T(x_{CM} - x_{CT})\delta_{yaw} + N_y\beta(x_{CP} - x_{CM}) - (m_N l_N(x_{CM} - x_{CT}) + I_N)\ddot{\delta}_{yaw})\phi \\
I_{zz}\ddot{\psi} &= (-T(x_{CM} - x_{CP})\delta_{pitch} - (m_N l_N(x_{CM} - x_{CP}) + I_N)\ddot{\delta}_{pitch} + N_z\alpha(x_{CP} - x_{CM}))\phi + \\
&\quad + (T(x_{CM} - x_{CT})\delta_{yaw} + N_y\beta(x_{CP} - x_{CM}) - (m_N l_N(x_{CM} - x_{CT}) + I_N)\ddot{\delta}_{yaw})
\end{aligned} \tag{5.43}$$

where the correspondence between angular velocities and Euler rates determined in Section 5.1 was used.

Next, the roll motion equation is omitted based on the common assumption that the roll rate is negligible, as supported by *Tapia* (2019), *Simplício* (2019), and *Greensite* (1967). Additionally, it is assumed that the LV is symmetric about its longitudinal axis. These assumptions allow the yaw and pitch axes to be treated as decoupled and identical. This means that a controller designed for the yaw axis will be applicable for the pitch axis, and vice-versa. Consequently, focusing on the pitch plane for further analysis eliminates two degrees of freedom: yaw and lateral translation in the yaw plane. Lastly, the longitudinal dynamics are excluded from consideration since they are minimally influenced by small perturbations, as demonstrated by *Bernard* (2009).

The remaining degrees of freedom are the pitch, and the motion perpendicular to the trajectory in the pitch plane (drift). Taking everything into account, equation 5.43 can be simplified to:

$$\begin{aligned}
m_0\ddot{z} &= -(T - D)\theta - T\delta_{pitch} - m_N l_N \ddot{\delta}_{pitch} - N_z\alpha \\
I_{yy}\ddot{\theta} &= -T(x_{CM} - x_{CT})\delta_{pitch} - (m_N l_N(x_{CM} - x_{CT}) + I_N)\ddot{\delta}_{pitch} + N_z\alpha(x_{CP} - x_{CM})
\end{aligned} \tag{5.44}$$

where $\dot{w}$ was replaced with $\ddot{z}$.

Now, by using eq. 5.44 and the rigid body terms of eq. 5.36, 5.37, 5.38 and 5.39, an LTI space-state model can be derived. For this purpose, it is assumed that for a particular flight point, all varying parameters are frozen (mass, moment of inertia, thrust, drag, position of center of mass, etc..). Additionally, a new equation pertaining the AoA is incorporated to balance the number of unknown variables and equations within the system:

$$\alpha = \alpha_{ground} - \alpha_w = \theta + \frac{\dot{z}}{V_{rel}} - \frac{v_w}{V_{rel}} \tag{5.45}$$

This equation specifically states the dependence of the AoA on two different components: one related to the ground and one caused by wind. Effectively, this allows for the establishing of a relation between the variables $\alpha, \theta, \dot{z}$, ensuring the consistency and solvability of the system.

Taking everything into consideration, the state-space model that represents the LTI system used for the rigid body controller analysis can now be computed. The resulting model is shown in equation 5.46. Note that in this model, the output α has been multiplied by the dynamic pressure in order to yield $\tilde{q}\alpha$ as the output. As will be shown later, this quantity represents an important metric for the lateral load applied to the LV.

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{z} \\ \ddot{z} \end{bmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ \mu_\alpha & -\frac{(x_{CP}-x_{CM})\mu_\alpha}{V_{rel}} & 0 & \frac{\mu_\alpha}{V_{rel}} \\ 0 & 0 & 0 & 1 \\ -\frac{N_z}{m_0}+a_0 & \frac{\mu_\alpha I_{yy}}{m_0 V_{rel}} & 0 & \frac{-N_z}{m_0 V_{rel}} \end{pmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ z \\ \dot{z} \end{bmatrix} + \begin{pmatrix} 0 & 0 \\ \mu_c & \mu_c' \\ 0 & 0 \\ -\frac{T}{m_0} & \frac{-m_N l_N}{m_0} \end{pmatrix} \begin{bmatrix} \delta_{pitch} \\ \ddot{\delta}_{pitch} \end{bmatrix} + \begin{pmatrix} 0 \\ -\frac{\mu_\alpha}{V_{rel}} \\ 0 \\ \frac{N_z}{m_0 V_{rel}} \end{pmatrix} \begin{bmatrix} v_w \end{bmatrix}$$

$$\begin{bmatrix} \tilde{q}\alpha \\ z_{IMU} \\ \dot{z}_{IMU} \\ \theta_{IMU} \\ \dot{\theta}_{IMU} \end{bmatrix} = \begin{pmatrix} \tilde{q} & 0 & 0 & \frac{\tilde{q}}{V_{rel}} \\ -(x_{IMU}-x_{CM}) & 0 & 1 & 0 \\ 0 & -(x_{IMU}-x_{CM}) & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ z \\ \dot{z} \end{bmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{bmatrix} \delta_{pitch} \\ \ddot{\delta}_{pitch} \end{bmatrix} + \begin{pmatrix} -\frac{\tilde{q}}{V_{rel}} \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \begin{bmatrix} v_w \end{bmatrix} \tag{5.46}$$

where

$$\begin{aligned} \mu_\alpha &= \tfrac{N_z(x_{CP}-x_{CM})}{I_{yy}} \\ \mu_c &= -\tfrac{T(x_{CM}-x_{CT})}{I_{yy}} \\ \mu_{c'} &= -\tfrac{m_N l_N(x_{CM}-x_{CT})+I_N}{I_{yy}} \\ a_0 &= -\tfrac{T-D}{m_0} \end{aligned} \tag{5.47}$$

5.3.2. Robust Design Model - DMR

The robust design model (DMR) is in everything equal to the Analysis model, with one key difference: the TWD effect is not included. The TWD effect introduces two non-minimum phase zeros, which would cause challenges during the controller design process (for an extensive list of these challenges the reader should refer to pages 185-193 and 235-239 of *Skogestad* and *Postlethwaite* (2005)). Therefore, if possible, it would be beneficial to remove this effect from the design model. Through $\mu-$sensitivity analysis it was possible to identify that the TWD effect posed little to no effect to the system's stability, and as such could be removed from the design model.

5.3.3. Analytical Design Model

Additionally, it is important to talk about one more design model, which will allow for the analytical design of a simple controller.

First, consider the equation from the first analysis model that pertains the pitch motion:

$$I_{yy}\ddot{\theta} = -T(x_{CM}-x_{CT})\delta_{pitch} - (m_N l_N(x_{CM}-x_{CT})+I_N)\ddot{\delta}_{pitch} + N_z\alpha(x_{CP}-x_{CM}) \tag{5.48}$$

If it is assumed that $\theta = \alpha$, and that the effect of nozzle inertia can be disregarded, it is possible to rewrite this equation as:

$$\ddot{\theta} = \mu_\alpha\,\theta - \mu_c\,\delta_{pitch} \tag{5.49}$$

This equation can be transformed into a transfer function, enabling an initial analysis of the system's stability. The transfer function is expressed in equation 5.50.

$$\frac{\theta}{\delta_{pitch}} = G(s) = \frac{-\mu_c}{s^2 - \mu_\alpha} \tag{5.50}$$

Observing eq. 5.50, and recalling the definition of μ_α, it is clear that the system will be unstable, as μ_α is **always** positive.

5.4. Design and Analysis Model - Rigid and Flexible Body

In this section, two models are derived: the design model and the analysis model, both incorporating rigid and flexible body behavior.

5.4.1. Analysis Model - AMF

Having derived the rigid body analysis model, it is trivial to derive the rigid/flexible body analysis model (AMF). This can be accomplished by incorporating equations 5.32, 5.34, 5.35, 5.36, 5.37, 5.38, 5.39 with equations 5.44 and 5.45. The resulting state-space representation is as follows:

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{z} \\ \ddot{z} \\ \dot{q} \\ \ddot{q} \end{bmatrix} = \left(\begin{array}{cccc|cc} 0 & 1 & 0 & 0 & 0_{1b} & 0_{1b} \\ \mu_\alpha & -\frac{(x_{CP}-x_{CM})\mu_\alpha}{V_{rel}} & 0 & \frac{\mu_\alpha}{V_{rel}} & f_\theta & 0_{1b} \\ 0 & 0 & 0 & 1 & 0_{1b} & 0_{1b} \\ -\frac{N_z}{m_0} + a_0 & \frac{\mu_\alpha I_{yy}}{m_0 V_{rel}} & 0 & \frac{-N_z}{m_0 V_{rel}} & f_z & 0_{1b} \\ \hline 0_{b1} & 0_{b1} & 0_{b1} & 0_{b1} & 0_{bb} & I_b \\ 0_{b1} & 0_{b1} & 0_{b1} & 0_{b1} & f_{q\ddot{q}} & f_{\dot{q}\ddot{q}} \end{array}\right) \begin{bmatrix} \theta \\ \dot{\theta} \\ z \\ \dot{z} \\ q \\ \dot{q} \end{bmatrix} + \left(\begin{array}{cc} 0 & 0 \\ \mu_c & \mu_c' \\ 0 & 0 \\ -\frac{T}{m_0} & \frac{-m_N l_N}{m_0} \\ \hline 0_{b1} & 0_{b1} \\ f_{\delta\ddot{q}} & f_{\dot{\delta}\ddot{q}} \end{array}\right) \begin{bmatrix} \delta_{pitch} \\ \ddot{\delta}_{pitch} \end{bmatrix} + \left(\begin{array}{c} 0 \\ -\frac{\mu_\alpha}{V_{rel}} \\ 0 \\ \frac{N_z}{m_0 V_{rel}} \\ \hline 0_{b1} \\ 0_{b1} \end{array}\right) [v_w]$$

$$\begin{bmatrix} \tilde{q}\alpha \\ z_{IMU} \\ \dot{z}_{IMU} \\ \theta_{IMU} \\ \dot{\theta}_{IMU} \end{bmatrix} = \left(\begin{array}{cccc|cc} \tilde{q} & 0 & 0 & \frac{\tilde{q}}{V_{rel}} & 0_{1b} & 0_{1b} \\ -(x_{IMU} - x_{CM}) & 0 & 1 & 0 & f_{qz} & 0_{1b} \\ 0 & -(x_{IMU} - x_{CM}) & 0 & 1 & 0_{1b} & f_{qz} \\ 1 & 0 & 0 & 0 & f_{q\theta} & 0_{1b} \\ 0 & 1 & 0 & 0 & 0_{1b} & f_{q\theta} \end{array}\right) \begin{bmatrix} \theta \\ \dot{\theta} \\ z \\ \dot{z} \\ q \\ \dot{q} \end{bmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{bmatrix} \delta_{pitch} \\ \ddot{\delta}_{pitch} \end{bmatrix} + \begin{pmatrix} -\frac{\tilde{q}}{V_{rel}} \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} [v_w] \tag{5.51}$$

where

$$\begin{aligned}
f_\theta &= [f_{\theta 1}, ..., f_{\theta b}] && \text{with} \quad f_{\theta i} = \frac{T}{I_{yy}} (\Psi_i'(x_{CT})(x_{CP} - x_{CM}) + \Psi_i(x_{CT})) \\
f_z &= [f_{z1}, ..., f_{zb}] && \text{with} \quad f_{zi} = \frac{T}{m_0} (\Psi_i'(x_{CT})) \\
f_{q\ddot{q}} &= \text{diag}(f_{q\ddot{q}1}, ..., f_{q\ddot{q}b}) && \text{with} \quad f_{q\ddot{q}i} = -{\omega_i}^2 \\
f_{\dot{q}\ddot{q}} &= \text{diag}(f_{\dot{q}\ddot{q}1}, ..., f_{\dot{q}\ddot{q}b}) && \text{with} \quad f_{\dot{q}\ddot{q}i} = -2\omega_i\zeta_i \\
f_{\delta\ddot{q}} &= [f_{\delta\ddot{q}1}, ..., f_{\delta\ddot{q}b}]^T && \text{with} \quad f_{\delta\ddot{q}i} = -T\Psi_i(x_{CT}) \\
f_{\dot{\delta}\ddot{q}} &= [f_{\dot{\delta}\ddot{q}1}, ..., f_{\dot{\delta}\ddot{q}b}]^T && \text{with} \quad f_{\dot{\delta}\ddot{q}i} = I_N\Psi_i'(x_{CT}) - m_N l_N \Psi_i(x_{CT}) \\
f_{qz} &= [f_{qz1}, ..., f_{qzb}] && \text{with} \quad f_{qzi} = \Psi_i(x_{IMU}) \\
f_{q\theta} &= [f_{q\theta 1}, ..., f_{q\theta b}] && \text{with} \quad f_{q\theta i} = -\Psi_i'(x_{IMU})
\end{aligned} \tag{5.52}$$

Additionally, b represents the number of bending modes, and the rigid lines were used to separate the rigid and flexible contributions.

With the state-space model described in equation 5.51, and by incorporating sufficient bending modes, it is possible to simulate both the displacement caused by flexible behavior in the IMU position and the flexible forces experienced by the launch vehicle. Following the approach outlined in *Tapia* (2019), only the first two bending modes will be included in the model.

5.4.2. Design Model - DMF

The rigid/flexible body design model (DMF) is equal to the respective analysis model. μ-sensitivity analysis and testing with controllers designed both with and without TWD included in the design model showed that the TWD effect could not be neglected during the design phase. The reasoning being that the interaction between bending modes and TWD is important and cannot be overlooked; otherwise, the system could exhibit unanticipated performance degradation or stability margin decrease when TWD is included in the model.

5.5. Actuator Model and Delay Model

5.5.1. Actuator Model

For most design and analysis models for launch vehicle control, actuators are either disregarded (see, for instance, *Hernandez* et al. (2021)) or included as a second order model (see page 25 of *Díaz* (2010)). They can also be included through the use of LFTs obtained from hardware in loop simulations (see page 38 of *Tapia* (2019)).

For this thesis, a second order model will be used to represent the actuators. The equation that describes the second order model is shown next:

$$\frac{\delta_{TVC}}{\delta_{cmd}} = \frac{{\omega_a}^2}{s^2 + 2\zeta_a\omega_a s + \omega_a^2} \tag{5.53}$$

where ω_a is the natural frequency, and where ζ_a is the damping ratio.

Next, in Figure 5.3, the bode plot of the second order TVC model, from δ_{cmd} to δ_{TVC} is shown.

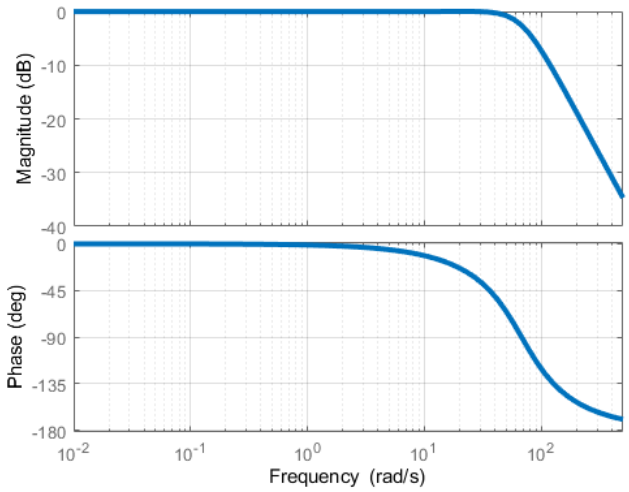

Figure 5.3: Bode plot from δ_{cmd} to δ_{TVC}.

5.5.2. Delay Model

Launch vehicles experience time delays due to the processing performed by on-board digital units. To account for these delays, a delay model can be employed. The delay model that will be used in this work was retrieved from *Tapia* (2019) and is presented below:

$$\frac{\delta_{delayed}}{\delta_{cmd}} = \frac{s^2\tau^2 - s6\tau + 12}{s^2\tau^2 + s6\tau + 12} \tag{5.54}$$

where τ is the time delay, and it was considered this parameter had an uncertainty of 25% (following the uncertainty defined in *Tapia* (2019)).

According to the same author, this model is valid within the frequency range relevant to LV control, spanning from 0.01 rad/s to 100 rad/s (up to the second bending mode). Figure 5.4 shows the bode plot from δ_{cmd} to $\delta_{delayed}$.

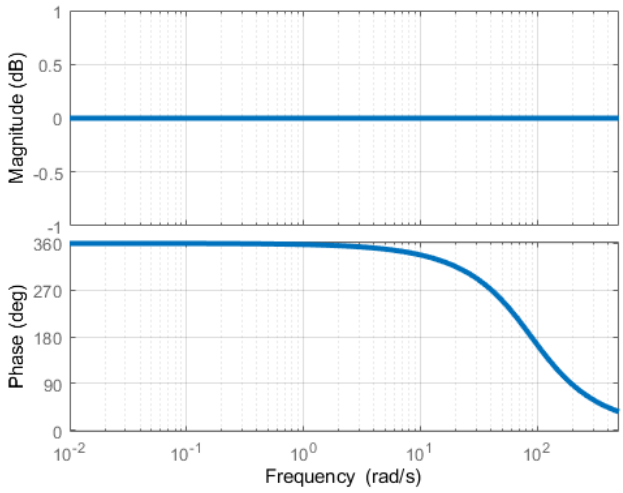

Figure 5.4: Bode plot from the δ_{cmd} to $\delta_{delayed}$.

5.6. LTI Parameter Values

This section provides the LTI models values at the flight point $t = 49$ s, which will be used for the controller design. A detailed discussion of the flight point selection for controller synthesis will be covered in the next chapter. However, by presenting the corresponding LTI parameter values here, all necessary information to reproduce the models is included within this chapter.

Table 5.1 presents the nominal LTI parameter values at $t = 49$s along with their associated uncertainties. The nominal parameter values and the uncertainty ranges for the flexible parameters are representative of Europe's small launcher (VEGA). Meanwhile, the uncertainties of the non-flexible parameters were defined following the work of *Simplício* (2019): 20% for aerodynamic parameters, 2% for mass, inertia, and center of mass, and 10% for the rest.

Table 5.1: LTI Parameters and Uncertainty Ranges.

Variable	Symbol	Units	Nominal Value	Uncert.
Time Delay	τ	s	0.039	$\pm 25.64\%$
Dynamic pressure	$\tilde{q}$	kPa	52897	$\pm 20\%$
Normal LV Force Gradient	C_{N_α}	-	2.2298	$\pm 20\%$
Thrust Force	T	kN	2489.3	$\pm 10\%$
Vehicle Velocity with respect to the ground	V_{rel}	m/s	697.92	$\pm 10\%$
Trim Velocity	a_0	m/s^2	6.5870	$\pm 10\%$
Longit. CP Coordinate	x_{CP}	m	18.3	$\pm 10\%$
Actuator Frequency	ω_a	rad/s	67.8	$\pm 10\%$
Actuator Damping Ratio	ζ_a	-	0.67	$\pm 10\%$
Distance between Nozzle's CM and CT	l_N	m	0.3	$\pm 10\%$
Longit. CM Coordinate	x_{CM}	m	9.07	$\pm 2\%$
LV Mass (total)	m_0	kg	87279	$\pm 2\%$
Normal LV MoI	I_{yy}	kg $\cdot$ m^2	$45.24{\cdot}10^5$	$\pm 2\%$
Nozzle Mass	m_N	kg	1616	$\pm 2\%$
Nozzle Inertia Moment	I_N	kg $\cdot$ m^2	1263	$\pm 2\%$
Long. IMU Coordinate	x_{IMU}	m	24.07	$\pm 2\%$
Reference LV Area	S_{ref-z}	m	7.14	0%
Longit. CT Coordinate	x_{CT}	m	0.96	0%
1st BM Frequency	ω_1	rad/s	32.8969	$[-20; 15]\%$
1st BM Damping	ζ_1	-	0.008	0%
Translational Length (x_{CT}) 1st BM	$\Psi_1(x_{CT})$	$1/$ sqrt (kg)	$6.0534 \cdot 10^{-3}$	$[-40; 30]\%$
Rotational Length (x_{CT}) 1st BM	$\Psi_1'(x_{CT})$	$1/$ sqrt (kg)	$-1.2118 \cdot 10^{-3}$	$[-40; 30]\%$
Translational Length (x_{IMU}) 1st BM	$\Psi_1(x_{IMU})$	$1/$ sqrt (kg)	$10.24 \cdot 10^{-3}$	$[-40; 30]\%$
Rotational Length (x_{IMU}) 1st BM	$\Psi_1'(x_{IMU})$	$1/$ sqrt (kg)	$2.0926 \cdot 10^{-3}$	$[-40; 30]\%$
2nd BM Frequency	ω_2	rad/s	74.8164	$[-25; 20]\%$
2nd BM Damping	ζ_2	-	0.009	0%
Translational Length (x_{CT}) 2nd BM	$\Psi_2(x_{CT})$	$1/$ sqrt (kg)	$2.8436 \cdot 10^{-3}$	$[-10; 20]\%$
Rotational Length (x_{CT}) 2nd BM	$\Psi_2'(x_{CT})$	$1/$ sqrt (kg)	$-1.3219 \cdot 10^{-3}$	$[-10; 20]\%$
Translational Length (x_{IMU}) 2nd BM	$\Psi_2(x_{IMU})$	$1/$ sqrt (kg)	$-5.7585 \cdot 10^{-3}$	$[-10; 20]\%$
Rotational Length (x_{IMU}) 2nd BM	$\Psi_2'(x_{IMU})$	$1/$ sqrt (kg)	$-2.3658 \cdot 10^{-3}$	$[-10; 20]\%$

6

$\mathcal{H}_\infty$ Closed Loop Shaping vs $\mathcal{H}_\infty$ Open Loop Shaping

This chapter presents the comparison between $\mathcal{H}_\infty$ Open Loop Shaping and $\mathcal{H}_\infty$ Closed Loop Shaping, as techniques used for the design of the attitude controller for the atmospheric ascent of a Launch Vehicle.

It starts with an analysis of the system's variability, as well as the definition of the analysis point in Section 6.1 and 6.2, respectively. Subsequently, the requirements for the TVC control system and the tests that will be used for the different controller comparisons, as well as to verify the requirements are defined in Section 6.3. Meanwhile, Section 6.4, contains an analysis of the rigid body model.

The next three sections entail three different controller designs. Firstly, in Section 6.5, a simple pole placement controller is designed. Secondly, in Section 6.6, a controller is designed with the use of $\mathcal{H}_\infty$ Closed Loop Shaping. Finally, in Section 6.7, a controller is designed with the use of $\mathcal{H}_\infty$ Open Loop Shaping. Each section provides a detailed explanation of the controller design process. For the Closed Loop Shaping method, this includes the different sets of constraints tested and uncertainty-aware design approaches. For the Open Loop Shaping method, it covers the newly developed weighting filter parameterization, the guidelines for applying $\mathcal{H}_\infty$ OLS to Lvs, and the allocation of requirements to specific frequency ranges.

In Section 6.8, the three controllers are compared in terms of performance, stability and robustness. This comparison raised questions regarding the impact of controller order, leading into the subsequent section. Section 6.9 presents the synthesis of higher-order $\mathcal{H}_\infty$ Closed Loop Shaping controllers that match the order of the $\mathcal{H}_\infty$ Open Loop controller. It explores which technique performs best when both controllers are of the same order.

Next, Section 6.10 summarizes the author's experience with each $\mathcal{H}_\infty$ method, highlighting the practical advantages and disadvantages of each approach. Finally, Section 6.11 encompasses the answers to Research Question 1 and its sub-Research Questions, as well as the chapter's conclusion.

6.1. System Variability Analysis

The goal of this section is to showcase the great variability inherent to the launch vehicle system. For that purpose, some of the most important parameters of the system were plotted either against time, or against time varying parameters.

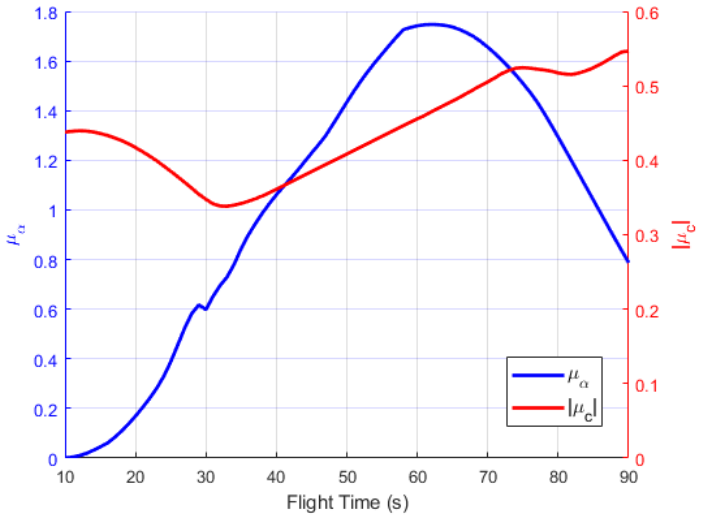

Figure 6.1: Time evolution of μ_α and μ_c (see eq. 5.47).

Figure 6.1 illustrates the scaled evolution of two key parameters, which govern the rotational motion of the launch vehicle through the following relation $\ddot{\theta} = \mu_\alpha \theta - \mu_c \delta$. Several observations can be made from this plot.

Firstly, both parameters exhibit significant variation over time, with μ_α in particular rising sharply from 0.2 to nearly 1.8 between 20 and 60 seconds. Secondly, since μ_α primarily dictates the position of the system's rightmost pole, as demonstrated in the simplified design model, it is evident that the system's instability will peak between 55 and 65 seconds. Furthermore, it is evident that the system remains unstable for the entire duration of the atmospheric flight, given the positivity of μ_α.

Figure 6.2 shows the variation of dynamic pressure against Mach number. It becomes apparent that the dynamic pressure increases up to approximately Mach 2.3. Beyond this point, despite the continued increase in Mach number, the dynamic pressure begins to decrease. This phenomenon can be explained

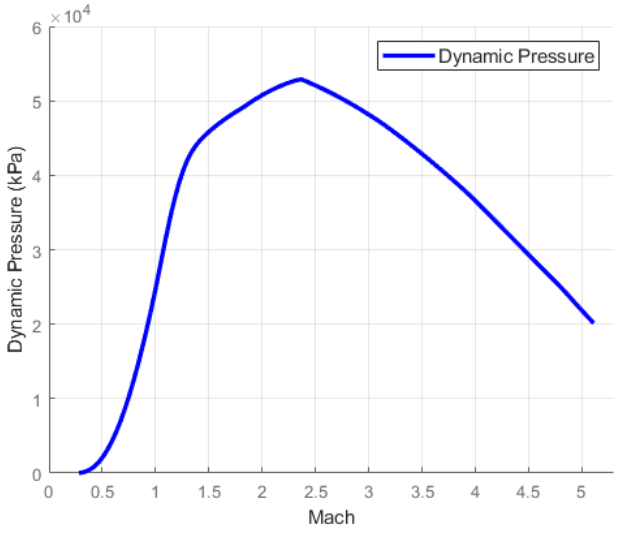

Figure 6.2: Dynamic Pressure vs Mach Number.

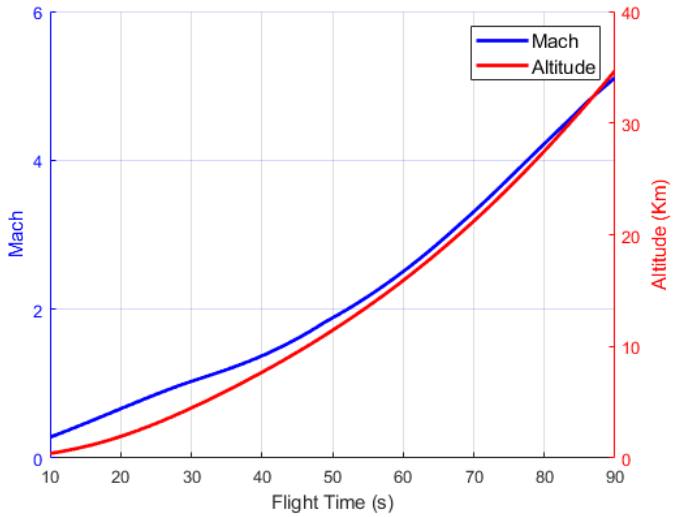

Figure 6.3: Mach and Altitude vs Flight Time.

by the fact that, at the moment the launch vehicle reaches Mach 2.3, the decrease in air density starts to outweigh the increases in the LV's velocity, leading to a subsequent decrease in dynamic pressure. It is worth noting that after the dynamic pressure begins decreasing, so will the structural load experienced by the LV. As it will be seen in Section 6.3, load is a function of $\tilde{q}\alpha$, and so the point where maximum dynamic pressure is achieved, is of particular interest. This peak in dynamic pressure occurs around $t = 49$ s, a critical moment, as the rightmost pole of the system also approaches its maximum norm, intensifying the control challenge. As discussed in *Skogestad* and *Postlethwaite* (2005) (pages 195-197), the higher the norm of a real right-hand-plane pole (as is the case for the aforementioned pole), the higher the lower limit placed on the bandwidth of the complementary sensitivity transfer function, meaning the roll-off of this transfer function must occur at a higher frequency.

The final two plots (Figure 6.3 and 6.4) aim at once again showing the time varying nature of the launch

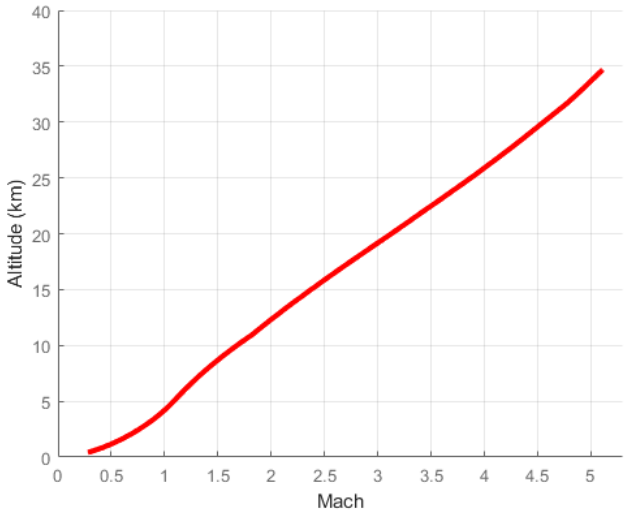

Figure 6.4: Mach vs Altitude.

vehicle system. In a mere 80 seconds, the vehicle mach number goes from being nearly 0 to 5. In the same time-frame, an altitude climb of roughly 35 km is also experienced. The relationship between mach and altitude is best visualized in figure 6.4.

6.2. Analysis Point Definition

In this thesis, the focus is on comparing two control design methods and introducing a new integrated design approach. To facilitate this comparison, the study will concentrate on a single design point that presents some of the most challenging requirements for the launch vehicle. This focus allows for a thorough evaluation of how well each controller design approach performs.

The selection of the design point is not trivial. The launch vehicle is a highly time-varying system, and different phases of atmospheric flight present distinct control challenges. For instance, in the early stages of launch, tracking accuracy is crucial, whereas during the phase of maximum dynamic pressure, the focus shifts to load relief. Nonetheless, the design of the controllers will focus on the point corresponding to the highest dynamic pressure, specifically $t = 49$ s. In this point, the significant dynamic pressure makes it particularly challenging to meet the load relief requirement. Additionally, robustness to uncertainty is crucial, as, for instance, load is a function of dynamic pressure, which has an uncertainty range of $\pm\, 20\%$. This uncertainty can severely affect the maximum allowable angle of attack (AoA). If the system is not robust to uncertainty, it may lead to violations of the load requirements. Furthermore, the rightmost pole of the system is near its maximum norm, making stabilization difficult. It is important to note that the location of this pole is also subject to considerable uncertainty, as it is primarily dictated by μ_α, which has an uncertainty range of approximately $[-50\%; +70\%]$ around its nominal value.

6.3. Requirements and Controller Tests

Before defining the requirements and the controller tests, one key remark is made. From here on, it will be assumed that, during the atmospheric phase, the LV utilizes open loop guidance. If the guidance system utilized a closed loop configuration it would interfere with the strategy of load alleviation, as load alleviation requires deviations from the reference trajectory. This assumption will make it so that a requirement on drift and drift rate will have to be established so as to limit trajectory deviations.

6.3.1. Requirement Definition

The set of requirements used in this thesis builds on those defined in *Tapia* (2019), *Simplício* et al. (2016) and *Díaz* (2010). In the first study, a comprehensive set of requirements for the TVC control system during the atmospheric phase of the VEGA launch vehicle is established, providing an excellent example of the stringent demands typical in launch vehicle TVC control systems. The second study outlines similar requirements for the same purpose. Notably, two key requirements from this latter work, related to tracking performance and disturbance rejection, have been incorporated into this thesis and will be applied in the analysis. Finally, a stability requirement for the nominal phase margin of the LV was taken from the third study.

The final set of requirements, as outlined in Table 6.1, establishes both quantitative and qualitative criteria for the LV's TVC control system, specifically for the atmospheric flight phase. Quantitative requirements include stability criteria, maximum allowable drift rate, and maximum actuation angle, while qualitative requirements encompass actuator usage minimization, command tracking, and disturbance rejection. Additionally, flight point specific requirements are outlined, notably the maximum allowable load

and maximum allowed drift. The drift requirement is based on *Tapia* (2019), which specifies a maximum expected drift value for VEGA at a flight point closely resembling the one analyzed in this study.

Table 6.1: Stability and performance requirements for the TVC control system.

	Requirements	Criterions	Bounds
Stability	Rigid body margins	Low Frequency (LF) Gain Margin (GM)	Nominal ≥ 6 dB
			Dispersed ≥ 0.5 dB
		High Frequency (HF) GM	Nominal ≤ -6 dB
			Dispersed ≤ -3 dB
		Phase Margin (PM)	Nominal ≥ 30 deg
			Dispersed ≥ 20 deg
	Flexible body margins	GM_f	Nominal ≤ -3 dB
			Dispersed ≤ -3 dB
		PM_f	Nominal ≥ 30 deg
			Dispersed ≥ 20 deg
Performance	Load	$\tilde{q}\alpha$	$< 2.75 \cdot 10^5$
	Lateral	z $\dot{z}$	< 167 m < 15 m/s
	Actuation	δ_{pitch} Actuator usage should be minimized	< 6.5 deg
	Tracking	TR1 (See below the table)	
	Disturbance Rejection	DR1 (See below the table)	

where the tracking and disturbance rejection requirements are defined as follows:

TR1 - The controller must be capable of tracking pitch angle commands, achieving zero steady-state error while constraining the transient response in terms of overshoot and maximum rate.

DR1 - The controller must be able to reject external disturbances (e.g. wind gusts) and internal dynamics (bending modes).

The requirements in Table 6.1 can be further categorized into stability and performance criteria. The stability requirements address both rigid and flexible body behavior, incorporating classical stability margins (gain and phase) for nominal and perturbed conditions.

Notably, the phase margin requirement for dispersed conditions is not derived from prior research but was specifically chosen to ensure sufficient relative stability across a broad range of operating scenarios. The perturbed conditions are characterized by the uncertainties outlined in Section 5.6, with the methods for evaluating these conditions described in a subsequent section.

Following the work of *Tapia* (2019), the controller is deemed to fulfill the stability requirements if the stability margins at the plant input meet the specified metrics. These metrics are most effectively visualized through a Nichols plot, as illustrated in Figure 6.5. Nevertheless, additional stability margins — such as loop-at-a-time plant output, MIMO plant output and plant input/output — remain valuable for comparison and provide further insights into the controller's robustness. Therefore, these margins will also be considered for the comparison of the different controllers produced.

The performance requirements for the launch vehicle thrust vector control system are divided into five categories. The first one, the load requirement, ensures that the loads applied to the LV remain below the

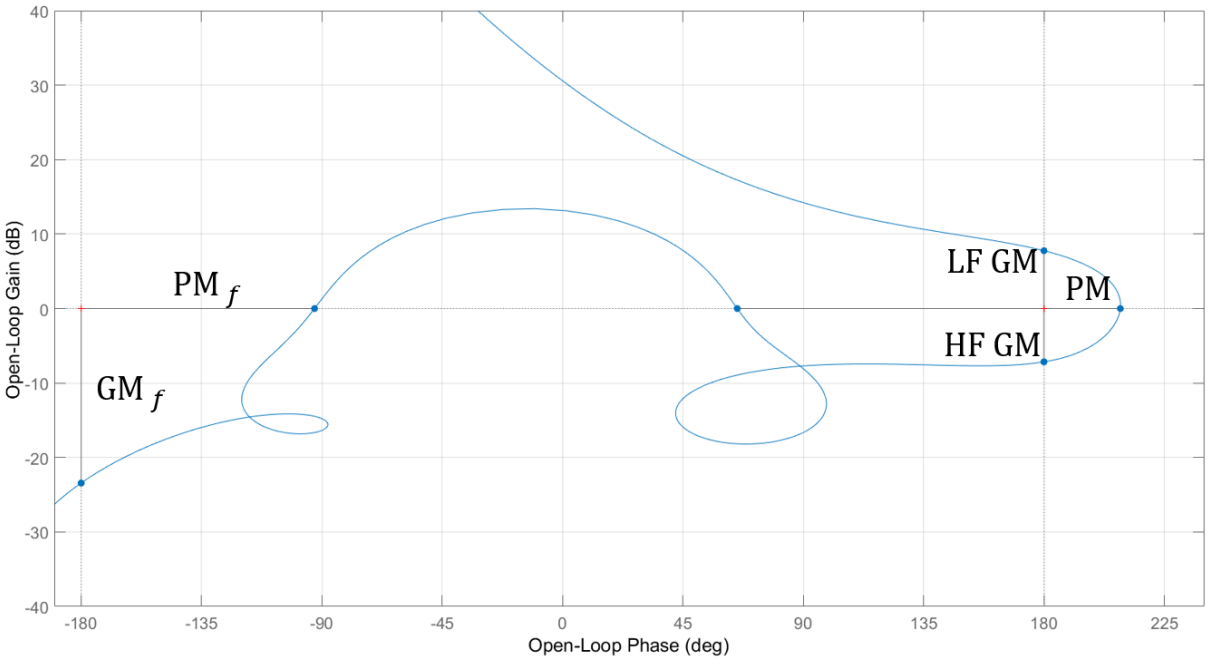

Figure 6.5: Launch vehicle stability margins requirements.

maximum allowable value. This requirement is expressed as a function of $\tilde{q}\alpha$, which is hereafter referred to as the load performance metric (LPM). As seen in *Tapia* (2019), the angle of attack is particularly sensitive to wind disturbances, highlighting the need for the control system to be robust against wind gusts in order to fulfill the load requirement.

Lateral requirements aim to bound the drift (z) and drift rate ($\dot{z}$) of the LV, which is especially critical due to the assumption that the LV employs open loop guidance. Actuation requirements, on the other hand, focus on limiting the maximum deflection angle (δ_{pitch}) and minimizing actuator usage. Tracking requirements ensure the controller can effectively track attitude commands while maintaining control over specific transient response parameters.

Finally, the controller must be capable of rejecting disturbances, with a primary emphasis on external disturbances such as wind gusts. Another type of disturbance to consider is the bending modes; however, their rejection will only be addressed in chapter 7.

6.3.2. Test Definition for Controller Comparison and Requirement Verification

The comparison between controllers and the controller requirement verification include several key tests:

- **Stability Margins:** Assessment of nominal and worst-case stability margins across various analysis points - plant input, loop-at-a-time (LAT) plant output, plant MIMO output, and plant input/output (I/O).

The input and loop-at-a-time stability margins will be visualized through Nichols plot. An example is given in Figure 6.6. This Nichols plot contains two lines, one representing the nominal Nichols (blue line), and one representing the Nichols correspondent to the worst-case S-T disk margin (red line). The red and blue dots represent classical gain and phase margins. Meanwhile, the blue and red disks represent the nominal S-T disk margin and the worst-case S-T disk margin, respectively. For the defined range of uncertainties, the worst-case disk margin is a portrayal of the region where the Nichols will never enter.

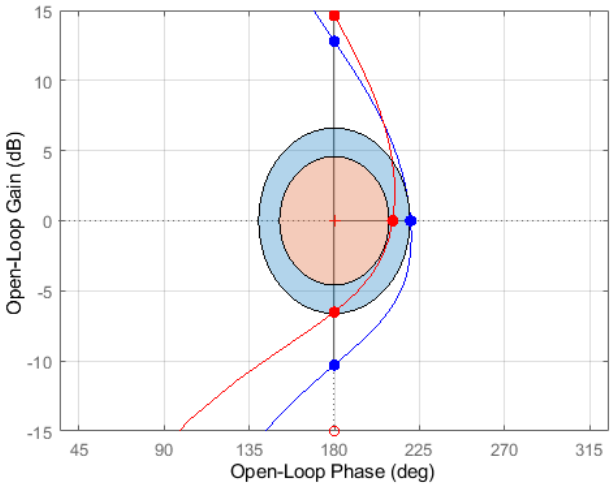

Figure 6.6: Example stability margin representation through nichols plot.

As discussed in Section 4.8, nominal disk margins are computed by applying a disk-based uncertainty to the nominal loop transfer function and determining how large that uncertainty can be while still preserving closed loop stability. On the other hand, the worst-case disk margin is computed by applying a disk-based uncertainty to the loop transfer function, combining it with the specified plant uncertainty, and using the Structured Singular Value (see Subsection 4.7.2) to identify the largest disk that ensures system stability.

- **Time Domain Performance:** Evaluation of three specific scenarios, based on the work of *Simplício (2019)*:
  - Evaluation of the impact of an external low frequency, high amplitude (LFHA) wind gust. This wind gust is characterized by a maximum peak of $40\,\mathrm{m/s}$ and has a shape typically observed in launch vehicle control system validation. The input wind signal for this test is shown in Figure 6.7.
  - Evaluation of the impact of an external high frequency, low to mid amplitude (HFLMA) wind

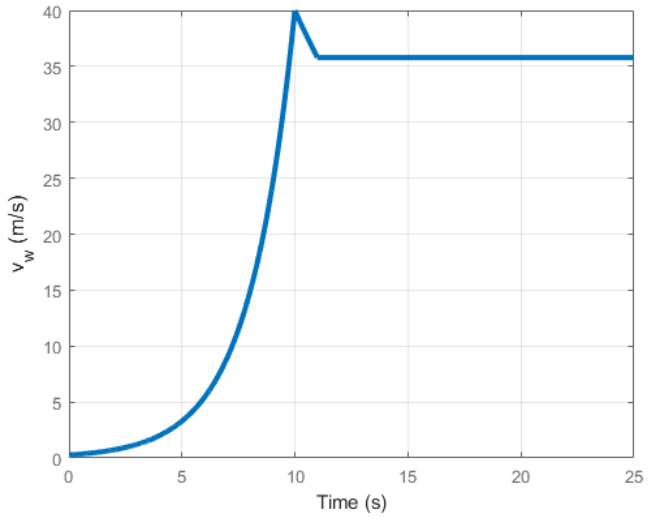

Figure 6.7: Low frequency, high amplitude wind input signal.

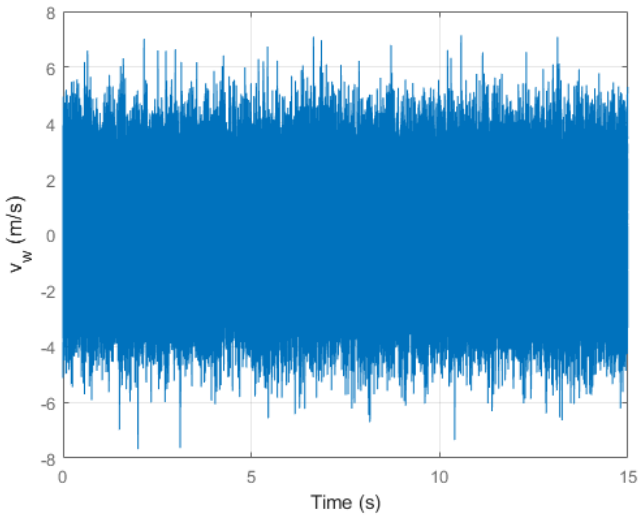

Figure 6.8: High frequency, low-to-mid amplitude wind input signal.

disturbance. The input signal is generated using a white noise signal, with the same seed applied each time to ensure consistent and reproducible results. An example input signal is shown in Figure 6.8.

- Evaluation of the system's response to a pitch angle tracking command of 0.5 degrees. This test uses a step input signal on the pitch angle command input with an amplitude of 0.5 degrees to assess the tracking performance.

The input signals used in the first two cases were chosen to align with the inputs used in *Simplício* (2019), facilitating verification of results. The tracking case has a specific implementation detail: following the methodology outlined in *Sánchez de la Llana* (2015), both drift and drift rate feedback were nulled during this case to enable proper testing of tracking. A detailed justification for this is given in the next section. Finally, it should be noted that all time-domain cases are repeated for 100 randomly sampled conditions [1].

While the time-domain cases provide insights into the performance of the controller in dispersed conditions, a more thorough evaluation could involve either testing additional randomly sampled cases or repeating the analysis in the corner cases. In this work, a different approach was employed so as to strengthen the analysis:

- **Worst-Case Gain Analysis:** This involves computing the highest gain in selected transfer functions with MATLAB's wcsigmaplot. This function utilizes the results presented in *Packard* et al. (2000) to compute the worst-case gain of a transfer function by evaluating the structured singular value (SSV) for an appropriate block structure. It then generates a plot of the system's worst-case gain, in other words, the highest possible gain, across the specified frequencies.

Through computing the worst-case gain of particular transfer functions it will be possible to further extend the comparison and testing of the produced controllers. For example, calculating the worst-case gain from the wind input to the load performance metric output ($\tilde{q}\alpha$) will help determine the maximum wind magnitude that ensures the load relief requirement is not violated. Finally, one more test is included:

[1] The choice for the number of simulations reflects a balance between computational burden and the need to adequately demonstrate the effect of dispersed conditions. It is important to note that this number of samples might not be sufficient for a thorough evaluation of the controller, however, that is where the worst-case gain analysis comes in.

- **Structured Singular Value (SSV) Analysis:** This analysis focuses on evaluating the robustness of the system against structured uncertainty (see Subsection 4.7.2).

These tests collectively provide a comprehensive comparison of the controllers' performance and robustness.

6.4. Rigid Body, Delay and TVC Actuator Models Analysis

The goal of this section is to implement and analyze the rigid body model, the delay model and the TVC actuator model, and perform an initial analysis on them. First, in Subsection 6.4.1, an analysis on the open loop rigid body analytical transfer functions is performed. Subsequently, in Subsection 6.4.2 the open loop system is implemented in SIMULINK and analyzed. Finally, in Subsection 6.4.3, the generalized plant, along with the control structure, are defined in preparation for the $\mathcal{H}_\infty$ controller synthesis. Additionally, the closed loop system is implemented in SIMULINK, and a key feature of the launch vehicle's closed loop model, which is crucial for controller analysis, is highlighted.

6.4.1. Open Loop Analytical Transfer Functions

The first step in analyzing the launch vehicle model was to derive the SISO transfer functions corresponding to the rigid body analysis model, AMR, which was discussed in Subsection 5.3.1. These can be found in Appendix A. The most important conclusions that can be drawn from these transfer functions are summarized next:

- Tail-Wag-Dog effect introduces two zeros per output. For the output $\tilde{q}\alpha$ one of them is non-minimum phase, for the outputs z_{IMU} and $\dot{z}_{IMU}$ both are non-minimum phase. These non-minimum phase zeros will impose limitations: inverse response behavior, high gain instability, and above all, bandwidth limitation (see *Skogestad* and *Postlethwaite* (2005), pages 185-193).
- The transfer functions that contain non-minimum phase zeros are: v_w to z_{IMU}, δ_{pitch} to z_{IMU}, $\ddot{\delta}_{pitch}$ to z_{IMU}, v_w to $\dot{z}_{IMU}$, δ_{pitch} to $\dot{z}_{IMU}$, $\ddot{\delta}_{pitch}$ to $\dot{z}_{IMU}$, δ_{pitch} to $\tilde{q}\alpha$ and $\ddot{\delta}_{pitch}$ to $\tilde{q}\alpha$.

6.4.2. Open Loop Model

The next step in this analysis was to implement the open loop model, which, specifically for this analysis, includes the rigid body analysis model (AMR), the delay model, and the TVC actuator model. To achieve this, a SIMULINK model was created and initialized with parameters corresponding to flight point $t = 49$ s. A representation of the model can be seen in Figure 6.9, where the block G_{LV} is given by the rigid body analysis model (AMR), as discussed in Subsection 5.3.1.

The SIMULINK model of the open loop system allows for the computation of the open loop poles, which are shown in table 6.2. It can be seen, that the launch vehicle is clearly unstable: it possesses two RHP poles. The rightmost pole is roughly at the location of $\sqrt{\mu_\alpha}$, meanwhile the other RHP pole, which is much slower, is caused by the translational motion of the LV. There are two other poles stemming directly from the rigid behavior of the launch vehicle, one at the origin and one located, roughly, at - $\sqrt{\mu_\alpha}$. The TVC actuator model introduces two poles with a frequency of $67.8\,\mathrm{rad/s}$, the delay model introduces another two poles, with a frequency of $88.8\,\mathrm{rad/s}$.

Finally, the magnitude plot of the nominal Bode plot from δ_{cmd} to θ_{IMU} is shown in Figure 6.10. This figure helps illustrate the impact of the right-half-plane zeros introduced by the TWD effect, which is evident in the Bode plot as a dip and a change in the roll-off rate at high frequencies.

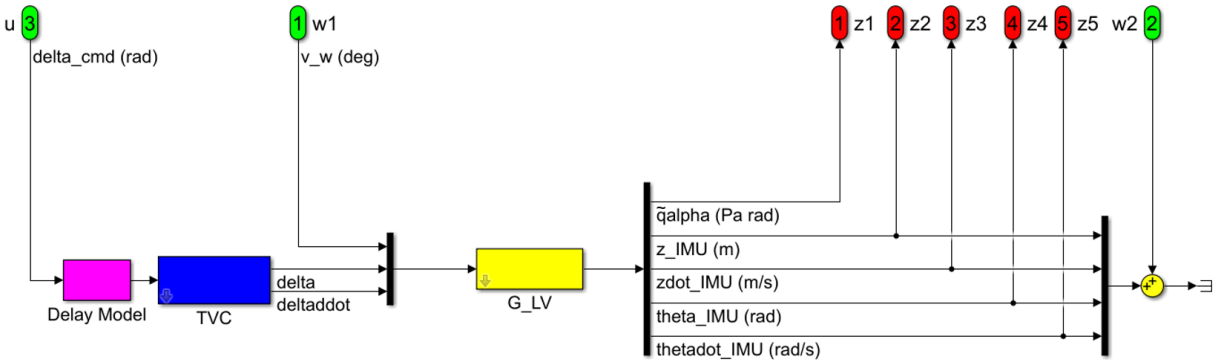

Figure 6.9: Simulink model of the open loop system.

Pole	Damping	Frequency (rad/s)	Time Constant (s)
0.00	-1.00	0.00	∞
$9.44 \cdot 10^{-3}$	-1.00	$9.44 \cdot 10^{-3}$	$-1.06 \cdot 10^{2}$
1.29	-1.00	1.29	$-7.75 \cdot 10^{-1}$
-1.34	1.00	1.34	$7.48 \cdot 10^{-1}$
$-45.4 + 50.3i$	0.67	67.80	$2.20 \cdot 10^{-2}$
$-45.4 - 50.3i$	0.67	67.80	$2.20 \cdot 10^{-2}$
$-76.9 + 44.4$	0.87	88.80	$1.30 \cdot 10^{-2}$
$-76.9 - 44.4$	0.87	88.80	$1.30 \cdot 10^{-2}$

Table 6.2: Open loop poles.

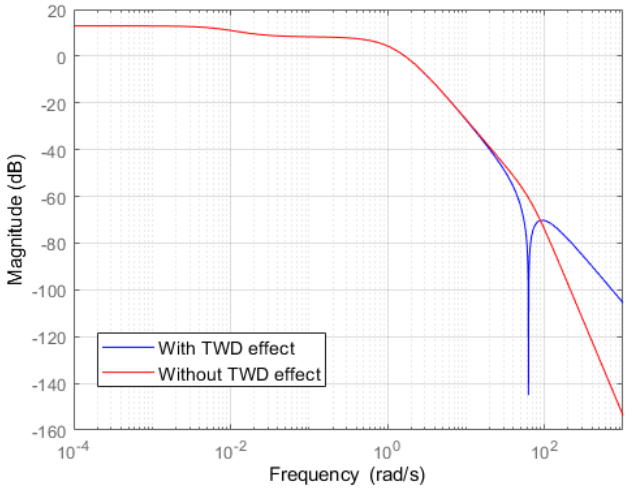

Figure 6.10: Magnitude part of the Bode plot from δ_{cmd} to θ_{IMU}.

6.4.3. Generalized Plant Definition and Implementation

In this work, the generalized plant's exogenous input signals consist of the wind disturbance input, sensor noise, and tracking command (pitch angle command - θ_{cmd}), denoted as $w1$, $w2$, and r, respectively. The plant contains a set of outputs called regulated outputs, they will be denoted as $z1 - z7$. Respectively, they represent the load performance metric ($\tilde{q}\alpha$), drift at the IMU location (z_{IMU}), drift rate at the IMU location ($\dot{z}_{IMU}$), pitch angle at the IMU location (θ_{IMU}), pitch rate at the IMU location ($\dot{\theta}_{IMU}$)[2], commanded TVC deflection (δ_{cmd}) and pitch angle error (e_θ). These were selected as the controlled outputs so as to be able to enforce the different requirements. The correspondence between each controlled output and the respective requirement will become clear during the discussion of the constraints utilized in the $\mathcal{H}_\infty$ CLS design. The generalized plant includes the commanded TVC deflection (δ_{cmd}) as the control variable (u) and the variables that the IMU measures - drift at the IMU location, drift rate at the IMU location, pitch angle at the IMU location, and pitch rate at the IMU location - as the sensed outputs (v).

Going forward, the subscript "IMU" will indicate quantities evaluated at the IMU's position, without always explicitly stating "at the location of the IMU".

The control law used to close the loop is based on the classic atmospheric flight TVC law for the VEGA launcher (see pages 30-32 of *Tapia* (2019)). The control law includes a PD controller on attitude, so as to stabilize and control the launch vehicle, and a PD controller on drift, so as to limit drift and minimize angle of attack (load relief). The classic VEGA flight control law also include a set of filters, which aim at improving rigid body stability margins, enforcing high-frequency attenuation on the drift/drift rate channels and performing a derivative action to compute the attitude rate signal. However, these will be excluded in order to simplify the design. The control law is shown next:

$$u = k_{p_\theta}\theta_{IMU} + k_{d_\theta}\dot{\theta}_{IMU} + k_{p_z}z_{IMU} + k_{d_z}\dot{z}_{IMU} \tag{6.1}$$

This control law will be tuned in the following sections with the use of pole placement and with the use of $\mathcal{H}_\infty$ CLS. For the section on $\mathcal{H}_\infty$ OLS, it will act as the robustifying controller.

With this, the closed loop model can be implemented in SIMULINK. Figure 6.11 illustrates the scheme used for the control design of both the $\mathcal{H}_\infty$ CLS and pole placement controllers. It is important to note that in this implementation, the block G_{LV} is given by DMR, the rigid body robust design model, which was described in Subsection 5.3.2.

Additionally, in Figure 6.11, all inputs and outputs are defined according to the generalized plant formulation. The gains applied to the input channels serve two functions: they scale the inputs based on their maximum expected values and convert degrees to radians where necessary. For the attitude command, a scaling factor of 1 deg is used; for the wind disturbance, a scaling factor of $33\,\mathrm{m/s}$ is applied; and for the sensor noise, the gain matrix is given by diag(0.02 deg, 0.1 deg/s, 0.01 m, 0.001 m/s). All of these values, except for the wind input scaling, were directly sourced from the input scaling used for $\mathcal{H}_\infty$ CLS in *Tapia* (2019), and they represent the maximum expected variation of each input.

The wind disturbance scaling factor corresponds to the highest wind speed recorded in a set of real measurements (within the altitude range of $0-20\,\mathrm{km}$) taken at the French Guiana, the Vega launch site, as documented on page 68 of *Tapia* (2019). Finally, output gains are applied to convert radians to degrees when necessary.

[2]The subscript "IMU" in these outputs indicates that the variables are evaluated at the IMU's position, accounting for its location relative to the center of mass, as described by the rigid body terms in eqs. 5.36 through 5.39.

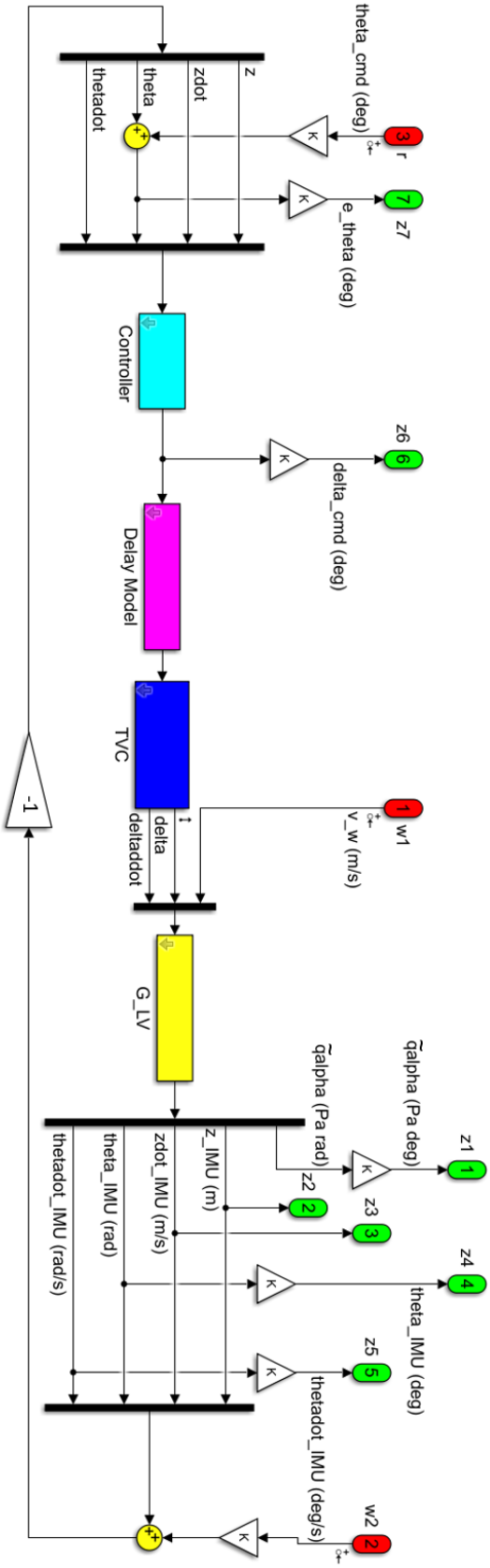

Figure 6.11: SIMULINK model of the closed loop control design scheme for $\mathcal{H}_\infty$ CLS and pole placement.

The specified scaling serves two main purposes: first, it converts the closed loop inputs and outputs into more convenient units (e.g., degrees instead of radians); and second, it facilitates the $\mathcal{H}_\infty$ CLS design. By selecting input scaling based on the maximum expected variation, the designer ensures that the $\mathcal{H}_\infty$ CLS constraints are applied for the scenario where the inputs reach their maximum expected values, thus providing the constraints with a clear physical interpretation. For instance, by applying this scaling, constraining the $\mathcal{H}_\infty$ norm of the transfer function from v_w to z_{IMU} to be smaller than 200 directly translates into ensuring that the system limits drift to below 200 m under the maximum expected wind conditions (for the nominal system).

One final remark needs to be made regarding the closed loop system. It was shown by *Tapia* (2019), analytically, that the double PD control law comes with constraints on the performance achieved by the controller. Namely, it was shown that by using this control law, the steady state gain at high and low frequencies of the transfer function from attitude command to attitude error (sensitivity function) is fixed to one. A physical explanation on this can be given: when the guidance system issues a command on pitch angle, the pitch feedback will take over the controller action. During this first instance, the controller will issue deflection commands leading to the tracking of the pitch angle command. This will lead to a drift motion of the LV, and eventually, the drift feedback will take over the controller action. The new controller commands will be issued so as to decrease drift, this will make it so that the pitch angle will start decreasing. Eventually, the conflicting orders caused by the drift error and pitch error will even out. The drift will stop increasing, and the pitch error will remain in a steady state value of zero degrees. This happens because, in the closed loop system, the drift reference is not adjusted when an attitude command is issued to the launch vehicle. When an LV is actually flying, this would not be the case, as the drift measurements would account for changes caused by the pitch commands, allowing the drift reference to be adjusted accordingly.

Nonetheless, it is crucial to be able to test the tracking capabilities of the LV with the closed loop system. For this purpose, the strategy used by *Sánchez de la Llana* (2015) will be applied: the drift and drift rate feedbacks are disconnected for the tracking tests. This will allow for a simplified tracking test, in which it is not possible to study the evolution of drift/drift rate. While this is a limitation of the results presented in this thesis, the consequences are small as drift and drift rate are evaluated in other time-domain tests, such as wind disturbance rejection tests. If the system can maintain drift and drift rate within acceptable bounds during high amplitude wind disturbance tests (which are expected to cause large variations in drift and drift rate), it is reasonable to expect that it will also maintain drift and drift rate within acceptable limits in the tracking tests. In these tests, only small tracking commands (maximum one degree) are issued, and the resulting disturbances are relatively minor, leading to smaller variations in drift and drift rate.

6.5. Pole Placement

The pole placement controller was designed using Matlab's systune and the pole placement goal. The model used for control design is illustrated in Figure 6.11, where the block G_{LV} contains the rigid body robust design model (DMR), as described in Subsection 5.3.2. It is worth mentioning that systune was specifically chosen to design the pole placement controller to allow the author to gain familiarity with the tool.

For the pole placement design, it was specified that the damping of the poles should be greater than 0.7 (a value normally used for aerospace control systems) and that the minimum decay frequency should be 0.75 rad/s. The minimum decay frequency was determined through a process of trial and error. First, a value was tested and used to compute a controller. Then, the worst-case stability input margin was tested.

If the stability margin failed to meet the requirements, the minimum decay frequency was reduced, and the process was repeated. With the selected minimum damping and decay frequency the following controller, which was named K_{PP}, was computed:

$$\begin{aligned} K_{p_\theta} &= -1.3628 \\ K_{d_\theta} &= -0.5884 \\ K_{p_z} &= 0.0053 \\ K_{d_z} &= 0.0218 \end{aligned} \tag{6.2}$$

This concluded the pole placement design. Once again it is important to emphasize that the goal of this controller is simply to serve as baseline.

6.6. $\mathcal{H}_\infty$ Closed Loop Shaping

The design of the $\mathcal{H}_\infty$ Closed Loop Shaping controller was carried out using systune, with the model used for control design depicted in Figure 6.11. In this model, the block G_{LV} contains the rigid body robust design model (DMR), as described in Subsection 5.3.2.

This design process followed three main steps. First, the constraints required for closed loop shaping were analyzed. Next, the specific values for these constraints were selected for controllers designed using three approaches: the Nominal Design (NOM), which utilized only the nominal model during optimization; the LFT Design (LFT)—a name used in this thesis to refer to the robust parametric control design method developed in *Apkarian* et al. (2015); and the Multi-model Design (MM), which used a Multi-model approach. The third step involved comparing all the controllers designed using the $\mathcal{H}_\infty$ CLS approach and determining which one should be used for comparison against the $\mathcal{H}_\infty$ Open Loop Shaping controller.

6.6.1. Closed Loop Shaping - Constraints

To determine the appropriate constraints for $\mathcal{H}_\infty$ Closed Loop Shaping, the following iterative process was used:

- **Start simple**: Begin with only two constraints, such as $||T_{v_w \to \tilde{q}\alpha}||_\infty < W$ and $\bar{\sigma}(T_{\theta_{cmd} \to e_\theta}) < X$.
- **Generate a controller**: Design an initial controller based on these constraints.
- **Plot singular value responses**: Plot all relevant singular value plots to examine the system's behavior.
- **Identify issues and refine constraints**: Identify any deficiencies in the controller behavior, add additional constraints, or adjust existing constraint values.
- **Repeat the process**: Iterate through this process, refining constraints and controller performance with each step. Stop when all deficiencies in the controller behavior are fixed.

where W and X represent static and dynamic constraints, respectively. $||T_{v_w \to \tilde{q}\alpha}||_\infty$ represents the $\mathcal{H}_\infty$ norm of the transfer function from input v_w to output $\tilde{q}\alpha$. $\bar{\sigma}(T_{\theta_{cmd} \to e_\theta})$ represents the maximum singular value of the transfer function from input θ_{cmd} to output e_θ.

With this approach it was possible to reach the following set of constraints:

$$\begin{aligned} &||T_{v_w \to \tilde{q}\alpha}||_\infty < W_1 \\ &||T_{v_w \to z_{IMU}}||_\infty < W_2 \\ &||T_{\theta_{cmd} \to \delta_{cmd}}||_\infty < W_3 \\ &||T_{\theta_{cmd} \to e_\theta}||_\infty < W_4 \end{aligned} \tag{6.3}$$

where W_1, W_2, W_3 and W_4 are static constraints that limit the $\mathcal{H}_\infty$ norm of the specified transfer functions.

Posteriorly, for reasons detailed in the next subsection, the constraint on $T_{\theta_{cmd}\to\delta_{cmd}}$ was revised to allow for either $||T_{\theta_{cmd}\to\delta_{cmd}}||\infty < W_3$ or $\bar{\sigma}(T_{\theta_{cmd}\to\delta_{cmd}}) < W_5$, where W_5 is a dynamic constraint.

Several remarks can be made regarding the final set of constraints. Subsection 6.6.2 details the relationship between launch vehicle requirements and each constraint. Additionally, this subsection contains the explanation as to why two alternatives are given for the constraint on $T_{\theta_{cmd}\to\delta_{cmd}}$. Subsequently, Subsection 6.6.3, and Subsection 6.6.4 address the unconventional nature of the constraint set and explore the additional constraints that were tested, respectively. These last two discussions are primarily of interest to readers who intend to perform $\mathcal{H}_\infty$ Closed Loop Shaping for a launch vehicle.

6.6.2. Constraint - Requirement Relation

The first constraint in eq. 6.3 ($||T_{v_w\to\tilde{q}\alpha}||_\infty < W_1$), directly addresses the load relief requirement by limiting the allowable amplification from the wind input to the load performance metric. It is worth noting that the $\tilde{q}\alpha$ output is particularly sensitive to wind disturbances.

The second constraint ($||T_{v_w\to z_{IMU}}||_\infty < W_2$) serves two purposes: it limits the amount of drift that can occur due to the external wind input, and indirectly constrains the maximum allowable drift rate caused by the same input. Furthermore, this constraint also indirectly restricts the drift and drift rate resulting from a pitch command input or from a sensor noise input. As a result, the second constraint is directly correlated with the requirements for drift and drift rate.

The third constraint ($||T_{\theta_{cmd}\to\delta_{cmd}}||_\infty < W_3$ or $\bar{\sigma}(T_{\theta_{cmd}\to\delta_{cmd}}) < W_5$) is implemented to limit actuation and is specifically placed on $T_{\theta_{cmd}\to\delta_{cmd}}$, as this transfer function is where actuation requirements were found to be most likely infringed. While this constraint does not limit the actuation caused by wind input, this is not a concern because the actuation due to external wind is typically much smaller.

As discussed in Subsection 6.6.3, for launch vehicles with the specific controller structure employed in this work, applying either dynamic or static shaping to $T_{\theta_{cmd}\to\delta_{cmd}}$ produces essentially equivalent results. Therefore, static shaping was deemed sufficient for this transfer function. However, for the Nominal Design controllers developed in Subsection 6.6.5, the constraint on $T_{\theta_{cmd}\to\delta_{cmd}}$ was aligned with the dynamic constraint used in *Tapia* (2019) to replicate VEGA-C's controller. While a static constraint could have yielded a similar outcome, determining such a constraint was unnecessary since the dynamic constraint was readily available. To address this, the formulation of the constraint on $T_{\theta_{cmd}\to\delta_{cmd}}$ was adapted to accommodate both dynamic and static shaping.

The fourth constraint ($||T_{\theta_{cmd}\to e_\theta}||_\infty < W_4$), henceforth referred to as sensitivity constraint, was selected to, indirectly, enforce stability margins. As is shown in page 37 of *Skogestad* and *Postlethwaite* (2005), the smaller the $\mathcal{H}_\infty$ norm of the sensitivity transfer function, the greater the lower bound on gain and phase margins. Therefore, this constraint directly relates to the stability margin requirements. Additionally, it relates to the tracking requirement, as it essentially limits the allowable tracking error.

Finally, the first three constraints directly relate to the disturbance rejection requirement. In essence, all the rigid body requirements are translated in the aforementioned set of constraints.

Additional constraints were added to the existing set; however, in all cases, this resulted in an over-constrained problem, rendering one of the constraints inactive. Thus, no more than four constraints were necessary. By observing which constraint became inactive when another was added, it was possible to evaluate whether replacing an existing constraint with a new one would be beneficial. Ultimately, it was determined that none of the original constraints should be replaced. A detailed discussion on this is provided in the third part of this analysis.

6.6.3. Static vs Dynamic Constraints

In many (if not most) $\mathcal{H}_\infty$ closed shaping works, the constraints used are a mix of dynamic (low-pass filters, high-pass filters, band-pass filters) and static (maximum singular value) - see for instance the works of *Hernandez* et al. (2021), *Tapia* (2019) for the case of LV. The reason being that dynamic constraints are used to shape the frequency response with attention to all frequencies, whilst static constraints simply push down the peak value of a transfer function.

Several transfer functions benefit from dynamic shaping. The most usual are S_o, T_o and input to actuation. The complementary output transfer function (T_o) is usually shaped as a low-pass filter to ensure noise attenuation and robustness to multiplicative uncertainty, at high frequencies. Similarly, the transfer function from inputs to actuation is often shaped as a low-pass filter to prevent actuation saturation, minimize high-frequency actuation and limit bandwidth. The output sensitivity function (S_o) is typically shaped as a high-pass filter to reduce steady-state tracking error and to position the sensitivity function's peak at higher frequencies (the waterbed effect).

However, for launch vehicles with the specific controller structure used in this work, dynamic shaping is not required for any of these transfer functions. The high and low frequency steady-state value of S_o are always fixed to one, due to the inclusion of drift feedback (as previously discussed); therefore, the high pass filter shape cannot be enforced and only S_o's peak value can be minimized.

On the other hand, testing has shown that dynamic constraints do not alter To's overall shape; they only adjust its peak value by shifting the entire curve up or down. Therefore, there is no reason for imposing a dynamic constraint rather than a static one.

Testing on input to actuation showed that the transfer function behaved similarly to T_o, with one key differences: the TF's singular value shape could be slightly modified. However, since the shape of the curve can only be minimally altered, dynamic shaping provides limited advantages, as a very similar result can be achieved using a static constraint.

Given that dynamic shaping was unnecessary for the above transfer functions — and other constrained transfer functions only required peak value limitations — only static shaping was applied. The exception was the Nominal Design controllers, where the actuation constraint was selected identical to the dynamic constraint used to reproduce VEGA-C's controller in *Tapia* (2019), as explained in the previous subsection.

6.6.4. Other Constraint Options

Other sets of constraints were tested and will be discussed here for completeness. Initially, to constrain drift and drift rate, several static constraints were tested on the following transfer functions: $T_{v_w \to z_{IMU}}$, $T_{\theta_{cmd} \to z_{IMU}}$, $T_{v_w \to z_{IMU}} + T_{\theta_{cmd} \to z_{IMU}}$, and $T_{v_w + \theta_{cmd} \to z_{IMU}}$. All options yielded similar results in limiting drift and drift rate, except for the final constraint. This constraint slightly worsened the wind to drift rejection, but improved the wind + pitch angle command to drift rejection by accounting for cross interactions between inputs. Ultimately, the decision was made to constrain the transfer function $T_{v_w \to z_{IMU}}$, as it directly addresses the input channel that can cause the drift and drift rate requirements to be infringed.

Another possibility that was tested was to replace the drift constraint with a drift rate constraint. However, this resulted in unbounded drift, as expected. Additionally, substituting the actuation constraint with a complementary sensitivity output constraint led to unbounded high-frequency actuation.

Another replacement test involved replacing the sensitivity constraint with a constraint applied to the transfer function $T_{\theta_{cmd} \to \dot{\theta}_{IMU}}$. This allowed for attaining the same results as those achieved with the original

set of constraints. The explanation lies in the fact that by constraining the attitude rate, the damping of the system's poles increases, reducing the system's response to perturbations, thereby achieving the same effect as limiting the sensitivity.

Finally, an alternative to the $\tilde{q}\alpha$ constraint was tested: constraint on $T_{wind+\theta_{cmd}\to\tilde{q}\alpha}$. This led to lower $\tilde{q}\alpha$ when considering cross interactions between inputs, but much worse results when considering only the wind input. Since the $\tilde{q}\alpha$ output is primarily influenced by the wind input, and load requirement violations are most likely to result from wind disturbances, this alternative was ultimately disregarded.

6.6.5. Selecting Values for the Constraints

After selecting the constraints, the next step was to determine how to use them optimally. The following discussion details the process and rationale behind designing various controllers. Three different approaches were used to design the controller: the Nominal Design (NOM), which utilized only the nominal model during optimization; the LFT Design (LFT), which employed the robust parametric control design method developed in *Apkarian* et al. (2015); and the Multi-model Design (MM), which used a Multi-model approach.

One final comment that is relevant for the following discussion is made: the Γ value for each constraint indicates whether the constraint was satisfied ($\Gamma < 1$) or violated ($\Gamma > 1$). When $\Gamma = 1$, the constraint is exactly at its limit (e.g., for a constraint $X \leq 2$, $X = 2$ results in $\Gamma = 1$). In general, all constraints will be tightened until $\Gamma = 1$ to ensure consistency in the design.

Nominal Design

The Nominal Design approach involves performing systune optimization using only the nominal model. Using this approach, two controllers were developed. For both controllers, the actuation constraint ($\bar{\sigma}(T_{\theta_{cmd}\to\delta_{cmd}}) < W_5$) and the drift constraint ($||T_{v_w\to z_{IMU}}||_\infty < W_2$) were selected to match those used to reproduce the original VEGA controller in *Tapia* (2019). These constraints are detailed in Table 6.3.

The remaining challenge was determining appropriate values for the load constraint ($||T_{v_w\to\tilde{q}\alpha}||_\infty < W_1$) and the sensitivity constraint ($||T_{\theta_{cmd}\to e_\theta}||_\infty < W_4$). However, this choice is not trivial.

Through testing different combinations of load/sensitivity constraints, it was found that a small loosening of the load constraint (e.g., 0.25%) allowed for a significant tightening of the sensitivity constraint, effectively enabling a substantial increase in the input stability S-T disk-based gain margin (1.9 dB). With this in mind, it was decided to test two different configurations.

The first configuration focused on attaining the maximum load relief possible. And so, the load constraint was tightened as much as possible (until the Γ value of all constraints, but the sensitivity constraint, were equal to one) and then, the sensitivity constraint was tightened until its Γ value was equal to one. The controller produced through this approach is coined $K_{CLS-NOM-LR}$, and the final set of constraints used to generate it can be seen in Table 6.3.

The second configuration, made use of the load/sensitivity trade-off, to attain larger stability margins, while almost maintaining the LR capabilities. In this design, the sensitivity constraint was first chosen to achieve S-T disk-based gain margins of at least 9 dB - which is simply a representative value, as other could have been chosen. Secondly, the load constraint was progressively tightened so as to force all the constraint Γ values equal to one. The controller produced through this approach is coined $K_{CLS-NOM-Stab}$, and the final set of constraints used to generate it can be seen in Table 6.3. Notably, for this second controller, the relaxation of the load constraint compared to the first design is minimal—just 0.78%. However, this small adjustment results in a significant improvement, increasing the S-T disk-based input gain margin by 2.5 dB.

Table 6.3: Nominal Design controllers constraints.

	$K_{CLS-NOM-LR}$	$K_{CLS-NOM-Stab}$
$\|\|T_{v_w \to \tilde{q}\alpha}\|\|_\infty < W_1$	$W_1 = 2.078 \cdot 10^5$	$W_1 = 2.094 \cdot 10^5$
$\|\|T_{v_w \to z_{IMU}}\|\|_\infty < W_2$	$W_2 = 100$	$W_2 = 100$
$\bar{\sigma}(T_{\theta_{cmd} \to \delta_{cmd}}) < W_5$	$W_5 = \frac{h_\theta s + \omega_\theta}{s + \omega_\theta / l_\theta}$	$W_5 = \frac{h_\theta s + \omega_\theta}{s + \omega_\theta / l_\theta}$
$\|\|T_{\theta_{cmd} \to e_\theta}\|\|_\infty < W_4$	$W_4 = 1.630$	$W_4 = 1.351$

where the actuation constraint is a low pass filter with a low frequency asymptote $l_\theta = 6.4$ dB, a high frequency asymptote $h_\theta = 2.5$ dB, and a crossover bandwidth $\omega_\theta = 30$ rad/s.

LFT Design

The LFT Design approach leverages the algorithm described in *Apkarian* et al. (2015), facilitating the incorporation of uncertainty during controller design. By incorporating uncertainty into the design, robust performance is achieved, effectively avoiding overtuning the controller for the nominal conditions. A brief description of the algorithm is given next. For a more detailed description, the reader should refer to *Apkarian* et al. (2015).

1. **Initialization:** The set of models used for the controller design begins with the nominal model.
2. **Structured $\mathcal{H}_\infty$ Controller Computation:** A structured $\mathcal{H}_\infty$ controller is computed by minimizing the $\mathcal{H}_\infty$ norm of the transfer functions that represent the designer's constraints.
3. **Stability Check:** The closed loop system, containing the computed controller, is tested for stability by varying the uncertain parameters. If a set of uncertain parameters that destabilizes the system is identified, the corresponding model is added to the collection of models for which the controller is being tuned, and the algorithm returns to step 2.
4. **Performance Degradation:** If the system remains stable, the next step is to try and degrade the performance of the system. This is done by varying the uncertain parameters, so as to increase the $\mathcal{H}_\infty$ norm of the transfer functions that are being constrained. If it is found that a set of parameters causes the violation of the constraints, the corresponding model is added to the set of models, and the algorithm returns to step 2.
5. **Algorithm Termination:** If the degradation in performance is marginal, the algorithm terminates, and the controller is finalized.

In practice, all the designer needs to do is specify the model's uncertain parameters (this can be done with ureal, for real parametric uncertainty), and then feed into systune the uss (uncertain state-space) model. systune will then attempt to perform the optimization while continuously seeking the combination of uncertain parameters that will most significantly violate the constraints. However, there is no guarantee of finding the worse case scenario for each goal.

The first controller designed with the LFT approach is coined $K_{CLS-LFT-LR}$. For this controller, the drift constraint ($||T_{v_w \to z_{IMU}}||_\infty < W_2$) and the actuation constraint ($||T_{\theta_{cmd} \to \delta_{cmd}}||_\infty < W_3$) were chosen based on their respective maximum allowable values. For drift, this was established at 167 meters, as per the requirements. For actuation, a maximum deflection of 6.5 degrees was used to specify the constraint. Subsequently, for the load and sensitivity constraints ($||T_{v_w \to \tilde{q}\alpha}||_\infty < W_1$ and $||T_{\theta_{cmd} \to e_\theta}||_\infty < W_4$, respectively), the approach was to tighten the load constraint as much as possible, and then tighten the sensitivity constraint to push the Γ values to one. The reasoning behind this is that

the LFT approach provides very strong stability margins (as will be shown later), making it unnecessary to sacrifice load relief for a tighter sensitivity constraint. The final constraints used to generate $K_{CLS-LFT-LR}$ can be seen in Table 6.4.

The second controller designed with the LFT approach is coined $K_{CLS-LFT-Act}$. For this controller, the same rationale was applied to the drift constraint. However, a different approach was taken for the other constraints. It was assumed that the algorithm would accurately identify the exact worst-case scenario, so the load constraint was set to the maximum allowable $\tilde{q}\alpha$ value in accordance with the requirements. Next, actuation was minimized as much as possible, and sensitivity was utilized to push all Γ values to one. If the algorithm successfully identifies the worst-case scenario, this approach would result in a controller capable of withstanding all wind conditions while minimizing actuation. The final constraints used to generate $K_{CLS-LFT-Act}$ can be seen in Table 6.4.

Table 6.4: LFT-based controllers constraints.

	$K_{CLS-LFT-LR}$	$K_{CLS-LFT-Act}$
$\|\|T_{v_w \to \tilde{q}\alpha}\|\|_\infty < W_1$	$W_1 = 2.511 \cdot 10^5$	$W_1 = 2.751 \cdot 10^5$
$\|\|T_{v_w \to z_{IMU}}\|\|_\infty < W_2$	$W_2 = 167$	$W_2 = 167$
$\|\|T_{\theta_{cmd} \to \delta_{cmd}}\|\|_\infty < W_3$	$W_3 = 6.5$	$W_3 = 4.6$
$\|\|T_{\theta_{cmd} \to e_\theta}\|\|_\infty < W_4$	$W_4 = 1.670$	$W_4 = 1.670$

Multi-model Design

The primary drawbacks of the LFT Design approach are the lack of guaranteed reproducibility of results and limited scalability for systems with a high number of parametric uncertainties. On one hand, the worst-case scenario identified in one run may differ from that found in another, and while such inconsistencies are infrequent, they remain a possibility. On the other hand, when the system contains a large number of parametric uncertainties, the computational burden can increase significantly. Therefore, the Multi-model Design approach, which addresses these issues, was employed to generate controllers. In this approach, multiple models are created by the designer, which are then fed into systune, and a controller is tuned to satisfy the requirements for all selected models.

As it might be clear, the key part of this approach lies in selecting the models that will be considered during design. In this project, a vertex case approach was used to select the models. It entails generating different models by utilizing all combinations of the maximum and minimum values for each uncertain parameter. However, the launch vehicle model has a large number of uncertain parameters, making it impractical to include every single one in the design process. If all were considered, the design process would become computationally burdensome, as the launch vehicle has a total of 15 rigid body uncertainties, resulting in 2^{15} possible models. In order to select the uncertainties that should be included, the μ-sensitivities of the system with the first LFT controller were computed (see *Marcos* et al. (2005) for a detailed explanation on μ-sensitivities). These sensitivities are presented in Figure 6.12.

The μ-sensitivities of the system are an effective measure of how much each uncertain parameter influences the structured singular value, and, consequently, the stability of the launch vehicle. For instance, if the μ-sensitivity of a certain parameter is 0.1, this implies that a given change in the uncertainty range of parameter, results in a change of the SSV that is one tenth as large. Ordered from largest to smallest μ-sensitivity peak, the first seven parameters are: C_{N_α}, $\tilde{q}$, x_{cp}, T, τ, V_{rel}, and I_{yy}.

Taking this into account, three controllers were developed using the Multi-model approach. The first controller incorporated the three uncertain parameters with the highest peak μ-sensitivity, the second

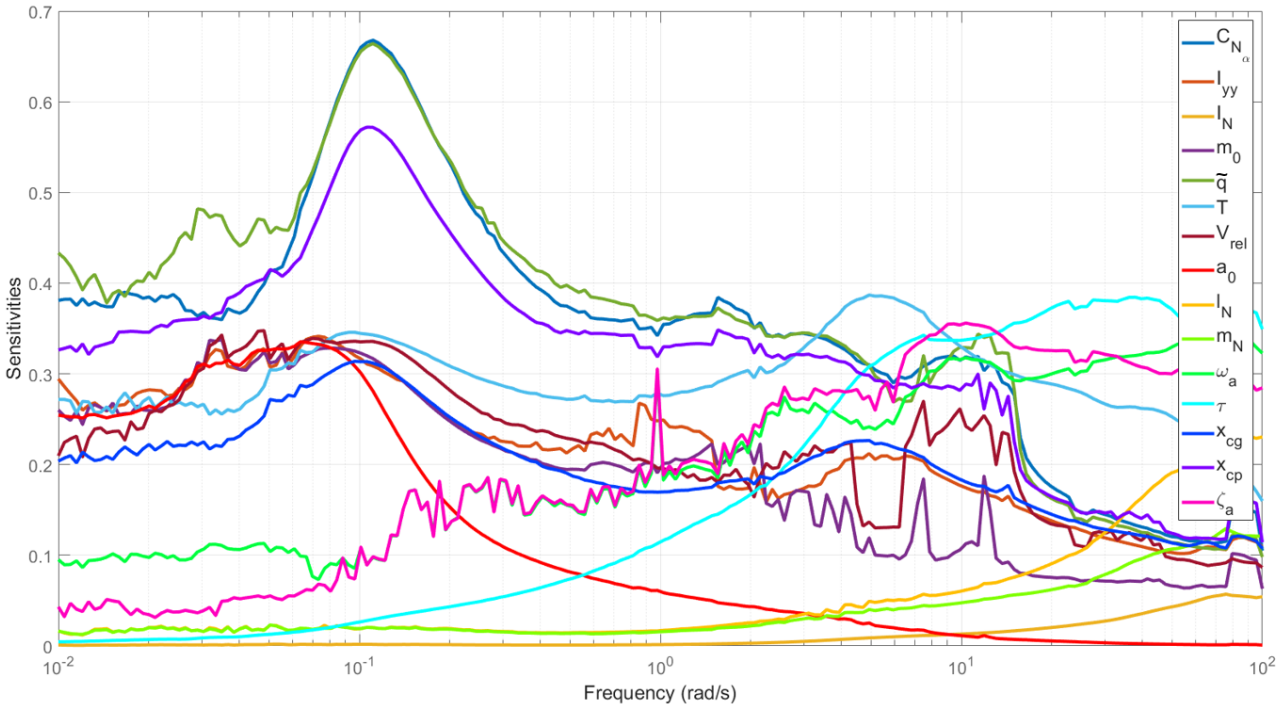

Figure 6.12: System's μ-sensitivities, utilizing the first LFT controller.

controller included five parameters, and the third controller utilized seven. In all iterations, the nominal model was included as well.

Utilizing the aforementioned models, the final constraint sets, as well as the uncertain parameters considered for each controller, can be seen in Table 6.5. The logic used to select the constraints was the same as the first LFT-based controller ($K_{CLS-LFT-LR}$). One remark should be made here: the Multi-model controllers were only designed after comparing the Nominal Design and the LFT-based controllers. Hence, at the time, it was already known that the first LFT-based controller was superior to the Nominal Design controllers and to the second LFT-based controller.

Table 6.5: $\mathcal{H}_\infty$ CLS Multi-model constraints and uncertain parameters.

	$K_{CLS-MM3-LR}$	$K_{CLS-MM5-LR}$	$K_{CLS-MM7-LR}$
Uncertain Parameters	C_{N_α}, $\tilde{q}$, x_{cp}	C_{N_α}, $\tilde{q}$, x_{cp}, T, τ	C_{N_α}, $\tilde{q}$, x_{cp}, T, τ, V_{rel}, I_{yy}
$\|\|T_{v_w \to \tilde{q}\alpha}\|\|_\infty < W_1$	$W_1 = 2.065 \cdot 10^5$	$W_1 = 2.164 \cdot 10^5$	$W_1 = 2.409 \cdot 10^5$
$\|\|T_{v_w \to z_{IMU}}\|\|_\infty < W_2$	$W_2 = 167$	$W_2 = 167$	$W_2 = 167$
$\|\|T_{\theta_{cmd} \to \delta_{cmd}}\|\|_\infty < W_3$	$W_3 = 6.5$	$W_3 = 6.5$	$W_3 = 6.5$
$\|\|T_{\theta_{cmd} \to e_\theta}\|\|_\infty < W_4$	$W_4 = 1.527$	$W_4 = 1.656$	$W_4 = 1.663$

6.6.6. Controller Comparison

After designing all the controllers (summarized in Table 6.6), they were implemented using a framework identical to that shown in Figure 6.11, with three key differences. First, the input scaling is now exclusively used to convert values from degrees to radians. Second, an additional output, denoted as δ_{pitch}, is included to represent the signal δ generated by the TVC block. Finally, the block G_{LV} now contains the rigid body analysis model (AMR), as described in Subsection 5.4.

Table 6.6: $\mathcal{H}_\infty$ CLS Controllers.

	K_{p_θ}	K_{d_θ}	K_{p_z}	K_{d_z}	Γ	Comp. Time
$K_{CLS-NOM-LR}$	-1.3336	-0.6018	0.000158	0.0017	1.00 for all constraints	3.2 s
$K_{CLS-NOM-Stab}$	-1.3179	-0.8303	0.000157	0.0022	1.00 for all constraints	3.4 s
$K_{CLS-LFT-LR}$	-3.6539	-1.1162	0.0011	0.0081	1.00 for all constraints	15.4 s
$K_{CLS-LFT-Act}$	-2.8198	-0.9902	0.000749	0.0058	1.00 for all constraints	15.6 s
$K_{CLS-MM3-LR}$	-4.1957	-1.2349	0.0010	0.0089	1.00 for all constraints	26.7 s
$K_{CLS-MM5-LR}$	-3.8174	-1.1352	0.000935	0.0080	1.00 for all constraints	83.4 s
$K_{CLS-MM7-LR}$	-3.7970	-1.1326	0.0010	0.0082	1.00 for all constraints	241.3 s

In Table 6.6, the last column shows the computation time required by systune to compute each controller, with 5 random starts employed per computation (following *Tapia* (2019)) and no parallel computing utilized. From the table, it is evident that the MM approach is slower than the LFT approach. This is primarily caused because the model does not include an excessive number of uncertainties. Additionally, it is clear that increasing the number of uncertain parameters in the MM approach significantly raises the computational time.

The $\mathcal{H}_\infty$ CLS controllers were compared in two stages: first, the two Nominal Design controllers were evaluated against the two LFT Design controllers. Next, the best-performing controller from this comparison was assessed against the three Multi-model Design controllers. For brevity, only selected analysis plots are shown here, with additional plots available in Appendix B.

Consider now the first comparison. The Nichols plot at the plant input, shown in Figure 6.13, provides valuable insights. This figure demonstrates the effectiveness of the LFT approach: the worst-case

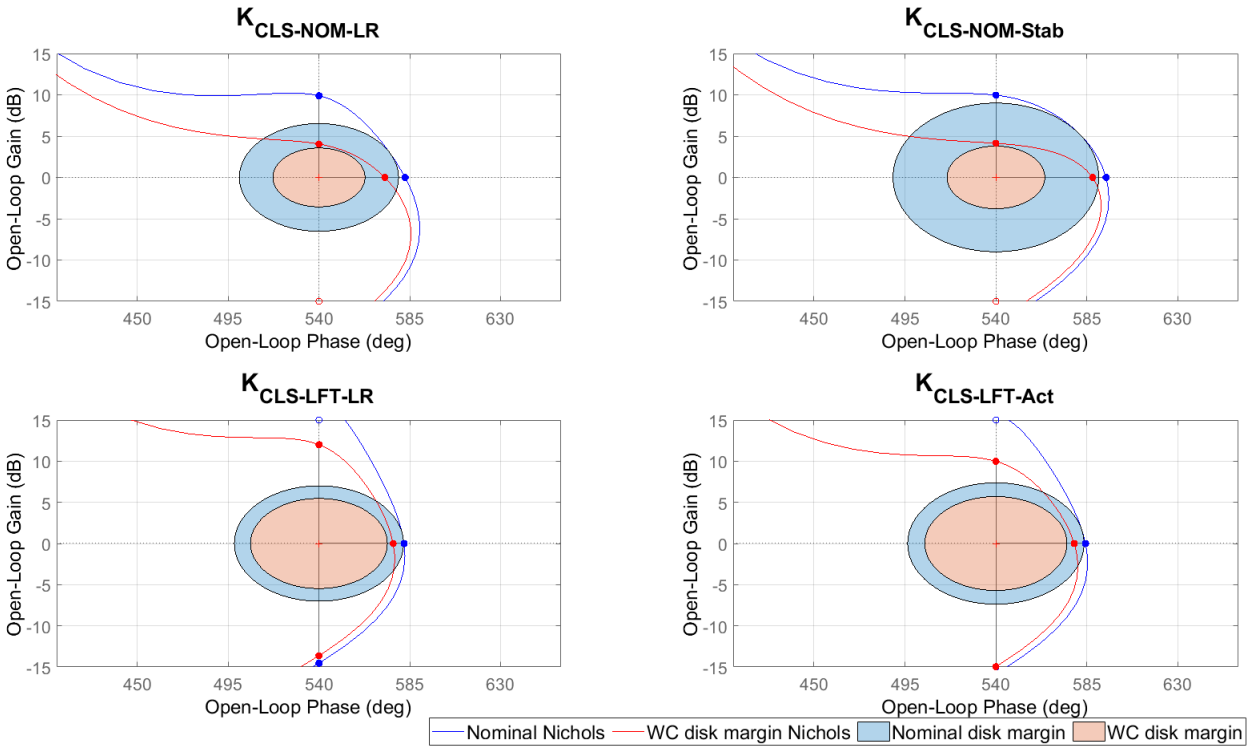

Figure 6.13: Input Nichols plot - Nominal Design vs LFT Design $\mathcal{H}_\infty$ CLS controllers.

scenario margins have improved tremendously, bringing the worst-case much closer to the nominal case, whilst maintaining good nominal case margins. While all controllers meet the rigid stability requirements, the overall conclusion is that the LFT controllers achieve better worst-case margins while maintaining comparable nominal performance. In Appendix B, the other stability margins can be observed. One more remark should be made: the nominal based controller whose goal was to improve stability margins ($K_{CLS-NOM-Stab}$), showed improved nominal margin results, but no improvement in the worst-case scenario. This makes sense as the imposition of a stricter sensitivity requirement is only being upheld for the nominal model.

Figure 6.14 presents the worst-case gain of the transfer function from v_w to $\tilde{q}\alpha$, normalized by the nominal dynamic pressure ($\tilde{q}_n$). In this figure, the thick black line represents the maximum allowed singular value, indicating that the load relief requirement is satisfied for a maximum wind velocity of 33 m/s as long as the curve remains below this threshold.

A key observation is the presence of two triangular peaks around 20 rad/s in the plots for both LFT

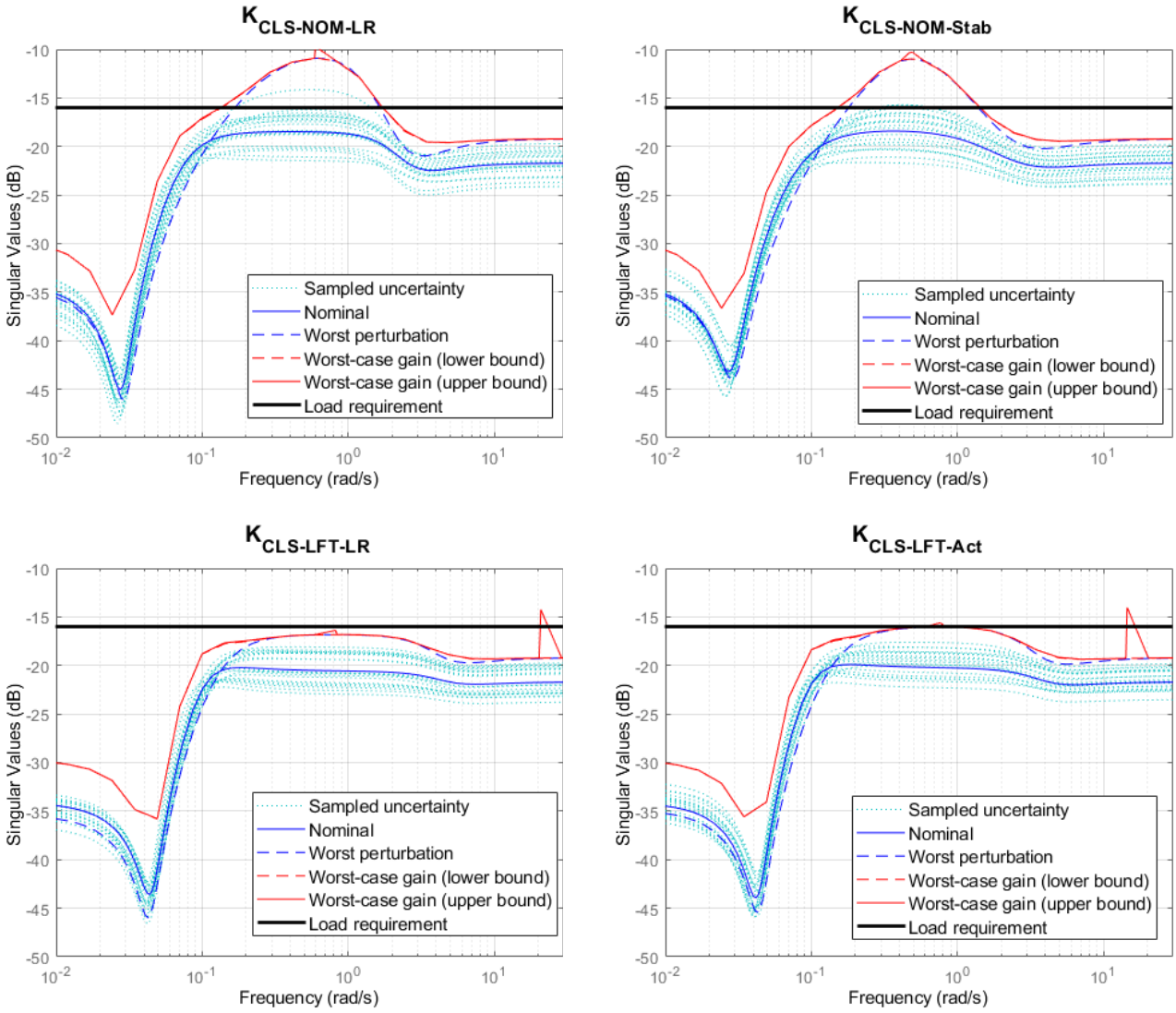

Figure 6.14: Worst-Case Gain of $T_{v_w \to \tilde{q}\alpha/\tilde{q}_n}$ - Nominal Design vs LFT Design $\mathcal{H}_\infty$ CLS controllers.

controllers. These peaks are numerical artifacts, as confirmed by the computation of the worst-case gain using more rigorous methods. In reality, for those regions, the upper bound is almost indistinguishable from the lower bound. With this clarification, it becomes evident that the only controller meeting the load relief requirement is $K_{CLS-LFT-LR}$.

Further analysis confirmed that $K_{CLS-LFT-LR}$ also satisfies the drift, drift rate and actuation requirements. Consequently, all other controllers can be eliminated as candidates, leaving $K_{CLS-LFT-LR}$ to proceed to the second comparison phase.

The second comparison evaluates $K_{CLS-LFT-LR}$ against the Multi-model controllers. Figure 6.15 illustrates the input Nichols plot pertaining the controllers of the second comparison. It can be observed that the input stability margins for all four controllers are closely comparable. Furthermore, an analysis of the remaining stability margins, shows that the trend is maintained: for each output, the nichols are basically indistinguishable; the same can be said regarding the MIMO output and I/O disk margins. As such, these results are not included in the project. If the reader is interested, he can simply refer to the stability margins of the LFT - LR controller, shown in Appendix B.

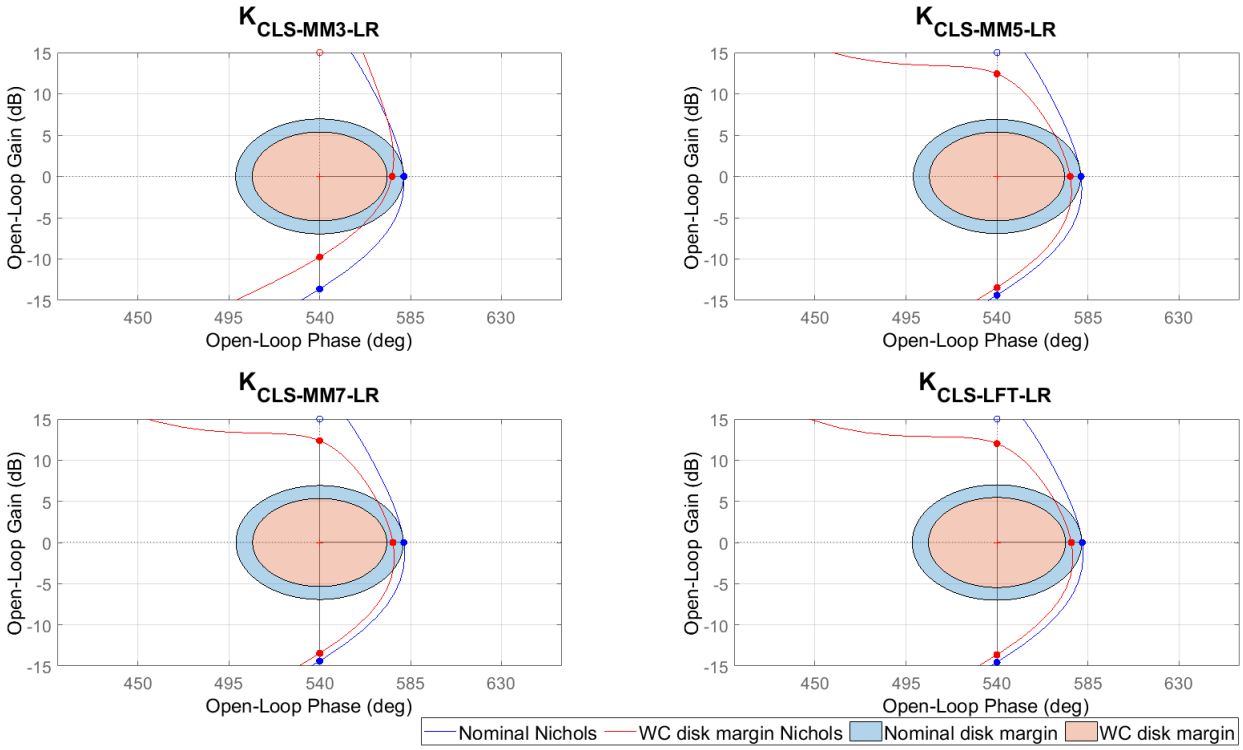

Figure 6.15: Input Nichols plot - $K_{CLS-LFT-LR}$ vs Multi-model Design $\mathcal{H}_\infty$ CLS controllers.

The only place where a distinction can be made is in the worst-case gain analysis. For the analysis on $T_{v_w \to \tilde{q}\alpha/\tilde{q}_n}$, all plots are essentially identical; however, the $\mathcal{H}_\infty$ norm of the worst-case gain for each controller shows slight differences, as presented in Table 6.7.

Table 6.7: $\mathcal{H}_\infty$ norm of the worst-case gain scenario of $T_{v_w \to \tilde{q}\alpha/\tilde{q}_n}$ - $K_{CLS-LFT-LR}$ vs Multi-model controllers.

	$K_{CLS-MM3-LR}$	$K_{CLS-MM5-LR}$	$K_{CLS-MM7-LR}$	$K_{CLS-LFT-LR}$
$\|\|T_{v_w \to \tilde{q}\alpha/\tilde{q}_n}\|\|_\infty$	-17.6	-17.2	-16.9	-16.8

In order for the load requirement to be upheld for a maximum wind velocity of $33\,\mathrm{m/s}$, the $\mathcal{H}_\infty$ norm of the WC gain of $T_{v_w \to \tilde{q}\alpha/\tilde{q}_n}$ must be smaller than $-16.05\,\mathrm{dB}$. This requirement, which is portrayed by the thick black line in the WC gain plots of $T_{v_w \to \tilde{q}\alpha/\tilde{q}_n}$, is computed with the following expression:

$$20log_{10}(\frac{Load_M}{Wind_M \cdot \tilde{q}}) \tag{6.4}$$

where $Load_M$ is the maximum allowable load and $Wind_M$ is the maximum expected wind velocity.

From Table 6.7, it is clear that all controllers uphold the load relief requirement. However, as the number of uncertainties being considered in the design increase, the load relief capabilities of the controller decrease.

Building on this analysis, Figure 6.16 presents the worst-case gain of $T_{v_w \to z_{IMU}}$. In this figure, the thick black line represents the maximum allowed singular value. When the plot remains below this line, it guarantees that drift will not exceed $167\,\mathrm{m}$ for a maximum wind velocity of $33\,\mathrm{m/s}$. Additionally, it should be

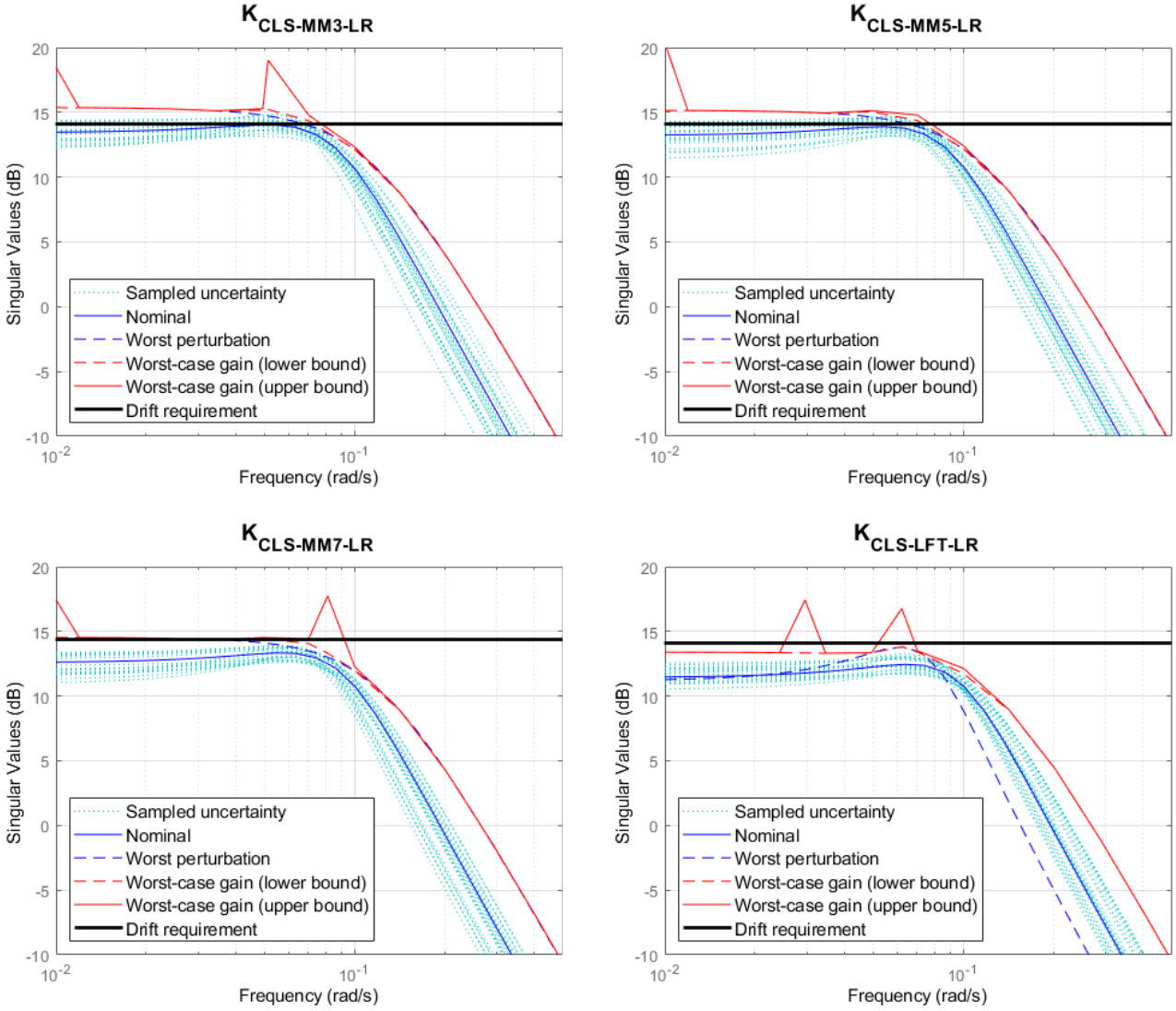

Figure 6.16: Worst-case Gain of $T_{v_w \to z_{IMU}}$ - $K_{CLS-LFT-LR}$ vs Multi-model Design $\mathcal{H}_\infty$ CLS controllers.

noted that the triangular peaks observed in the plots are numerical artifacts. Considering this information, it should be clear that only $K_{CLS-LFT-LR}$ satisfies the drift requirement, although $K_{CLS-MM7-LR}$ comes very close to meeting it. Additionally, the plots reveal that as the number of uncertainties increases, drift rejection improves (illustrated by the fact that the worst-case gain upper bound curve progressively shifts down). When both observations are considered together, it becomes clear that while the first three uncertainties are sufficient to address the worst-case scenario for load relief, additional uncertainties are required to achieve effective drift rejection.

Finally, Figure 6.17 illustrates the worst-case gain of $T_{\theta_{cmd}\to\delta_{cmd}}$. Once again, the thick black line shown in the figure represents the maximum allowable singular value, ensuring that if the plot stays below this line, the actuator deflection will be smaller than 6.5 degrees. Additionally, it should be noted that for the case of the worst-case gain of $T_{\theta_{cmd}\to\delta_{cmd}}$, the drift and drift rate feedbacks were disconnected. This is due to the particularities of the LV model, as described in Subsection 6.4.3. With this in mind, it is easy to understand that more than the three first uncertainties are necessary to adequately cover the worst-case

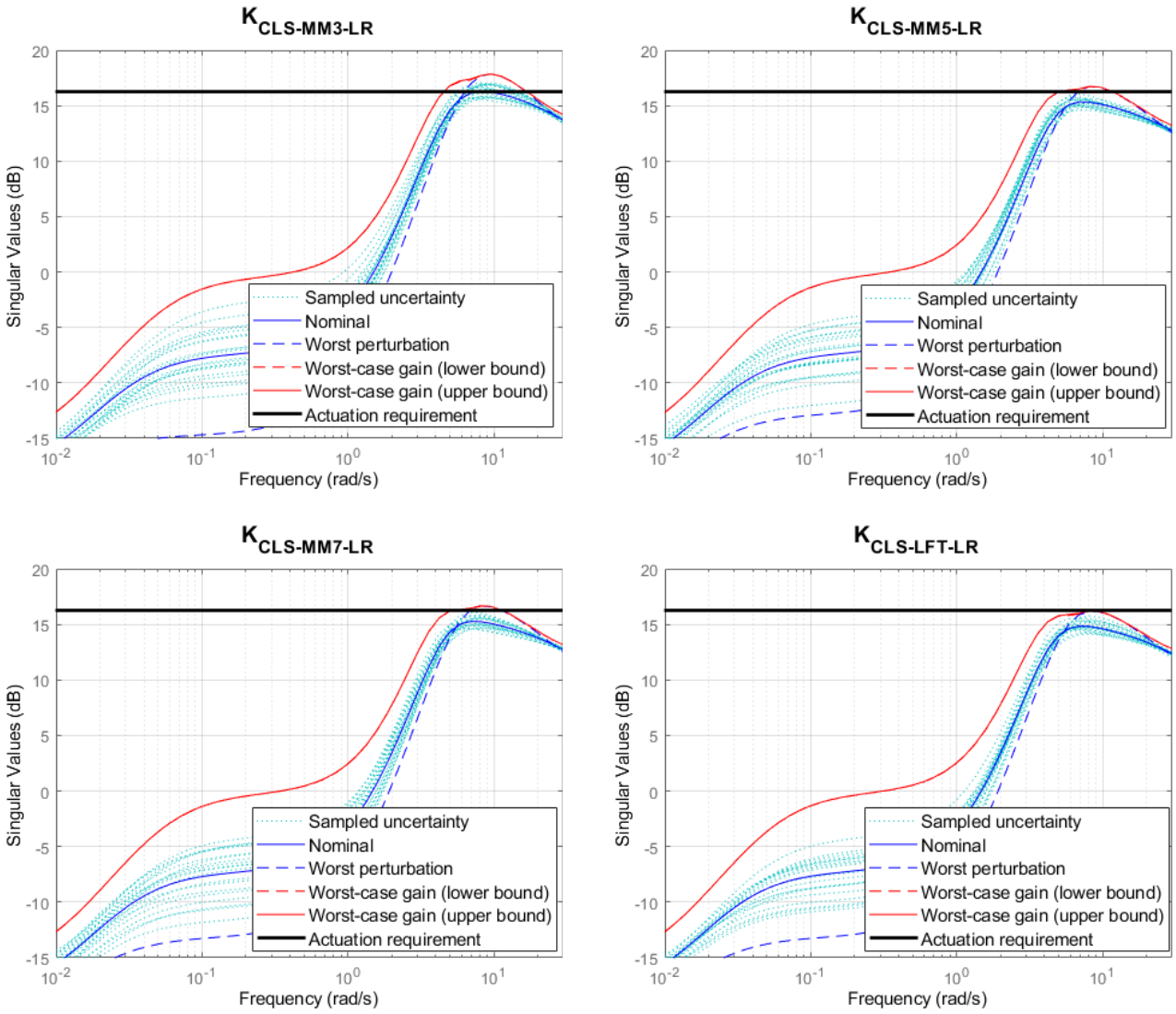

Figure 6.17: Worst-case gain of $T_{\theta_{cmd}\to\delta_{cmd}}$ - $K_{CLS-LFT-LR}$ vs Multi-model Design $\mathcal{H}_\infty$ CLS controllers.

scenario for actuation. Notably, $K_{CLS-MM3-LR}$ completely fails to meet the actuation requirement, while $K_{CLS-MM5-LR}$ performs significantly better, almost meeting it. $K_{CLS-MM7-LR}$ performs slightly better than $K_{CLS-MM5-LR}$, failing to satisfy the requirement by a mere 0.5 dB. Only the LFT controller meets the actuation requirement.

The final comparison of the $\mathcal{H}_\infty$ CLS controllers yields several key conclusions. Notably, $K_{CLS-LFT-LR}$ should serve as the benchmark for evaluating the $\mathcal{H}_\infty$ OLS controller. Based on the earlier worst-case plots, input stability analysis, and the additional stability assessments and time-domain cases presented in Appendix B, only $K_{CLS-LFT-LR}$ is expected to satisfy all the specified requirements.

Secondly, the $K_{CLS-MM7-LR}$ controller came close to meeting the performance requirements and could potentially rival $K_{CLS-LFT-LR}$ performance if recomputed with additional uncertainties. However, this would come at the cost of significantly increased computational time. As shown in Table 6.6, the Multi-model approach exhibits a rapid growth in computation time as more uncertain parameters are introduced. Further testing showed that increasing the number of uncertain parameters from 7 to 8 caused the computation time to rise from approximately 4 minutes to approximately 10 minutes. Further increasing the number of uncertain parameters to 9 pushed the computational time to 21 minutes. Therefore, while the Multi-model approach has the potential to approximate LFT performance while offering more reproducible results, further increasing the number of uncertainties to fully match the LFT controller's performance could render the design process impractically time-consuming.

Finally, for the Multi-model approach, it should be noted that whilst the first three uncertainties with higher μ-sensitivities are enough to achieve the desirable load relief and stability, they are not sufficient to achieve the desired drift and actuation requirements. If one remembers the fact that the first three uncertainties are C_{N_α}, $\tilde{q}$ and x_{cp}, it becomes clear why they are sufficient for load relief and for stability but not for the other criteria.

6.7. $\mathcal{H}_\infty$ Open Loop Shaping

This section details the use of $\mathcal{H}_\infty$ Open Loop Shaping to design the atmospheric attitude controller for $t = 49$ s. First, the open loop is scaled and analyzed in Subsection 6.7.1. Next, in Subsection 6.7.2, a correspondence is established between the LV performance requirements and the associated open loop shape and controller shape. Additionally, the frequency ranges where these requirements are most relevant are discussed. This analysis provides valuable insights into how the loop shape should be defined, effectively bridging the gap between closed loop performance requirements and open loop gain specifications. Subsequently, in Subsection 6.7.3, the parametrization of the weighting filters used in this work is detailed. Guidelines on how to use the parametrization to compute the $\mathcal{H}_\infty$ OLS robust controller are given in Subsection 6.7.4. Finally, in Subsection 6.7.5, the parametrization is used to generate $\mathcal{H}_\infty$ OLS controllers for the selected flight point.

6.7.1. Open Loop Scaling and Analysis

As discussed in most control theory textbooks, scaling is an essential step of designing a controller, as it makes model analysis and controller design much simpler (*Skogestad* & *Postlethwaite*, 2005). Additionally, improper scaling can lead to issues, particularly when minimizing certain cost functions (see pages 34-35 of *Papageorgiou* (1998) for an example).

As there is only one input, the main focus of the scaling is the output of the plant. The scaling was performed using the procedure described in *Skogestad* and *Postlethwaite* (2005): the input was divided by

the largest allowable input change, and the outputs were divided by the largest allowable error. Additionally, the scaling factors for the input, attitude output, and attitude rate output were each multiplied by a factor to ensure that the respective channels could receive/output values in degrees. The scaling factors are as follows:

$$S_{in} = \frac{\pi}{180} \cdot \frac{1}{6.5} \quad S_z = \frac{1}{150} \quad S_{\dot{z}} = \frac{1}{15} \quad S_\theta = \frac{180}{\pi} \quad S_{\dot{\theta}} = \frac{180}{\pi} \cdot \frac{1}{0.25} \tag{6.5}$$

where the maximum expected attitude rate error was estimated from the low frequency, high amplitude wind rejection time domain case of the final $\mathcal{H}_\infty$ closed loop controller. The pitch maximum value was used equal to the maximum allowable command of the VEGA LV (retrieved from *Tapia* (2019)). The drift, drift rate and input maximum values come directly from the requirements.

Typically, after applying scaling, the condition number of the plant [3] can be examined to evaluate the effectiveness of the scaling. However, since the plant is SIMO (single-input-multiple-output) and has only one singular value, this assessment is not possible. Therefore, the gain portion of the Bode plots from the plant input to each individual output has been analyzed and is illustrated in Figure 6.18.

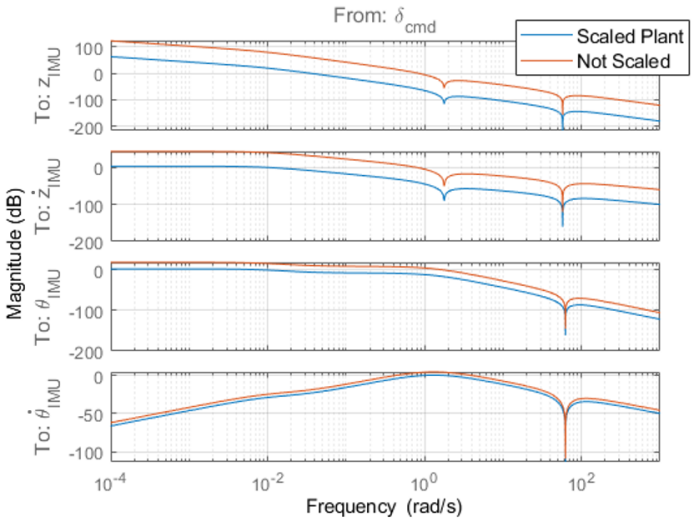

Figure 6.18: Magnitude part of bode plot from plant input to individual outputs.

As observed, all outputs are now much closer in magnitude. However, the drift output still exhibits a significantly higher magnitude. At this stage, it was decided to leave the scaling as is, with the option to revisit it later if necessary. Subsequent work confirmed that this scaling approach was effective.

Some important considerations can be made regarding the output channels by looking at the previously shown bode plot. It is clear that the different outputs have quite different crossover values. With appropriate weighting filter usage, it will be possible to utilize each channel to manipulate the open loop gain at a particular frequency. This will be further discussed in Subsection 6.7.3.

Finally, it is important to consider the open loop gains of the plant and compare them to those of the $\mathcal{H}_\infty$ CLS controller, as this might give insights into the desired shape. Figure 6.19 shows the singular values of the input and output open loops of the system with the final $\mathcal{H}_\infty$ CLS controller, along with the singular values of the scaled plant.

[3] The ratio between the plant's largest and smallest singular values - see page 94 of *Skogestad* and *Postlethwaite* (2005), for a more detailed explanation.

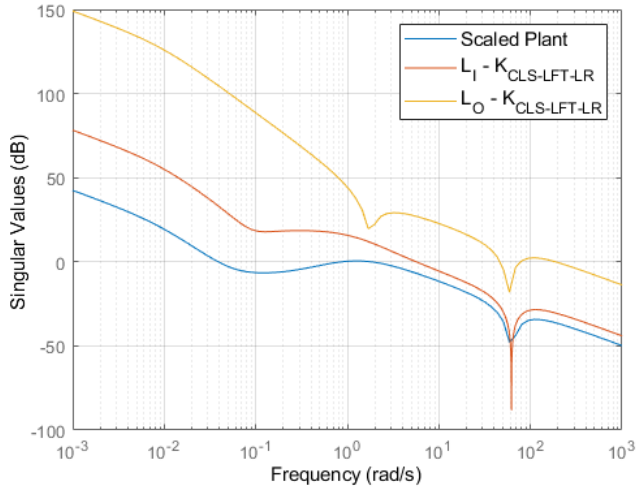

Figure 6.19: Open loop SV comparison.

First, it should be noted that the scaled plant exhibits a decent roll-off rate at both low and high frequencies, with a satisfactory roll-off at crossover (close to -20 dB/dec). However, it shows low gain at low and mid frequencies. This indicates that there is significant room for improvement, particularly in the lower frequency range, where higher gain and better roll-off could be achieved. The rationale behind these desired goals will be discussed in the next subsection.

The second point of interest is the presence of dips (resonances) in the plots. The dip around 2 rad/s in L_O is attributed to a non-minimum phase zero in the drift channel. Moreover, all plots display a common dip at a higher frequency, which is caused by a non-minimum phase zero introduced by the tail-wag-dog effect. The TWD effect will be neglected during the $\mathcal{H}_\infty$ OLS controller design, just as it was during the $\mathcal{H}_\infty$ CLS controller design.

Finally, the third notable observation is the low, flat gain of the scaled plant observed between 0.1 rad/s and 2 rad/s. The low gain indicates challenges in rejecting disturbances within this specific range, which is particularly important for load relief, as will be discussed in the next subsection.

6.7.2. Launch Vehicle Requirements vs Open Loop Shape and Controller Shape

The correspondence between closed loop requirements and the singular values of the open loop and controller transfer functions is well-documented in the literature, as detailed in *McFarlane* and *Glover* (1992), *Papageorgiou* (1998), and *Bates* and *Postlethwaite* (2002), among others. In this subsection, this knowledge is applied to translate the launch vehicle requirements into open loop and controller specifications. However, before proceeding, a general overview of the desired loop shape and controller shape is provided. It should be noted that the in the following discussion the relation between the singular values of the open loop transfer functions and closed loop transfer functions are merely stated, but not mathematically demonstrated. The reader who is interested in said demonstration can refer to pages 13-15 of *Papageorgiou* (1998).

In general, the desired open loop gains shape consists of a high minimum singular value at low frequencies (LFs), a low maximum singular value at high frequencies (HFs), and a crossover frequency that meets the bandwidth requirements. Additionally, it is desirable to achieve a high roll-off rate at both high and low frequencies and a roll-off rate of -20 dB/dec at crossover.

On one hand, increasing the minimum singular values of the loop gains, $\underline{\sigma}(L_O)$ and $\underline{\sigma}(L_I)$, at low frequencies reduces the maximum singular values of S_O and S_I in the same frequency region, respectively. Reducing the maximum SV of S_I, $\bar{\sigma}(S_I)$, results in improved input disturbance rejection at the plant input. Meanwhile, reducing the maximum SV of S_O, $\bar{\sigma}(S_O)$, results in improved output disturbance rejection at the plant output, but also enhances tracking performance [4]. Similarly, reducing the maximum singular value of the loop gains, $\bar{\sigma}(L_O)$ and $\bar{\sigma}(L_I)$, at high frequencies pushes down the maximum singular values of T_O and T_I, respectively. This results in improved noise rejection at the plant output through the reduction of $\bar{\sigma}(T_O)$, and enhanced control signal attenuation through the reduction of $\bar{\sigma}(T_I)$.

In terms of controller shape, it is desirable for the controller's singular values to exhibit an integrator-like shape. By increasing the minimum singular value of the controller, $\underline{\sigma}(K)$, at low frequencies, $\bar{\sigma}(S_O G)$ is reduced, leading to improved input disturbance rejection at the plant output. Conversely, by decreasing $\bar{\sigma}(K)$, at high frequencies, $\bar{\sigma}(KS_O)$ is reduced, resulting in better noise attenuation at the plant input and improved control signal attenuation.

By achieving an open loop shape and controller as described, disturbance rejection, tracking performance, noise attenuation, and low control effort can be ensured. Next, each LV performance requirement is translated into an open loop specification.

1 - Load Relief (LR)

The main goal of the atmospheric attitude controller is to provide load relief, this equates to maintaining the load performance metric ($\tilde{q}\alpha$) as low as possible. To achieve a reduction in the load performance metric (LPM), the aim is to minimize the angle of attack (AoA). Consider, the equation that describes the angle of attack:

$$\alpha = \frac{\dot{z}}{V} + \theta - \frac{v_w}{V} \tag{6.6}$$

Since there are no measurements or estimations of AoA or wind, minimizing the AoA, can only be achieved through minimizing drift rate and pitch angle. This involves maintaining both parameters as close as possible to their steady-state values: 0 m/s for the drift rate and 0 degrees for the pitch angle. Consequently, minimizing drift rate and pitch angle equates to achieving effective disturbance rejection in these channels. Thus, the load relief problem can be recast as a challenge of ensuring disturbance rejection in the drift rate and pitch angle outputs. Disturbance rejection at the plant output is a well known specification of open loop shaping, it can be achieved by ensuring that, at the required frequency range, $\underline{\sigma}(L_O) >> 1$ and $\underline{\sigma}(K) >> 1$. Additionally, by satisfying the first specification, reference tracking is also ensured, as explained in the previous page.

If one recalls the worst-case gain plots of $T_{v_w \to \tilde{q}\alpha/\tilde{q}_n}$ (see figure 6.14), it is clear where LR is critical. Specifically, in the frequency range of approximately 0.2 to 2.2 rad/s, the gain in the aforementioned transfer function is the highest, making this range particularly important for load relief. Achieving $\underline{\sigma}(L_O) >> 1$ and $\underline{\sigma}(K) >> 1$ in this frequency range (hereafter referred to as the critical frequency range) will result in improved disturbance rejection capabilities for both pitch angle and drift rate, thereby effectively accomplishing load relief.

2 - Drift and Drift Rate Bounding

Once again, the problem is one of disturbance rejection at the plant output, as bounding drift and drift rate is equivalent to keeping these parameters within a range around their steady-state values (both of

[4] Recalling that $S_O + T_O = I$, it is easy to see that if $\bar{\sigma}(S_O)$ is decreased, the minimum and maximum SV of T_O are pushed to one, resulting in improved tracking.

which are equal to zero).

The worst-case gain of $T_{v_w \to z_{IMU}}$ exhibits the highest singular value in the frequency range 0 to $0.1\,\text{rad/s}$ (see Figure 6.16), making this range crucial for bounding drift, since it is where the drift requirement is most likely to be violated. Ensuring proper drift bounding in this range will, in principle [5], guarantee the fulfillment of the drift requirement. Therefore, drift bounding, can be achieved through attaining $\underline{\sigma}(L_O) >> 1$ and $\underline{\sigma}(K) >> 1$ in the aforementioned frequency range.

Similarly, the peak singular value of the worst-case gain of $T_{v_w \to \dot{z}_{IMU}}$ occurs in the frequency range of 0.07 to $0.2\,\text{rad/s}$, highlighting the importance of this range for drift rate bounding. Therefore, following a rational similar to the one used for drift bounding, drift rate bounding can be achieved through attaining $\underline{\sigma}(L_O) >> 1$ and $\underline{\sigma}(K) >> 1$ in the aforementioned frequency range.

3 - Actuation Minimization and Bounding

Actuation can be minimized and bounded considering the following equation (*Papageorgiou*, 1998):

$$u = KS_o(r - n) - KS_o d_o - T_i d_i \tag{6.7}$$

where r is the reference command, n is noise, d_o is the plant output disturbance and d_i is the plant input disturbance.

To achieve lower actuation, it is important to minimize the contributions arising from reference commands, noise, and output/input disturbances. Specifically, to reduce the impact of noise and reference commands at high frequencies, it is essential to minimize $\bar{\sigma}(KS_O)$ in this frequency region. This objective can be accomplished effectively by ensuring that $\bar{\sigma}(K)$ is minimized at high frequencies.

To reduce the contribution from output disturbances and reference commands at low frequencies, one should minimize $\bar{\sigma}(KS_O)$ at low frequencies. However, it can be shown that if $\underline{\sigma}(L_O) >> 1$ (which is normally the case at low frequencies), then $\bar{\sigma}(KS_O) \simeq \frac{1}{\underline{\sigma}(G)}$. Therefore, reducing the contribution from output disturbances and reference commands at low frequencies is not possible, as this solely depends on the plant.

For input disturbances, minimizing $\bar{\sigma}(L_I)$ at low frequencies is desirable, as this would decrease $\bar{\sigma}(T_I)$. However, this cannot be done as it will conflict with other specifications. Instead, the crossover frequency can be adjusted, as reducing the crossover frequency and therefore the gain right before crossover will decrease $\bar{\sigma}(L_I)$ in those frequencies, which will help mitigate input disturbance effects. However, care must be taken when decreasing the crossover frequency, as this will affect the bandwidth of the system and therefore its tracking capabilities.

In summary, to achieve lower actuation, one must minimize $\bar{\sigma}(K)$ at high frequencies, and decrease the crossover frequency as well the gain near it.

4 - Disturbance Rejection (output decoupling)

Attaining output decoupling is relevant at low frequencies and is essentially the same as rejecting disturbances at the plant output (see page 15 of *Papageorgiou* (1998)). As previously discussed, rejecting disturbances at the plant output can be achieved through attaining $\underline{\sigma}(L_O) >> 1$ and $\underline{\sigma}(K) >> 1$ at low frequencies.

[5] Recalling the worst-case gain plot of $T_{v_w \to z_{IMU}}$, it is clear that outside the aforementioned range it is still possible for drift requirement violations to occur, tho far less likely, due to the natural roll-off of the transfer function.

Desired Loop Shape and Controller

Keeping the aforementioned discussion in mind, it is clear that the controller must have a high minimum singular value at LF and a low maximum singular value at HF (this can be attained through a controller with an integrator-like shape). In terms of open loop shaping, the desired loop shape is illustrated in Figure 6.20. In this figure, it is possible to see where drift and drift rate rejection is most important, as shown by the large double red and dark blue arrow. The region where load relief is most important is shown by the large light blue double arrow. The thin red arrows above these areas show that by pushing the curve upwards (increasing the singular values), these particular objectives can be improved. On the rightmost part of the plot, it can be seen that in order to get better actuation, the gain after LR zone can be decreased, and the crossover frequency reduced.

Beneath the plot, three large arrows showcase the other goals attained by shaping the open loop as shown, as well as in what frequencies they are most relevant. The orange double arrow aims at showing the zone where the trade-off between tracking/disturbance rejection/decoupling (dec) and actuation happens: increasing gain in that region might improve tracking, DR and decoupling, but it will worsen actuation. The pinkish double arrow shows the zone where noise attenuation can be achieved. This would be done through lowering the maximum singular values in that region, as illustrated by the thin green arrow.

Finally, the reader should also notice the evolution of the curve slope. At low and high frequencies the slope is high, whilst for crossover the slope is very close to -20 dB/dec.

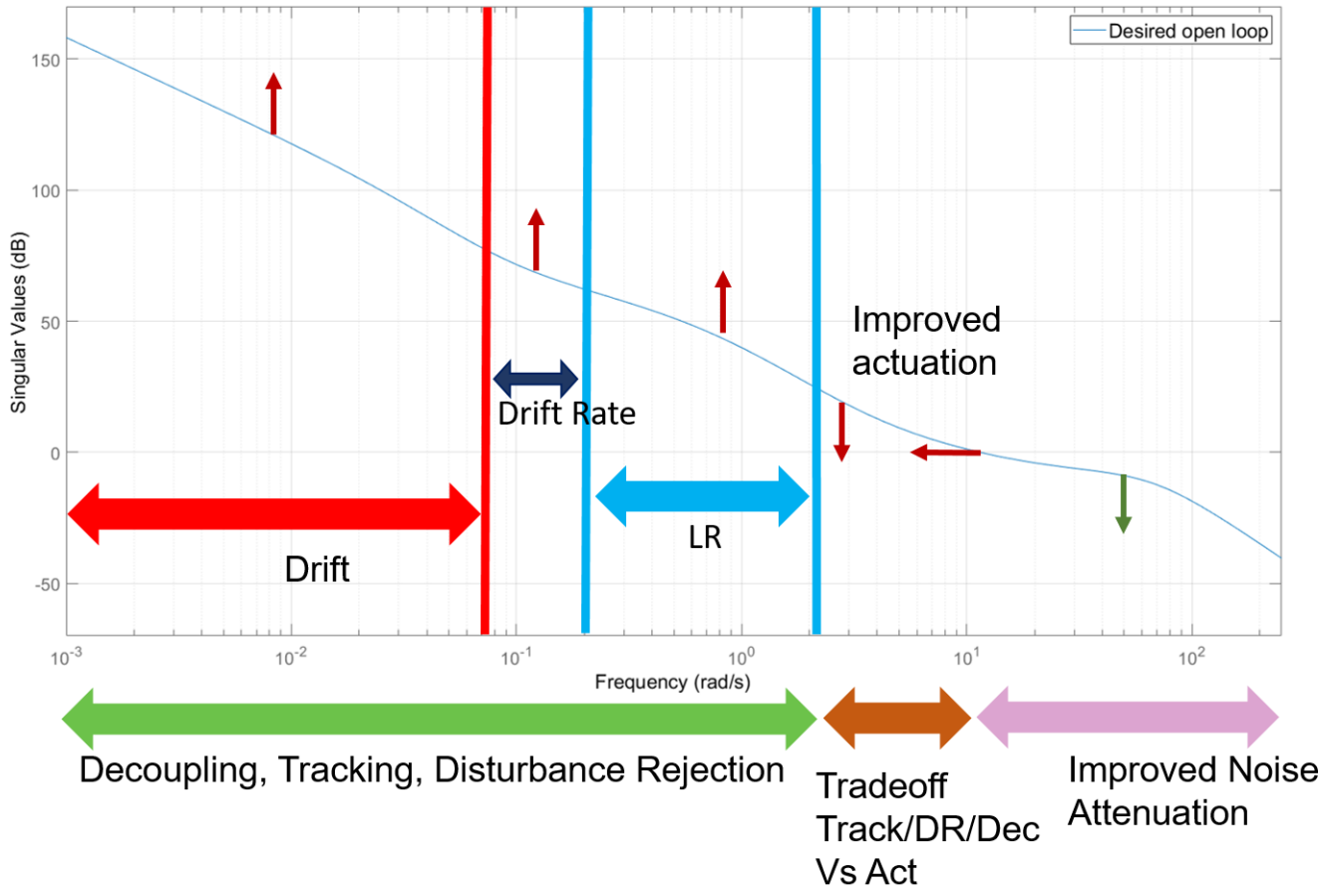

Figure 6.20: Example open loop desired shape.

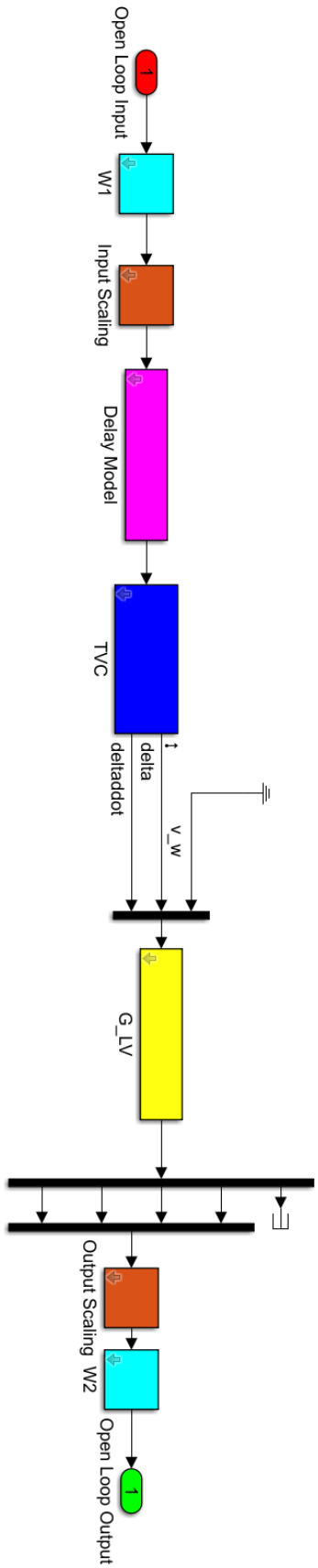

Figure 6.21: SIMULINK scheme for open loop shaping.

6.7.3. Weighting Filter Parametrization

The development of the weighting filter parametrization (and how to use it) is by far the most critical and difficult part of $\mathcal{H}_\infty$ Open Loop Shaping. This is particularly true for LVs, as no other work has built such set of weighting filters.

To help visualize the placement of the weighting filters, Figure 6.21 illustrates the scheme used for open loop shaping, as discussed in Subsection 4.10.1. The figure shows the pre-compensator and post-compensator filters, W_1 and W_2, respectively, which shape the open loop singular values. The diagram also highlights the placement of the input and output scaling.

The final set of filters follows the guidelines established by *Papageorgiou* (1998) for $\mathcal{H}_\infty$ Open Loop Shaping, which recommend placing proportional-integral action in the pre-compensator and lead-lag action in the post-compensator. The weighting filter parametrization is as follows:

$$\begin{aligned} W_1 &= Proportional - Integral \\ W_{2-z} &= Gain \\ W_{2-\dot{z}} &= Lead - Lag \\ W_{2-\theta} &= Gain \\ W_{2-\dot{\theta}} &= Lead - Lag \end{aligned} \tag{6.8}$$

As it can be seen, the pre-compensator is used to add proportional integral (PI) action on all channels. This is crucial as, in the LV case, all channels benefit from PI action (this will be further discussed later on). Moreover, this also ensures that the robust controller has the desired integrator-like shape. The post-compensator contains two gains: the drift gain is used to adjust the open loop SV at LF, whilst the pitch angle channel gain, is used to adjust the SV in the CF range. The two lead-lag (LL) filters in the post-compensator are used to delay the crossover and smoothen the curve around it.

In order to explain the effects and benefits of applying PI action on the pre-compensator, the reader must first understand the frequency dependency of the shaped plant's SV on each of the shaped plant's outputs. For that purpose, consider Figure 6.22, which compares the singular values of the open loop system, compensated with weighting filters according to the parametrization, to the singular values of the

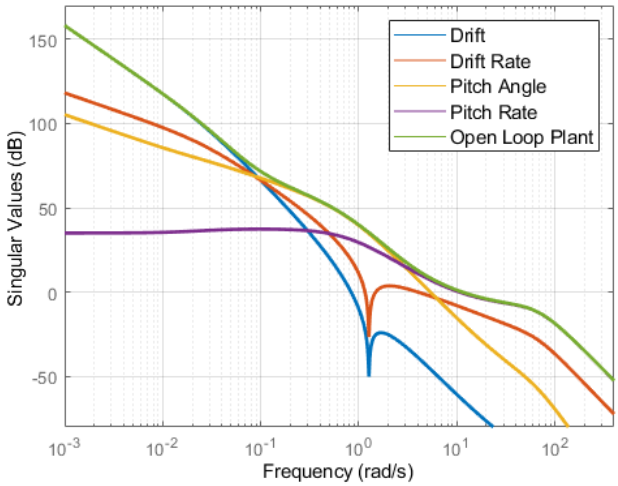

Figure 6.22: SV of compensated open loop vs SV of compensated individual input-output channels.

individual input-output channels of the same compensated system.

From the figure, it is evident that initially, the drift channel defines the open loop system response. As the frequency increases, the drift rate channel becomes dominant, until the effects of the non-minimum phase zero take place. Following this, the pitch channel takes over, and eventually the pitch rate becomes dominant for the rest of the frequency domain. It should be noted, that after the transition from pitch to pitch rate, and up until shortly after crossover, the drift rate channel has an influence on the response.

Having understood that each output channel is dominant at a certain frequency range, it is now possible to discuss the importance of the PI pre-compensator. Additionally, the role of the LL filters will also be discussed.

Consider Figure 6.23, which shows the singular value plots of the weighted system's individual input-output channels. Each subplot presents singular values for a specific input-output pair, illustrating the effect of progressively applied filter sets:

- Blue: SVs of the plant shaped with only static gains.
- Red: SVs of the shaped plant when a PI controller is included in the pre-compensator, and static gains are kept equal to those of the blue curve.
- Yellow: SVs of the shaped plant when a PI controller is included in the pre-compensator, lead-lag filters are added to the output rate channels, and static gains are kept equal to those of the blue curve.

This layered approach — from gains only, to PI control, to the full parametrization — highlights the frequency response modifications introduced by each filter set across the different input-output channels.

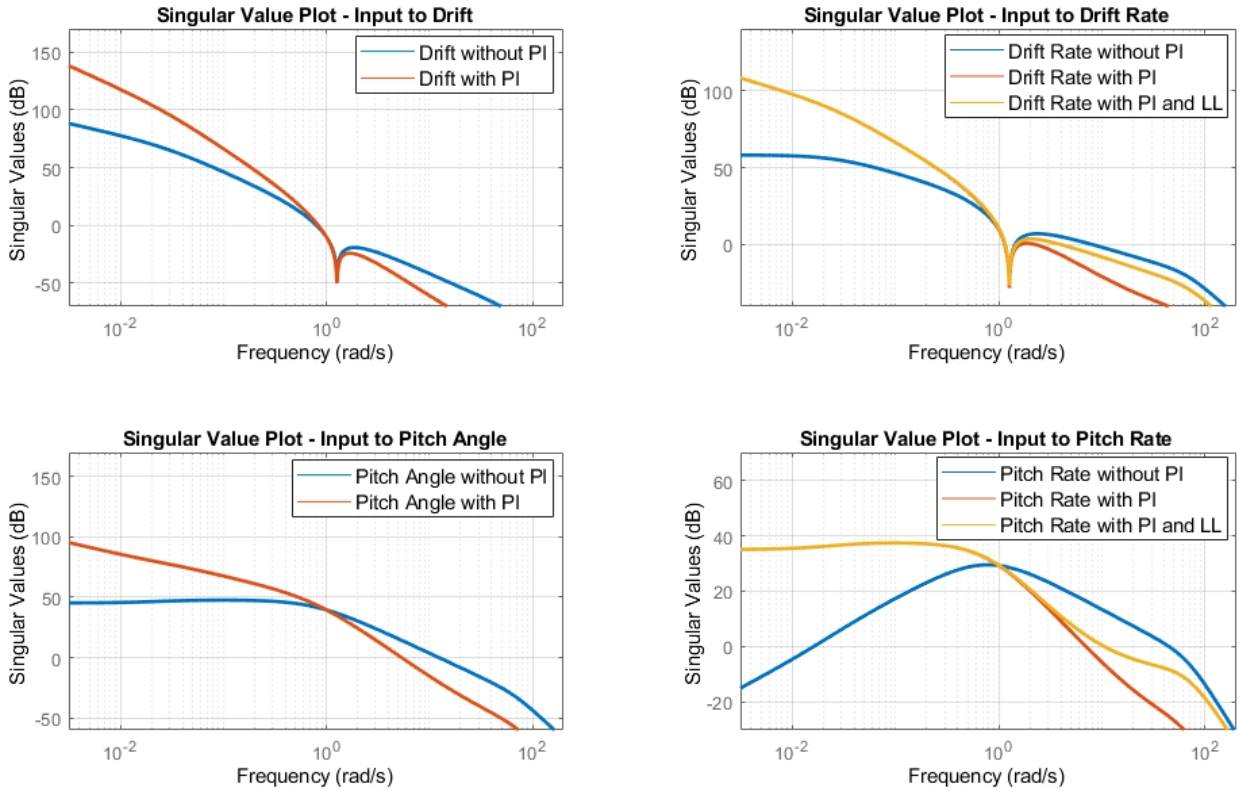

Figure 6.23: PI filter and LL filter effect on the individual channels.

It is now easier to look at each individual channel and explain the contributions of the weighting filters.

Adding integral action to the drift channel ensures a high low frequency gain and a steeper roll-off at low frequencies. As this output is dominant at low frequencies, the integral action guarantees that the open loop response achieves the desired shape in this frequency range.

Adding integral action to the drift rate channel achieves the same results as the drift channel. Moreover, it ensures that the open loop gain maintains an integrator-like shape as the drift rate channel becomes dominant. Additionally, the integral action decreases the gain after the non-minimum phase zero, which is beneficial, as it allows to further increase drift rate gain without delaying the crossover. This adjustment results in improved load relief while maintaining actuation levels, as the increased drift rate gain raises the open loop singular value in the critical frequency range.

In the pitch channel, adding integral action results in a steep SV slope, allowing for an increase of the pitch channel gain, whilst maintaining crossover. Once again, enabling better load relief.

Adding integral action to the pitch rate channel results in a steeper SV slope, enabling an increase in gain while maintaining the crossover frequency. While this does not directly contribute to load relief, it facilitates a smoother transition from the pitch-dominated region to the pitch rate-dominated region.

The role of the PI in the pitch and pitch rate channel is particularly important due to the non-minimum phase zero in the drift rate channel. The NMP zero causes a sudden decrease in the open loop SV at a frequency critical for load relief. This could only be solved through an increase in pitch/pitch rate output channel gains, as these are the channels active in the relevant frequency range. Without the integral action, the increase in gain would cause a tremendous increase in crossover frequency, leading to unacceptable actuation levels.

Finally, the lead-lag action allows for the adjustment of both crossovers in the output rate channels. As shown in Figure 6.22, these are the dominant channels near the crossover. Testing has shown that both lead-lag filters are necessary to achieve the desired open loop specifications. It should be noted, that the lead lags are only required, due to the low order of the robustifying controller (only four gains). In essence, the lead lags are aiding the robustifying controller in performing the NCF robustification.

6.7.4. How to Compute the $\mathcal{H}_\infty$ OLS Robust Controller

In practice, designing the $\mathcal{H}_\infty$ Open Loop Shaping robust controller (weighting filters plus robustifying controller) is simple. Next, the process that should be followed is described, and some guidelines are given for the selection of the weighting filters:

- Step 1 - Add Pre-Compensator, and leave Post-Compensator as identity matrix.
- Step 2 - Select W_{2_θ} to shape open loop SV in the critical frequency range.
- Step 3 - Select W_{2_z} to shape open loop SV at low frequencies.
- Step 4 - Add lead-lag filters.
- Step 5 - Compute robustifying controller
- Step 6 - Evaluate results, if needed readjust theta and drift post-compensator gains.

Note: In this work, the robustifying controller is composed of the two PDs shown in Subsection 6.4.3. This choice was motivated by the goal of maintaining the robust controller's order as close as possible to that of the $\mathcal{H}_\infty$ CLS controller while preserving the structure of the original two PDs. By selecting the two

PDs as the robustifying controller, only the weighting filters contribute to increasing the order of the robust controller.

In step 1, the zero of the pre-compensator **should** be placed slightly after the crossover frequency, so as to not affect open loop shape in the CF region. The placing of the PI zero is relatively easy to optimize - by keeping all gains and LL filters fixed, the PI zero can be varied while recomputing the robustifying controller and comparing the attained NCF robustness margin. This simple and fast analysis quickly allows for an estimation of the optimal PI zero location.

For steps two and three, the reader should recall figure 6.20: using a high value for $W_{2-\theta}$ leads to high CF open loop SV, which translates into attaining more load relief and better tracking; using a high value for W_{2-z} leads to high LF open loop gain, which translates into attaining good drift/drift rate rejection.

For the lead-lag filters, both zeros should be positioned before the crossover frequency. The placement of the poles, however, is more nuanced: the drift rate pole should be located close to the crossover frequency, either slightly before or after, while the pitch rate pole should be set at a higher frequency. Fine-tuning the lead-lag zero and pole locations may require some trial and error, but this process only needs to be done once. During this thesis, the same PI and lead-lag filters were consistently used to generate controllers for different flight points, yielding satisfactory results across all cases.

Alternatively, the lead-lag filters can be computed simultaneously with the robustifying controller to provide an initial estimate for the zero and pole locations. In this approach, Steps 1-3 are followed as described. Next, the designer sets the lead-lag filters and robustifying controller as tunable parameters in systune, optimizing both at once with the goal of maximizing the NCF robustness margin. When using this approach, it is important to carefully specify allowable ranges for the lead-lag pole and zero locations to prevent them from being placed at infinity.

Having described the general guidelines for the use of the parametrization, specific values are given for the weighting filters. These should serve as initial guesses for launch vehicles with similar configurations, launching in similar atmospheric conditions.

$$\begin{aligned} W_1 &= \frac{35(s+20)}{s} \\ W_{2-z} &= z_{range} \\ W_{2-z} &= \frac{30(s+8)}{s+16} \\ W_{2-\theta} &= \theta_{range} \\ W_{2-\dot{\theta}} &= \frac{0.44(s+1.6)}{s+80} \end{aligned} \tag{6.9}$$

where $z_{range} = [1 - 130]$, and where $\theta_{range} = [0.1 - 0.5]$.

Some ranges for W_{2-z} and for $W_{2-\theta}$ are given, testing showed them to be valid for nominal design. On the other hand, the zero location of the PI controller was chosen to improve the robustness margin. It was observed that, as long as this zero is positioned after the crossover frequency, its precise placement does not affect the loop shape prior to crossover. The lead-lag filters were similarly tuned by trial and error to maximize the robustness margin. Since these filters primarily affect the loop after the critical frequency range, they do not influence load relief.

While the defined ranges for $W_{2-\theta}$ and W_{2-z} offer significant design flexibility, there might be instances where these limits are insufficient. For instance, there might be a flight point for which the drift requirement

is not fulfilled, even if W_{2-z} is pushed to the upper end of its range. To address this, the lead-lag filters can be re-tuned to accommodate values outside the specified $W_{2-\theta}$ and W_{2-z} ranges. This is done by following the lead-lag tuning procedure that makes use of systune and that was previously described in this subsection. This adaptability enhances design freedom and ensures the system meets more stringent performance requirements across a wider range of flight conditions.

6.7.5. Computing the Controller

This section describes the process of computing the $\mathcal{H}_\infty$ Open Loop Shaping controller. It is important to note three key aspects. First, the loop shaping process is conducted using the open loop configuration shown in Figure 6.21, where the block G_{LV} contains the DMR model — the rigid body robust design model that neglects the TWD effect (recall Subsection 5.3.2).

Second, the NCF robustification process makes use of the scheme depicted in Figure 6.24, where the block G_{LV} again contains the DMR model. This scheme directly implements the approach outlined in Subsection 4.10.2.

Third, the tracker configuration discussed in Subsection 4.10.3, illustrated in Figure 6.25, is used to implement the $\mathcal{H}_\infty$ OLS controllers. In this implementation, the block G_{LV} corresponds to the AMR model — the rigid body analysis model that incorporates the TWD effect (recall Subsection 5.3.1). Additionally, the output gains and the input gains in this implementation are responsible for converting signal to and from degrees, respectively. The only exception is the command input gain, which serves two functions: it converts degrees to radians and acts as the prefilter. With this discussion in mind, the controller computation process can now be explained.

The first step was to design a controller using only the nominal model. To accomplish this, a pre-compensator, as defined in equation 6.9, was added to the input of the open loop plant. Then, $W_{2-\theta}$ was adjusted to refine the system's performance in the CF region. Specifically, the adjustment ensured that, in the CF region, the SV of the shaped plant were slightly higher than the SV of $L_O - K_{CLS-NOM-LR}$ (L_O corresponding to the nominal load relief focused $\mathcal{H}_\infty$ CLS controller). This adjustment led to $W_{2-\theta} = 0.5$.

Next, W_{2-z} was set to 50 (a random value) and an initial estimation of the lead-lags was computed with systune, utilizing the process described in the previous section. From there, W_{2-z} was progressively increased, with the lead-lag filters manually adjusted as needed. The maximum achievable drift post-compensator gain, while maintaining $\gamma \leq 4$, was determined to be 130. The final set of filters and the robustifying controller [6] are shown in table 6.8. This robust controller is coined $K_{OLS-NOM}$.

Table 6.8: Weighting filters and robustifying controller values - $K_{OLS-NOM}$.

	$K_{OLS-NOM}$
W_{2-z}	130
$W_{2-\dot{z}}$	$\frac{29.78(s+7.474)}{s+15}$
$W_{2-\theta}$	0.5
$W_{2-\dot{\theta}}$	$\frac{0.44(s+1.615)}{s+80}$
K_{p_z}	0.3810
K_{d_z}	0.6998
K_{p_θ}	-0.4694
K_{d_θ}	-0.6770

[6] The reader should recall that the robustifying controller is comprised of the two PDs shown in Subsection 6.4.3

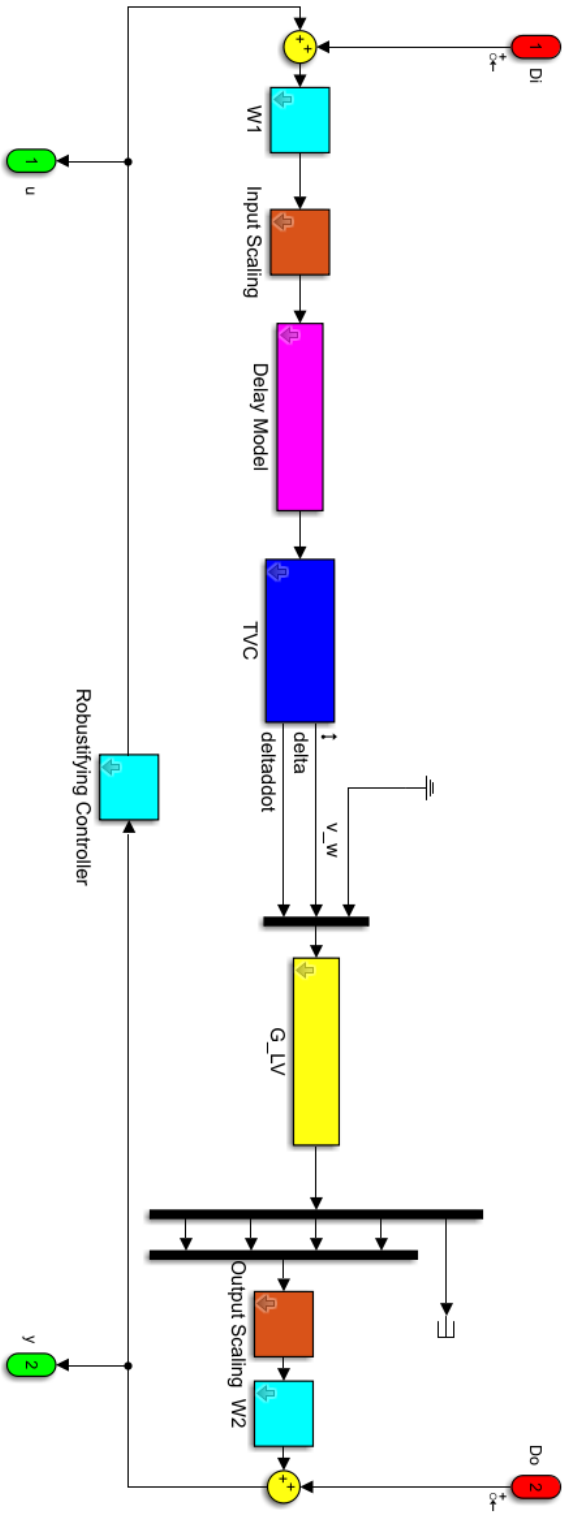

Figure 6.24: SIMULINK scheme for NCF robustification process

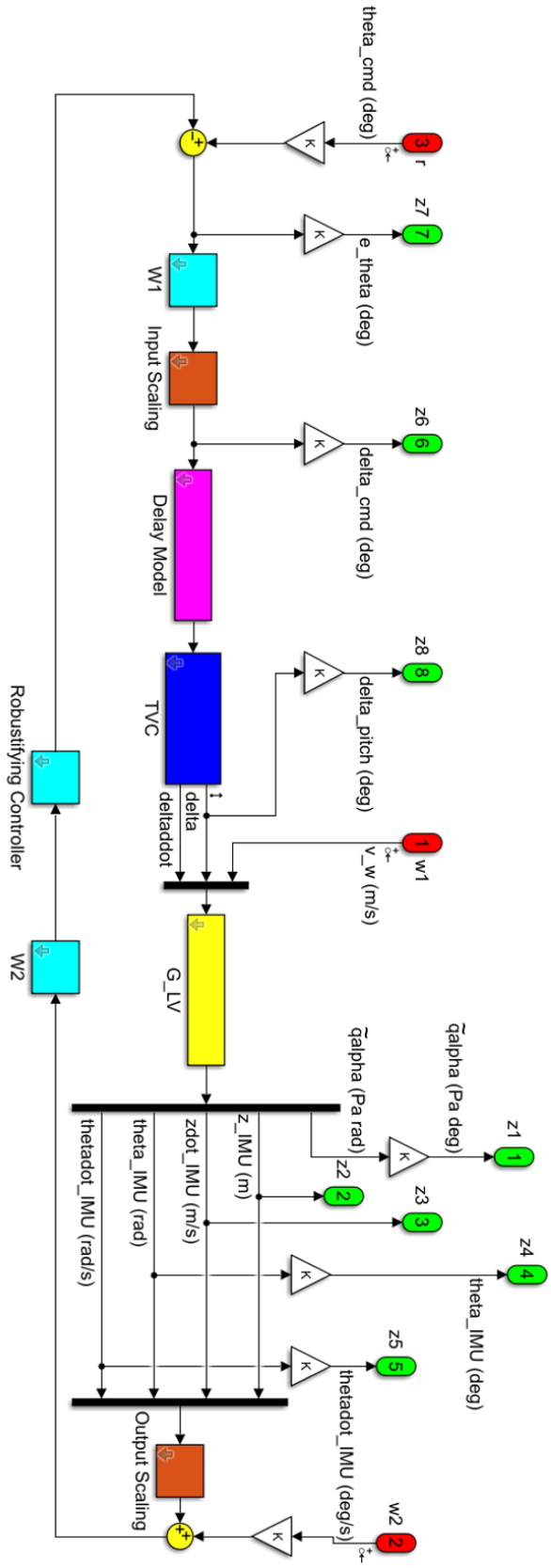

Figure 6.25: SIMULINK scheme of tracker implementation for the $\mathcal{H}_\infty$ OLS controllers.

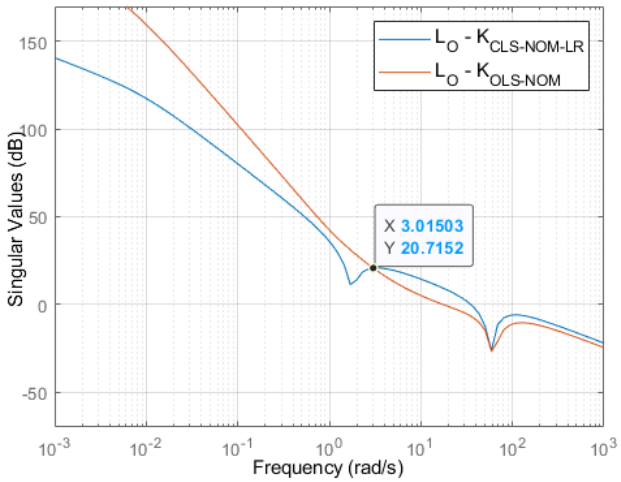

Figure 6.26: L_O SV comparison - nominal $\mathcal{H}_\infty$ CLS vs nominal $\mathcal{H}_\infty$ OLS.

In Figure 6.26, the SVs of $L_O - K_{OLS-NOM}$ are shown and compared to the SVs of $L_O - K_{CLS-NOM-LR}$. It is clear that, up to the end of the CF region, the singular values of $L_O - K_{OLS-NOM}$ are higher than those of $L_O - K_{CLS-NOM-LR}$. This led to the expectation that the nominal $\mathcal{H}_\infty$ OLS controller would outperform the nominal $\mathcal{H}_\infty$ CLS controller. Subsequent analysis confirmed this expectation, showing that $K_{OLS-NOM}$ significantly outperformed $K_{CLS-NOM-LR}$ in most tests, whilst attaining comparable results in the remaining. Detailed results can be found in Appendix C.

However, this controller was not satisfactory, as the worst-case gain analysis revealed that the launch vehicle requirements for load relief were not being upheld. This is illustrated in Figure 6.27, where the thick black line represents the maximum allowable singular value, ensuring that if the worst-case gain remains under the line, the load requirements will be upheld for a maximum wind velocity of $33\,\mathrm{m/s}$.

The solution that was found was to design the controller with the help of the Multi-model strategy. Note that the LFT-based approach was not applicable in this case, as the sole tuning objective provided to

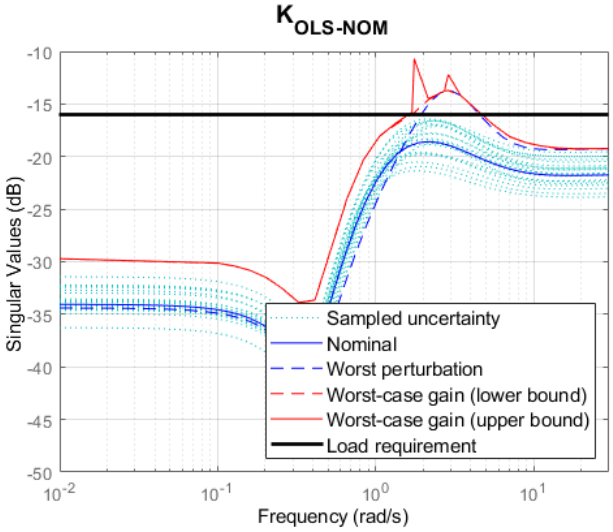

Figure 6.27: Worst-case gain of $T_{v_w \to \tilde{q}\alpha/\tilde{q}_n}$ - $K_{OLS-NOM}$.

Table 6.9: Weighting filters and robustifying controller values - $\mathcal{H}_\infty$ OLS Multi-model design.

	$K_{OLS-MM3}$	$K_{OLS-MM5}$	$K_{OLS-MM7}$
W_{2-z}	1	1	1
$W_{2-\dot{z}}$	$\frac{0.06(s+2.16)}{s+11.63}$	$\frac{12.5(s+1.8)}{s+9.3}$	$\frac{12.1(s+1.7)}{s+9.0}$
$W_{2-\theta}$	0.96	0.395	0.36
$W_{2-\dot{\theta}}$	$\frac{0.44(s+7.3)}{s+80}$	$\frac{0.37(s+5.6)}{s+80}$	$\frac{0.36(s+5.5)}{s+80}$
K_{p_z}	0.3617	0.3685	0.3687
K_{d_z}	0.3997	0.4329	0.4411
K_{p_θ}	−0.3613	−0.3765	−0.3761
K_{d_θ}	−0.8168	−0.8321	−0.8296

systune during the $\mathcal{H}_\infty$ OLS design process is the robustification against NCF uncertainty. As a result, systune would prioritize finding models that most challenge the given constraint, which may not align with cases where load relief or other performance objectives are critical.

Using the Multi-model approach, three controllers were designed, each corresponding to a different model set: the first model set consisted of the nominal model along with corner cases generated from the three uncertainties with the highest peak μ-sensitivity (see figure 6.12); the second and third model sets followed a similar approach but incorporated five and seven uncertainties, respectively. The three controllers were named $K_{OLS-MM3}$, $K_{OLS-MM5}$ and $K_{OLS-MM7}$, reflecting the number of uncertainties included in each model set.

For each model, the controller was produced to maximize load relief. This was done by selecting $W_{2-z} = 1$, and then increasing $W_{2-\theta}$ as much as possible, while slightly adapting the lead-lag filters, so that the NCF robustness margin was smaller than four. A note should be made regarding the choice of the drift post-compensator gain: it was initially selected randomly. The objective was to perform a worst-case analysis for the computed Multi-model controllers of the $T_{v_w \to z_{IMU}}$ transfer function, aiming to identify the smallest possible drift post-compensator gain (or slightly above it) that would satisfy the drift performance requirements. During the analysis, it was found that selecting $W_{2-z} = 1$ was already very close to the minimum allowable value, meaning no further adjustments were necessary.

The post compensator values utilized for each Multi-model controller, and the robustifying controller [7] values are shown in Table 6.9. It should be noted that for all three controllers, the pre-compensator used is the same as the one described in equation 6.9.

With these controllers computed, the worst-case gains of the transfer functions $T_{v_w \to \tilde{q}\alpha/\tilde{q}_n}$ and $T_{v_w \to z_{IMU}}$ were evaluated. Results showed that while $K_{OLS-MM3}$ performed adequately in terms of load relief, it failed to meet the drift requirements. This outcome was expected due to the results presented in Subsection 6.6.6, where it was shown that while the first three uncertainties with the highest μ-sensitivities are sufficient to achieve the desired load relief, they are insufficient to meet the drift requirements.

On the other hand, both $K_{OLS-MM5}$ and $K_{OLS-MM7}$ were shown to meet the load and drift requirements, as can be visualized in Figures 6.28 and 6.29. In these figures the large triangular peaks were confirmed to be numerical artifacts. And the thick black lines still represent the maximum allowed singular values, ensuring that if the plots remain bellow them, the drift and load relief requirements are upheld.

[7] The reader should recall that the robustifying controller is comprised of the two PDs, as discussed in Subsection 6.4.3

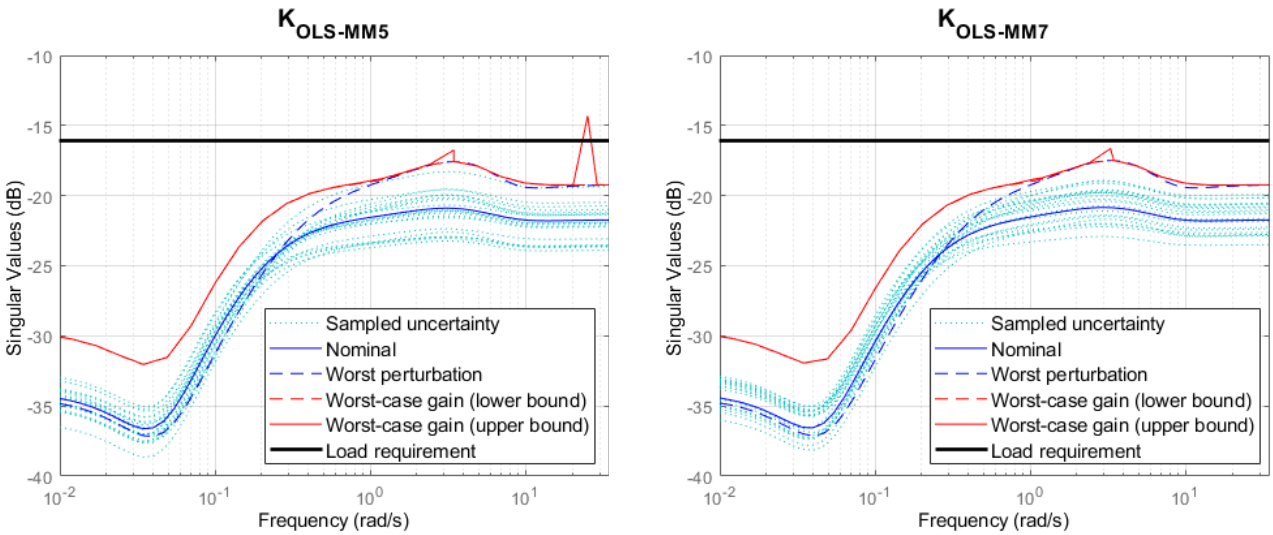

Figure 6.28: Worst-case gain of $T_{v_w \to \tilde{q}\alpha/\tilde{q}_n}$ - $K_{OLS-MM5}$ vs $K_{OLS-MM7}$.

In terms of load (see Figure 6.28), $K_{OLS-MM5}$ and $K_{OLS-MM7}$ deliver comparable performance, as is demonstrated by the similar worst-case gain upper bound. However, in terms of drift (see Figure 6.29), $K_{OLS-MM7}$ performs slightly better than $K_{OLS-MM5}$, as evidenced by the smaller singular values of the worst-case gain upper bound for $K_{OLS-MM7}$.

Finally, an evaluation of worst-case actuation for $K_{OLS-MM5}$ and for $K_{OLS-MM7}$, revealed that both perform equally well. The actuation comparison in Figure 6.30 shows similar maximum actuator demands, indicating comparable performance under worst-case conditions.

Additional testing confirmed that $K_{OLS-MM7}$ is expected to meet all requirements. When comparing it to $K_{OLS-MM5}$, $K_{OLS-MM7}$ showed similar performance in terms of load relief and actuation, but superior results in terms of drift rejection. Considering everything, $K_{OLS-MM7}$ was chosen for comparison with $K_{CLS-LFT-LR}$.

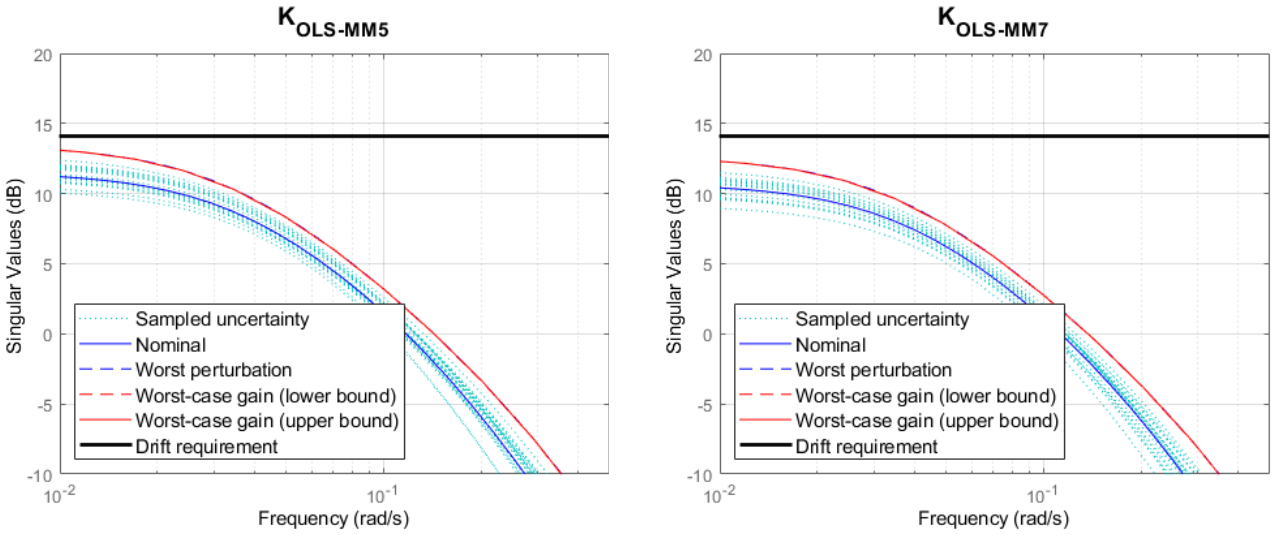

Figure 6.29: Worst-case gain of $T_{v_w \to z_{IMU}}$ - $K_{OLS-MM5}$ vs $K_{OLS-MM7}$.

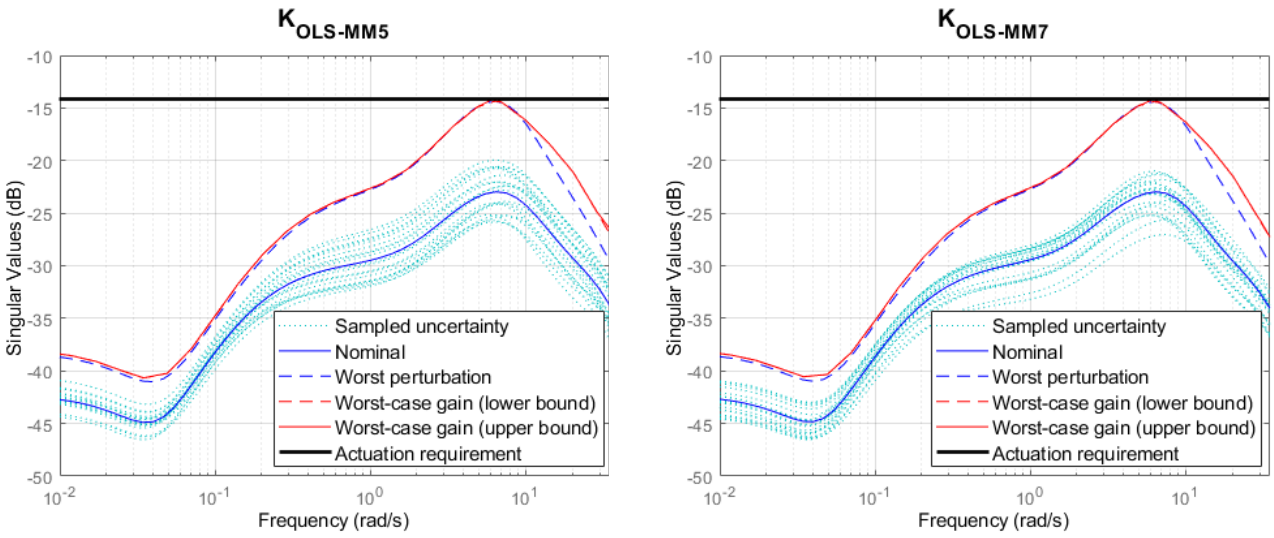

Figure 6.30: Worst-case gain of $T_{v_w \to \delta_{cmd}}$ - $K_{OLS-MM5}$ vs $K_{OLS-MM7}$.

6.8. Controller Comparison and Analysis

In this section, the pole placement controller (K_{PP}), the 7 uncertainty Multi-model $\mathcal{H}_\infty$ Open Loop Shaping controller ($K_{OLS-MM7}$) and the load relief focused LFT $\mathcal{H}_\infty$ Closed Loop Shaping controller ($K_{CLS-LFT-LR}$) will be compared. The comparison entails the tests described in Section 6.3. Before presenting the comparison, two important remarks are highlighted in the following paragraph.

As previously discussed, the implementation of K_{PP} and $K_{CLS-LFT-LR}$ follows a framework similar to the control design scheme illustrated in Figure 6.11, with three key differences. First, the input scaling is now exclusively responsible for converting values from degrees to radians. Second, an additional output, denoted as δ_{pitch}, is introduced to represent the signal δ generated by the TVC block. Finally, the block G_{LV} incorporates the rigid body analysis model (AMR), as detailed in Subsection 5.4. In contrast, $K_{OLS-MM7}$ is implemented using the tracker implementation shown in Figure 6.25, where the block G_{LV} also contains the rigid body analysis model (AMR).

6.8.1. Stability Margins

Only the input Nichols and the table summarizing the stability margins are shown here. The rest of the Nichols plot can be found in Appendix D.

Input Stability Margins

As illustrated in Figure 6.31, the pole placement controller has good nominal margins, but very small worst-case (WC) margins. It should be noted, that they are large enough to fulfill the LV requirements.

Both $\mathcal{H}_\infty$ controllers have good nominal and very good worst-case margins. Nonetheless, there is a slight superiority of the $\mathcal{H}_\infty$ CLS controller in the worst-case disk margin.

Stability Margin Summary

In Table 6.10, the nominal and worst-case disk-based margins for the three controllers are presented. The gain margin are the S-T disk-based gain margins and they are in dB. The phase margin are the correspondent S-T disk-based phase margins. The frequency is in rad/s, and represents the frequency in

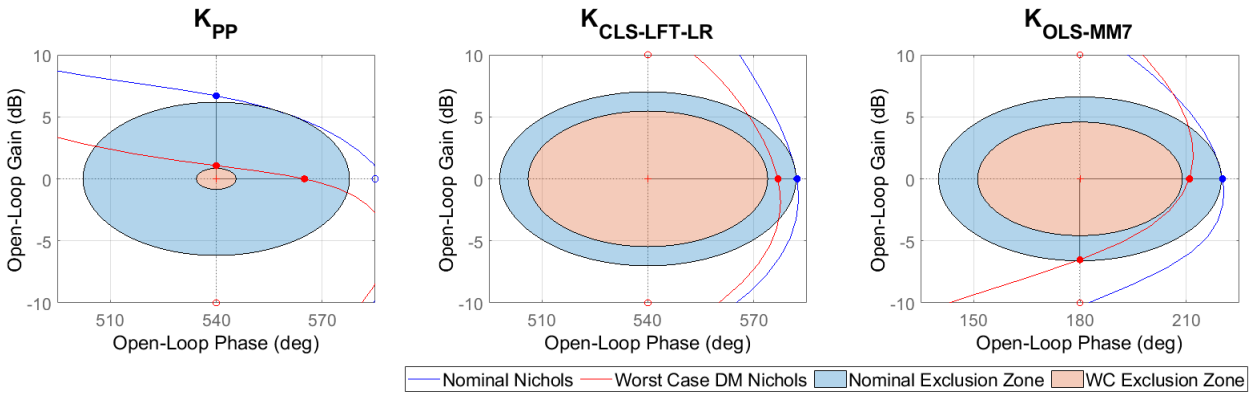

Figure 6.31: Input nichols - K_{PP} vs $K_{CLS-LFT-LR}$ vs $K_{OLS-MM7}$.

which the disk margin was computed.

There are several reasons to use S-T disk-based margins instead of classical ones— for a full discussion, the reader is encouraged to consult *Seiler* et al. (2020). However, the main reason for their use in this work is the fact that they are balanced (the allowable gain/phase increase and decrease are symmetric, as was discussed in Section 4.8).

It should be noted that the gain/phase S-T disk margins will always be equal to or smaller than the classical gain/phase margins. Therefore, if the gain/phase S-T disk margins exceed the established values for the classical margins in the controller requirements, it can be concluded that the requirements are being satisfied.

It is clear from Table 6.10, that the $\mathcal{H}_\infty$ CLS controller has the advantage at the input margin, despite the $\mathcal{H}_\infty$ OLS controller having very large margins. In the worst-case scenario the system with the $\mathcal{H}_\infty$ OLS

Table 6.10: Stability margins summary - K_{PP} vs $K_{CLS-LFT-LR}$ vs $K_{OLS-MM7}$.

		K_{PP}			$K_{CLS-LFT-LR}$			$K_{OLS-MM7}$		
		GM	PM	Freq	GM	PM	Freq	GM	PM	Freq
Nominal	Input	6.18	37.72	1.34	7.01	41.93	5.11	6.61	39.93	8.88
WC	Input	0.65	5.60	1.21	5.47	33.91	4.18	4.58	28.90	14.35
Nominal	LAT drift	11.31	59.58	0.48	8.56	49.06	0.10	18.99	77.19	0.09
	LAT drift rate	11.67	60.75	0.86	35.46	88.07	16.53	19.02	77.23	15.12
	LAT pitch angle	5.99	36.70	1.05	8.31	47.96	4.38	11.91	61.52	3.61
	LAT pitch rate	10.41	56.44	0.85	10.90	58.17	8.72	8.29	47.88	9.48
WC	LAT drift	6.98	41.76	1.15	5.82	35.80	0.07	15.31	70.52	0.07
	LAT drift rate	2.00	13.10	1.20	22.23	81.16	0.94	14.29	68.15	15.71
	LAT pitch angle	0.76	5.02	1.21	5.74	35.38	0.08	5.45	33.78	3.66
	LAT pitch rate	1.59	10.44	1.20	7.78	45.56	9.22	5.40	33.55	15.99
Nominal	MIMO Output	2.46	16.02	0.76	3.21	20.69	0.10	4.59	28.97	3.60
WC	MIMO Output	0.38	2.47	1.19	2.22	14.50	0.08	3.74	23.91	4.99
Nominal	I/O	1.96	12.79	0.93	2.66	17.30	4.80	2.82	18.30	8.25
WC	I/O	0.26	1.72	1.20	2.09	13.63	4.03	1.99	12.99	13.68

controller has a 4.58 dB S-T disk-based gain margin, which is much larger than the requirement of 0.5 dB.

For the loop-at-a-time (LAT) outputs, the $\mathcal{H}_\infty$ OLS controller has the advantage in the drift channel, whilst the $\mathcal{H}_\infty$ CLS controller has the advantage in the remainder channels. However, it should be noted that the $\mathcal{H}_\infty$ OLS controller margins are still very large in these cases.

For the MIMO output margins, it can be seen that the pole placement controller has very small worst-case margins, notably only 0.38 dB S-T disk gain margin. Both $\mathcal{H}_\infty$ controllers have good margins, but the $\mathcal{H}_\infty$ OLS controller takes the clear advantage, having more 1.5 dB S-T disk gain margin in the worst-case scenario, when compared to the $\mathcal{H}_\infty$ CLS controller.

For the input/output (I/O) margins, once again, the pole placement controller has very small worst-case margins - 0.26 dB S-T disk gain margin. Both $\mathcal{H}_\infty$ controllers have similar nominal and worst-case margins, with the CL controller having more 0.1 dB disk gain margin in the worst-case scenario.

Taking everything into consideration, it is clear that both $\mathcal{H}_\infty$ controllers have great results in both the nominal and worst-case scenarios. Meanwhile, the pole placement controller has much worse results in the worst-case scenarios. Nonetheless, all three controllers uphold the rigid body stability margin requirements defined in Section 6.3. Out of both $\mathcal{H}_\infty$ controllers, it is hard to say which is superior in terms of stability, as each has the upper hand in different stability margins. Moreover, even when one performs better, the other still has quite large margins. So, in terms of stability, it was concluded that the $\mathcal{H}_\infty$ controllers are evenly matched.

6.8.2. Time Domain Analysis

For the following analysis, the color coding for the controllers follows the scheme defined in Figure 6.32. Additionally, the shaded regions in the plots represent the results of simulations under dispersed conditions.

Figure 6.32: Color code for time domain analysis - K_{PP} vs $K_{CLS-LFT-LR}$ vs $K_{OLS-MM7}$.

Low Frequency, High Amplitude (LFHA) Wind Gust

From Figure 6.33, it is clear that the controller with the smallest load performance metric (LPM) peak value is the pole placement controller. This is not surprising, as both of $\mathcal{H}_\infty$ controllers focus on robust performance, whilst the PP controller was designed without knowledge of the uncertainty. When comparing the $\mathcal{H}_\infty$ controllers, the $\mathcal{H}_\infty$ OLS controller has a much smaller peak value, and its shaded region also has overall smaller values.

For both drift and drift rate (Figures 6.34 and 6.35), the pole placement controller performs the best again, for the same reasons. Meanwhile, the $\mathcal{H}_\infty$ OLS controller outperforms the $\mathcal{H}_\infty$ CLS controller, having much smaller drift and drift rate in nominal and dispersed conditions.

By observing Figure 6.36, it becomes clear that for the pitch angle, under both nominal and dispersed conditions, the $\mathcal{H}_\infty$ CLS controller consistently shows the lowest values throughout most of the simulation, with the $\mathcal{H}_\infty$ OLS controller and the PP controller showing progressively higher values.

For the pitch rate (Figure 6.37), the $\mathcal{H}_\infty$ CLS controller demonstrates the smallest peak but exhibits a slower response compared to the $\mathcal{H}_\infty$ OLS controller, in which pitch rate returns to 0 deg/s much more

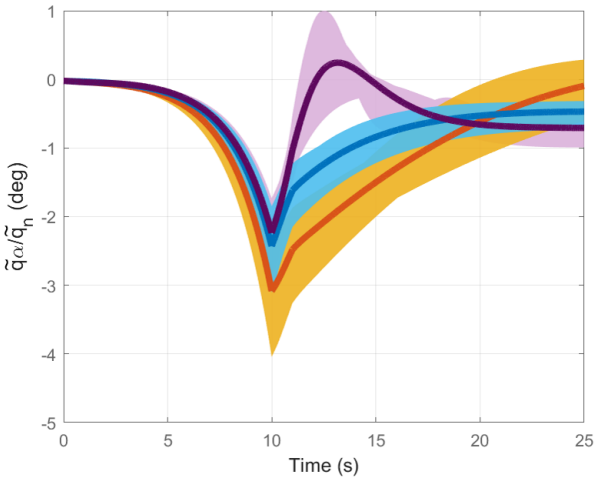

Figure 6.33: LFHA Wind - Load performance metric (LPM), normalized by the nominal dynamic pressure.

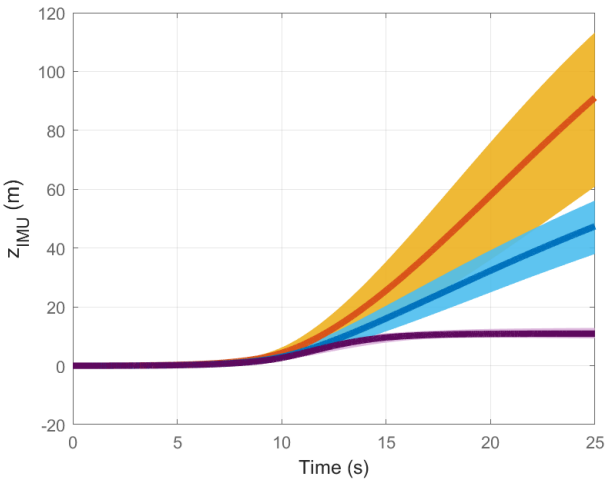

Figure 6.34: LFHA Wind - Drift.

rapidly. In contrast, the PP controller shows a significantly higher peak value in pitch rate and greater sensitivity to dispersed conditions.

In terms of actuation (Figure 6.38), the $\mathcal{H}_\infty$ OLS and PP controllers achieve the smallest peak values, with the $\mathcal{H}_\infty$ OLS controller also exhibiting the least dispersed variation and a lower actuation rate. By comparison, the $\mathcal{H}_\infty$ CLS controller shows the largest peak actuation value, which is significantly higher than the others.

Overall, the pole placement controller is the controller with best results in terms of load relief, drift and drift rate rejection, however this is at the expense of large pitch angle and pitch rate peak values. Following this, the $\mathcal{H}_\infty$ OLS controller presents the second best results in terms of load relief, drift and drift rate rejection, it also performs better than the pole placement controller in terms of pitch angle and pitch rate. Meanwhile, the $\mathcal{H}_\infty$ CLS controller has the worse results in terms of LPM, drift, drift rate, and actuation, but the best in terms of pitch angle and pitch rate.

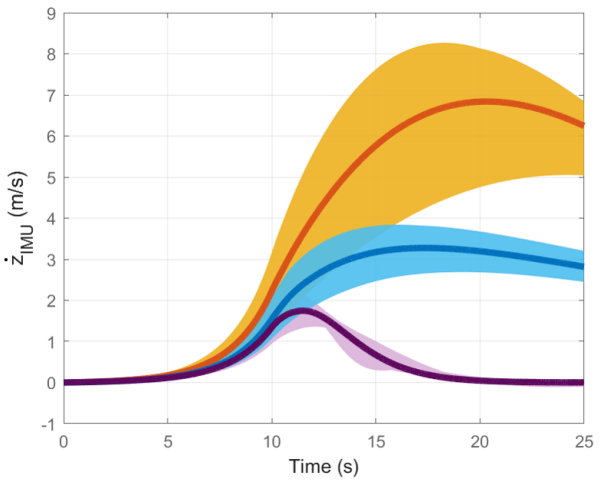

Figure 6.35: LFHA Wind - Drift rate.

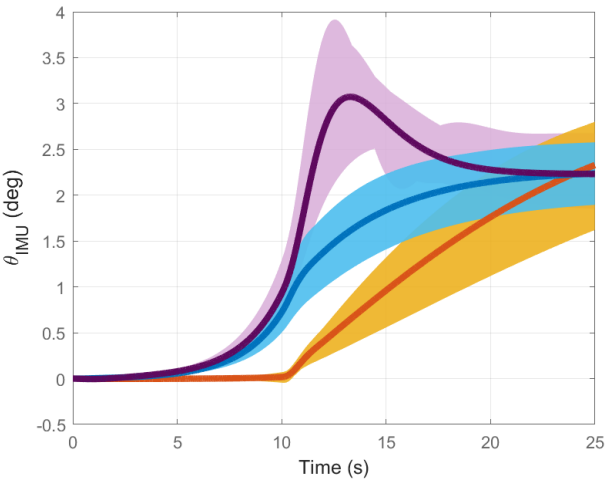

Figure 6.36: LFHA Wind - Pitch angle.

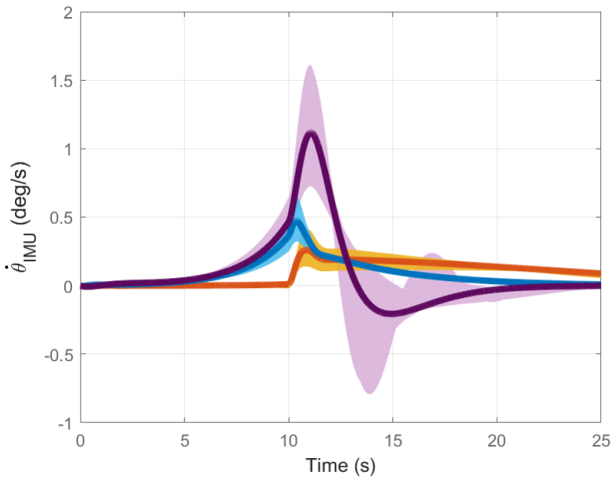

Figure 6.37: LFHA Wind - Pitch rate.

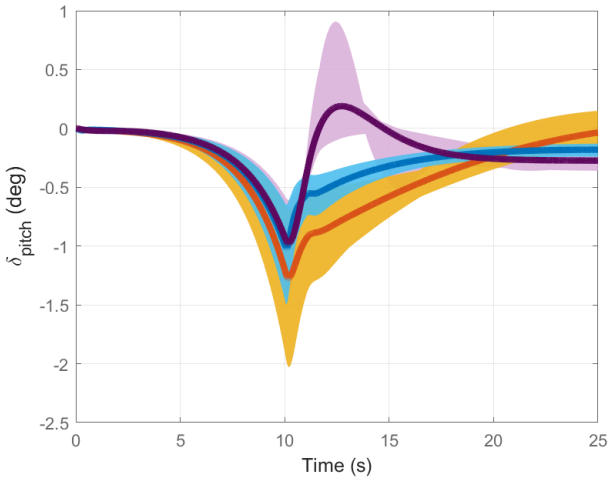

Figure 6.38: LFHA Wind - TVC deflection.

Tracking Test - 0.5 deg step on pitch angle command

It is clear from Figure 6.39, that the $\mathcal{H}_\infty$ OLS controller has the smallest LPM throughout the response, with $\mathcal{H}_\infty$ CLS being the second best, and the PP controller the worse.

In terms of pitch angle tracking (see Figure 6.40), the $\mathcal{H}_\infty$ OLS controller is the only capable of tracking without overshoot, and with zero steady-state tracking error (a results of the integral action and the pre-filter). Additionally, it also has the shortest settling time, and the second shortest rising time.

From Figure 6.41, it is clear that the $\mathcal{H}_\infty$ OLS controller and PP have similar peak pitch rate, with the $\mathcal{H}_\infty$ CLS controller having much higher peak value.

In terms of actuation (see Figure 6.42), the $\mathcal{H}_\infty$ OLS controller and the PP controller have similar peak value and actuation rate. Both controllers have much smaller peak value than the $\mathcal{H}_\infty$ CLS controller, which also has much higher actuation rate.

It should be noted that the observed trends are also seen in the dispersed conditions of each re-

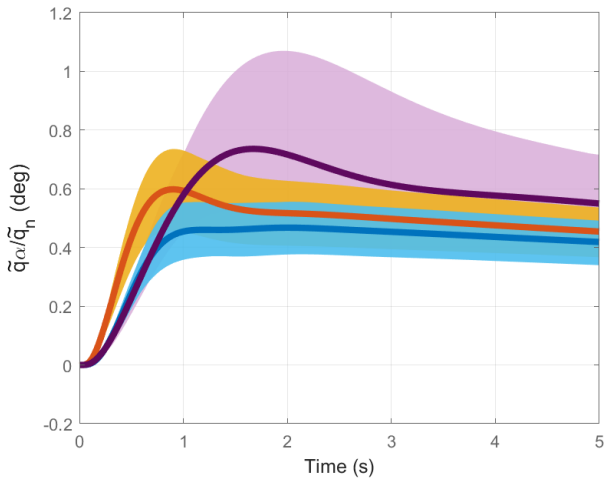

Figure 6.39: Pitch Angle Tracking - LPM, normalized by the nominal dynamic pressure.

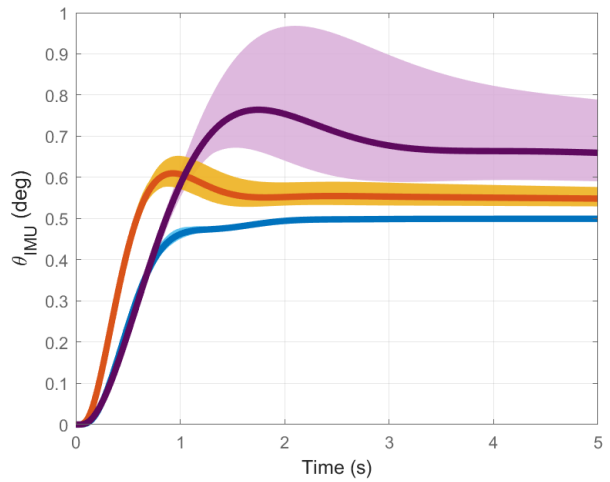

Figure 6.40: Pitch Angle Tracking - Pitch angle.

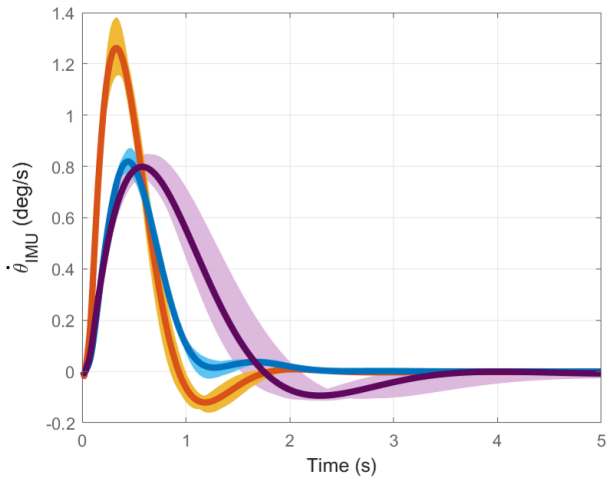

Figure 6.41: Pitch Angle Tracking - Pitch rate.

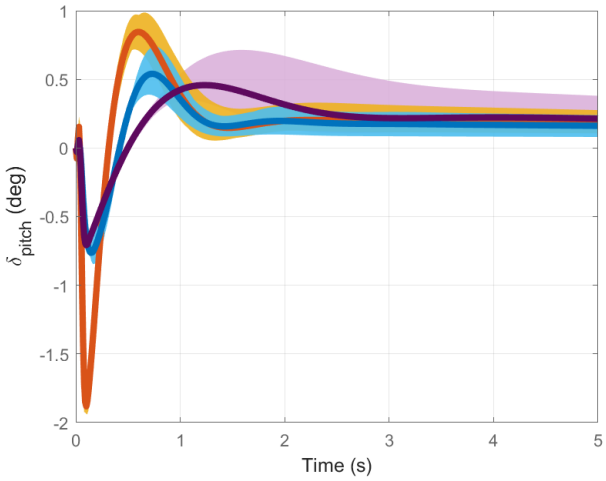

Figure 6.42: Pitch Angle Tracking - TVC deflection.

sponse. Overall, the $\mathcal{H}_\infty$ OLS controller performs by far the best, having smaller LPM, better tracking, and comparable actuation to the PP controller.

High Frequency, Low to Mid Amplitude (HFLMA) Wind Gust

In the High frequency wind gust simulation, not much can be said regarding the LPM (Figure 6.43). If the plot is magnified, it will be seen that there is not much difference between the different controllers.

In drift and drift rate (Figures 6.44 and 6.45), the controllers have similar nominal response. In terms of dispersed conditions, the pole placement controller performs the best in drift, and the $\mathcal{H}_\infty$ OLS controller performs the best in drift rate.

From Figures 6.46 and 6.47, it can be seen that for pitch angle and pitch rate, the $\mathcal{H}_\infty$ controllers have similar nominal and dispersed responses, meanwhile the PP controllers has much worse results.

For the actuation (Figure 6.48), the $\mathcal{H}_\infty$ OLS controller has the largest nominal and dispersed response. The $\mathcal{H}_\infty$ CLS controller and the PP controller have similar results.

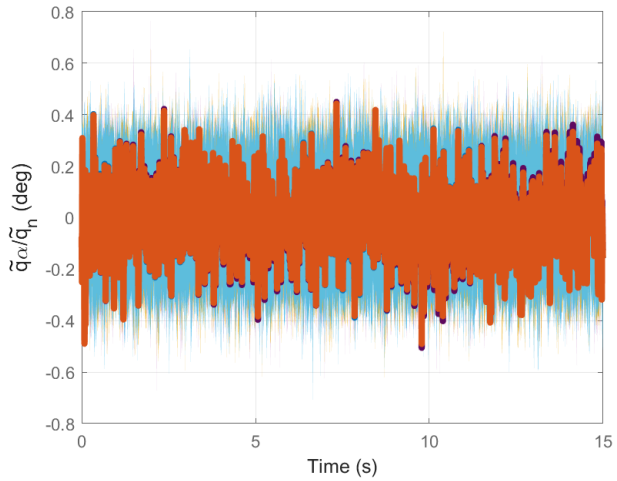

Figure 6.43: HFLMA Wind - LPM, normalized by the nominal dynamic pressure.

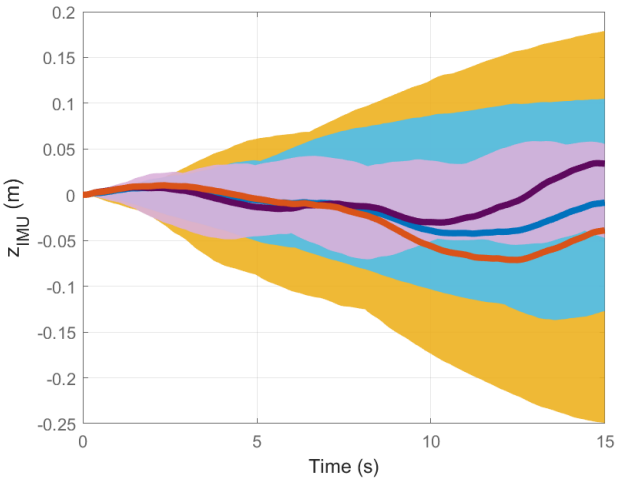

Figure 6.44: HFLMA Wind - Drift.

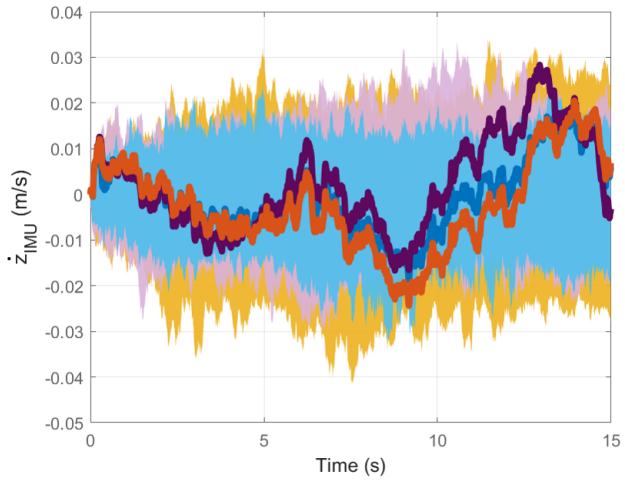

Figure 6.45: HFLMA Wind - Drift rate.

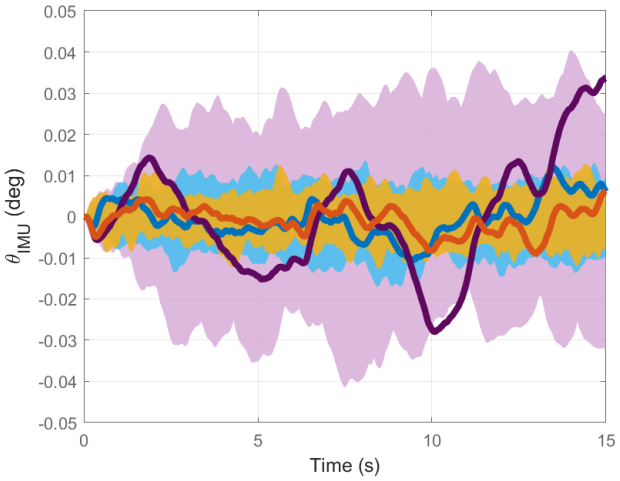

Figure 6.46: HFLMA Wind - Pitch angle.

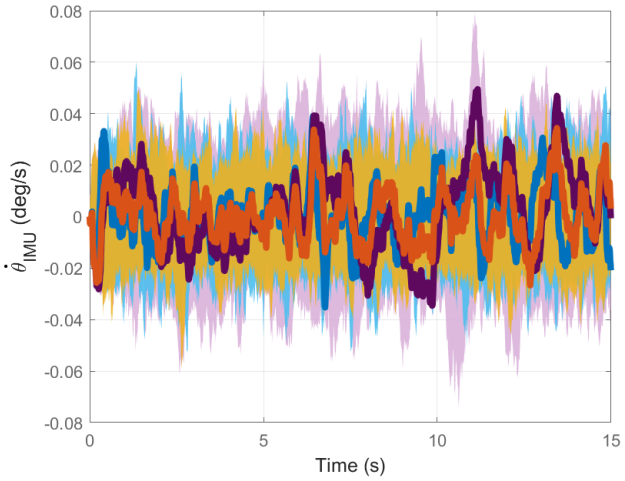

Figure 6.47: HFLMA Wind - Pitch rate.

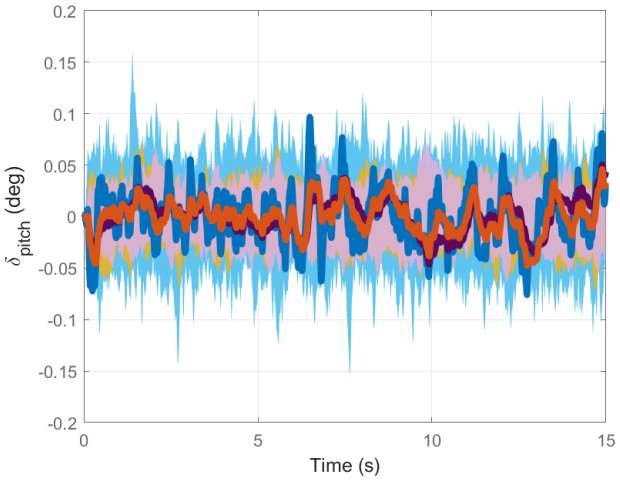

Figure 6.48: HFLMA Wind - TVC deflection.

Overall, the $\mathcal{H}_\infty$ OLS controller performs better than the $\mathcal{H}_\infty$ CLS controller, having better or equal results in five out of six plots, with only slightly worse results in the sixth plot. However, when compared to the pole placement controller, the comparison is more narrow, as the $\mathcal{H}_\infty$ OLS controller only has superior or equal results in four of the six plots, with significantly worse results in the other two.

6.8.3. Worst-Case Gain Analysis

In the following worst-case analysis, the thick black lines represent the maximum allowable singular values for the respective test. Meaning that if the upper bound of the worst-case gain remains bellow the black line, then the respective requirement is fulfilled. Additionally, the large triangular peaks where the worst-case gain upper bound significantly differs from the lower bound are numerical artifacts. This has been confirmed through the computation of the worst-case gain using more rigorous methods. In reality, for these regions, the upper bound is almost indistinguishable from the lower bound.

Load Relief

The strength of the $\mathcal{H}_\infty$ controllers, particularly the $\mathcal{H}_\infty$ OLS controller, is best seen in the worst-case analysis. From Figures 6.49, 6.50 and 6.51, it is clear that only the $\mathcal{H}_\infty$ controllers are capable of withstanding the expected winds, with the modeled uncertainty. Moreover, the $\mathcal{H}_\infty$ OLS controller is able to withstand higher winds than the $\mathcal{H}_\infty$ CLS controller, 37 m/s vs 34 m/s, respectively.

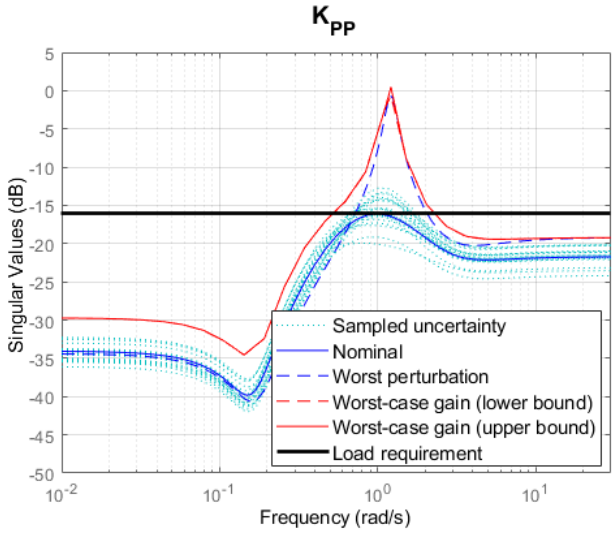

Figure 6.49: Worst-case gain of $T_{v_w \to \tilde{q}\alpha/\tilde{q}_n}$ - K_{PP}.

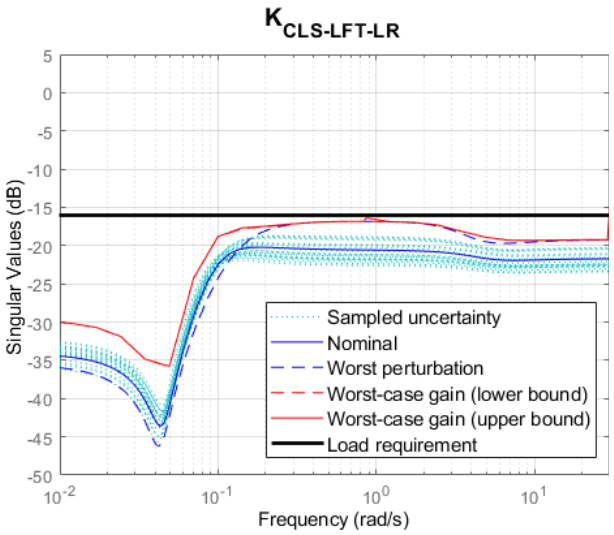

Figure 6.50: Worst-case gain of $T_{v_w \to \tilde{q}\alpha/\tilde{q}_n}$ - $K_{CLS-LFT-LR}$.

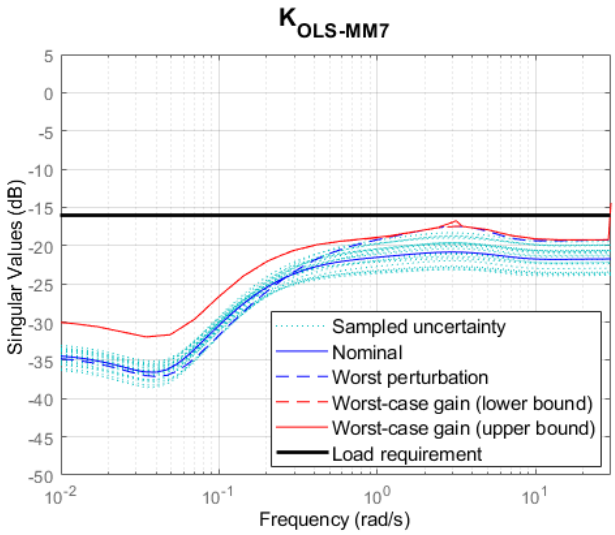

Figure 6.51: Worst-case gain of $T_{v_w \to \tilde{q}\alpha/\tilde{q}_n}$ - $K_{OLS-MM7}$.

Drift and Drift Rate

From the worst-case gain plots of $T_{v_w \to z_{IMU}}$ (Figure 6.52) and $T_{v_w \to \dot{z}IMU}$ (Figure 6.53), it is clear that the $\mathcal{H}_\infty$ OLS controller clearly outperforms the $\mathcal{H}_\infty$ CLS controller. For the WC gain upper bound of $T_{v_w \to z_{IMU}}$, the $\mathcal{H}_\infty$ OLS controller achieves a slightly lower peak (12 dB vs. 13 dB) and shows an earlier, sharper reduction in singular values, when compared to the $\mathcal{H}_\infty$ CLS controller. Moreover, while the $\mathcal{H}_\infty$ CLS controller barely meets the drift rate requirement (second plot of Figure 6.53), the $\mathcal{H}_\infty$ OLS controller stays well below the threshold, as is evidenced by the third plot of Figure 6.53. Interestingly, the

PP controller also demonstrates very strong drift and strong drift rate rejection (see the first plot of Figures 6.52 and 6.53), consistent with the high amplitude wind analysis in Figures 6.34 and 6.35.

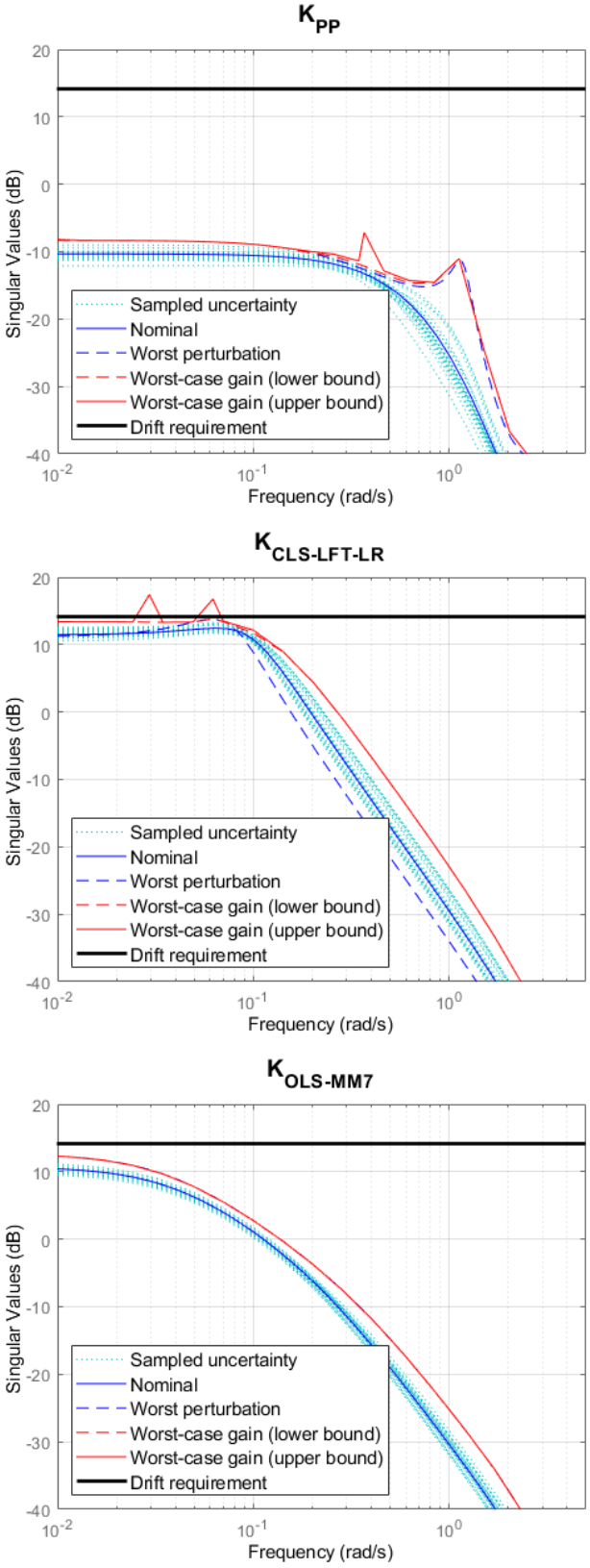

Figure 6.52: Worst-case gain - $T_{v_w \to z_{IMU}}$ - K_{PP} vs $K_{CLS-LFT-LR}$ vs $K_{OLS-MM7}$.

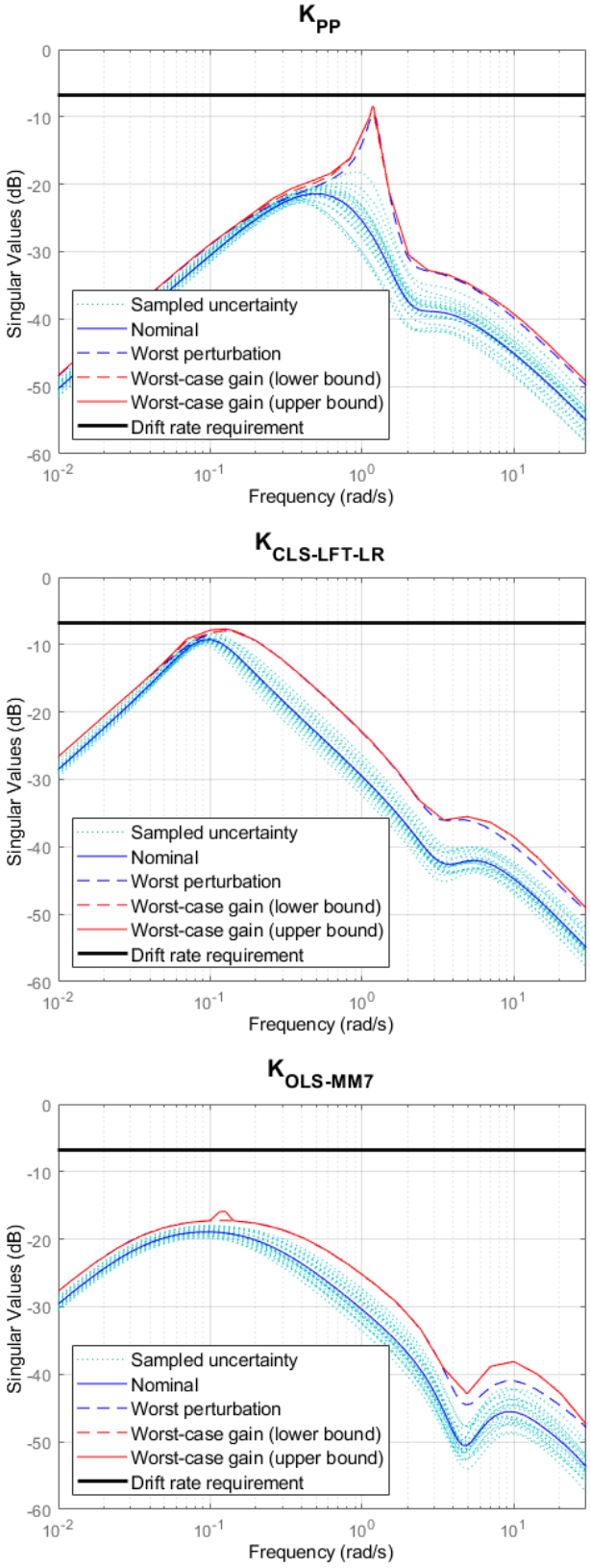

Figure 6.53: Worst-case gain - $T_{v_w \to \dot{z}_{IMU}}$ - K_{PP} vs $K_{CLS-LFT-LR}$ vs $K_{OLS-MM7}$.

Actuation

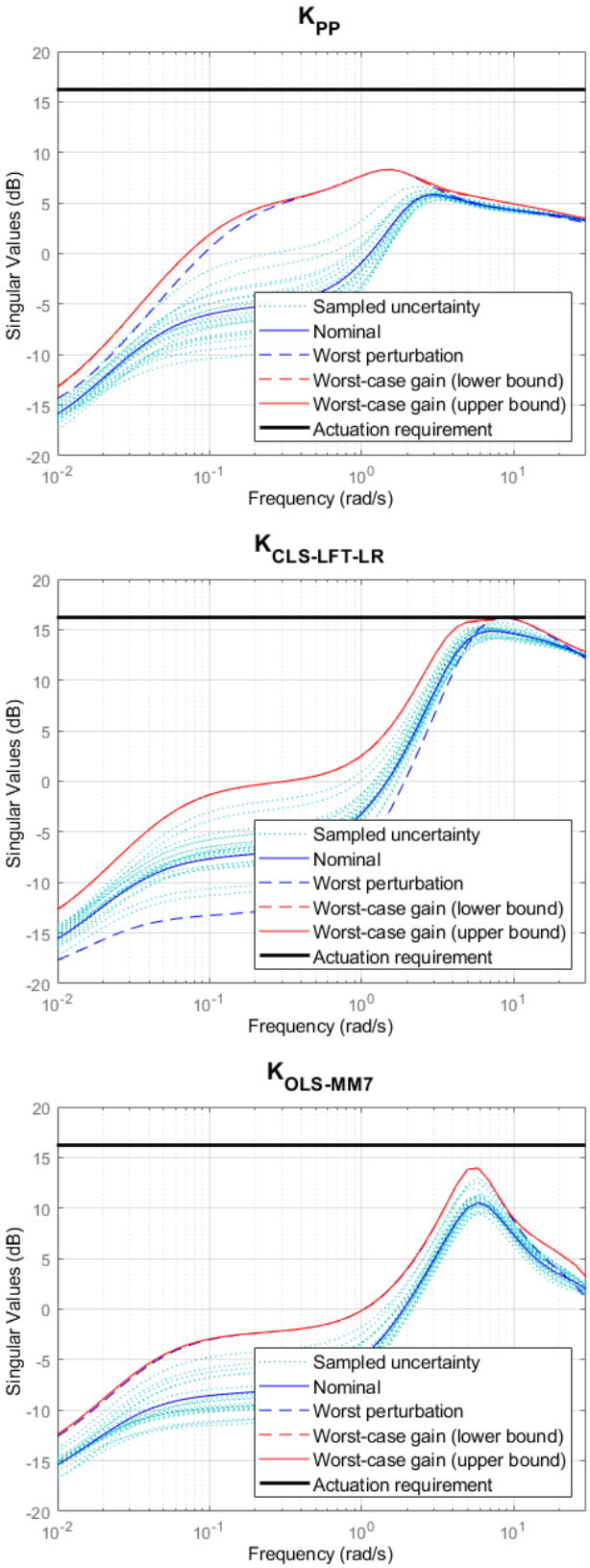

Figure 6.54: Worst-case gain - $T_{\theta_{cmd} \to \delta_{cmd}}$ - K_{PP} vs $K_{CLS-LFT-LR}$ vs $K_{OLS-MM7}$.

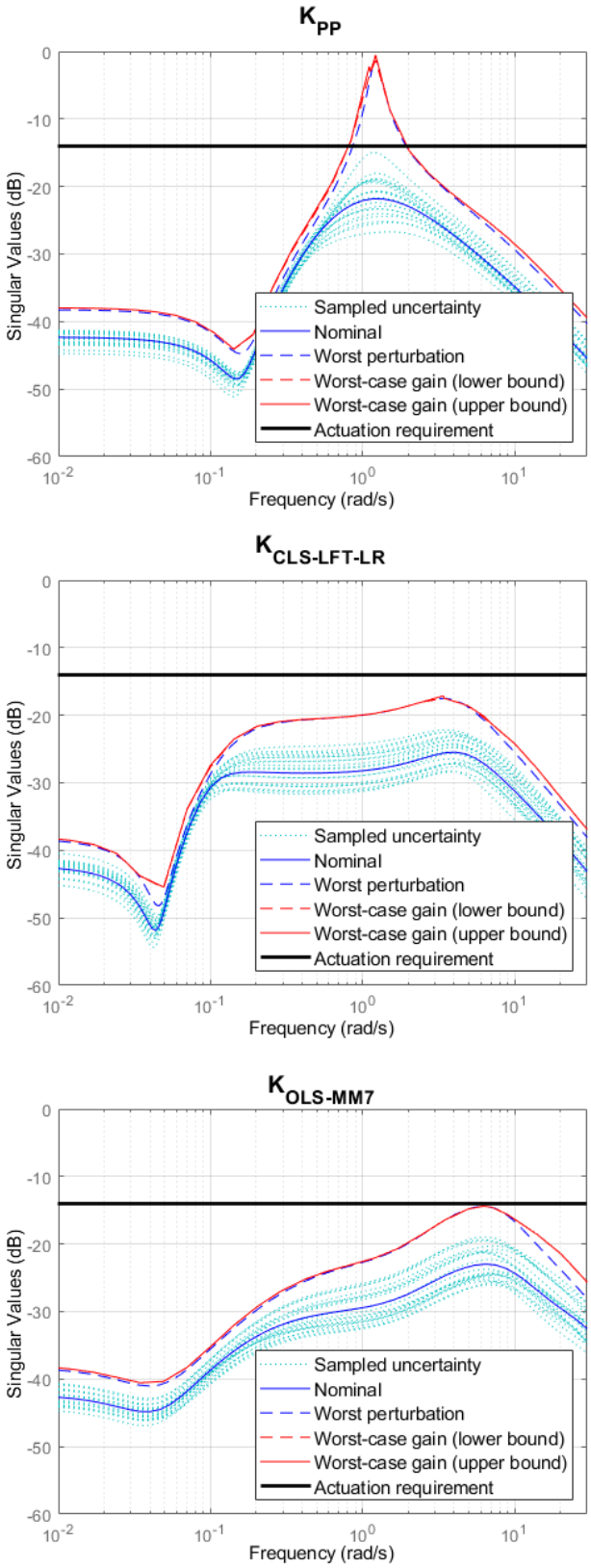

Figure 6.55: Worst-case gain - $T_{v_w \to \delta_{cmd}}$ - K_{PP} vs $K_{CLS-LFT-LR}$ vs $K_{OLS-MM7}$.

In terms of actuation, the PP controller clearly infringes one of the requirements, as shown in the worst-case gain of $T_{v_w \to \delta_{cmd}}$ (first plot of Figure 6.55). This is caused by its very high drift rejection.

Both $\mathcal{H}_\infty$ controllers meet the actuation requirements (see second and third plots of Figures 6.54 and 6.55), but the $\mathcal{H}_\infty$ CLS controller only marginally satisfies the requirement in the worst-case gain of $T_{\theta_{cmd} \to \delta_{cmd}}$ (second plot of Figure 6.54). A similar issue occurs for the $\mathcal{H}_\infty$ OLS controller in the worst-case gain of $T_{v_w \to \delta_{cmd}}$ (third plot of Figure 6.55).

Note that for the worst-case gain of $T_{\theta_{cmd} \to \delta_{cmd}}$, the drift and drift rate feedbacks were disconnected due to the specific characteristics of the LV model - see Subsection 6.4.3 for a detailed explanation.

6.8.4. Structured Singular Value

The peak value of the upper bound of the Structured Singular Value of the system with each controller is shown in Table 6.11.

Table 6.11: Structured Singular Value comparison - K_{PP} vs $K_{CLS-LFT-LR}$ vs $K_{OLS-MM7}$.

	PP	$\mathcal{H}_\infty$ CLS	$\mathcal{H}_\infty$ OLS
SSV Upper Bound Peak Value	0.86	0.365	0.376

As shown in Table 6.11, the PP controller is very close to instability, with its peak μ value near one. In contrast, the $\mathcal{H}_\infty$ controllers demonstrate strong robustness against structured uncertainty, with both having μ peak values below 0.4. The frequency-dependent SSV plots for the $\mathcal{H}_\infty$ CLS and $\mathcal{H}_\infty$ OLS controllers are shown in Figure 6.56. The figure clearly illustrate that both controllers not only have small peak values but also exhibit even lower values throughout the rest of the evaluated frequency range, further highlighting their robustness. The frequency-dependent SSV plot for the PP controller is not shown for brevity, but can be found in Appendix D.

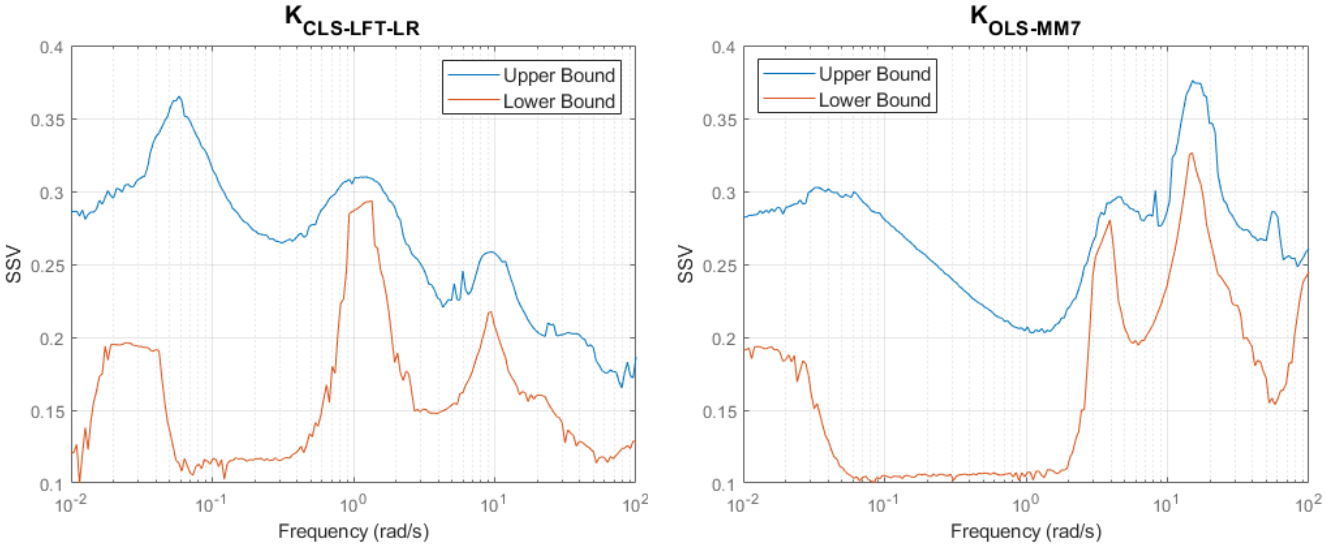

Figure 6.56: Structured Singular Value - $K_{CLS-LFT-LR}$ vs $K_{OLS-MM7}$.

6.8.5. Comparison Conclusion

The previous subsections presented a comprehensive comparison of the controllers designed using Pole Placement, $\mathcal{H}_\infty$ Closed Loop Shaping, and $\mathcal{H}_\infty$ Open Loop Shaping. This subsection provides an overall conclusion based on the results of that comparison.

In terms of stability margins, it was demonstrated that the two $\mathcal{H}_\infty$ controllers performed equally well, and significantly better than the PP controller.

In terms of the time domain simulations, it was shown that the $\mathcal{H}_\infty$ OLS controller completely outperformed the $\mathcal{H}_\infty$ CLS controller in all three simulations. Compared to the PP controller, the $\mathcal{H}_\infty$ OLS controller performed best in tracking, arguably performed the best in high-frequency wind rejection tests, though it showed slightly worse performance in high amplitude wind rejection. This outcome is primarily because the PP controller is overtuned for the nominal case.

The $\mathcal{H}_\infty$ Open Loop Shaping approach demonstrated its advantages most clearly in the worst-case gain analysis. This controller tolerated 9% higher wind speeds under worst-case conditions than the $\mathcal{H}_\infty$ CLS controller, showed superior drift rejection, significantly outperformed in drift rate disturbance rejection, and maintained similar actuation performance. A comparison to the PP controller in the worst-case analysis is not relevant, as the PP controller cannot meet all performance requirements.

Overall, the $\mathcal{H}_\infty$ Open Loop Shaping approach demonstrated a significant performance advantage over the $\mathcal{H}_\infty$ Closed Loop Shaping approach. It is important to note, however, that the controllers differ in order: the $\mathcal{H}_\infty$ CLS controller consists of four gains, whereas the $\mathcal{H}_\infty$ OLS controller includes three additional first-order filters to shape the system's open loop response. While $\mathcal{H}_\infty$ OLS achieves superior performance, it does so at the cost of increased controller order.

This raises an important question: is the observed difference in performance a result of the increased controller order? The following section will address this question.

6.9. Higher Order $\mathcal{H}_\infty$ Closed Loop Shaping

To address the question raised in the previous section, a controller matching the order of the $\mathcal{H}_\infty$ OLS controller will be designed using $\mathcal{H}_\infty$ CLS. To achieve this, $\mathcal{H}_\infty$ CLS will be used to simultaneously tune the original PD controllers and a set of filters with the same parametrization as the one used in the $\mathcal{H}_\infty$ OLS design. The design model used for this process is illustrated in Figure 6.57. In this model, an input channel labeled di has been added to represent the plant input disturbance. Additionally, an output channel labeled u_p has been introduced to represent the plant input. The block "Controller" contains the two original PDs. Meanwhile, the block G_{LV} contains the rigid body robust design model (DMR), as described in Subsection 5.3.2. The input and output gains are set equal to the ones used earlier for the $\mathcal{H}_\infty$ CLS design, as described in Subsection 6.4.3. It is also important to note that the open loop scaling employed throughout the $\mathcal{H}_\infty$ OLS designs is incorporated here. This inclusion simplifies the result comparison without diminishing the significance or validity of the results.

The first step in this approach was to reproduce the controller generated with $\mathcal{H}_\infty$ OLS. However, this was not trivial, as $\mathcal{H}_\infty$ CLS inherently produces controllers with unbalanced L_I/L_O gains, whereas $\mathcal{H}_\infty$ OLS automatically balances both gains. Utilizing the constraint set used for the original $\mathcal{H}_\infty$ CLS design (but changing the value of each constraint), and adding either one or two additional constraints to specifically shape L_I, it was possible to reproduce the nominal $\mathcal{H}_\infty$ OLS controller ($K_{OLS-NOM}$) in two different ways. It should be noted that for both optimizations, the drift and theta angle post-compensator gains were set equal to those of the nominal $\mathcal{H}_\infty$ OLS controller. This simplification does not affect the

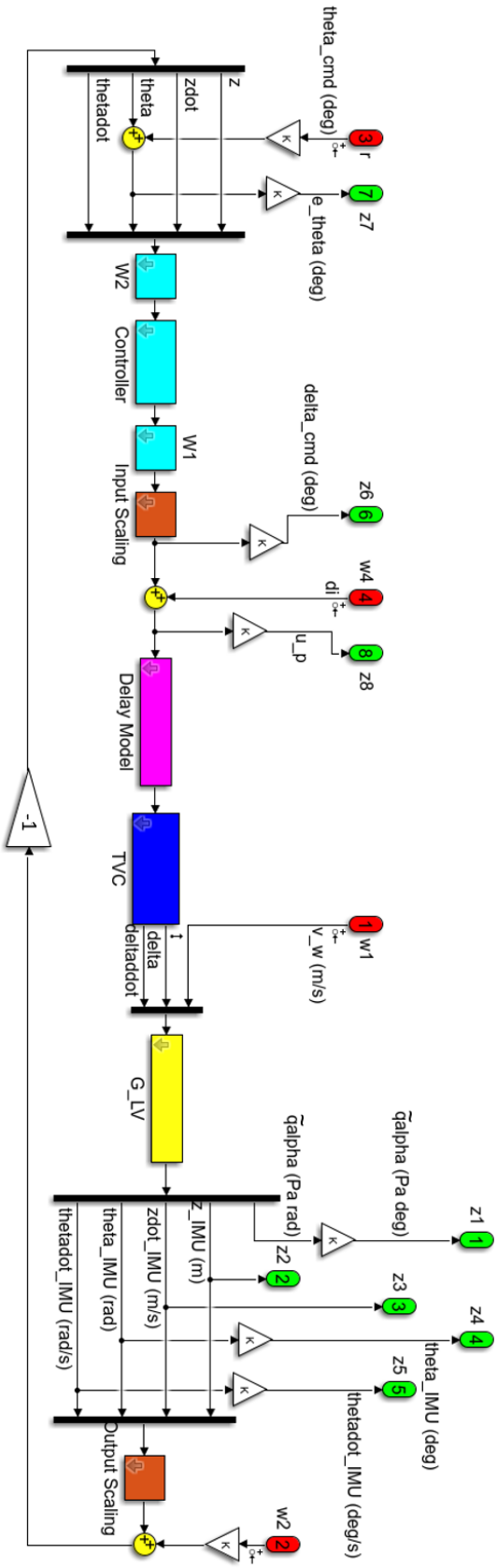

Figure 6.57: SIMULINK implementation of the control design scheme for the higher order $\mathcal{H}_\infty$ CLS controller.

Table 6.12: Higher order (HO) nominal design $\mathcal{H}_\infty$ CLS optimization constraints - set 1 and 2.

Condition	Constraint Set 1	Constraint Set 2
$\|\|T_{v_w \to \tilde{q}\alpha}\|\|_\infty < W_1$	$W_1 = 2.05 \cdot 10^5$	$W_1 = 2.05 \cdot 10^5$
$\|\|T_{v_w \to z_{IMU}}\|\|_\infty < W_2$	$W_2 = 0.0099$	$W_2 = 0.0099$
$\|\|T_{\theta_{cmd} \to \delta_{cmd}}\|\|_\infty < W_3$	$W_3 = 0.0668$	$W_3 = 0.0668$
$\|\|T_{\theta_{cmd} \to e_\theta}\|\|_\infty < W_4$	$W_4 = 1.3656$	$W_4 = 1.3656$
$\bar{\sigma}(T_{di \to u_p}) < W_6$	$W_6 = \left(\frac{h_{di}s+\omega_{di}}{s+\omega_{di}/l_{di}}\right)^2$	$W_6 = 0.2570$
Input Stability Disk Margin:	Gain Margin > 5.18 dB Phase Margin > 32.2 deg	– –

validity of the results, as testing has shown that, regardless of the constraints on these two gains, the resulting loop gains (L_O and L_I) remain essentially the same. The only differences are in the ratios $K_{p_\theta}/W_{2-\theta}$ and K_{p_z}/W_{2-z}.

The sets of constraints used to reproduce the nominal $\mathcal{H}_\infty$ OLS controller are shown in Table 6.12. For the first constraint set, the parameters for the high-pass filter W_6 are: $h_{di} = 0.822$, $\omega_{di} = 10^{-3}$: rad/s, and $l_{di} = 8.6 \cdot 10^{-6}$. The first constraint set, though slightly more complex than the second, yields a controller (denoted as $K_{CLS-NOM-HO-1}$) that closely matches the nominal $\mathcal{H}_\infty$ OLS controller, as will be demonstrated next. In comparison, the simpler second constraint set results in a controller (denoted as $K_{CLS-NOM-HO-2}$) that provides a somewhat less precise match.

Figure 6.58 compares the loop gains of the controllers designed using the first and second sets of constraints with those of the nominal $\mathcal{H}_\infty$ OLS controller. The figure also includes the loop gains of the nominal load relief-focused $\mathcal{H}_\infty$ CLS controller ($K_{CLS\text{-}NOM\text{-}LR}$), highlighting the improvements achieved by increasing the controller order. From the figure, it is evident that the controller produced using the first constraint set ($K_{CLS-NOM-HO-1}$) closely matches the nominal $\mathcal{H}_\infty$ OLS controller, while the second controller ($K_{CLS-NOM-HO-2}$) provides a slightly less accurate match. Notably, the second controller demonstrates that by adding a single constraint ($\bar{\sigma}(T_{di \to u_p}) < W_6$) to the original $\mathcal{H}_\infty$ CLS constraint set, it

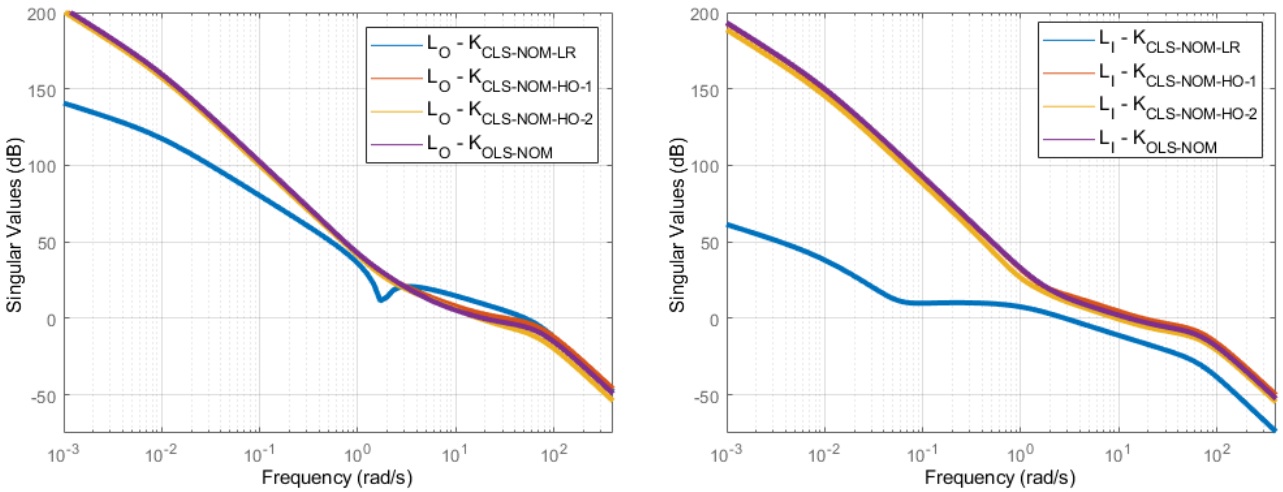

Figure 6.58: Output open loop (on the left) and input open loop (on the right) SV comparison - $K_{OLS-NOM}$ vs $K_{CLS-NOM-LR}$ vs $K_{CLS-NOM-HO-1}$ vs $K_{CLS-NOM-HO-2}$.

Table 6.13: Weighting filters and robustifying controller comparison - $K_{OLS-NOM}$ vs $K_{CLS-NOM-HO-1}$ vs $K_{CLS-NOM-HO-2}$.

	$K_{OLS-NOM}$	$K_{CLS-NOM-HO-1}$	$K_{CLS-NOM-HO-2}$
W_1	$\frac{35\,(s+20)}{s}$	$\frac{30.328\,(s+20.12)}{s}$	$\frac{35\,(s+16.39)}{s}$
$W_{2-\dot{z}}$	$\frac{29.778\,(s+7.474)}{(s+15)}$	$\frac{38.97\,(s+5.005)}{(s+10.34)}$	$\frac{49.709\,(s+4.285)}{(s+10.21)}$
$W_{2-\dot{\theta}}$	$\frac{0.44271\,(s+1.615)}{(s+80)}$	$\frac{0.74664\,(s+2.428)}{(s+79.92)}$	$\frac{0.2453\,(s+3.711)}{(s+80)}$
K_{p_θ}	-0.4694	-0.5272	-0.3292
K_{d_θ}	-0.6770	-0.6325	-0.9280
K_{p_z}	0.3810	0.4198	0.2731
K_{d_z}	0.6998	0.6333	0.3099

is possible to achieve results that closely align with those of the nominal $\mathcal{H}_\infty$ OLS controller.

Table 6.13 further compares the weighting filters and robustifying controllers of $K_{CLS-NOM-HO-1}$ and $K_{CLS-NOM-HO-2}$ with those of $K_{OLS-NOM}$. The data presented in this table clearly indicates that the results obtained with the $\mathcal{H}_\infty$ CLS approach are very similar to those of the nominal $\mathcal{H}_\infty$ OLS approach.

After reproducing the nominal $\mathcal{H}_\infty$ OLS controller, the 7-uncertainty Multi-model $\mathcal{H}_\infty$ OLS controller ($K_{OLS-MM7}$) was also recreated using the constraint set shown in Table 6.14. The parameters of the high-pass filter W_6 for this optimization are: $h_{di} = 0.891$, $\omega_{di} = 10^{-3}$: rad/s, and $l_{di} = 1.6181 \cdot 10^{-5}$. During the optimization, the drift and theta post-compensator gains were selected to be equal to those of $K_{OLS-MM7}$ for the same reasons as before. The controller obtained from this optimization is referred to as $K_{CLS-MM7-HO}$.

Figure 6.59 compares the L_I and L_O loop gains of $K_{OLS-MM7}$ and $K_{CLS-MM7-HO}$. As shown, the loop gains of the two controllers align almost perfectly. This alignment is further confirmed in Table 6.15, which compares the computed weighting filters and robustifying controllers, highlighting the close match between the two designs.

Having reproduced both the nominal and MM $\mathcal{H}_\infty$ OLS controllers using the $\mathcal{H}_\infty$ CLS approach, the next step was to use the latter method to explore potential improvements over the $\mathcal{H}_\infty$ OLS controllers. For this purpose, using solely the nominal model, many attempts were made to improve on the achieved loop

Table 6.14: Higher order MM $\mathcal{H}_\infty$ CLS optimization constraints.

Condition	Constraint Set
$\|\|T_{v_w \to \tilde{q}\alpha}\|\|_\infty < W_1$	$W_1 = 2.317 \cdot 10^5$
$\|\|T_{v_w \to z_{IMU}}\|\|_\infty < W_2$	$W_2 = 0.93$
$\|\|T_{\theta \to \delta_{cmd}}\|\|_\infty < W_3$	$W_3 = 0.0562$
$\|\|T_{\theta \to \theta_e}\|\|_\infty < W_4$	$W_4 = 1.38$
$\bar{\sigma}(T_{di \to u_p}) < W_6$	$W_6 = \left(\frac{h_{di}s+\omega_{di}}{s+\omega_{di}/l_{di}}\right)^2$
Input Stability Disk Margin:	Gain Margin > 5 dB Phase Margin> 31 deg

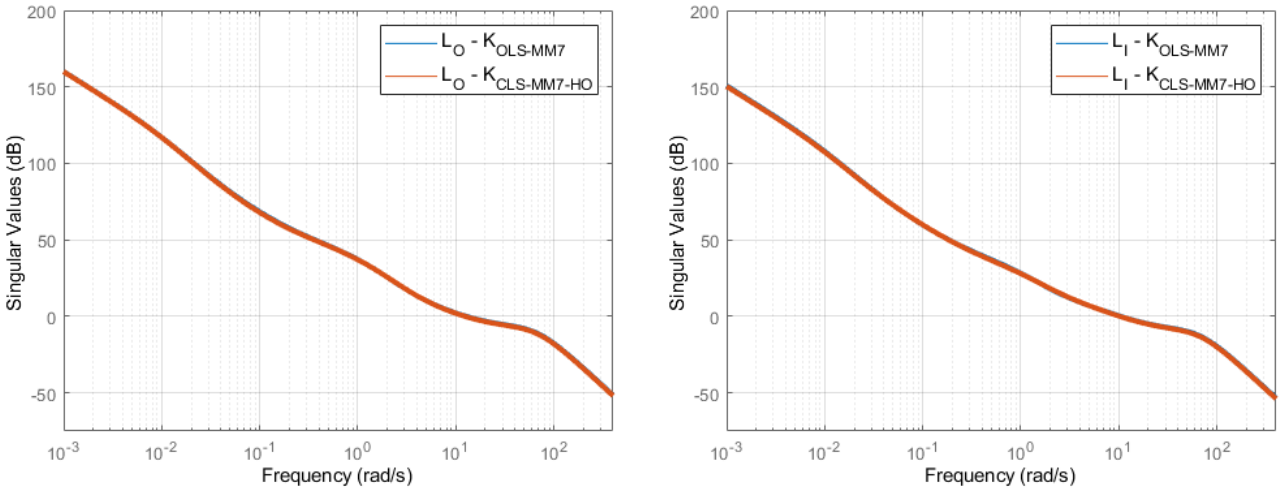

Figure 6.59: Output open loop (on the left) and input open loop (on the right) SV comparison - $K_{OLS-MM7}$ vs $K_{CLS-MM7-HO}$.

shape. However, it was concluded that this was not possible, as any improvement in load relief, or drift rejection, or tracking capabilities (or any other parameter) was always offset with a worsening somewhere else (worse actuation, worse bandwidth, worse robustness to input disturbances, etc...). This is further evidenced in the next two examples.

In the first example, the goal was to improve load relief and drift rejection. This was achieved through tightening the load and the drift constraints and loosening the sensitivity and actuation constraints. Note that the constraints had to be loosened so as to ensure that Γ values of the optimization remained equal to one. In Figure 6.60, the L_I and L_O of the resulting controller are compared to those of nominal $\mathcal{H}_\infty$ OLS controller. As it can be seen, not only the load relief and the drift/drift rate rejection were improved, but the L_I shape was also improved: the new controller has essentially the same L_I crossover, with higher L_I SV prior to crossover and slightly lower L_I SV post crossover. However, these improvements come at the cost of a significantly later crossover in the L_O SV plot, as well as much higher L_O SVs at higher frequencies. This results in a controller with worse robustness to output multiplicative uncertainty, as well as worse noise attenuation.

In the second example, the goal was, again, to improve load relief, drift rejection, whilst maintaining

Table 6.15: Weighting filters and robustifying controller comparison - $K_{OLS-MM7}$ vs $K_{CLS-MM7-HO}$.

	$K_{OLS-MM7}$	$K_{CLS-MM7-HO}$
W_1	$\frac{35\,(s+20)}{s}$	$\frac{35.09\,(s+19.95)}{s}$
$W_{2-\dot{z}}$	$\frac{12.099\,(s+1.71)}{(s+9.024)}$	$\frac{12\,(s+1.833)}{(s+10.17)}$
$W_{2-\dot{\theta}}$	$\frac{0.3595\,(s+5.532)}{(s+80)}$	$\frac{0.35\,(s+5.9971)}{(s+80)}$
K_{p_θ}	-0.3761	-0.3564
K_{d_θ}	-0.8296	-0.8102
K_{p_z}	0.3687	0.3511
K_{d_z}	0.4411	0.4748

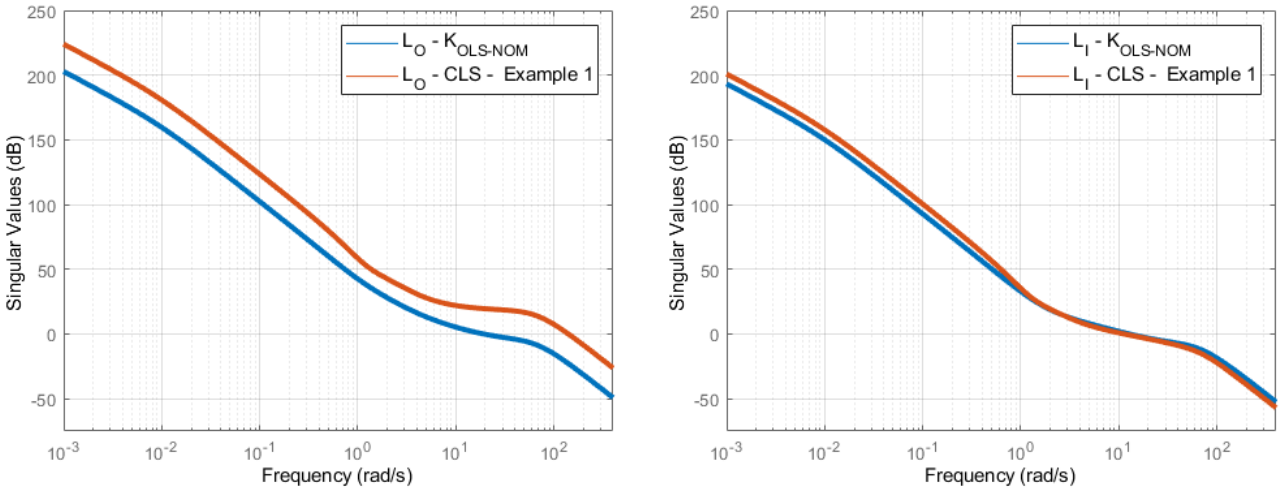

Figure 6.60: Output open loop (on the left) and input open loop (on the right) SV comparison - $K_{OLS-NOM}$ vs 1st improvement attempt with nominal higher order $\mathcal{H}_\infty$ CLS.

high frequency L_O gain low. This was achieved through tightening the load and drift constraints, loosening the sensitivity and actuation constraints, and replacing the constraint applied to $di \rightarrow u_p$ with a constraint limiting the $\mathcal{H}_\infty$ norm of the transfer function from the noise input to all of the outputs. In Figure 6.61, the L_I and L_O of the new controller are compared to those of nominal $\mathcal{H}_\infty$ OLS controller. As shown, the new controller demonstrates a clear improvement in the L_O shape. Its L_O achieves the same crossover frequency as the L_O of the $\mathcal{H}_\infty$ OLS controller, but displays equal or higher gain at frequencies prior crossover and essentially the same gain afterward. However, the L_I transfer function of the new controller exhibits significantly lower gain at low frequencies, while maintaining comparable gain to $K_{OLS-NOM}$ around and beyond the crossover frequency. Overall, the new controller has improved load relief and drift/rate rejection, but reduced attenuation of input disturbances at the plant input.

Both examples help to convey that any improvement in the loop gain of the $\mathcal{H}_\infty$ OLS controller is always offset with worsening of the loop gain somewhere else. This is further evidenced by the fact that to tighten any closed loop constraint, other closed loop constraints need to be loosened.

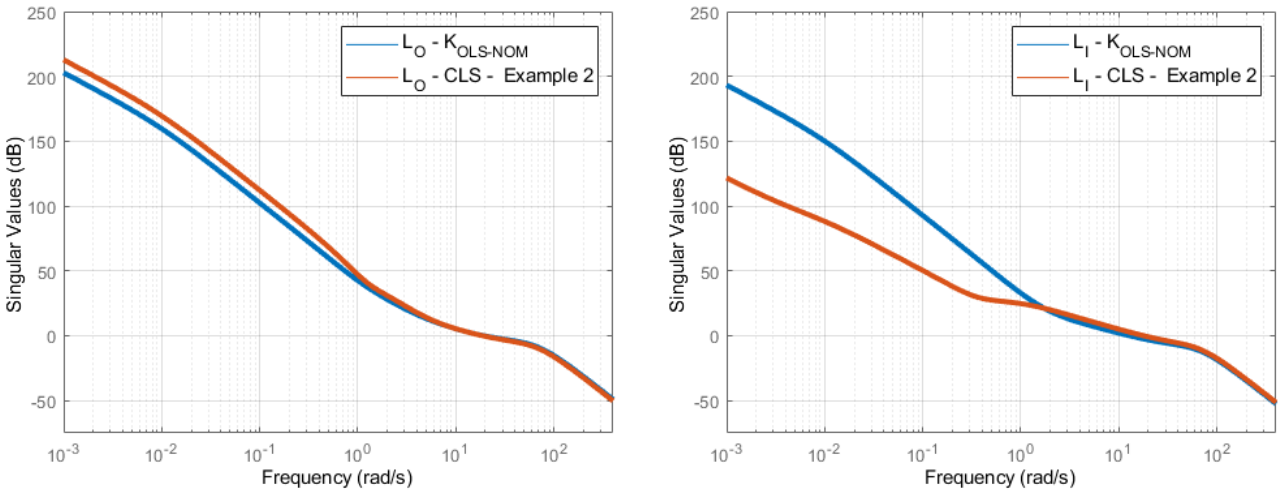

Figure 6.61: Output open loop (on the left) and input open loop (on the right) SV comparison - $K_{OLS-NOM}$ vs 2nd improvement attempt with nominal higher order $\mathcal{H}_\infty$ CLS.

In conclusion, it was possible to show that $\mathcal{H}_\infty$ Closed Loop Shaping and $\mathcal{H}_\infty$ Open Loop Shaping can attain (almost) the same result, when the controller order is equal. However, the complexity needed to operate each technique (and attain similar results) is severely different. This is primarily because $\mathcal{H}_\infty$ OLS automatically produces balanced L_I/L_O results, whereas $\mathcal{H}_\infty$ CLS demands greater expertise, requiring simultaneous manual tuning of multiple closed loop transfer functions to achieve comparable L_I/L_O balance. This simultaneous tuning is not trivial, as balancing constraints on multiple transfer functions is inherently complex and challenging. Consequently, designing the attitude controller with $\mathcal{H}_\infty$ OLS yields results as effective as $\mathcal{H}_\infty$ CLS, while being simpler to implement and requiring less design time.

6.10. $\mathcal{H}_\infty$ Closed Loop Shaping and $\mathcal{H}_\infty$ Open Loop Shaping - Property Summary

This section aims at summarizing the author's experience with $\mathcal{H}_\infty$ Closed Loop Shaping (CLS) and $\mathcal{H}_\infty$ Open Loop Shaping (OLS), as methods for the design of an attitude controller for the atmospheric ascent of a Launch Vehicle. Table 6.16 provides an overview of the results, which are then discussed in detail to highlight the key findings and considerations.

Table 6.16: Comparison between $\mathcal{H}_\infty$ Open Loop Shaping and $\mathcal{H}_\infty$ Closed Loop Shaping, from a user perspective.

	$\mathcal{H}_\infty$ **OLS**	$\mathcal{H}_\infty$ **CLS**
Final Result	Equal	Equal
L_I **/** L_O	Balanced	Unbalanced but can be made balanced
Learning Curve	Higher	Lower
Difficulty to attain better Robustness	Lower	Higher
Difficulty to attain better Performance	Lower	Higher
Difficulty in expressing requirements as constraints	Higher	Lower
Reproducibility for other flight points	Higher	Lower
Guarantees	Several	Several

The first point in Table 6.16 is straightforward: with the same controller order, both methods achieved (almost) identical results.

The second point highlights that, when the loop shaping is well done [8], $\mathcal{H}_\infty$ OLS can automatically produce controllers with balanced L_I and L_O gains, thanks to the NCF robustification process. As a result, the controller produced by $\mathcal{H}_\infty$ OLS inherently benefits from the advantages of well-shaped, balanced L_I and L_O gains, such as good disturbance rejection at both the input and output. In contrast, $\mathcal{H}_\infty$ CLS does not automatically balance these gains, requiring the designer to manually tune them, often involving extensive trial and error to achieve similar results. However, $\mathcal{H}_\infty$ CLS provides greater flexibility when focusing on a single loop gain (L_I or L_O), allowing for targeted optimization of that particular gain. For instance, it would be possible to increase the L_O gain and achieve better load relief, at the expense of a worst result in L_I loop gain.

The third point is particularly relevant for those deciding which method to learn. On one hand, starting

[8] In other words, when $\gamma_{min} < 4$.

with $\mathcal{H}_\infty$ OLS can be quite challenging, as it is not an intuitive method at first. The difficulty increases if a weighting filter parametrization for the aerospace vehicle is not already available, as the user would need to create this parametrization from scratch — a complex task that requires significant trial and error. In such cases, an alternative approach would be to design a controller using the LQG approach first. This allows the user to utilize the loop gains of the system with the LQG controller as initial loop gain targets, which can then be used to guide the design of weighting filters for the $\mathcal{H}_\infty$ OLS method. By providing these initial loop gain goals, the weighting filter design process, although still not trivial, becomes more manageable and less prone to trial and error. On the other hand, $\mathcal{H}_\infty$ CLS (particularly when using a framework that stems from the Signal-based $\mathcal{H}_\infty$ method, as in this work; see Section 4.9), offers a more user-friendly and intuitive approach, often yielding reasonably good results with relative ease, as most requirements can be directly represented as simple closed loop constraints. Of course, this assumes that the user will adopt a careful, methodical approach with this second method, rather than simply loading systune with numerous constraints and expecting optimal results.

The fourth and fifth points address the challenges of improving performance and robustness after overcoming the initial learning curve. It is clear from this work that achieving improved robustness is much easier with the $\mathcal{H}_\infty$ OLS process, thanks to the second step, in which the shaped system is robustified against NCF uncertainty. Due to the robustification against NCF, the resulting solution automatically exhibits equal and symmetric stability margins at both the plant input and output, as well as robustness against different types of unstructured uncertainty. On the other hand, $\mathcal{H}_\infty$ CLS requires manual tuning of multiple transfer functions to achieve robustness and equal stability margins at both the input and output, making the process significantly more challenging.

When it comes to performance, some may argue that it is easier to achieve better results with $\mathcal{H}_\infty$ CLS, thanks to the straightforward translation of requirements into closed loop specifications. However, this perspective overlooks a significant challenge: $\mathcal{H}_\infty$ CLS requires the manipulation of multiple transfer functions, making it easy to specify conflicting requirements. This often leads to a time consuming process of adjusting each transfer function constraint, so as to attain ”optimal” performance. In contrast, $\mathcal{H}_\infty$ OLS mitigates this issue as only the loop gains are being manipulated. Additionally, the parametrization developed in this work simplifies the manipulation of loop gains, reducing the complex task to the straightforward selection of just two parameters, all while ensuring excellent performance.

The sixth point is related to the expressing of requirements as constraints. As mentioned earlier, most requirements can be directly written as closed loop constraints, making it easier to define them with $\mathcal{H}_\infty$ CLS. However, converting closed loop specifications into open loop ones is generally straightforward. Many common specifications (such as disturbance rejection, noise attenuation, control signal reduction, and reference tracking) can be transformed into open loop specifications through simple mathematical manipulation of their closed loop counterparts. These conversions are not only straightforward but also well-documented in numerous references, which provide not only the final results but also detailed step-by-step instructions on how to derive them (e.g., pages 81-85 of *K. Zhou* and *Doyle* (1998)). Furthermore, when these straightforward transformations are combined with the relationships outlined in this work between performance requirements and their relevant frequency ranges (see Subsection 6.7.2), it becomes evident that expressing requirements as constraints for $\mathcal{H}_\infty$ OLS is manageable, though still not as simple as with $\mathcal{H}_\infty$ CLS.

The seventh point pertains the reproducibility of the controller design methods for other flight points. Utilizing $\mathcal{H}_\infty$ OLS for the other flight points is easier, as it is essentially the same as just changing the weighting filters pertaining drift and pitch angle (two gains). On the other hand, $\mathcal{H}_\infty$ CLS necessitates

the manipulation of nearly all constraints to maximize controller performance/robustness. For example, if $\mathcal{H}_\infty$ CLS were used to design a controller for a flight point where reference tracking is the primary goal, the constraints would need significant adjustments: the load relief and drift constraints would have to be relaxed, the sensitivity constraint tightened, and potentially the constraint on actuation would have to be loosened. This means that even a simple goal change requires $\mathcal{H}_\infty$ CLS to adjust several constraints, leading to a time-consuming trial-and-error process. In contrast, achieving the same goal with $\mathcal{H}_\infty$ OLS is easier, requiring only a reduction in the drift weighting filter gain and an increase in the pitch angle weighting filter gain.

The final point highlights the guarantees offered by each technique. $\mathcal{H}_\infty$ OLS ensures simultaneous and symmetric stability margins at both the plant input and output and is capable of handling systems where parameter variations could render them open loop unstable (*Bates* & *Postlethwaite*, 2002). In contrast, $\mathcal{H}_\infty$ CLS cannot accommodate systems that become open loop unstable due to parameter variations (*Bates* & *Postlethwaite*, 2002). However, it still guarantees simultaneous and symmetric stability margins at the plant input and output, albeit with the added complexity of requiring the designer to manually manipulate multiple closed loop transfer functions — an effort that is handled automatically in $\mathcal{H}_\infty$ OLS. Additionally, $\mathcal{H}_\infty$ CLS provides the assurance that, when the closed loop requirements are properly translated into constraints for the controller synthesis, and the resulting controller satisfies all these constraints, the closed loop requirements are (at least) nominally fulfilled. In contrast, $\mathcal{H}_\infty$ OLS does not inherently offer such guarantees, requiring the resulting controller to be tested to verify whether the closed loop requirements have been met.

6.11. First Research Question and Chapter Conclusion

The goal of this chapter was to compare $\mathcal{H}_\infty$ Open Loop Shaping and $\mathcal{H}_\infty$ Closed Loop Shaping as techniques for designing the attitude controller for the atmospheric ascent of a launch vehicle, ultimately addressing Research Question 1 and its sub-Research Questions. These are restated bellow to refresh the reader's memory and provide context.

Research Question 1

Can $\mathcal{H}_\infty$ Open Loop Shaping be used to generate attitude controllers for launch vehicles that exhibit equal or superior robustness, stability and performance compared to controllers derived using $\mathcal{H}_\infty$ Closed Loop Shaping, while reducing design time/complexity?

Research Question 1.1

How can $\mathcal{H}_\infty$ Open Loop Shaping be applied to design an attitude controller for a launch vehicle? Specifically, what form should the weighting filters take to properly shape the plant?

Research Question 1.2

Can the $\mathcal{H}_\infty$ OLS design methodology reduce overall design time and complexity compared to the $\mathcal{H}_\infty$ CLS design methodology?

Research Question 1.3

Can the attitude controller designed using $\mathcal{H}_\infty$ Open Loop Shaping achieve comparable or superior results in terms of uncertainty handling, disturbance rejection, stability margins, tracking accuracy, drift control, load relief, and actuator usage, compared to the controller produced using $\mathcal{H}_\infty$ Closed Loop Shaping?

With the research questions restated, the subsequent discussion addresses the answers. Firstly, Research Question 1.1 was answered in Section 6.7. Specifically, the weighting filters should take the form outlined in the parametrization described in Subsection 6.7.3. Additionally, $\mathcal{H}_\infty$ OLS can be applied to design an attitude controller for a launch vehicle by following the process detailed in Subsection 6.7.4.

Answering Research Questions 1.2 and 1.3 was a multi-step process. First, it entailed an initial design of a controller using $\mathcal{H}_\infty$ CLS, this was described in Section 6.6. Secondly, the $\mathcal{H}_\infty$ OLS controller was designed in Subsection 6.7.5. With this, it was possible to compare both produced controllers in Subsection 6.8. The comparison showed the superiority of the $\mathcal{H}_\infty$ OLS controller in disturbance rejection, tracking accuracy, drift minimization, load relief, whilst achieving comparable results in actuator usage and stability margins - both for nominal and dispersed conditions. In terms of SSV both controllers performed equally well. However, the two controllers that were compared were of different order. And the question of whether the different controller order was the cause of the results arose. So, in Section 6.9, a controller with the same order of the $\mathcal{H}_\infty$ OLS controller, was tuned using $\mathcal{H}_\infty$ CLS. For this purpose, the controller to be tuned was composed of the original PD controllers and a set of filters with the same parametrization that was used in the $\mathcal{H}_\infty$ OLS design. This section demonstrated that both methods achieve identical results when the same controller order and structure are employed, with $\mathcal{H}_\infty$ OLS offering a simpler and faster design process. Furthermore, it was shown that the $\mathcal{H}_\infty$ CLS methodology cannot improve upon the results achieved by $\mathcal{H}_\infty$ OLS.

This analysis effectively addressed Research Question 1.2 by demonstrating that the $\mathcal{H}_\infty$ OLS methodology results in a simpler and faster design process compared to $\mathcal{H}_\infty$ CLS. For Research Question 1.3, it was shown that both $\mathcal{H}_\infty$ OLS and $\mathcal{H}_\infty$ CLS yield identical controllers with consistent results across all specified parameters. Finally, regarding Research Question 1, the study confirmed that $\mathcal{H}_\infty$ OLS can be used to generate launch vehicle attitude controllers that exhibit equal robustness, stability, and performance as those generated using $\mathcal{H}_\infty$ CLS, while also reducing design time and complexity.

Having answered Research Question 1 and its sub-questions, all the knowledge gained throughout the different design stages culminated in Section 6.10, where a summary of the author's experience with each $\mathcal{H}_\infty$ method is provided.

7

Integrated Rigid/Flexible Controller Design

In the traditional approach to TVC control, the design of the rigid body controller and the bending filter for the flexible modes is treated separately. The design process has three steps: firstly, the rigid body controller is designed; secondly, the flexible interactions are added to the model and the bending filter is tuned; thirdly, the rigid body controller and bending filter are manually re-tuned to ensure proper integration and compliance with all requirements.

As described in *Tapia* (2019), efforts have been made to streamline this process. In particular, *Tapia* (2019) developed and implemented a method that allows for the simultaneous design of both the rigid body controller and the bending filter using $\mathcal{H}_\infty$ Closed Loop Shaping. Building on this concept, the current work will develop an integrated approach for designing the Rigid/Flexible controller using the $\mathcal{H}_\infty$ Open Loop Shaping method.

And so, this chapter is structured as follows: Section 7.1 covers the parametrization and implementation of the bending filter; Section 7.2 presents the rigid controller and bending filter design using the separate design approach, based solely on the nominal model; Section 7.3 describes and implements the integrated design approach, again considering only the nominal model; Section 7.4 compares the results from both approaches; Section 7.5 applies the integrated design approach to develop a Multi-model based controller; Section 7.6 presents the results for both the nominal and Multi-model integrated controllers; finally, Section 7.7 entails the answers to Research Question 2 and its subquestions, as well as the chapter's conclusion.

7.1. Bending Filter Parametrization and Implementation

For this work, the bending filter parametrization that will be used is the one outlined in *Tapia* (2019). This parametrization, which is detailed in eq. 7.1, was developed to enable the application of structured $\mathcal{H}_\infty$ design, while preserving the legacy knowledge of the original bending filter used by the VEGA GNC team.

$$H_{BF}(s) = \prod_{i=1}^{4} \frac{s^2 + \eta_i s + (\omega_i)^2}{s^2 + \eta_i/\epsilon_i s + (\omega_i)^2} \cdot \left(\frac{\epsilon_{LP} s^2 + \eta_{LP} s + (0.6\omega_{q_2})^2}{s^2 + \eta_{LP} s + (0.6\omega_{q_2})^2} \right)^3 \tag{7.1}$$

This bending filter consists of four notch filters and a low-pass filter. The notch filters are centered around the minimum expected frequency of the 1st bending mode (BM), the nominal frequency of the 1st BM, the maximum expected frequency of the 1st BM and the minimum expected frequency of the 2nd BM, respectively. These filters aim at notching the first and second BM, whilst the low-pass filter aims at attenuating the upper BMs.

Each notch filter is parameterized by two variables, ϵ_i and η_i. The first parameter is tunable and corresponds to the attenuation at the center frequency of the notch filter, while the second is fixed and describes the width of the filter. The low-pass filter is fixed as well, with ϵ_{LP} representing the attenuation at high frequencies, and η_{LP} being defined as in eq. 7.2.

$$\eta_{LP} = 2\,\zeta\,(0.6\,\omega_{q_2}) \tag{7.2}$$

where ζ is the low pass filter's damping factor, and ω_{q_2} is the nominal frequency of the second BM.

In this work, both the fixed and tunable variables are set according to the values and limits specified in *Tapia* (2019). These can be seen in Table 7.1.

Table 7.1: Fixed and tunable Variables of the bending filter parametrization.

Fixed Variables		Tunable Variables		
Variable	Value	Variable	Min	Max
ζ_1	2	ϵ_1	-25 dB	-10 dB
ζ_2	5	ϵ_2	-10 dB	-4 dB
ζ_3	2	ϵ_3	-25 dB	-10 dB
ζ_4	12	ϵ_4	-20 dB	-15 dB
η_{LP}	40			
ϵ_{LP}	-8.3 dB			

With this in mind, the implementation of the bending filter can be seen in Figure 7.1. In this scheme the output gains convert signals from radians to degrees, while the input gains convert signals from degrees to radians. The one exception is the command input gain, which has two functions: to convert degrees to radians, and to act as the prefilter.

7.2. Separate Rigid/Flexible Body Controller Design

The goal of the separate rigid/flexible body controller design is to serve as baseline for the comparison. In this step, the rigid body controller will still be designed with $\mathcal{H}_\infty$ Open Loop Shaping, whilst the bending filter is tuned by hand. In addition to ensuring that the final controller meets all the requirements defined in Section 6.3, another key objective is to ensure that the 1st BM peak gain in the input open loop gain was similar to that of the controller developed in *Tapia* (2019). Achieving the latter condition is expected to ensure proper damping of the 1st BM, as the strong performance of this reference controller has already been validated through nonlinear simulation.

Figure 7.2 illustrates the design process for the separate rigid/flexible body controller. This process is divided into three main phases: initial, intermediate, and final. Several important remarks need to be made regarding each phase.

In the first step of the initial phase, a loose rigid body shape was employed (i.e., $W_{2-\theta} = 0.01$, $W_{2-z} = 0.01$, $W_1 = 4\left(\frac{s+11.5}{s}\right)$), as experimental observations revealed that the introduction of the bending filter required lower loop gain specifications to achieve a sufficiently large robustness margin. The SIMULINK models used for this step are identical to those employed for the $\mathcal{H}_\infty$ OLS design in the previous chapter (see Figures 6.9 and 6.24). It should be noted that in these models the block G_{LV} is given by the rigid body robust design model (DMR), as detailed in Subsection 5.3.2.

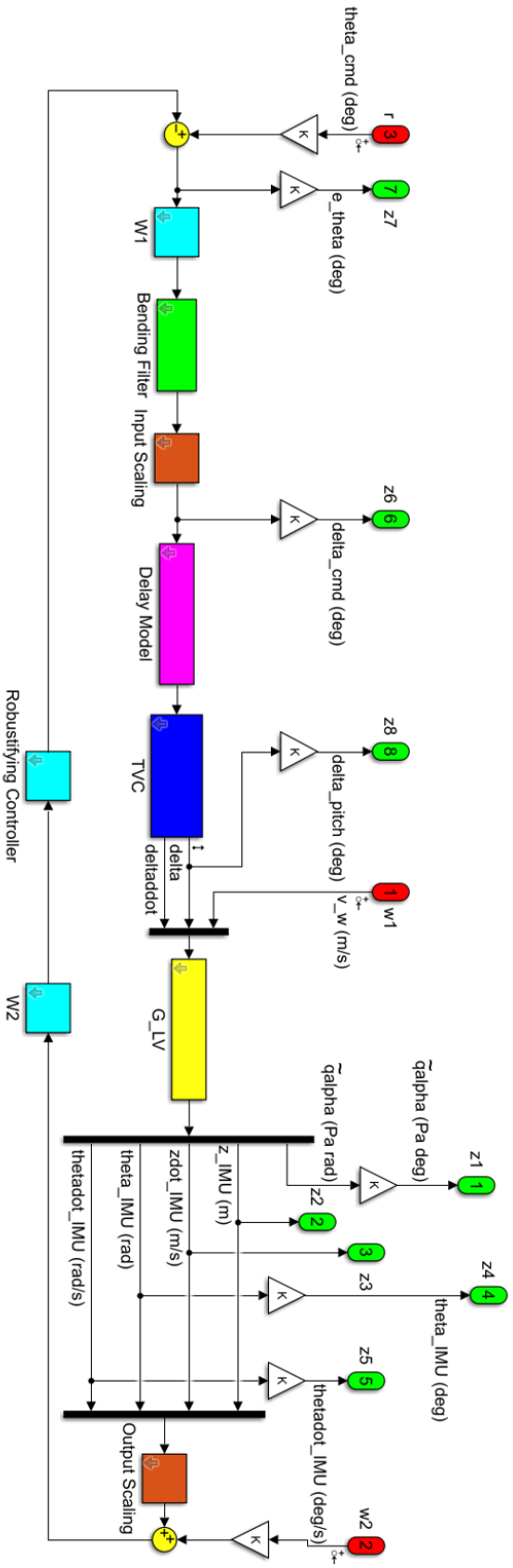

Figure 7.1: Tracker implementation scheme for the rigid/flexible body $\mathcal{H}_\infty$ OLS controllers.

In the second step of the initial phase, the bending filter was incorporated, with the ϵ_i parameter of each notch filter initially set to the midpoint of its allowable range. For this step and all subsequent steps, the SIMULINK models used for design are identical to those used for $\mathcal{H}_\infty$ OLS in the previous chapter (see Figures 6.9 and 6.24), with two key differences: the inclusion of the bending filter between the pre-compensator and the input scaling, and the fact that the G_{LV} block now contains the rigid/flexible body design model (DMF), as discussed in Subsection 5.4.2.

In the intermediate phase, fine-tuning the bending filter to achieve the desired flexible stability margins and 1st BM peak gain involved minor trial and error. The condition $\gamma \leq 4$ was applied to ensure that the lead-lags were appropriately selected, minimizing the need for significant adjustments in the final phase.

In the final phase, selecting the bending filter and determining the drift and pitch angle post-compensator gains (W_{2_θ} and W_{2_z}) proved to be a non-trivial task, requiring extensive experimentation. Adjusting the drift and pitch angle post-compensator gains impacted the loop gain at the 1st BM frequencies, creating the need for additional notching at those frequencies. This adjustment process was particularly challenging, as excessive notching led to an overly steep slope near the crossover frequency. Moreover, finding the optimal combination of drift and pitch angle post-compensator gains added complexity, as multiple configurations were possible.

Through testing various controllers, it was concluded that the drift post-compensator gain should be set to 0.21, and the pitch angle post-compensator gain increased as much as possible, while maintaining $\gamma \leq 4$. This drift post-compensator gain is the smallest which allowed for sufficient drift rejection (as can be seen by the results shown in Subsection 7.6). Therefore, by selecting this drift post-compensator gain and increasing pitch angle post-compensator gain as much as possible, while maintaining $\gamma \leq 4$, it will be possible to maximize load relief while attaining sufficient drift rejection. With this in mind, extensive trial and error was carried out, and the final bending filter configuration was determined: ϵ_2 and ϵ_4 were set to their minimum values, while ϵ_1 and ϵ_3 were adjusted to -21.3 dB and -11.9 dB, respectively. This configuration was selected because it supported the chosen drift post-compensator gain, and allowed for the highest possible pitch angle post-compensator gain, while ensuring that the 1st BM peak gain and the nominal flexible stability margins were as desired.

Finally, one important remark is made regarding the rationale of maintaining $\gamma \leq 4$. From the rigid body design, it was known that stability requirements for the LV were generally easy to meet, while the load relief and drift rejection requirements were the most likely to be infringed. Moreover, using the previously presented $\mathcal{H}_\infty$ OLS methodology (Subsection 6.7.4) and targeting a robustness margin smaller or equal to four typically resulted in fulfilling the LV stability margins requirements. Testing showed this to be true for the rigid/flexible body controller design as well. Therefore, by ensuring the robustness margin remained below four and maximizing the drift and pitch angle post-compensator gains (which are directly related to improving drift and load rejection as discussed in Subsection 6.7.4) it was possible to achieve a controller with sufficiently large stability margins and optimal load/drift rejection. This controller is coined $K_{OLS-NOM-SEP}$ and will be shown in Section 7.4. However, before that, the integrated design approach will be discussed in the following subsection.

7.3. Integrated Rigid/Flexible Body Nominal Design

This section discusses a process for designing an integrated rigid/flexible body controller, with the primary goal of simultaneously tuning the robustifying controller and bending filter. The SIMULINK models used for design are identical to those used for $\mathcal{H}_\infty$ OLS in the previous chapter (recall Figure 6.9 and 6.24), with

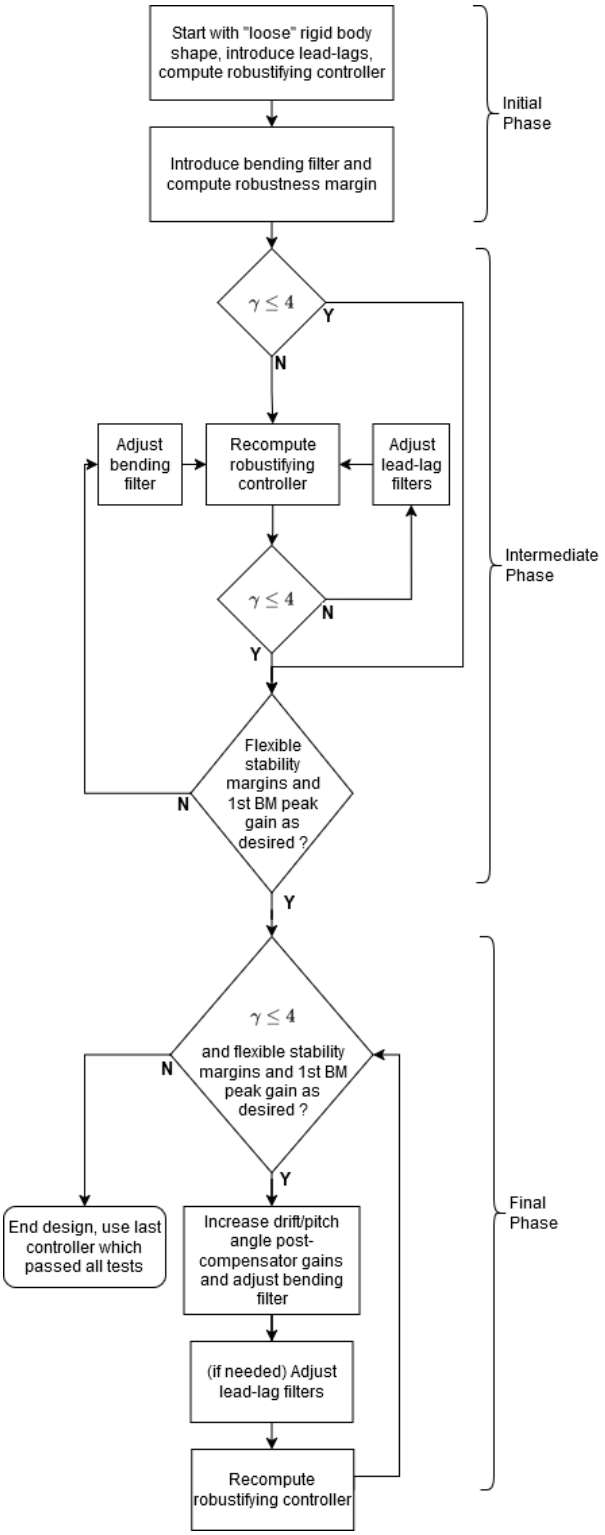

Figure 7.2: Workflow of the separate rigid/flexible body controller design.

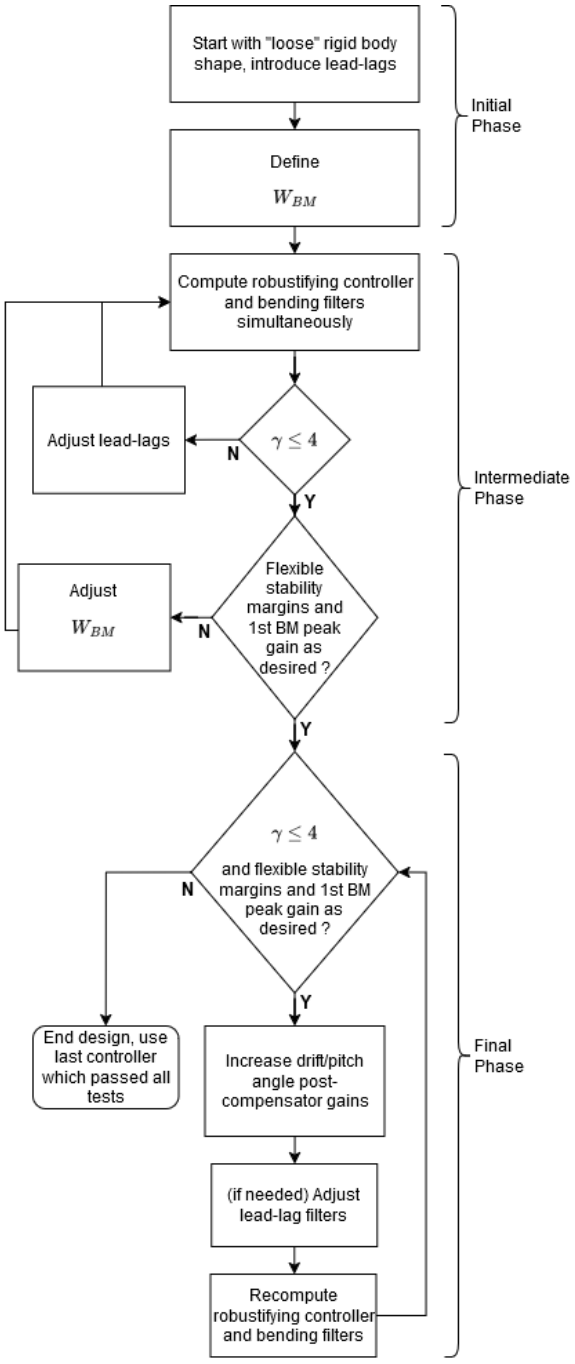

Figure 7.3: Workflow of the integrated rigid/flexible body controller design.

two key differences: the introduction of a bending filter between the pre-compensator and the input scaling in the NCF robustification scheme (Figure 6.24), and the modification of the block G_{LV} in both models, which now incorporates the rigid/flexible body design model (DMF), as detailed in Subsection 5.4.2.

To achieve the simultaneous tuning of the robustifying controller and bending filter, two optimization objectives are employed during the use of systune: NCF robustification and maximum allowable loop gain. The last objective is only applied at bending mode frequencies and is formally defined in eq. 7.3.

$$\bar{\sigma}(L_I(\omega)) < W_{BM}, \quad \forall \omega \in \omega_R \tag{7.3}$$

where W_{BM} is a static filter and where ω_R is a frequency range spanning from $20\,\text{rad}/\text{s}$ (slightly before the first BM), to infinity. It should be noted that this constraint was applied on L_I, as the requirement for the 1st BM peak gain pertains to the input open loop gain.

The step-by-step procedure for designing the integrated controller is detailed in Figure 7.3. As it can be seen, there are once again three phases (initial, intermediate and final) during the design. Some important remarks regarding the intermediate and final design phases are given next.

In the intermediate design phase, the goal is to select the static filter (W_{BM}) used for the maximum allowable loop gain constraint. If done properly (i.e. fulfilling the conditions given in the workflow of Figure 7.3), the flexible stability margins and 1st BM peak gain will be as desired during the rest of the design. Subsequently, the designer can move on to the final phase, and focus solely on optimizing the performance of the controller.

In the final phase, the drift post-compensator gain was set to 0.21, as testing showed it to be the minimum drift post-compensator gain that provided adequate drift rejection (this is demonstrated by the results presented in Subsection 7.6). Subsequently, pitch post-compensator gain was increased as much as possible, while maintaining $\gamma < 4$, for the same reasons as in the separate design.

Finally, it is important to mention that it is possible to tune the lead-lag filters simultaneously with the bending filter and robustifying controller. To do so, in the intermediate and final phase of the integrated design, the designer must set the lead-lags, bending filters and robustifying controller as tunable parameters in systune. Subsequently, all of them can be computed simultaneously, utilizing as optimization goals the NCF robustification and the aforementioned maximum allowable loop gain. Care should be taken when opting for this design approach, as the lead-lags must be properly parametrized and constrained.

With the integrated approach it was possible to design a rigid body controller and bending filter in a much simpler way than with the separate approach. In fact, it should be noted that the application of the integrated design methodology took far less time than the separate design methodology. This is mainly a result of the simultaneous robustifying controller/bending filter tuning, which not only eliminates the need for manual bending filter adjustments, but also enables a smoother and automatic integration of the two components. The controller computed with this method is coined $K_{OLS-NOM-INT}$ and will be shown in the next section.

7.4. Comparison of the Results

First, both the computed rigid body controllers and bending filters are presented in Table 7.2. As shown, all the filters and robustifying controllers are quite similar. However, several key observations should be made.

Table 7.2: Robustifying controller, bending filter and weighting filters comparison - $K_{OLS-NOM-SEP}$ vs $K_{OLS-NOM-INT}$.

	$K_{OLS-NOM-SEP}$	$K_{OLS-NOM-INT}$
W_1	$\frac{4(s+11.5)}{s}$	$\frac{4(s+11.5)}{s}$
W_{2-z}	0.21	0.21
$W_{2-\dot{z}}$	$\frac{30(s+0.23)}{(s+14.49)}$	$\frac{30(s+0.24)}{(s+14.95)}$
$W_{2-\theta}$	0.12	0.12
$W_{2-\dot{\theta}}$	$\frac{0.21(s+1.92)}{(s+10.07)}$	$\frac{0.21(s+1.91)}{(s+10.08)}$
K_{p_θ}	-0.2928	-0.3007
K_{d_θ}	-1.1526	-1.1532
K_{p_z}	0.4253	0.4255
K_{d_z}	0.5451	0.5422
ϵ_1	-21.3	-21.3407
ϵ_2	-10	-10
ϵ_3	-11.9	-11.9035
ϵ_4	-20	-20

In both designs, the pre-compensator was initialized with the same value and remained unchanged throughout the process. Additionally, the drift post-compensator gains were selected based on testing controllers produced by each respective approach, ensuring they did not influence the choice of one another. The pitch angle post-compensator gains were maximized independently for each design, with no interaction between them, as the objective was to achieve the highest possible gains in each approach.

Furthermore, although the lead-lag compensators were manually tuned in both approaches — potentially raising concerns that the results of the integrated design were influenced by the separate design — testing in the integrated design confirmed that they were the optimal choice for fulfilling the tuning requirements. Notably, when the lead-lags were simultaneously tuned with the bending filter and robustifying controller in the integrated approach, they naturally converged to nearly the same values as those determined during

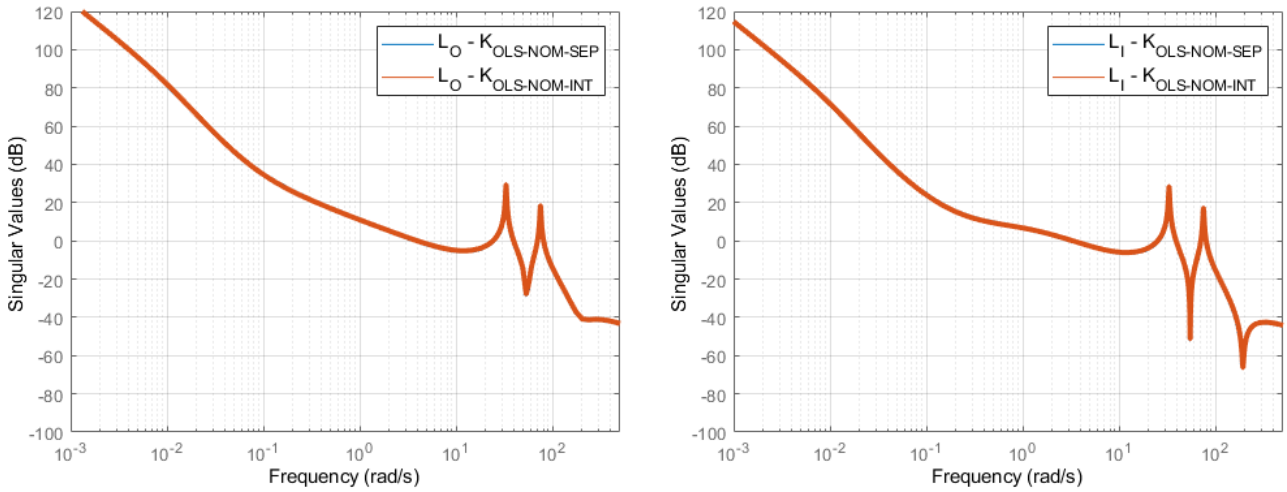

Figure 7.4: Output and input open loop (L_O on the left and L_I on the right) SV comparison - $K_{OLS-NOM-SEP}$ vs $K_{OLS-NOM-INT}$.

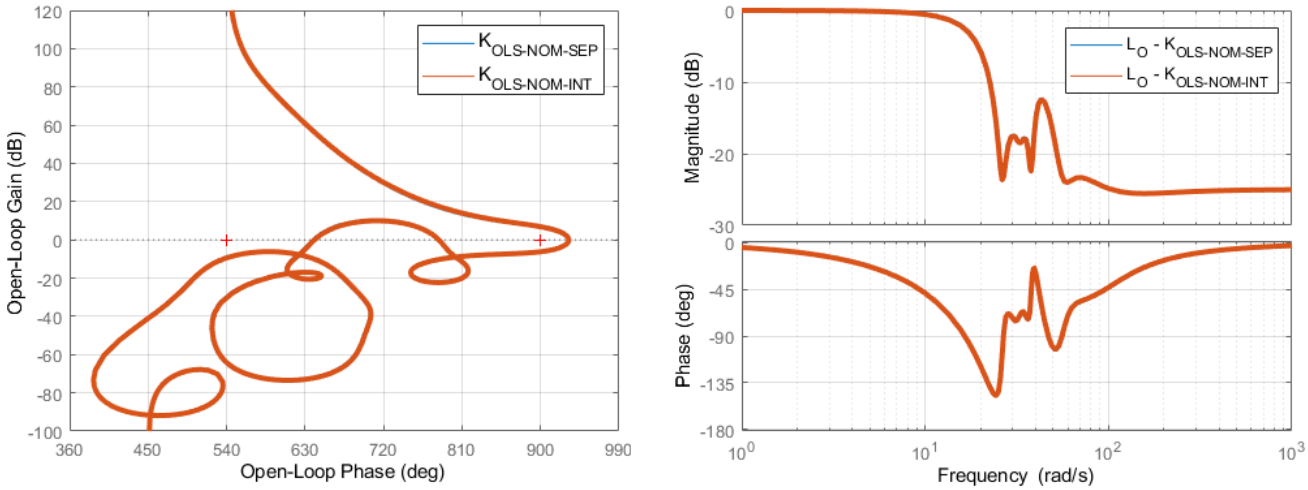

Figure 7.5: Input Nichols plot (on the left) and Bode plot of the bending filters (on the right) - $K_{OLS-NOM-SEP}$ vs $K_{OLS-NOM-INT}$.

the manual tuning. This convergence highlights that their selection and tuning were driven by the design objectives, rather than by any prior knowledge from the separate design process. With this discussion in mind, it becomes clear that the experience from the first design did not influence the second design.

Despite the differences in design approach, both methods produced nearly identical results. Figure 7.4 highlights this by comparing the open loop gains of the two controllers, which are shown to be identical. Furthermore, Figure 7.5 presents two comparisons: the first compares the input Nichols plots of each controller, while the second compares the Bode plots of the bending filters of each controller. Both comparisons show indistinguishable results, further validating the consistency of the design approaches.

In essence, both methods yield the same results, which is a crucial finding given that the integrated design method is significantly simpler and faster to implement. With this conclusion established, the next section will focus on applying the integrated design method to develop a Multi-model rigid body controller and bending filter capable of meeting all the specified requirements.

7.5. Integrated Rigid/Flexible Body Multi-model Controller Design

The objective of the Multi-model design approach was to ensure that the bending filter and rigid body controller were robustly tuned, meeting stability and performance requirements even under uncertainty. To accomplish this, the design process accounted for both the nominal model and the corner cases defined by five uncertain parameters. These parameters included the frequencies of the first two bending modes, as well as the three rigid body parameters with the highest mu-sensitivity: Cn_α, $\tilde{q}$ and x_{cp}.

After defining the models to be used for the design, the integrated rigid/flexible body controller design approach was applied, resulting in the controller labeled $K_{OLS-MM5-INT}$. The weighting filters, robustifying controller, and bending filter of $K_{OLS-MM5-INT}$ are compared with those of $K_{OLS-NOM-INT}$ in Table 7.3.

It should be noted that, for the Multi-model design, the pre-compensator and lead-lags were adjusted to ensure $\gamma = 4$. The drift post-compensator gain was set equal to that of the nominal integrated controller, as testing has shown this ensures sufficient drift rejection. Additionally, the pitch angle post-compensator

Table 7.3: Robustifying controller, bending filter and weighting filters comparison - $K_{OLS-MM5-INT}$ vs $K_{OLS-NOM-INT}$.

	$K_{OLS-MM5-INT}$	$K_{OLS-NOM-INT}$
W_1	$\frac{6.26(s+10.2)}{s}$	$\frac{4(s+11.5)}{s}$
W_{2-z}	0.21	0.21
$W_{2-\dot{z}}$	$\frac{30(s+0.30)}{(s+19.09)}$	$\frac{30(s+0.24)}{(s+14.95)}$
$W_{2-\theta}$	0.1	0.12
$W_{2-\dot{\theta}}$	$\frac{0.12(s+1.8)}{(s+5.78)}$	$\frac{0.21(s+1.91)}{(s+10.08)}$
K_{p_θ}	-0.3000	-0.3007
K_{d_θ}	-0.9864	-1.1532
K_{p_z}	0.3441	0.4255
K_{d_z}	0.4556	0.5422
ϵ_1	-16.52	-21.3407
ϵ_2	-10	-10
ϵ_3	-10	-11.9035
ϵ_4	-17.91	-20

gain was slightly reduced compared to the nominal controller in order to satisfy the desired stability margin requirements.

7.6. Evaluation of the Nominal and Multi-model Integrated Controllers

This section presents the tests performed on the two integrated controllers developed in the previous sections. The implementation of these controllers is carried out using the SIMULINK model illustrated in Figure 7.1, where the block G_{LV} contains the rigid/flexible body analysis model (AMF), as discussed in Subsection 5.4.1.

While the tests in this section are similar to those in the previous chapter, there are three key differences: worst-case analysis has been replaced by Monte Carlo analysis, and additional comparisons are made between the bending filters of each controller, as well as between the 1st BM peak gains of each controller (evaluated using the input Nichols). The rationale for the first change is that, due to the large number of uncertain parameters (15 rigid and 10 flexible), worst-case analysis could have resulted in evaluating the controller under overly pessimistic scenarios.

For each test case, 1000 random samples were generated, reflecting a balance between computational burden and the need to adequately demonstrate the effect of dispersed conditions. It is important to note that the number of samples used may not be sufficient for a thorough evaluation of the controller, as a more detailed analysis would require determining the optimal number of Monte Carlo simulations. However, the primary goal was simply to conduct an initial analysis in order to make preliminary observations regarding the success of the approaches used. While the sample size may limit the validity of the results, it provides a reasonable starting point for assessing the effectiveness of the controller under a range of conditions.

7.6.1. Bending Filters

Figure 7.6 illustrates the Bode plot of the bending filter for each controller. The Multi-model approach led to a bending filter design characterized by a less pronounced negative phase peak and an increased filter

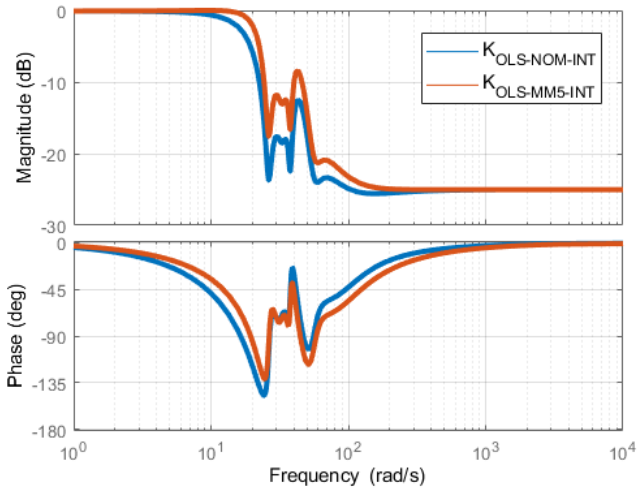

Figure 7.6: Bode plot of the bending filters - $K_{OLS-NOM-INT}$ vs $K_{OLS-MM5-INT}$.

magnitude. This adjustment arose because the constraint on the 1st BM peak gain was slightly relaxed to account for uncertainties in the BM frequency. As shown in the next subsection, this relaxation reduces the attenuation of the first BM peak under both nominal and dispersed conditions. However, as will be explained, this is not expected to pose a significant issue.

7.6.2. First Bending Mode Peak Gain

Figure 7.7 presents the input Nichols plots for both controllers. The plot on the left corresponds to $K_{OLS-NOM-INT}$, while the plot on the right corresponds to $K_{OLS-MM5-INT}$. In these plots, the thick blue line represents the nominal Nichols, and the cyan lines depict the Nichols under dispersed conditions. Additionally, Figure 7.8 shows the input Nichols plot of the rigid/flexible body controller designed in *Tapia* (2019), which served as the basis for establishing the 1st BM peak gain requirement. It is worth noting that, in all three figures, the 1st BM peak gain occurs near the point where the open loop phase approaches zero degrees.

The nominal controller's 1st BM peak gain under nominal conditions exhibits a slight increase compared to the results reported in *Tapia* (2019). Under dispersed conditions, both controllers achieve very similar performance, with a worst-sampled (WS) 1st BM gain of approximately 16 dB. In contrast, the Multi-model controller shows a significantly higher 1st BM peak gain under both nominal and dispersed conditions relative to *Tapia* (2019), with increases of 3.5 dB for the nominal case and 2.5 dB for the worst-sampled case. Testing indicates that this discrepancy does not pose an issue. As demonstrated in the following subsections, the stability margins of $K_{OLS-MM5-INT}$ satisfy the specified requirements. Additionally, time-domain tests and worst-sampled case analyses confirm its robust performance.

It is important to note that while the Multi-model controller exhibits a higher 1st BM peak gain in both nominal and dispersed conditions compared to *Tapia* (2019), this outcome is partly influenced by the number of random samples used. A comparison between Figure 7.7 and Figure 7.8 clearly shows a higher number of random sampled cases in this work (evident from the increased line density). When the number of samples is reduced to match the line density of Figure 7.8, the Multi-model controller shows only a 1 dB increase in the worst-sampled case compared to *Tapia* (2019), bringing the results much closer to the intended target.

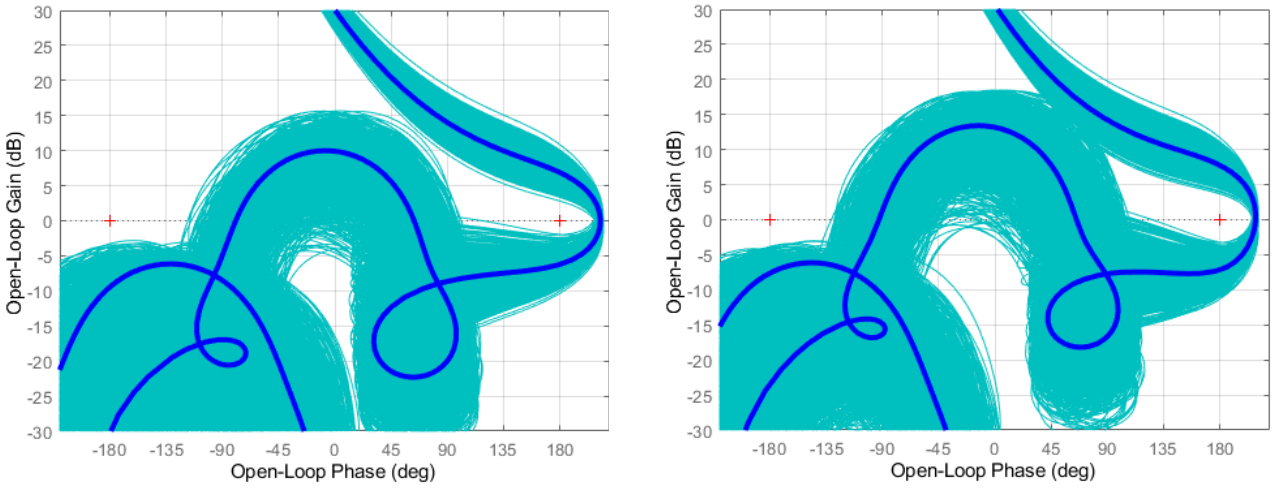

Figure 7.7: Input Nichols plot comparison - $K_{OLS-NOM-INT}$ (on the left) vs $K_{OLS-MM5-INT}$ (on the right)

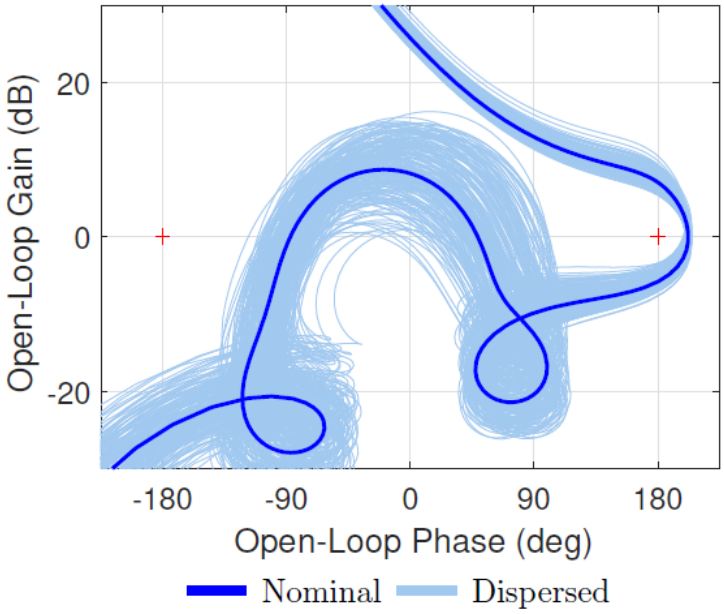

Figure 7.8: Input Nichols plot of the rigid/flexible body controller designed in *Tapia* (2019) (image retrieved directly from that work).

7.6.3. Stability Margins

For brevity, only the input Nichols is shown. The rest of the stability results are summarized in Table 7.4.

Input Stability Margins

The input Nichols plot (Figure 7.9) contains three lines, one representing the nominal Nichols (blue line), one representing the Nichols corresponding to the worst-sampled rigid disk margin (red line), one representing the Nichols corresponding to the worst-sampled (WS) flexible disk margin (black line). The blue disks represents the nominal disk margins, while the red and purple disks represent the worst-sampled rigid/flexible disk margins, respectively. The worst-sampled disk margins are the regions in which, according to the testing, the Nichols plot is not expected to enter.

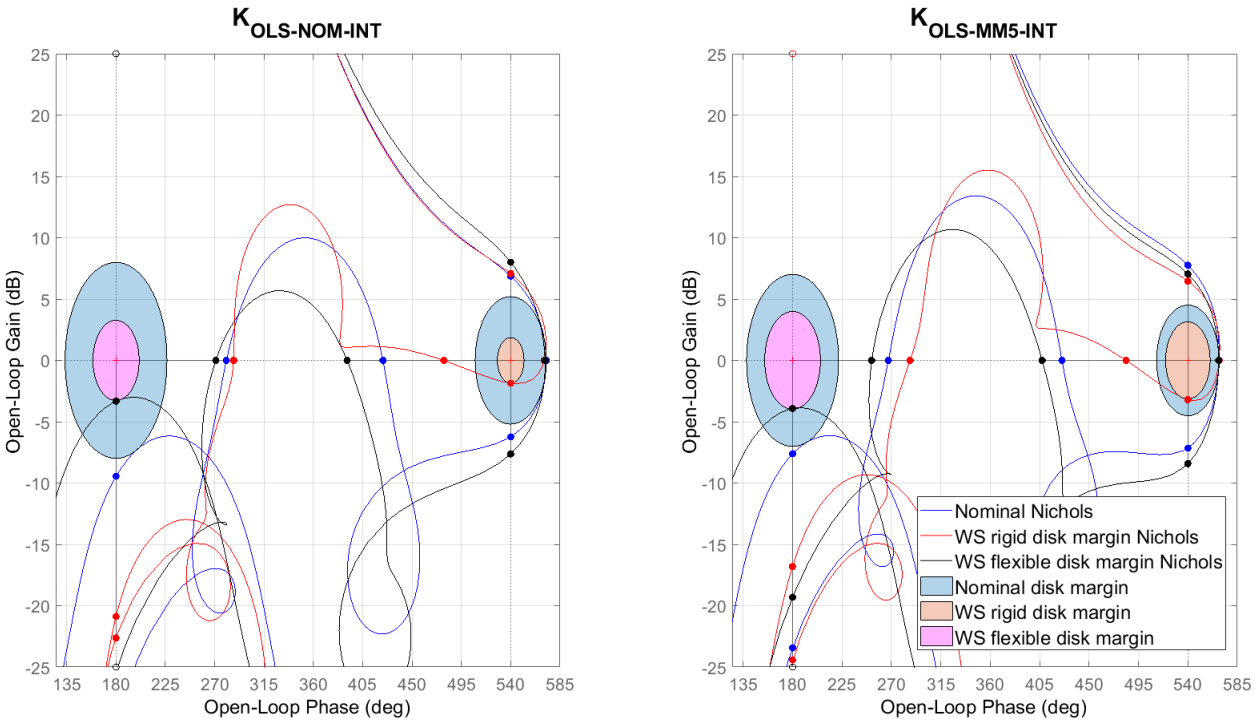

Figure 7.9: Input Nichols - $K_{OLS-NOM-INT}$ vs $K_{OLS-MM5-INT}$.

As shown in Figure 7.9, the nominal controller exhibits good nominal rigid and flexible margins, as well as strong worst-sampled case flexible margins. However, its rigid margins in the worst-sampled cases are smaller and unacceptable according to the requirements (smaller than 3 dB and 20 deg). In contrast, the Multi-model controller has slightly smaller nominal margins, but significantly larger rigid and flexible margins in the worst-sampled cases. In fact, while both controllers meet the stability requirements for nominal rigid and flexible margins as well as dispersed flexible margins, only $K_{OLS-MM5-INT}$ achieves sufficient rigid margins under dispersed conditions.

Stability Margin Summary

Table 7.4 presents the nominal and worst-sampled case S-T disk-based margins for both controllers. The gain margins correspond to the S-T disk-based gain margins (in dB), while the phase margins represent the corresponding S-T disk-based phase margins (in degrees). The frequency, given in rad/s, indicates where the disk margin was computed. The table uses color coding for clarity: green highlights the input stability margins for both nominal and worst-sampled cases, red highlights the worst stability margins among the loop-at-a-time (LAT) cases, and cyan highlights the MIMO output and input/output (I/O) stability margins for the worst-sampled cases.

For $K_{OLS-MM5-INT}$, all loop-at-a-time disk-based stability margins, except for those highlighted in red, exceed 12 dB and 63 deg . For the red-highlighted margins, the nominal disk-based stability margins exceed 4.5 dB and 29 deg, while the worst-sampled disk-based margins exceed 2.5 dB and 17 deg. The cyan-highlighted MIMO output disk-based stability margins are greater than 1.8 dB and 12 deg, and the input/output disk-based stability margins surpass 1 dB and 7 deg. These results confirm that the Multi-model controller not only meets the stability requirements for input margins but also demonstrates robustness across multiple metrics.

When comparing the Multi-model controller to the nominal controller, it is evident that the Multi-model controller improves the loop-at-a-time stability margins in all but three cases, where the disk-based margins were already greater than 20 dB and 80 deg. The Multi-model controller significantly improves the worst-sampled cases of the input, MIMO output, and input/output stability margins (the latter two highlighted in cyan), and also shows improvements in the nominal MIMO output stability margin. In contrast, the nominal controller slightly outperforms the Multi-model controller in the nominal input stability margins and the nominal input/output stability margin. Overall, the Multi-model controller demonstrates superior robustness, achieving significant improvements in almost all worst-sampled cases while maintaining strong margins in the nominal cases.

These findings underscore the significant advantages of the Multi-model approach in addressing the challenge of designing robustly tuned controllers. This approach demonstrates a clear ability to enhance stability margins, particularly in worst-sampled scenarios, while maintaining acceptable margins in nominal cases.

Table 7.4: Stability Margin Summary - $K_{OLS-NOM-INT}$ vs $K_{OLS-MM5-INT}$.

			$K_{OLS-NOM-INT}$			$K_{OLS-MM5-INT}$		
			GM	PM	Freq	GM	PM	Freq
Rigid	Nominal	Input	5.20	32.42	3.06	4.51	28.49	3.64
	Worst-sampled case	Input	1.86	12.16	8.96	3.14	20.28	10.08
	Nominal	LAT drift	14.72	69.20	0.07	15.38	70.68	0.07
		LAT drift rate	18.74	76.82	0.90	20.87	79.66	0.79
		LAT pitch angle	16.25	72.50	1.33	17.59	74.96	1.42
		LAT pitch rate	5.23	32.60	1.88	4.65	29.29	3.58
	Worst-sampled case	LAT drift	13.31	65.62	0.06	13.76	66.82	0.07
		LAT drift rate	10.46	56.62	1.09	13.24	65.43	11.41
		LAT pitch angle	11.03	58.61	1.21	12.54	63.45	1.29
		LAT pitch rate	2.30	15.03	9.46	2.68	17.41	11.24
Flexible	Nominal	Input	7.98	46.51	75.20	7.01	41.92	75.06
	Worst-sampled case	Input	3.27	21.07	76.83	3.98	25.39	81.08
	Nominal	LAT drift	122.18	90.00	33.80	117.17	90.00	34.29
		LAT drift rate	31.72	87.03	75.28	26.31	84.46	75.28
		LAT pitch angle	65.10	89.94	33.80	59.82	89.88	34.29
		LAT pitch rate	8.39	48.33	75.28	7.44	43.98	75.01
	Worst-sampled case	LAT drift	115.21	90.00	31.91	109.86	90.00	30.46
		LAT drift rate	23.30	82.18	82.65	20.16	78.79	82.94
		LAT pitch angle	60.96	89.90	30.24	53.16	89.75	36.71
		LAT pitch rate	3.84	24.55	88.17	4.15	26.37	80.02
	Nominal	MIMO output	3.68	23.58	0.84	4.00	25.50	0.59
	Worst-sampled case	MIMO output	1.42	9.31	9.75	1.84	12.06	11.04
	Nominal	I/O	2.39	15.56	2.52	2.11	13.78	3.57
	Worst-sampled case	I/O	0.88	5.79	8.89	1.17	7.70	10.77

7.6.4. Time Domain Analysis

For the following analysis, the color coding for the integrated controllers follows the scheme defined in Figure 7.10. Additionally, the shaded regions in the plots represent the results of simulations under dispersed conditions.

Figure 7.10: Color code for time domain analysis - $K_{OLS-NOM-INT}$ vs $K_{OLS-MM5-INT}$.

Low Frequency, High Amplitude (LFHA) Wind Gust

In terms of load performance metric (LPM), drift, and drift rate (Figures 7.11, 7.12, and 7.13, respectively), both controllers exhibit very similar responses under nominal and dispersed conditions, with the nominal controller showing a slight advantage. This difference can be attributed to the Multi-model controller's marginally improved pitch angle performance in both nominal and dispersed conditions (see Figure 7.14).

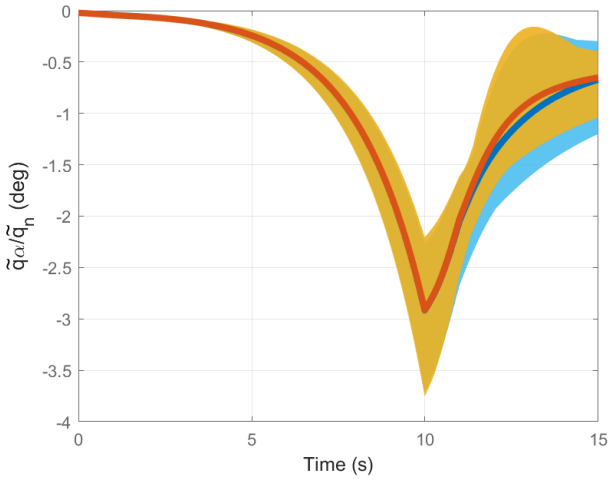

Figure 7.11: LFHA Wind - Load performance metric (LPM), normalized by the nominal dynamic pressure.

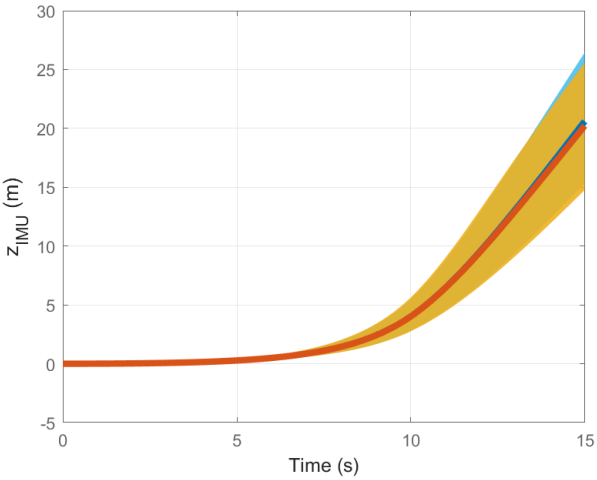

Figure 7.12: LFHA Wind - Drift.

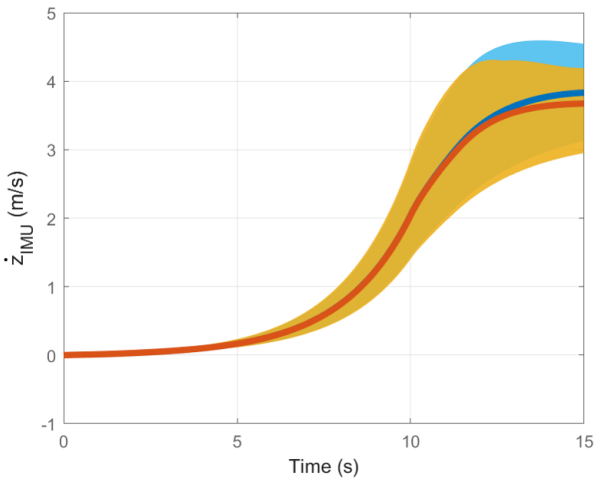

Figure 7.13: LFHA Wind - Drift rate.

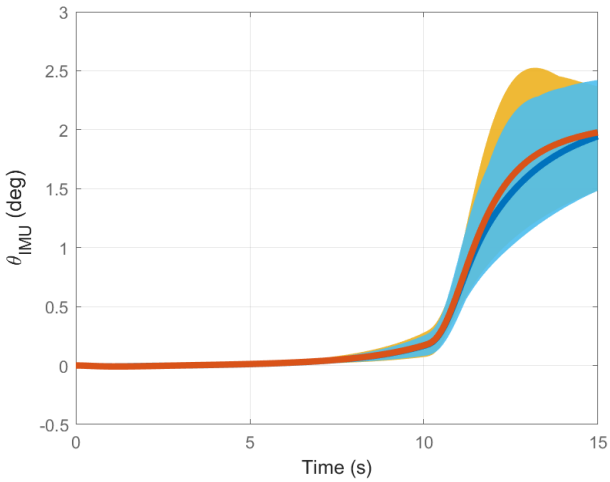

Figure 7.14: LFHA Wind - Pitch angle.

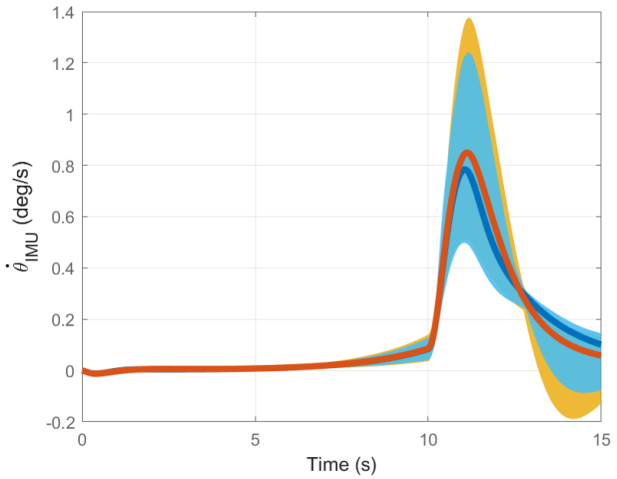

Figure 7.15: LFHA Wind - Pitch rate.

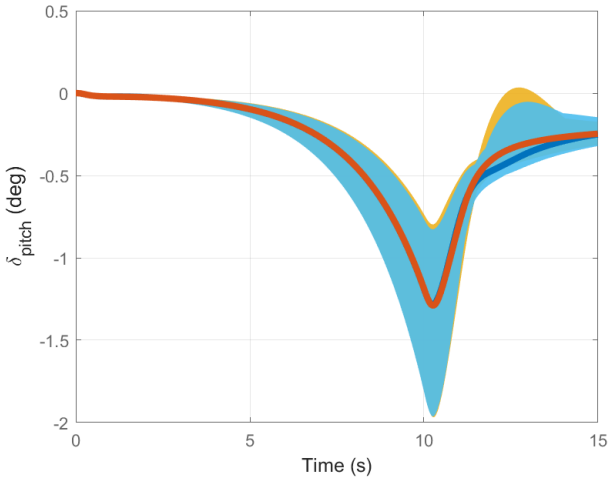

Figure 7.16: LFHA Wind - TVC deflection.

The Multi-model controller also exhibits a smaller pitch rate peak value, as illustrated in Figure 7.15, and nearly indistinguishable nominal and dispersed conditions results in actuation, as shown in Figure 7.16.

Overall, while there is a notable decrease in performance when comparing the integrated controllers to the $\mathcal{H}_\infty$ OLS rigid-only controller from the previous chapter, all plots demonstrate results that meet the performance requirements.

Tracking Test - 0.5 deg step on pitch angle command

In terms of tracking, Figure 7.18 demonstrates that both controllers effectively track the reference signal, with similar nominal responses. However, the Multi-model controller slightly outperforms the nominal controller under dispersed conditions. The same observations apply to the LPM in Figure 7.17 and actuation in Figure 7.20. The only notable exception is the pitch rate (Figure 7.19), where the nominal controller achieves a slightly lower peak value for both nominal and dispersed conditions. Other than this difference, the pitch rate responses remain largely comparable.

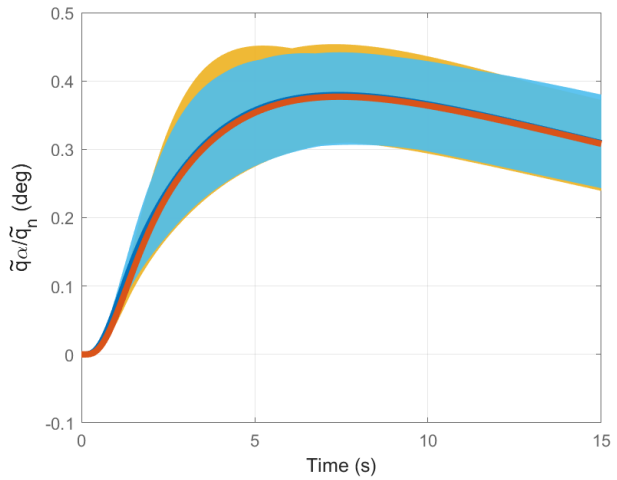

Figure 7.17: Pitch Angle Tracking - LPM, normalized by the nominal dynamic pressure.

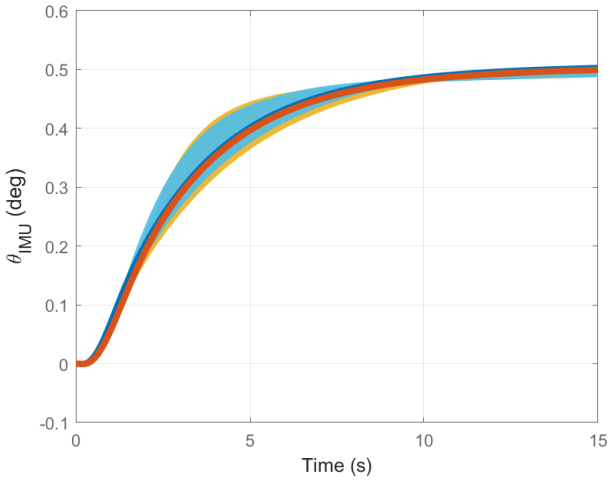

Figure 7.18: Pitch Angle Tracking - Pitch angle.

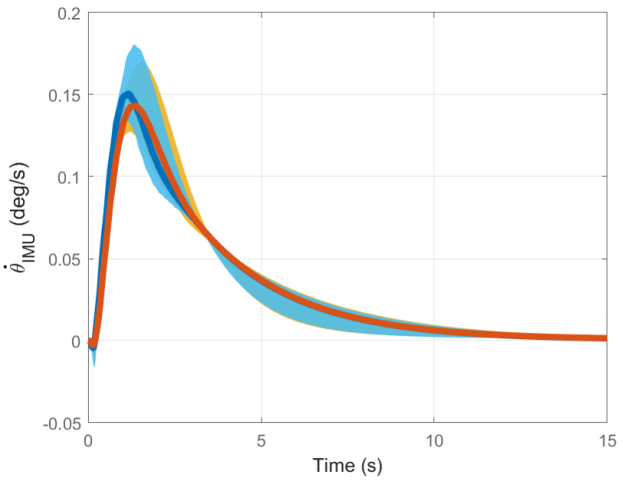

Figure 7.19: Pitch Angle Tracking - Pitch rate.

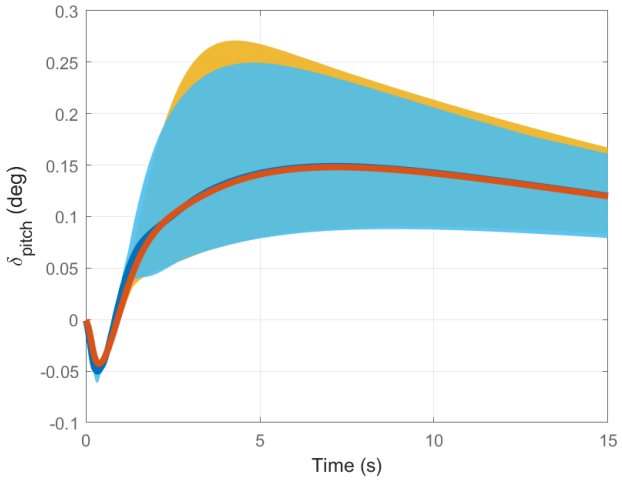

Figure 7.20: Pitch Angle Tracking - TVC deflection.

This test highlights one of the most significant effects of the flexible body: as the first bending mode is very close to the controller's original bandwidth, the controller bandwidth had to be significantly reduced, leading to slower tracking of commands (when compared to the $\mathcal{H}_\infty$ OLS rigid-only controller from the previous chapter). Nevertheless, all plots demonstrate results that meet the performance requirements.

High Frequency, Low to Mid Amplitude (HFLMA) Wind Gust

From Figures 7.21 through 7.26, it is evident that both controllers exhibit very similar nominal responses across all quantities. Under dispersed conditions, the Multi-model controller delivers superior results for drift, drift rate, and pitch angle (Figures 7.22 through 7.24), while achieving comparable performance to the nominal controller for the remaining quantities.

Overall, all the results fall within acceptable bounds. However, it is important to highlight a significant decrease in performance across all results when compared to the $\mathcal{H}_\infty$ OLS rigid-only controller from the previous chapter.

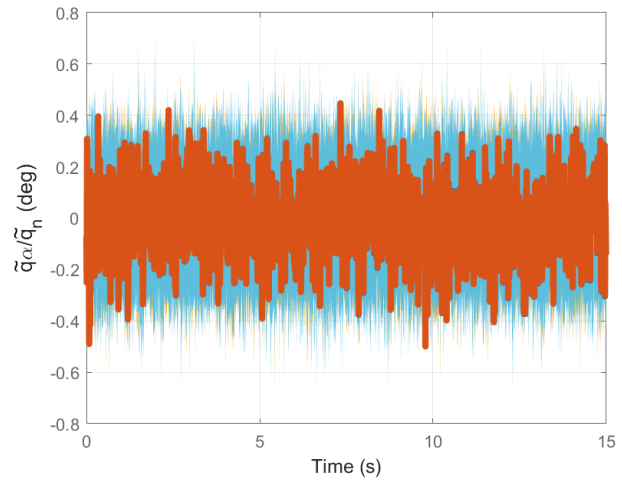

Figure 7.21: HFLMA Wind - LPM, normalized by the nominal dynamic pressure.

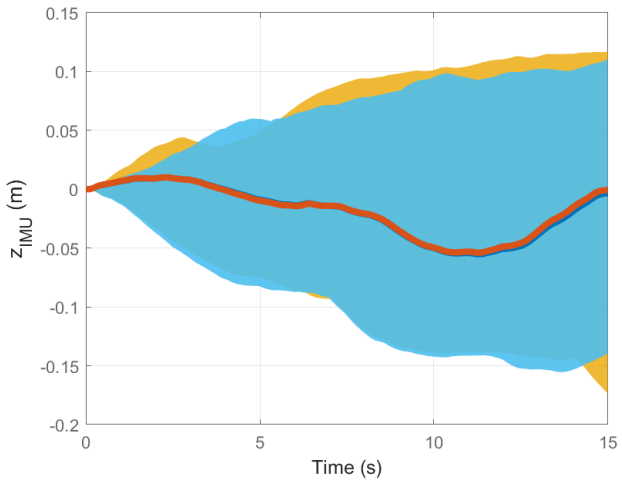

Figure 7.22: HFLMA Wind - Drift.

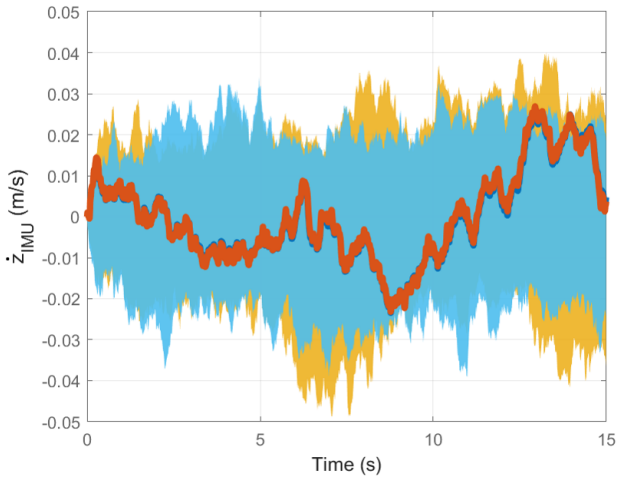

Figure 7.23: HFLMA Wind - Drift rate.

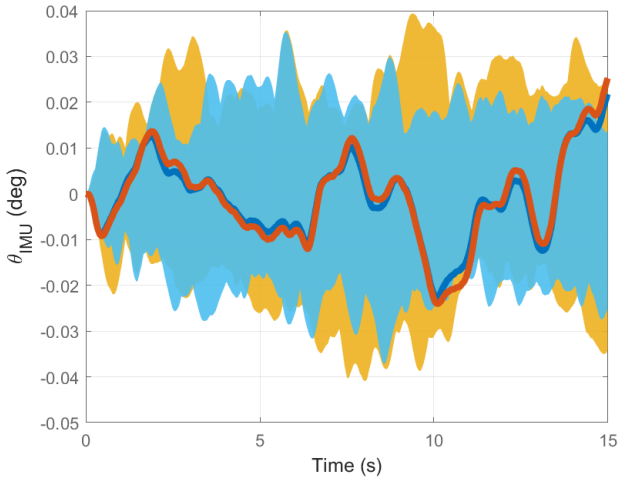

Figure 7.24: HFLMA Wind - Pitch angle.

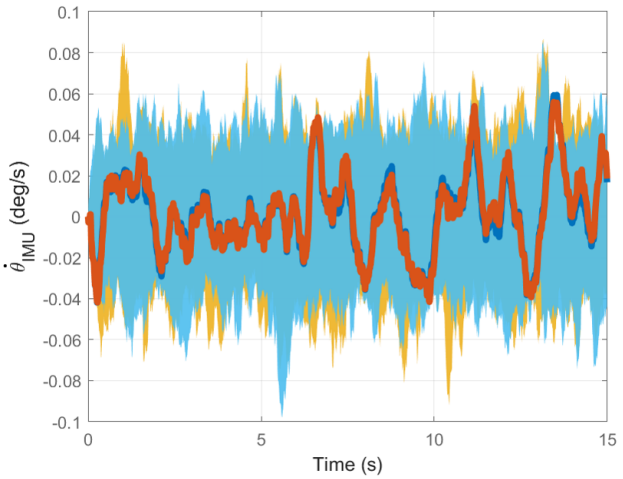

Figure 7.25: HFLMA - Pitch rate.

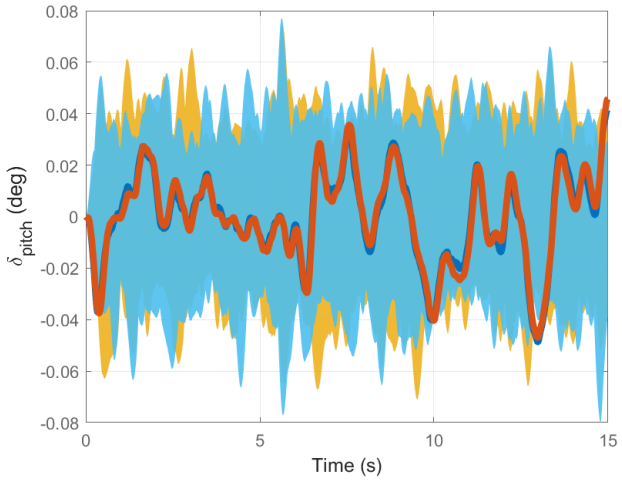

Figure 7.26: HFLMA Wind - TVC deflection.

7.6.5. Worst Sampled Case Gain Analysis

In the following analysis, the thick black lines represent the maximum allowable singular values for the respective test. Meaning that if the highest singular value of all the sampled cases remains bellow the black line, then the respective requirement is fulfilled.

Load Relief

Figure 7.27 demonstrates that the Multi-model controller design approach successfully improves load relief, as evidenced by the reduced worst-sampled gain compared to the nominal controller. However, the figure also reveals instances where the load requirement may be exceeded. This was expected, as the pitch angle post-compensator gain had to be significantly reduced compared to that of the $\mathcal{H}_\infty$ OLS rigid-only controllers to account for the effects of flexible body behavior.

It is important to emphasize that this does not necessarily indicate a failure to meet the load relief requirement, as the test is inherently conservative. A non-linear simulation would be necessary to fully validate compliance with this requirement; however, such an analysis falls outside the scope of this work. Taking into account this limitation and the positive outcomes of the previous time-domain analysis, it is still

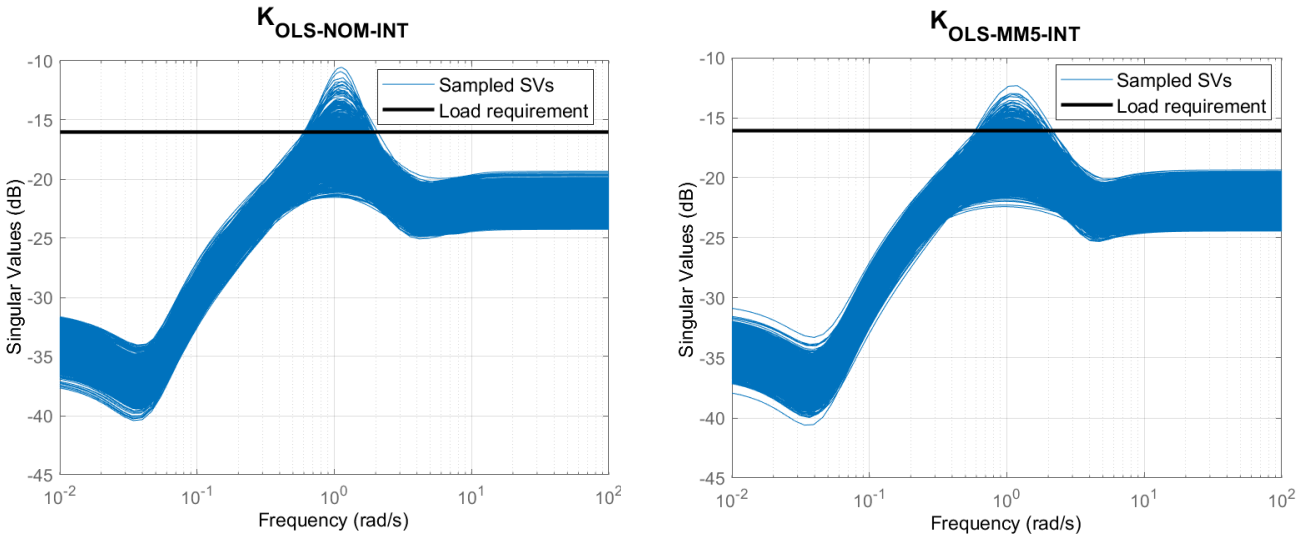

Figure 7.27: Worst-Sampled gain of $T_{v_w \to \tilde{q}\alpha/\tilde{q}_n}$ - $K_{OLS-NOM-INT}$ vs $K_{OLS-MM5-INT}$.

expected that the Multi-model controller provides adequate load relief.

Drift and Drift Rate

In the drift plots (Figure 7.28), both controllers clearly meet the requirement, with the worst-sampled gain staying below the thick black line. Additionally, both controllers demonstrate comparable performance, achieving similarly strong results. In the drift rate plots (Figure 7.29), it is possible to see that that both controllers attain equally good results and fulfill the requirements.

Actuation

In terms of actuation, Figure 7.30 illustrates that for $T_{\theta_{cmd} \to \delta_{cmd}}$, both controllers achieve similar performance and meet the specified requirements. However, Figure 7.31 highlights the superiority of the Multi-model controller. While the Multi-model controller satisfies the requirement imposed on $T_{v_w \to \delta_{cmd}}$, as evidenced by its worst-sampled gain remaining below the black line, the nominal controller fails to do so in some cases, violating the requirement.

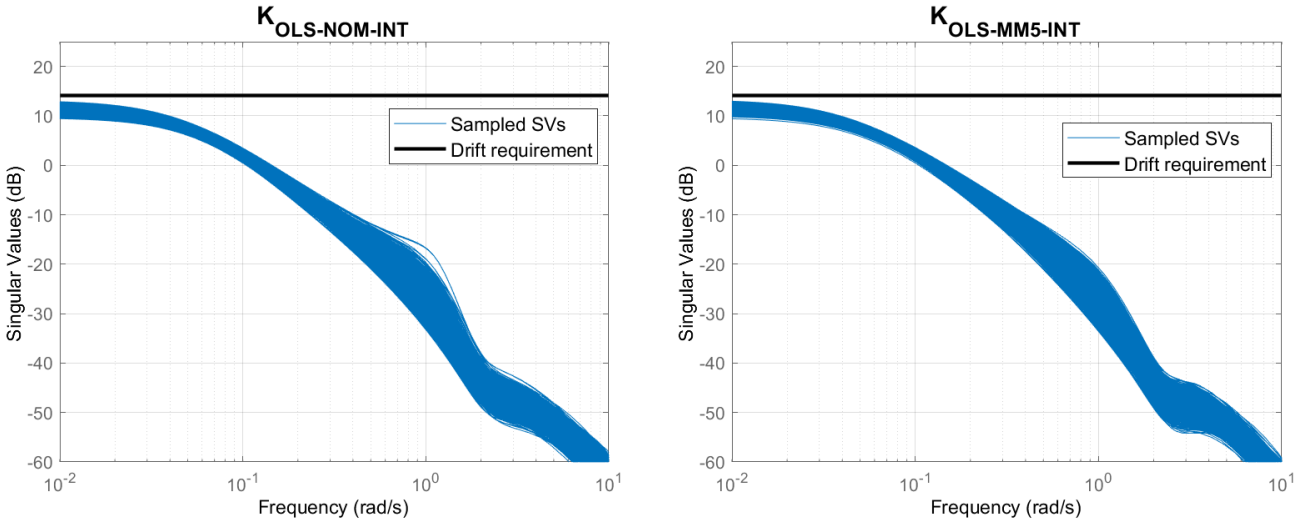

Figure 7.28: Worst-Sampled gain of $T_{v_w \to z_{IMU}}$ - $K_{OLS-NOM-INT}$ vs $K_{OLS-MM5-INT}$.

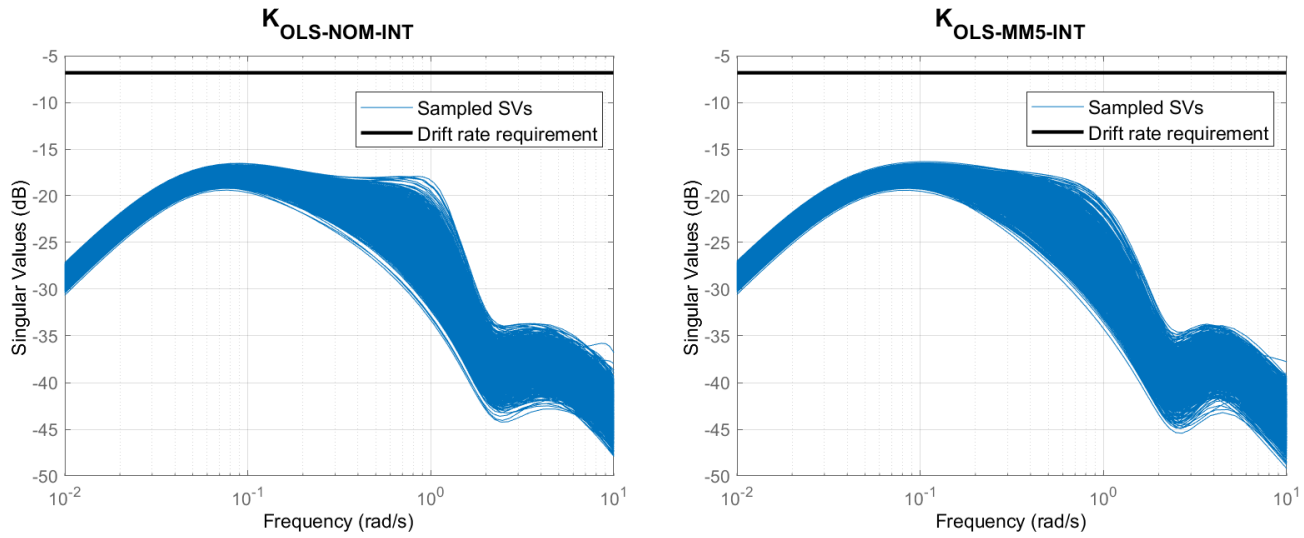

Figure 7.29: Worst-Sampled gain of $T_{v_w \to \dot{z}_{IMU}}$ - $K_{OLS-NOM-INT}$ vs $K_{OLS-MM5-INT}$.

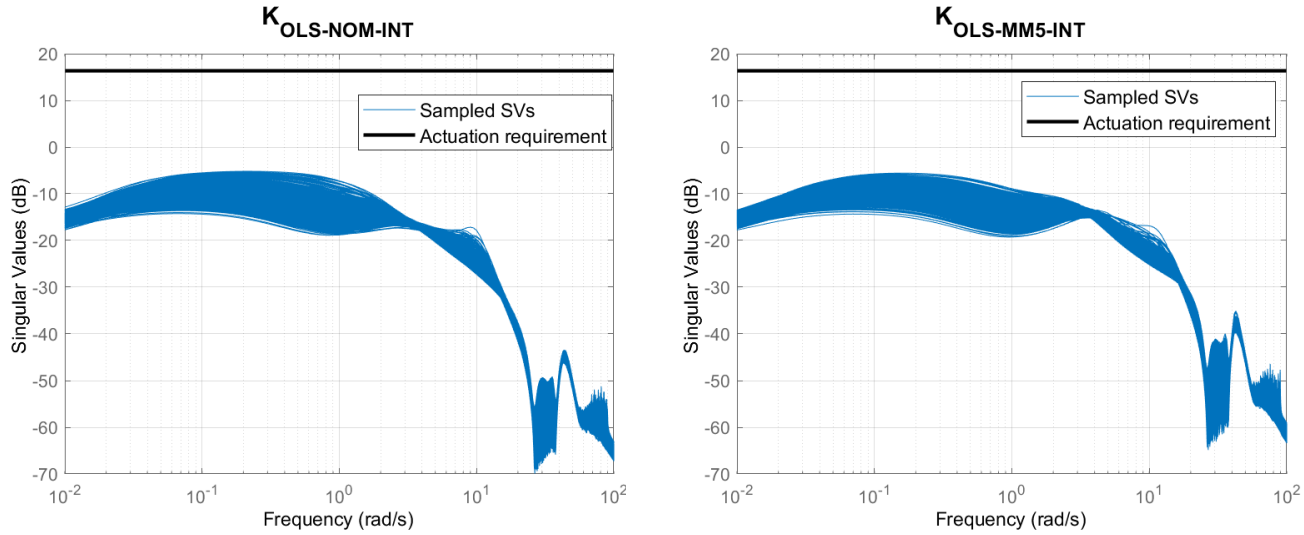

Figure 7.30: Worst-Sampled gain of $T_{\theta_{cmd} \to \delta_{cmd}}$ - $K_{OLS-NOM-INT}$ vs $K_{OLS-MM5-INT}$.

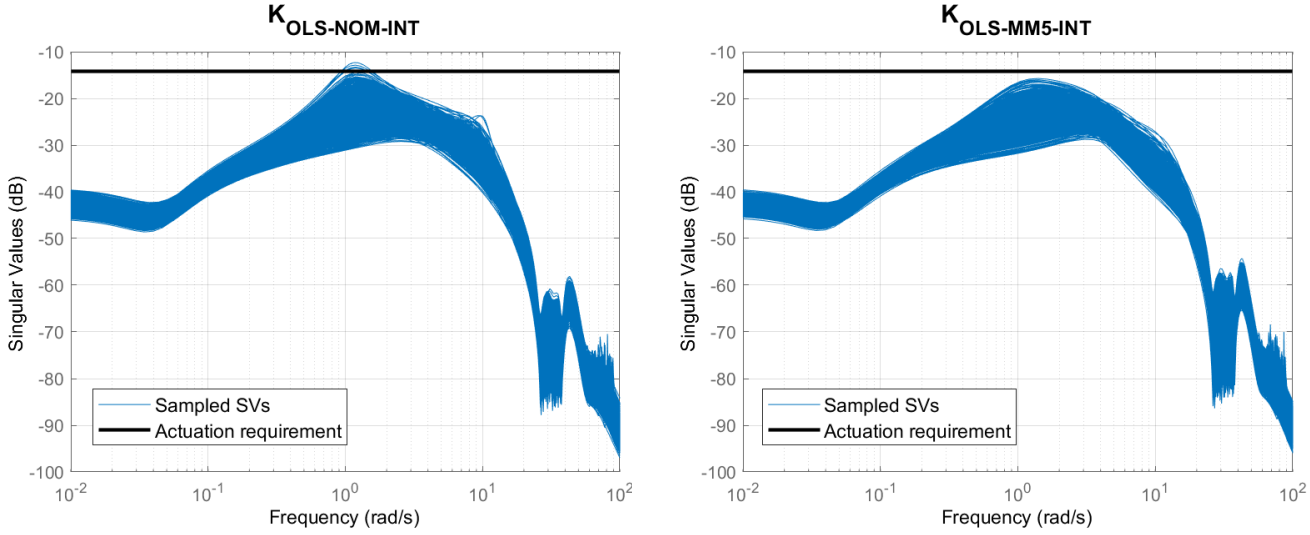

Figure 7.31: Worst-Sampled gain of $T_{v_W \to \delta_{cmd}}$ - $K_{OLS-NOM-INT}$ vs $K_{OLS-MM5-INT}$.

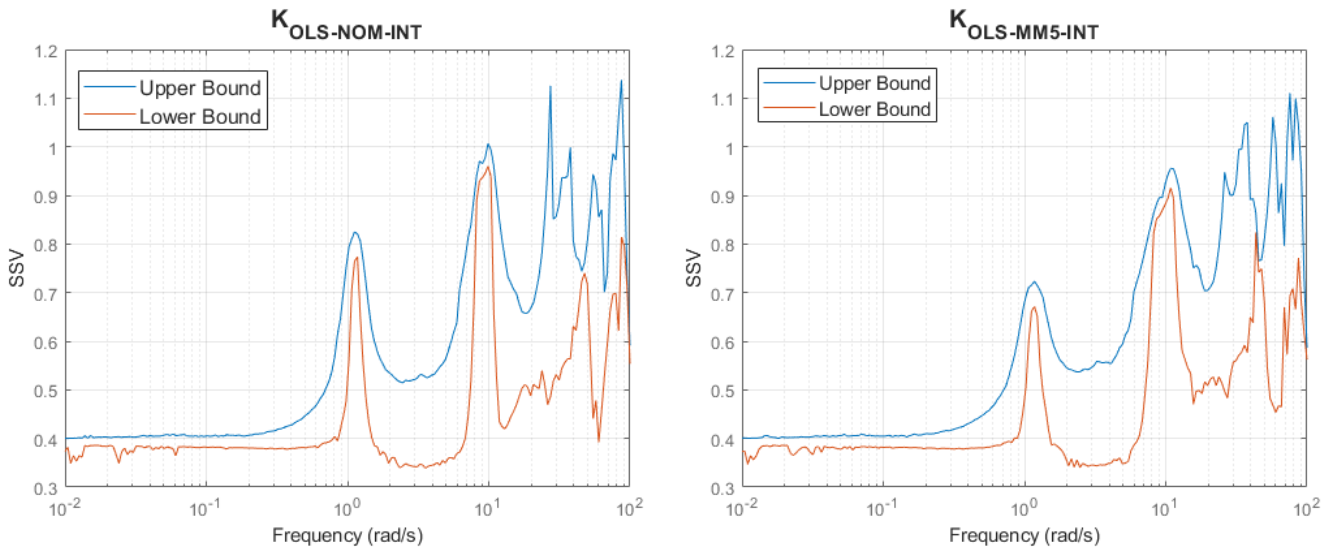

Figure 7.32: Structured Singular Value - $K_{OLS-NOM-INT}$ vs $K_{OLS-MM5-INT}$.

7.6.6. Structured Singular Value

From Figure 7.32, it can be observed that, in general, the Multi-model controller exhibits lower upper and lower bounds for the Structured Singular Value (SSV) compared to the nominal controller. While the lower bound of the SSV remains below 1 for both controllers, the upper bound exceeds 1, introducing some uncertainty regarding stability across the full range of modeled uncertainties. Despite this, the higher SSV bound does not appear to be concerning, as none of the results from Subsections 7.6.3 and 7.6.4 indicate instability. The elevated SSV values can be attributed to the stringent loop shaping applied to enhance controller performance, along with the large number of uncertain parameters — 25 in total, including 10 more than in the rigid-only case.

Overall, based on the previous analysis, it can be reasonably expected, though not guaranteed due to the limited test campaign, that the Multi-model controller fulfills all the requirements. This outcome is promising, as it indicates that the integrated rigid/flexible body $\mathcal{H}_\infty$ OLS methodology developed for launch vehicles is effective. However, to conclusively verify compliance with all requirements, it would be necessary to design a controller capable of covering the full flight envelope, followed by conducting non-linear simulations. Both tasks fall beyond the scope of the current work and are identified as objectives for future research.

7.7. Second Research Question and Chapter Conclusion

The goal of this chapter was to develop an integrated rigid body controller and bending filter design methodology using $\mathcal{H}_\infty$ OLS, ultimately addressing Research Question 2 and its sub-research questions. These are restated bellow to refresh the reader's memory and provide context.

Research Question 2

Can the integrated attitude controller/bending filter, designed with $\mathcal{H}_\infty$ OLS, achieve similar results to the controller designed with the traditional separate approach, whilst improving design time?

Research Question 2.1

Does the integrated controller demonstrate comparable performance in terms of handling uncertainties, disturbance rejection, stability margins, tracking accuracy, drift minimization, load relief, and actuator usage, when compared to the separate design approach?

Research Question 2.2

Can the integrated design methodology reduce the overall design time compared to the traditional separate design process?

With the research questions restated, the subsequent discussion addresses the answers. Firstly, Research Question 2.2 was answered through the work developed in Section 7.2 and in Section 7.3. In the former, the separate rigid/flexible body controller design methodology was described, and in the latter, the integrated rigid/flexible body controller design methodology was introduced. From these two sections it was possible to conclude that the integrated design approach effectively reduced design time, therefore answering Research Question 2.2.

Research Question 2.1 was examined in Section 7.4, where controllers developed through the separate and integrated methods were directly compared. The results demonstrated that both methods yielded identical controllers, with equivalent performance, robustness, and stability, thereby addressing Research Question 2.1.

Taking all factors into account, Research Question 2 can be answered as follows: The integrated design approach for the rigid controller and bending filter using $\mathcal{H}_\infty$ OLS yields results identical to those achieved by the traditional separate design method, while offering a substantial reduction in design time.

Additionally, two sections (Section 7.5 and Section 7.6) were included to further validate the integrated design methodology. In Section 7.5, the integrated design methodology was applied to develop a Multi-model based controller. In Section 7.6, the MM integrated controller and nominal integrated controller were tested to assess their ability to meet the launch vehicle's stringent requirements. Although the dispersed condition testing was limited, preliminary results provide a reasonable expectation, though not a guarantee, that the integrated Multi-model controller fulfills all the requirements, supporting the viability of the integrated approach.

8

Conclusion and Future Research

8.1. Conclusion

This thesis investigates the application of $\mathcal{H}_\infty$ Open Loop Shaping (OLS) as a method for designing the Thrust Vector Control (TVC) system for the atmospheric ascent of flexible launch vehicles. The research is structured around two key objectives:

Research Objective

To evaluate the capability of $\mathcal{H}_\infty$ Open Loop Shaping in producing launch vehicle attitude controllers with equal or superior robustness, stability, and performance compared to those designed using $\mathcal{H}_\infty$ Closed Loop Shaping, while reducing design time and complexity.

Research Objective

To investigate the potential of utilizing an integrated rigid body controller and bending filter design strategy based on $\mathcal{H}_\infty$ Open Loop Shaping, with the goal of reducing design time while maintaining robustness, stability, and performance compared to the separate design approach.

For the first objective, this thesis demonstrates that both $\mathcal{H}_\infty$ OLS and $\mathcal{H}_\infty$ CLS (Closed Loop Shaping), when applied to design controllers of the same order and structure, result in identical controllers with equal robustness, stability, and performance. However, the $\mathcal{H}_\infty$ OLS approach reduces design time and complexity.

The $\mathcal{H}_\infty$ OLS design methodology developed in this work introduces a systematic two-step approach. The first step involves shaping the open loop gain by defining a proportional-integral (PI) filter, two lead-lag (LL) filters, and two gains. The second step consists of robustifying the shaped open loop system against normalized coprime factor uncertainty. While the initial loop shaping steps may seem non-trivial, the guidelines established in Subsection 6.7.4 significantly simplify the process. Furthermore, of the five parameters involved in the loop shaping step — PI filter, two lead-lag filters, and two gains — the PI and lead-lag filters only need to be configured once. Notably, during this thesis, the same PI and lead-lag filters were used consistently across different flight points, which resulted in satisfactory performance in all cases. This significantly simplifies the process of designing the controller for additional flight points, as the filters designed for the first flight point can be reused, reducing the design process for subsequent flight points to

selecting two gains.

In addition to these advantages, the $\mathcal{H}_\infty$ OLS approach also integrates seamlessly with the Multi-model approach, facilitating the achievement of robustly tuned controllers with minimal additional effort. Moreover, when implemented correctly, the $\mathcal{H}_\infty$ OLS approach automatically balances the input and output loop gains and ensures simultaneous and symmetric stability margins at both the plant input and output. These features make the $\mathcal{H}_\infty$ OLS approach particularly advantageous for applications where robustness and efficiency are key.

However, the $\mathcal{H}_\infty$ OLS approach does come with certain limitations. Notably, it is only through testing that it can be verified whether the controllers produced with $\mathcal{H}_\infty$ OLS meet the closed loop requirements. Additionally, the need to convert closed loop requirements into open loop specifications introduces some complexity. To address this, this thesis provides a detailed mapping between launch vehicle requirements, their corresponding open loop specifications, and their relevant frequency ranges, offering practical guidance for shaping loop gains to meet specific objectives.

With regard to $\mathcal{H}_\infty$ OLS, $\mathcal{H}_\infty$ CLS has both advantages and disadvantages, although overall, it was found to be a more challenging design method. On one hand, $\mathcal{H}_\infty$ CLS integrates seamlessly with the Multi-model approach and the robust parametric controller design approach (referred to as the LFT approach in this work). Additionally, it allows for the direct translation of closed loop requirements into constraints. This provides the assurance that if the controller satisfies these constraints, it will meet the closed loop requirements, at least nominally. On the other hand, $\mathcal{H}_\infty$ CLS requires the designer to manipulate multiple transfer functions simultaneously (ranging from four to six in this work) to achieve all performance requirements and ensure simultaneous symmetric stability margins at both the plant input and output. This tuning is not trivial, as specifying conflicting requirements can easily occur, requiring repeated adjustments and iterations to balance all the constraints effectively. Furthermore, when designing controllers for other flight points, nearly all constraints must be readjusted, which can be time-consuming and requires careful consideration to maintain performance, stability, and robustness.

Overall, both methods have their advantages and disadvantages. Since they reach the same result, selecting one over the other can be considered a matter of preference. However, this is not the author's view. While both methods yield the same outcome, the $\mathcal{H}_\infty$ OLS design methodology is simpler and reduces design time for three key reasons: it avoids manipulating multiple transfer functions, offers higher reproducibility between flight points, and automatically ensures robustness at both the plant input and output. Although these benefits come at the cost of converting requirements into open loop specifications and setting loop shaping weighting filters to meet the specifications, both challenges have been addressed and simplified in this thesis. Therefore, the author believes that the $\mathcal{H}_\infty$ OLS methodology developed in this work improves on $\mathcal{H}_\infty$ CLS, as it simplifies and accelerates the design process, while delivering equivalent results in terms of robustness, stability, and performance.

It is important to note that this research was specifically limited to the case of controlling a single thrust vector along one axis for a particular launch vehicle configuration and launch environment. Future research can address this limitation by testing the $\mathcal{H}_\infty$ OLS methodology in other control scenarios, such as multi-axis and multi-engine systems, as well as in different vehicle configurations and launch environments.

For the second research objective, the thesis demonstrates that the rigid body $\mathcal{H}_\infty$ OLS design process can be extended into an integrated methodology that simultaneously tunes both the rigid body controller and the bending filter. This approach yield identical results to the traditional separate design method, while significantly reducing design time and effort. Moreover, since this approach builds directly on the rigid

body $\mathcal{H}_\infty$ OLS process, it inherits all of its benefits, including its simplicity and efficiency. Finally, the test campaign conducted for the integrated controllers suggests that the Multi-model integrated controller is capable of meeting the rigorous performance and robustness requirements for launch vehicles. While these findings are promising, further validation through more comprehensive methods, such as non-linear analysis, is necessary to fully confirm that the controller meets all requirements and to strengthen confidence in the applicability and reliability of the proposed methodologies.

In conclusion, this research presents two key takeaways. First, $\mathcal{H}_\infty$ OLS offers a robust framework for simplifying the design of TVC systems for launch vehicles, without compromising performance and robustness. Second, the $\mathcal{H}_\infty$ OLS integrated design methodology for the rigid body controller and bending filter effectively streamlines the design process, while achieving identical results to the separate approach. These findings not only validate $\mathcal{H}_\infty$ OLS as a viable alternative to $\mathcal{H}_\infty$ CLS but also introduce a more efficient methodology for designing both the rigid body controller and the bending filter in launch vehicle TVC control systems.

8.2. Future Research

This section outlines several potential research directions that could build upon this work.

Extend the design techniques for the multi-axis and multi-engine case.

In this work, the control strategy was developed for a single thrust vector along one axis, specifically focusing on the pitch axis. Future research could extend this by adapting the design techniques to account for multiple engines and their motion across both pitch and yaw axes. This would involve refining the developed $\mathcal{H}_\infty$ Open Loop Shaping design methodology to effectively manage the interactions between the axes and the complexities introduced by multi-engine dynamics, ensuring more robust and efficient control for launch vehicles.

Extend the design for all flight points, perform gain scheduling and fully evaluate the results in a non-linear simulator.

Another potential research direction would be to implement the design method for all flight points, followed by the application of gain scheduling to create a full-envelope controller. Testing this controller using a non-linear simulator, such as the one used in *Tapia* (2019), would enable further verification of the design methodologies developed in this thesis, ensuring their effectiveness across the entire flight envelope.

Test the viability of the developed $\mathcal{H}_\infty$ OLS weighting filter parametrization and guidelines for launch vehicles with very different configurations and launch environments.

As a final direction, this research could be extended by testing the adaptability of the developed $\mathcal{H}_\infty$ OLS methodology on launch vehicles with different structural configurations (for instance a launch vehicle with sloshing) and operational environments. Such an investigation would determine the flexibility and applicability of the current design methodology across a wider range of vehicle architectures and mission profiles.

References

AElfwine. (2004). V-2 rocket photograph [Licensed under CC BY-SA 3.0, available at: https://creativecommons.org/licenses/by-sa/3.0/. Image downloaded from: https://en.wikipedia.org/wiki/V-2_rocket#. Accessed: 2025-01-07.].

Ansari, *U.*, & *Bajodah*, *A. H.* (2015). Generalized dynamic inversion scheme for satellite launch vehicle attitude control. *IFAC-PapersOnLine*, *vol. 48*(no. 9), pp. 114–119.

Apkarian, *P.*, & *Noll*, *D.* (2006). Nonsmooth $\mathcal{H}_\infty$ synthesis. *IEEE Transactions on Automatic Control*, *vol. 51*(no. 1), pp. 71–86.

Apkarian, *P.*, & *Noll*, *D.* (2017). The $\mathcal{H}_\infty$ Control Problem is Solved. *Aerospace Lab*, (no. 13), pp. 1–11.

Apkarian, *P.*, *Dao*, *M.*, & *Noll*, *D.* (2015). Parametric robust structured control design. *IEEE Transactions on Automatic Control*, *vol. 60*, pp. 1857–1869.

Bacon, *B.*, & *Ostroff*, *A.* (2000). Reconfigurable flight control using nonlinear dynamic inversion with a special accelerometer implementation. *AIAA Guidance, Navigation, and Control Conference and Exhibit*, pp. 4565.

Bates, *D.*, & *Postlethwaite*, *I.* (2002). *Robust multivariable control of aerospace systems*. DUP Science.

Belletti Araque, *J. P.*, *Zavoli*, *A.*, *Trotta*, *D.*, & *De Matteis*, *G.* (2021). Advanced $\mathcal{H}_\infty$ synthesis for launch vehicle attitude control in atmospheric flight. *AIAA Science and Technology Forum and Exposition, AIAA SciTech Forum 2021*, pp. 1–19.

Bernard, *M.* (2009). *VEGA missionization and post flight analyses* [Doctoral dissertation, Sapienza Universita di Roma].

Biertümpfel, *F.*, *Pfifer*, *H.*, & *Theis*, *J.* (2024). Robust space launcher control with time-varying objectives. *Journal of Guidance, Control, and Dynamics*, pp. 1–11.

Blackburn, *T.*, & *Vaughan*, *D.* (1971). Application of linear optimal control and filtering theory to the Saturn V launch vehicle. *IEEE Transactions on Automatic Control*, *vol. 16*(no. 6), pp. 799–806.

Brito, *A.*, *Filho*, *W. L.*, & *Ramos*, *F.* (2006). Approach comparison for controller design of a launcher. *Proceedings of the 6th ESA International Conference on Spacecraft Guidance, Navigation and Control Systems*.

Brockett, *R.* (1978). Feedback invariants for nonlinear systems. *IFAC Proceedings Volumes*, *vol. 11*(no. 1), pp. 1115–1120.

Buțu, *F.-A.*, *Lungu*, *R.*, & *Bărbulescu*, *L.-F.* (2016). Adaptive flight control for a launch vehicle based on the concept of dynamic inversion. *20th International Conference on System Theory, Control and Computing (ICSTCC)*, pp. 812–817.

Byrnes, *C. I.*, & *Isidori*, *A.* (1989). New results and examples in nonlinear feedback stabilization. *Systems & Control Letters*, *vol. 12*, pp. 437–442.

Carril, *E. .-. P.* (2024). Artist's impression of Copernicus Sentinel-1C in its VEGA-C launcher. [Licensed under ESA Standard Licence for educational, editorial, and informational use. Image downloaded from: https://www.esa.int/Applications/Observing_the_Earth/Copernicus/Sentinel-1/Watch_live_VEGA-C_to_launch_Sentinel-1C. Accessed: 2025-01-07.].

Chiang, *R. Y.*, & *Safonov*, *M. G.* (1988). *Robust control toolbox*. MathWorks.

Choi, *H. D.*, & *Kim*, *J.* (2001). Adaptive notch filter design for bending vibration of a sounding rocket. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, *vol. 215*(no. 1), pp. 13–23.

Costa, *B. A.*, *Parente*, *F. L.*, *Belfo*, *J.*, *Somma*, *N.*, *Rosa*, *P.*, *Igreja*, *J. M.*, *Belhadj*, *J.*, & *Lemos*, *J. M.* (2024). A reinforcement learning approach for adaptive tracking control of a reusable rocket model in a landing scenario. *Neurocomputing*, *vol. 577*, pp. 127377.

David, *S.*, *Baldesi*, *G.*, *Döll*, *C.*, & *Bérard*, *C.* (2008). Self-scheduling controller for a launcher in atmospheric ascent. *IFAC Proceedings Volumes*, *vol. 41*(no. 2), pp. 14005–14010.

Díaz, *E.* (2010). *Design of a robust controller for the VEGA TVC using the μ synthesis technique* [Master's thesis, Universitat Politècnica de Catalunya].

Doyle, *J. C.* (1978a). Guaranteed margins for LQG regulators. *IEEE Transactions on Automatic Control*, *vol. AC-23*(no. 4), pp. 756–757.

Doyle, *J. C.* (1978b). Robustness of multiloop linear feedback systems. *1978 IEEE Conference on Decision and Control including the 17th Symposium on Adaptive Processes*, pp. 12–18.

Doyle, *J. C.* (1984). Lecture notes on advances in multivariable control [ONR/Honeywell Workshop].

Doyle, *J. C.*, *Glover*, *K.*, *Khargonekar*, *P.*, & *Francis*, *B. A.* (1989). State space solutions to standard $\mathcal{H}_2$ and $\mathcal{H}_\infty$ control problems. *IEEE Transactions on Automatic Control*, *vol. AC-34*(no. 8), pp. 731–747.

Doyle, *J.*, *Wall*, *J.*, & *Stein*, *G.* (1982). Performance and robustness analysis for structured uncertainty. *Proceedings of the IEEE Conference on Decision and Control*, pp. 629–636.

Doyle, *J.* (1982). Analysis of feedback systems with structured uncertainties. *IEE Proceedings D (Control Theory and Applications)*, *vol. 129*(no. 6), pp. 242–250.

Doyle, *J. C.* (1983). Synthesis of robust controllers and filters. *Proceedings of the IEEE Conference on Decision and Control*, pp. 109–114.

Drake, *D.*, *Xin*, *M.*, & *Balakrishnan*, *S. N.* (2004). New nonlinear control technique for ascent phase of reusable launch vehicles. *Journal of Guidance, Control, and Dynamics*, *vol. 27*(no. 6), pp. 930–937.

Elmelhi, *A.* (2014). Modified adaptive notch filter based on neural network for flexible dynamic control. *International Journal of Computer and Electrical Engineering*, *vol. 6*, pp. 185–190.

Emelyanov, *S.* (1957). *Variable structure control systems*. Nauka.

Emelyanov, *S.* (1986). Higher order sliding modes in the binary control systems. *Sov. Phys. Dokl.*, *vol. 4*, pp. 291–293.

Encyclopedia Britannica. (2024). Launch vehicle [Accessed: 20 February 2024]. https://www.britannica.com/technology/launch-vehicle

ESA/CNES/Arianespace. (2019). Launch of Ariane V [Licensed under ESA Standard Licence for educational, editorial, and informational use. Image downloaded from: https://www.esa.int/Enabling_Support/Space_Transportation/Europe_s_Spaceport/Ariane_5_launch_campaigns. Accessed: 2025-01-07.].

Falcoz, *A.*, *Pittet*, *C.*, *Bennani*, *S.*, *Guignard*, *A.*, *Bayart*, *C.*, & *Frapard*, *B.* (2015). Systematic design methods of robust and structured controllers for satellites. Application to the refinement of Rosetta's orbit controller. *CEAS Space Journal*, *vol. 7*(no. 3), pp. 319–334.

Fari, *S.*, *Seelbinder*, *D.*, *Theil*, *S.*, & *Simplicio*, *P.* (2024). Robust fault detection and isolation algorithms for TVC systems: An experimental test. *Proceedings of the 75th International Astronautical Congress*.

Gahinet, *P.*, & *Apkarian*, *P.* (1994). A linear matrix inequality approach to $\mathcal{H}_\infty$ control. *International Journal of Robust and Nonlinear Control*, *vol. 4*(no. 4), pp. 421–448.

George, *K. E.*, *Mary*, *A.*, *Varghese*, *A. M.*, *Sivadas*, *M.*, *Mohan*, *A.*, & *Santhosh*, *A.* (2021). Robust attitude control of launch vehicle using $\mathcal{H}_\infty$ controller. *2021 International Conference on Advances in Computing and Communications (ICACC)*, pp. 1–6.

Glover, *K.*, & *Doyle*, *J. C.* (1988a). State-space formulae for all stabilizing controllers that satisfy an $\mathcal{H}_\infty$ -norm bound and relations to risk sensitivity. *Systems and Control Letters*, *vol. 11*, pp. 167–172.

Glover, *K.*, *Limebeer*, *D. J.*, *Doyle*, *J. C.*, *Kasenally*, *E. M.*, & *Safonov*, *M. G.* (1991). A characterization of all solutions to the four block general distance problem. *SIAM Journal of Control and Optimization*, *vol. 29*, pp. 283–324.

Glover, *K.*, & *McFarlane*, *D.* (1989). Robust stabilization of normalized coprime factor plant descriptions with $\mathcal{H}_\infty$ -bounded uncertainty. *IEEE Transactions on Automatic Control*, *vol. 34*(no. 8), pp. 821–830.

Glover, *K.*, *Vinnicombe*, *G.*, & *Papageorgiou*, *G.* (2000). Guaranteed multi-loop stability margins and the gap metric. *Proceedings of the 39th IEEE Conference on Decision and Control*, *vol. 4*, pp. 4084–4085 vol.4.

Glover, *K.* (1984). All optimal hankel-norm approximations of linear multivariable systems and their L^∞ -error bounds. *International Journal of Control*, *vol. 39*(no. 6), pp. 1115–1193.

Glover, *K.*, & *Doyle*, *J. C.* (1988b). State-space formulae for all stabilizing controllers that satisfy an $\mathcal{H}_\infty$ -norm bound and relations to relations to risk sensitivity. *Systems & Control Letters*, *vol. 11*, pp. 167–172.

Greensite, *A. L.* (1967a). *Analysis and design of space vehicle flight control systems. Volume I - short period dynamics* (NASA Contractor Report No. CR-820). NASA.

Greensite, *A. L.* (1967b). *Analysis and design of space vehicle flight control systems. Volume VII - attitude control during launch* (NASA Contractor Report No. CR-826). NASA.

Hague, *T. N.* (2000). *An application of robust $\mathcal{H}_2$/ $\mathcal{H}_\infty$ control synthesis to launch vehicle ascent by tyler nicklaus hague.* [Doctoral dissertation, Massachusetts Institute of Technology].

Hernandez, *J. A. M.*, *Sagliano*, *M.*, *Heidecker*, *A.*, *Seelbinder*, *D.*, *Fari*, *S.*, *Theil*, *S.*, *Woicke*, *S.*, & *Dumont*, *E.* (2021). Ascent flight control system for reusable launch vehicles: Full order and structured $\mathcal{H}_\infty$ designs. *11th International ESA Conference on Guidance, Navigation & Control Systems*.

Hunt, *L. R.*, & *Su*, *R.* (1981). Control of nonlinear time-varying systems. *20th IEEE Conference on Decision and Control including the Symposium on Adaptive Processes*, pp. 558–563.

Jang, *J. W.*, *Hall*, *R.*, *Bedrossian*, *N.*, & *Hall*, *C.* (2008). Ares-I bending filter design using a constrained optimization approach. *AIAA Guidance, Navigation and Control Conference and Exhibit*, pp. 6289.

Kharisov, *E.*, *Gregory*, *I.*, *Cao*, *C.*, & *Hovakimyan*, *N.* (2008). L1 adaptive control for flexible space launch vehicle and proposed plan for flight validation. *AIAA Guidance, Navigation and Control Conference and Exhibit*, pp. 7128.

Kokotović, *P.*, & *Arcak*, *M.* (2001). Nonlinear and adaptive control: An abbreviated status report. *The 9th Mediterranean Conference on Control and Automation*.

Kunusch, *C.*, *Puleston*, *P.*, & *Mayosky*, *M.* (2012). Fundamentals of sliding-mode control design. In *Sliding-mode control of PEM fuel cells* (pp. 35–71). Springer London.

Levantovsky, *L.* (1985). Second order sliding algorithms: Their realization. *Dynamics of Heterogeneous Systems*, pp. 32–43.

Limebeer, *D.*, *Kasenally*, *E.*, & *Perkins*, *J.* (1993). On the design of robust two degree of freedom controllers. *Automatica*, *vol. 29*(no. 1), pp. 157–168.

Marcos, *A.*, *Bates*, *D.*, & *Postlethwaite*, *I.* (2005). Control oriented uncertainty modelling using μ sensitivities and skewed μ analysis tools. *Proceedings of the 44th IEEE Conference on Decision and Control*, pp. 6436–6441.

Marcos, *A.*, *Roux*, *C.*, *Rotunno*, *M.*, *Joos*, *H.-D.*, *Bennani*, *S.*, *Penin*, *L.*, & *Caramagno*, *A.* (2011). The V&V problematic for launchers: Current practise and potential advantages on the application of modern analysis techniques. *Proceedings of ESA GNC, 2011*.

McFarlane, *D.* (1989). *Robust controller design using normalized coprime factor plant descriptions*. Springer.

McFarlane, *D.*, & *Glover*, *K.* (1988). An $\mathcal{H}_\infty$ design procedure using robust stabilization of normalized coprime factors. *Proceedings 1988 Conference on Decision and Control*.

McFarlane, *D.*, & *Glover*, *K.* (1992). A loop-shaping design procedure using $\mathcal{H}_\infty$ synthesis. *IEEE Transactions on Automatic Control*, *vol. 37*(no. 6), pp. 759–769.

McFarlane, *D.*, *Glover*, *K.*, & *Noton*, *M.* (1988). Robust stabilization of a flexible space-platform: An $\mathcal{H}_\infty$ coprime factorization approach. *Proceedings 1988 International Conference on Control*.

Meyer, *G.*, *Su*, *R.*, & *Hunt*, *L.* (1984). Application of nonlinear transformations to automatic flight control. *Automatica*, *vol. 20*(no. 1), pp. 103–107.

Mooij, *E.* (2020). Robust control of a conventional aeroelastic launch vehicle. *AIAA Scitech 2020 Forum*, pp. 1103.

Morita, *Y.*, & *Goto*, *S.* (2008). Design for robustness using the μ-synthesis applied to launcher attitude and vibration control. *56th International Astronautical Congress of the International Astronautical Federation, the International Academy of Astronautics, and the International Institute of Space Law*, pp. C1–3.

Muse, *J.*, & *Calise*, *A.* (2010). $\mathcal{H}_\infty$ Norm minimization approach for adaptive control of the Crew Launch Vehicle. *AIAA Guidance, Navigation, and Control Conference*, pp. 7566.

Muse, *J.*, *Kim*, *K.*, *Qin*, *L.*, *Calise*, *A.*, & *Craig*, *J.* (2012). Adaptive attitude and vibration control of the NASA Ares Crew Launch Vehicle. *AIAA Guidance, Navigation and Control Conference and Exhibit*, pp. 7129.

NASA/MSFC. (2024). Artist's impression of Ares-I Crew LV. [Public Domain image. Image downloaded from: https://commons.wikimedia.org/wiki/File:Aiaa2.jpg. Accessed: 2025-01-07.].

Navarro-Tapia, *D.*, *Marcos*, *A.*, *Bennani*, *S.*, & *Roux*, *C.* (2017). Structured $\mathcal{H}_\infty$ and linear parameter varying control design for the VEGA launch vehicle. *Eucass-7th European Conference for Aeronautics and Space Sciences*.

Nikonaft. (2012). Artist's impression of the flight of a launch vehicle [Retrieved and modified for use as cover image under iStockphoto's standard license. URL: https://www.istockphoto.com/photo/rocket-gm158584803-22476768].

Noll, *B.*, *Zvara*, *J.*, & *Deyst*, *J. J.* (1970). *Space vehicle design criteria (guidance and control), effects of structural flexibility on launch vehicle control systems* (NASA Technical Report No. SP-8036). NASA.

Orr, *J.*, & *Zwieten*, *T. V.* (2012). Robust, practical adaptive control for launch vehicles. *AIAA Guidance, Navigation, and Control Conference*, pp. 4549.

Packard, *A.*, *Balas*, *G.*, *Liu*, *R.*, & *Shin*, *J.-Y.* (2000). Results on worst-case performance assessment. *Proceedings of the 2000 American Control Conference.*, *vol. 4*, pp. 2425–2427.

Papageorgiou, *G.* (1998). *Robust control system design: $\mathcal{H}_\infty$ Loop shaping and aerospace applications* [Dissertation for the degree of Doctor of Philosophy]. Darwin College, Cambridge.

Passino, *K. M.* (2001). Intelligent control: An overview of techniques. *Perspectives in control engineering: Technologies, applications, and new directions*, pp. 104–133.

Pignie, *G.* (2002). Ariane 5 and Ariane 5 evolution GN&C overview. *IAF abstracts, 34th COSPAR Scientific Assembly*, pp. 6.

Rugh, *W. J.* (1990). Analytical framework for gain scheduling. *1990 American Control Conference*, pp. 1688–1694.

Rynaski, *E.* (1966). Optimal control of a flexible launch vehicle. *Control and Flight Dynamics Conference*, pp. 592.

Safonov, *M.* (1977). *Robustness and stability aspects of stochastic multivariable feedback system design* [Ph.D. thesis]. MIT [Supervised by M. Athans].

Safonov, *M.* (1980). *Stability and robustness of multivariable feedback systems*. MIT Press.

Safonov, *M.*, & *Athans*, *M.* (1981). A multiloop generalization of the circle criterion for stability margin analysis. *IEEE Transactions on Automatic Control*, *vol. AC-26*(no. 2), pp. 415–422.

Sánchez de la Llana, *D.* (2015). *Design of $\mathcal{H}_\infty$ controllers for ill-conditioned plants in presence of disturbances* [Doctoral dissertation, Universidad Nacional de Educación a Distancia].

Sandell, *N. R. (* (1979). *Recent developments in the robustness theory of multivariable systems* (tech. rep.). MIT - Laboratory for Information and Decision Systems.

Schoon, *A.* (2024). *Review of $\mathcal{H}_\infty$ static output feedback controller synthesis methods: Application to fighter aircraft control* [Master's thesis, Delft University of Technology].

Sefton, *J. A.* (1991). *A geometrical approach to feedback stability* [Doctoral dissertation, University of Cambridge].

Seiler, *P.*, *Packard*, *A.*, & *Gahinet*, *P.* (2020). An introduction to disk margins [lecture notes]. *IEEE Control Systems Magazine*, *vol. 40*(no. 5), pp. 78–95.

Shtessel, *Y.*, & *Krupp*, *D.* (1997). Sliding mode control of reusable launch vehicle in launch and re-entry modes. *Proceedings The Twenty-Ninth Southeastern Symposium on System Theory*, pp. 533–539.

Shtessel, *Y.*, *Zhu*, *J.*, & *Daniels*, *D.* (2002). Reusable launch vehicle attitude control using time-varying sliding modes. *AIAA Guidance, Navigation, and Control Conference and Exhibit*, pp. 81 –85.

Simplício, *P.* (2019). *Guidance and control elements for improved access to space from planetary landers to reusable launchers* [Doctoral dissertation, University of Bristol].

Simplício, *P.*, *Acquatella*, *P.*, & *Bennani*, *S.* (2023). Launcher attitude control based on incremental nonlinear dynamic inversion: A feasibility study towards fast and robust design approaches. *ESA GNC and ICATT Conference*.

Simplício, *P.*, *Bennani*, *S.*, *Marcos*, *A.*, *Roux*, *C.*, & *Lefort*, *X.* (2016). Structured Singular Value analysis of the VEGA launcher in atmospheric flight. *Journal of Guidance, Control, and Dynamics*, *vol. 39*(no. 6), pp. 1342–1355.

Skogestad, *S.*, & *Postlethwaite*, *I.* (2005). *Multivariable feedback control: Analysis and design*. John Wiley & Sons.

Slotine, *J.*, & *Li*, *W.* (1991). *Applied nonlinear control*. Prentice Hall.

Smith, *P.* (1998). A simplified approach to nonlinear dynamic inversion based flight control. *23rd Atmospheric Flight Mechanics Conference*, pp. 4461.

Sontag, *E. D.*, & *Sussmann*, *H. J.* (1988). Further comments on the stabilizability of the angular velocity of a rigid body. *Systems & Control Letters*, *vol. 12*, pp. 437–442.

Stein, *G.* (2003). Respect the unstable. *IEEE Control Systems Magazine*, *23*(no. 4), pp. 12–25.

Stewart, *G. W.* (1993). On the early history of the singular value decomposition. *SIAM Review*, *vol. 35*(no. 4), pp. 551–566.

Stott, *J. E.*, & *Shtessel*, *Y. B.* (2012). Launch vehicle attitude control using sliding mode control and observation techniques. *Journal of the Franklin Institute*, *vol. 349*(no. 2), pp. 397–412.

Tapia, *D. N.* (2019). *Robust and adaptive TVC control design approaches for the VEGA launcher* [Doctoral dissertation, University of Bristol].

Toker, *O.*, & *Özbay*, *H.* (1995). On the np hardness of the purely complex computation, analysis/synthesis, and some related problems in multidimensional systems. *Proceedings of the 1995 American Control Conference*, *vol. 1*, pp. 447–451.

Tony, *L. A.*, & *C.*, *A. K. W.* (2015). μ-synthesis in state space; applied to a typical satellite launch vehicle. *2015 International Conference on Control Communication & Computing India (ICCC)*, pp. 113–118.

Trotta, *D.*, *Zavoli*, *A.*, *De Matteis*, *G.*, & *Neri*, *A.* (2020). Adaptive attitude control of launch vehicles in atmospheric flight. *Proceedings of the AAS/AIAA Astrodynamics Specialist Conference*, pp. 9–12.

Tsinias, *J.* (1989). Sufficient lyapunov-like conditions for stabilization. *Mathematics of Control, Signals, and Systems*, *vol. 2*, pp. 343–358.

Utkin, *V.*, *Poznyak*, *A.*, *Orlov*, *Y.*, & *Polyakov*, *A.* (2020). Conventional and high order sliding mode control. *Journal of the Franklin Institute*, *vol. 357*(no. 15), pp. 10244–10261.

Vidyasagar, *M.*, & *Kimura*, *H.* (1986). Robust controllers for uncertain linear multivariable systems. *Automatica*, *vol. 22*(no. 1), pp. 85–94.

Vinod, *V.*, *Geetha*, *S.*, & *Dasgupta*, *S.* (2012). Controller design for ascent phase of reusable launch vehicle using backstepping. *Journal of The Institution of Engineers (India): Series C*, *vol. 93*, pp. 41–45.

Wall, *J. H.*, *Orr*, *J. S.*, & *VanZwieten*, *T. S.* (2014). Space launch system implementation of adaptive augmenting control. *Annual American Astronautical Society (AAS) Guidance, Navigation, and Control Conference*, (no. AAS 14-051).

Wang, *X.*, *Zhang*, *F.*, *Hu*, *D.*, *Chen*, *R.*, & *Song*, *Z.* (2023). Review, prospect and technical challenge of launch vehicle. In *Z. Song*, *D. Zhao*, & *S. Theil* (Eds.), *Autonomous trajectory planning and guidance control for launch vehicles* (pp. 1–31).

Wie, *B.*, *Du*, *W.*, & *Whorton*, *M.* (2008). Analysis and design of launch vehicle flight control systems. *AIAA Guidance, Navigation and Control Conference and Exhibit*, pp. 6291.

Wong, *P.*, & *Athans*, *M.* (1975). *On the interaction structure of linear multi-input feedback control systems* [Master's Thesis]. MIT [Supervised by M. Athans].

Xing, *G.*, & *Bainum*, *P.* (1996). $\mathcal{H}_\infty$ Robust controller design and comparison with LQG, PID controllers for an expendable launch vehicle. *Astrodynamics Conference*, pp. 3591.

Zames, *G.* (1966). On the input–output stability of time-varying nonlinear feedback systems—part i: Conditions derived using concepts of loop gain, conicity, and positivity. *IEEE Transactions on Automatic Control*, *vol. AC-11*(no. 2), pp. 228-238 and 465–476.

Zames, *G.* (1981). Feedback and optimal sensitivity: Model reference transformations, multiplicative seminorms, and approximate inverses. *IEEE Transactions on Automatic Control*, *vol. AC-26*(no. 2), pp. 301–320.

Zhou, *K.*, & *Doyle*, *J.* (1998). *Essentials of robust control* (Vol. 104). Prentice Hall.

Zhou, *S.*, *Bai*, *X.*, *Zhang*, *S.*, & *Yang*, *C.* (2024). Research on attitude control of launch vehicle at ascent stage through reinforcement learning. *3rd International Conference on Innovations and Development of Information Technologies and Robotics (IDITR)*, pp. 47–52.

Zipfel, *P.* (2007). *Modeling and simulation of aerospace vehicle dynamics*. American Institute of Aeronautics; Astronautics.

A

Rigid Body Analytical Transfer Functions

In order to perform an initial analysis on the LV model, the SISO open loop transfer functions for the rigid body analysis model (AMR) were analytically derived. Equations A.1, A.2, A.3, A.4, and A.5 contain, respectively, the transfer functions from the different inputs (TVC deflection, acceleration of the TVC deflection, and wind velocity) to the pitch angle, pitch rate, drift, drift rate, and angle of attack. It should be noted that in these equations, the subscript IMU is used for certain outputs to indicate that these outputs are evaluated at the position of the IMU.

$$\begin{aligned}
\frac{\theta_{IMU}}{\delta_{pitch}} &= \frac{s\,(\mu_c) - c5\,\mu_c + \frac{\mu_\alpha}{V_{rel}}\,c7}{Den(s)} \\
\frac{\theta_{IMU}}{\ddot{\delta}_{pitch}} &= \frac{\mu_{c'}\,s - c5\,\mu_{c'} + \frac{\mu_\alpha}{V_{rel}}\,c8}{Den(s)} \\
\frac{\theta_{IMU}}{\alpha_w} &= \frac{-\frac{\mu_\alpha}{V_{rel}}\,s}{Den(s)}
\end{aligned} \tag{A.1}$$

$$\begin{aligned}
\frac{\dot{\theta}_{IMU}}{\delta_{pitch}} &= \frac{s^2\,(\mu_c) - s\,(c5\,\mu_c - \frac{\mu_\alpha}{V_{rel}}\,c7)}{Den(s)} \\
\frac{\dot{\theta}_{IMU}}{\ddot{\delta}_{pitch}} &= \frac{\mu_{c'}\,s^2 - (c5\,\mu_{c'} - \frac{\mu_\alpha}{V_{rel}}\,c8)\,s}{Den(s)} \\
\frac{\dot{\theta}_{IMU}}{\alpha_w} &= \frac{-\frac{\mu_\alpha}{V_{rel}}\,s^2}{Den(s)}
\end{aligned} \tag{A.2}$$

$$\begin{aligned}
\frac{z_{IMU}}{\delta_{pitch}} &= \frac{s^2\,(-x_{IMU}\,\mu_c + c7) - s\,(c1\,c7 - c4\,\mu_c + \frac{\mu_\alpha}{V_{rel}}\,c7\,x_{IMU} - c5\,\mu_c\,x_{IMU}) + \mu_c\,c3 - c7\,\mu_\alpha}{Den(s)\,s} \\
\frac{z_{IMU}}{\ddot{\delta}_{pitch}} &= \frac{s^2\,(-\mu_{c'}\,x_{IMU} + c8) - s\,(c1\,c8 - c4\,\mu_{c'} + \frac{\mu_\alpha}{V_{rel}}\,c8\,x_{IMU} - c5\,\mu_{c'}\,x_{IMU}) + c3\,\mu_{c'} - c8\,\mu_\alpha}{Den(s)\,s} \\
\frac{z_{IMU}}{\alpha_w} &= \frac{s^2\,(\frac{\mu_\alpha}{V_{rel}}\,x_{IMU} - c5) - \frac{\mu_\alpha}{V_{rel}}\,c3 + c5\,\mu_\alpha}{Den(s)\,s}
\end{aligned} \tag{A.3}$$

$$\begin{aligned}
\frac{\dot{z}_{IMU}}{\delta_{pitch}} &= \frac{s^2\left(-x_{IMU}\,\mu_c + c7\right) - s\left(c1\,c7 - c4\,\mu_c + \frac{\mu_\alpha}{V_{rel}}\,c7\,x_{IMU} - c5\,\mu_c\,x_{IMU}\right) + \mu_c\,c3 - c7\,\mu_\alpha}{Den(s)} \\
\frac{\dot{z}_{IMU}}{\ddot{\delta}_{pitch}} &= \frac{s^2\left(-\mu_{c'}\,x_{IMU} + c8\right) - s\left(c1\,c8 - c4\,\mu_{c'} + \frac{\mu_\alpha}{V_{rel}}\,c8\,x_{IMU} - c5\,\mu_{c'}\,x_{IMU}\right) + c3\,\mu_{c'} - c8\,\mu_\alpha}{Den(s)} \\
\frac{\dot{z}_{IMU}}{\alpha_w} &= \frac{s^2\left(\frac{\mu_\alpha}{V_{rel}}\,x_{IMU} - c5\right) - \frac{\mu_\alpha}{V_{rel}}\,c3 + c5\,\mu_\alpha}{Den(s)}
\end{aligned} \tag{A.4}$$

$$\begin{aligned}
\frac{\alpha}{\delta_{pitch}} &= \frac{c7\,s^2 - s\left(c1\,c7 - V_{rel}\,\mu_c - c4\,\mu_c\right) + c3\,\mu_c - c7\,\mu_\alpha + V_{rel}\,\frac{\mu_\alpha}{V_{rel}}\,c7 - V_{rel}\,c5\,\mu_c}{Den(s)\,V_{rel}} \\
\frac{\alpha}{\ddot{\delta}_{pitch}} &= \frac{c8\,s^2 - s\left(c1\,c8 - V_{rel}\,\mu_{c'} - c4\,\mu_{c'}\right) + c3\,\mu_{c'} - c8\,\mu_\alpha + V_{rel}\,\frac{\mu_\alpha}{V_{rel}}\,c8 - V_{rel}\,c5\,\mu_{c'}}{Den(s)\,V_{rel}} \\
\frac{\alpha}{\alpha_w} &= \frac{-s^3 + c1\,s^2}{Den(s)\,V_{rel}}
\end{aligned} \tag{A.5}$$

where:

$$\begin{aligned}
Den(s) &= s^3 - (c1 + c5)s^2 - (\mu_\alpha)\,s - \frac{\mu_\alpha}{V_{rel}}\,c3 + c5\,\mu_\alpha \\
c1(s) &= -\frac{(x_{CP} - x_{CM})\,\mu_\alpha}{V_{rel}} \\
c3(s) &= -\frac{N_z}{m_0} + a_0 \\
c4(s) &= \frac{I_{yy}\mu_\alpha}{m_0 V_{rel}} \\
c5(s) &= \frac{-N_z}{m_0 V_{rel}} \\
c7(s) &= -\frac{T}{m_0} \\
c8(s) &= -\frac{m_N l_N}{m_0}
\end{aligned} \tag{A.6}$$

B

$\mathcal{H}_\infty$ CLS Controller Comparison - Nominal Design Vs LFT Design

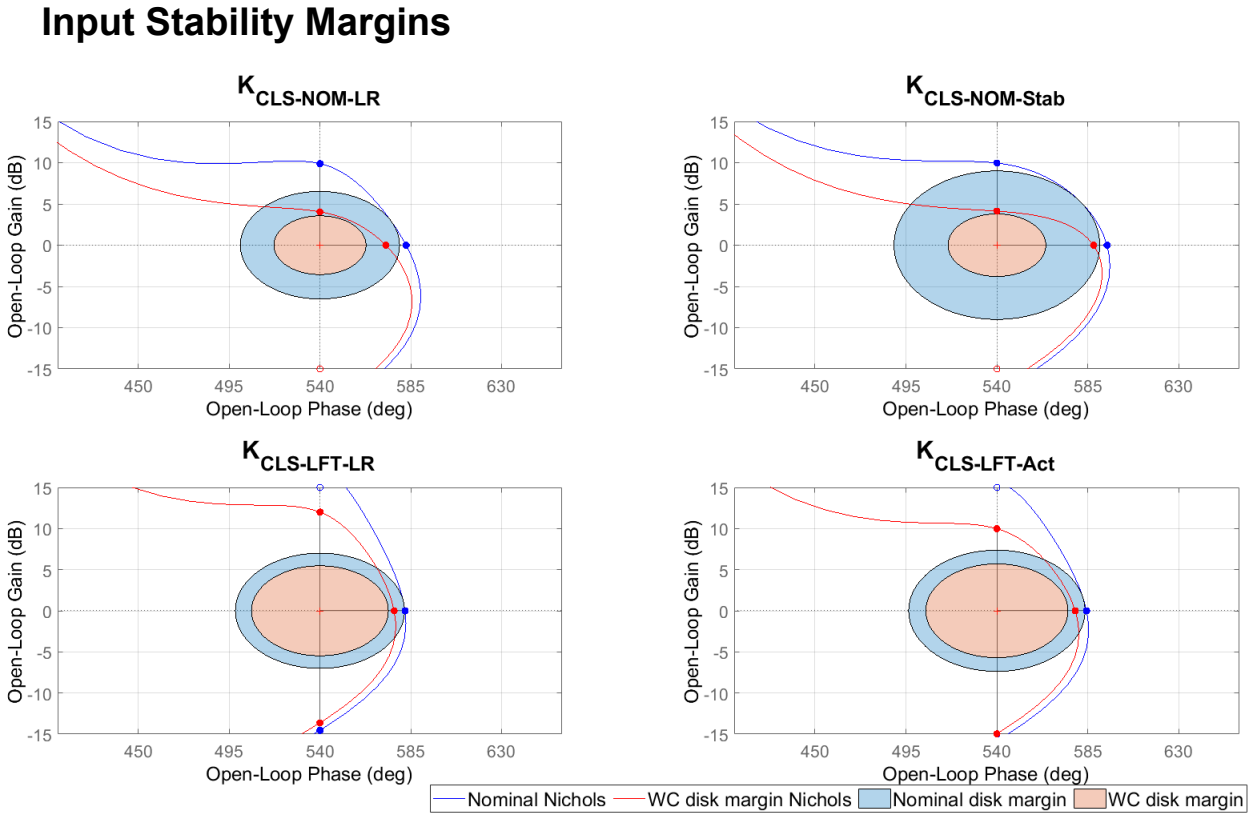

Figure B.1: Input Nichols - Nominal Design vs LFT Design $\mathcal{H}_\infty$ CLS controllers.

Output Stability Margins

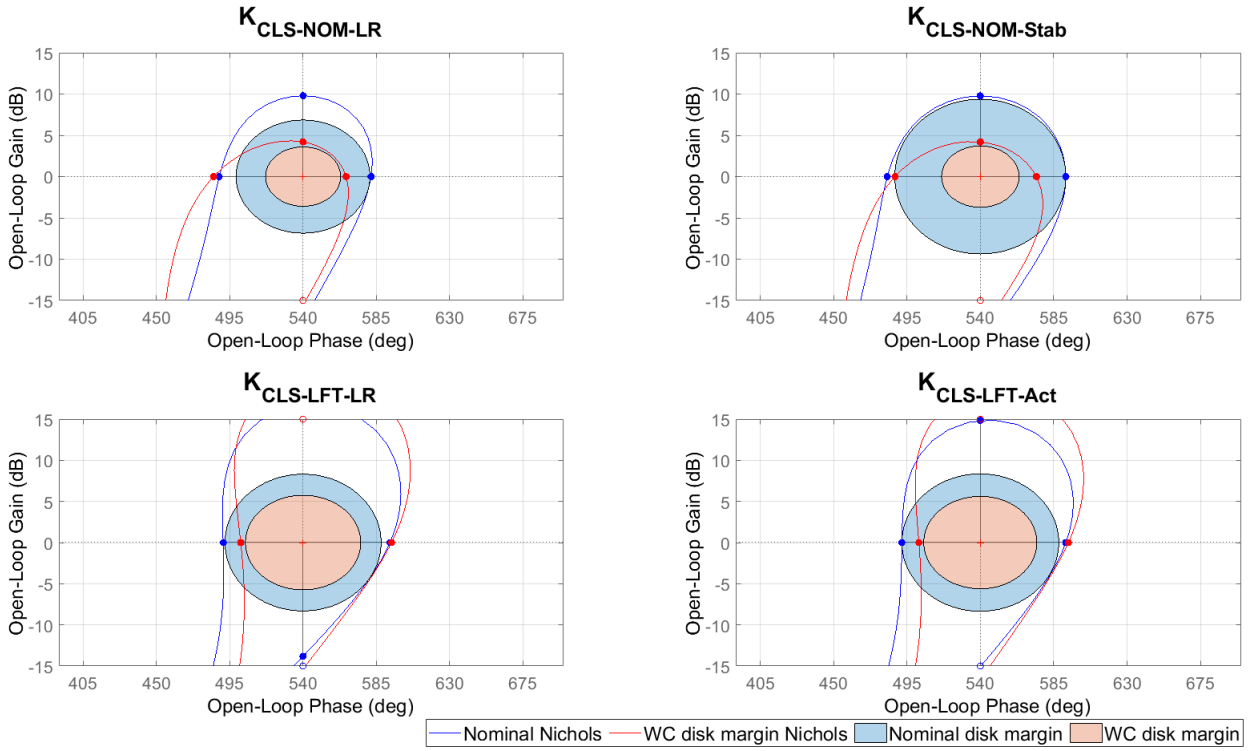

Figure B.2: Loop-at-a-time (LAT) pitch angle Nichols - Nominal Design vs LFT Design $\mathcal{H}_\infty$ CLS controllers.

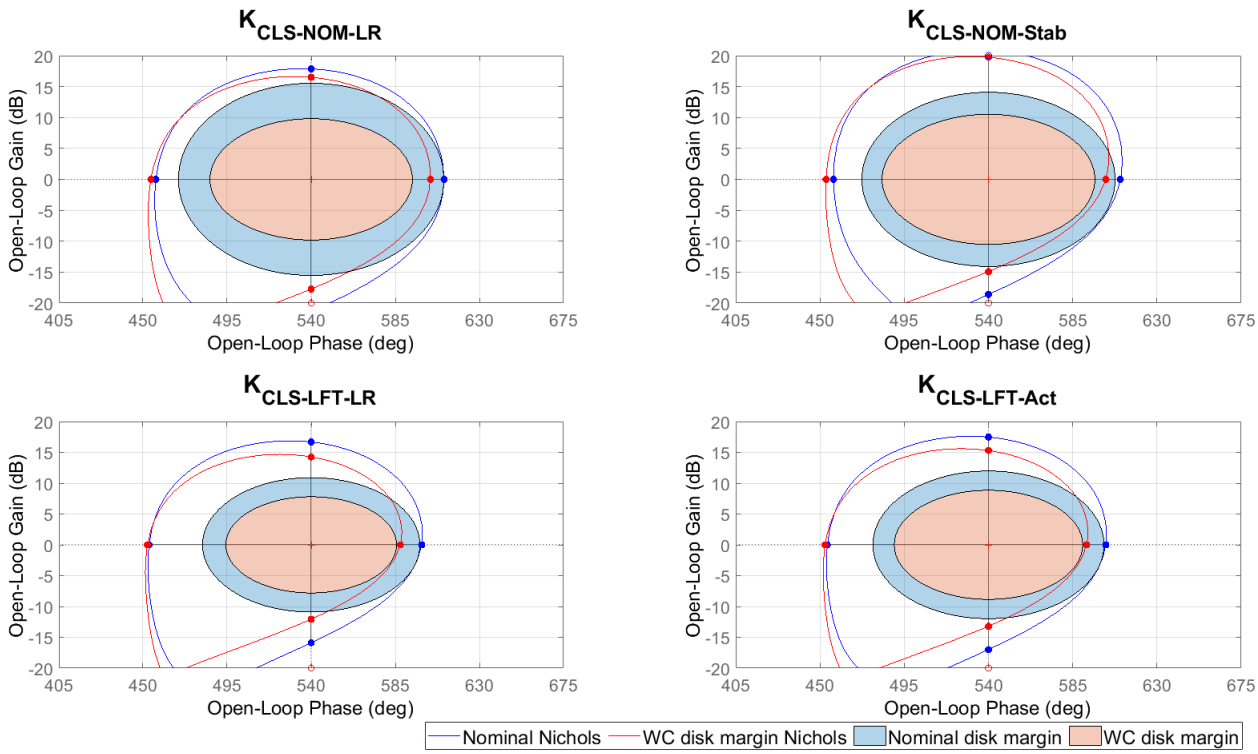

Figure B.3: LAT pitch rate Nichols - Nominal Design vs LFT Design $\mathcal{H}_\infty$ CLS controllers.

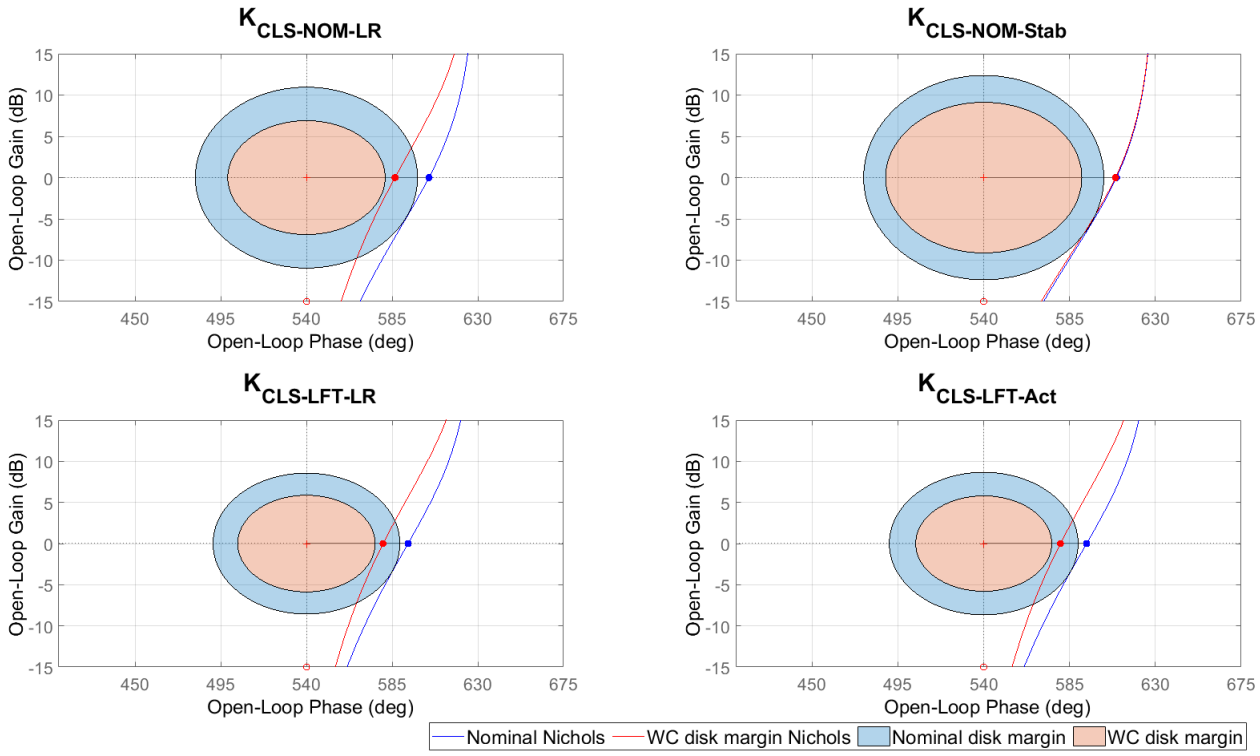

Figure B.4: LAT drift Nichols - Nominal Design vs LFT Design $\mathcal{H}_\infty$ CLS controllers.

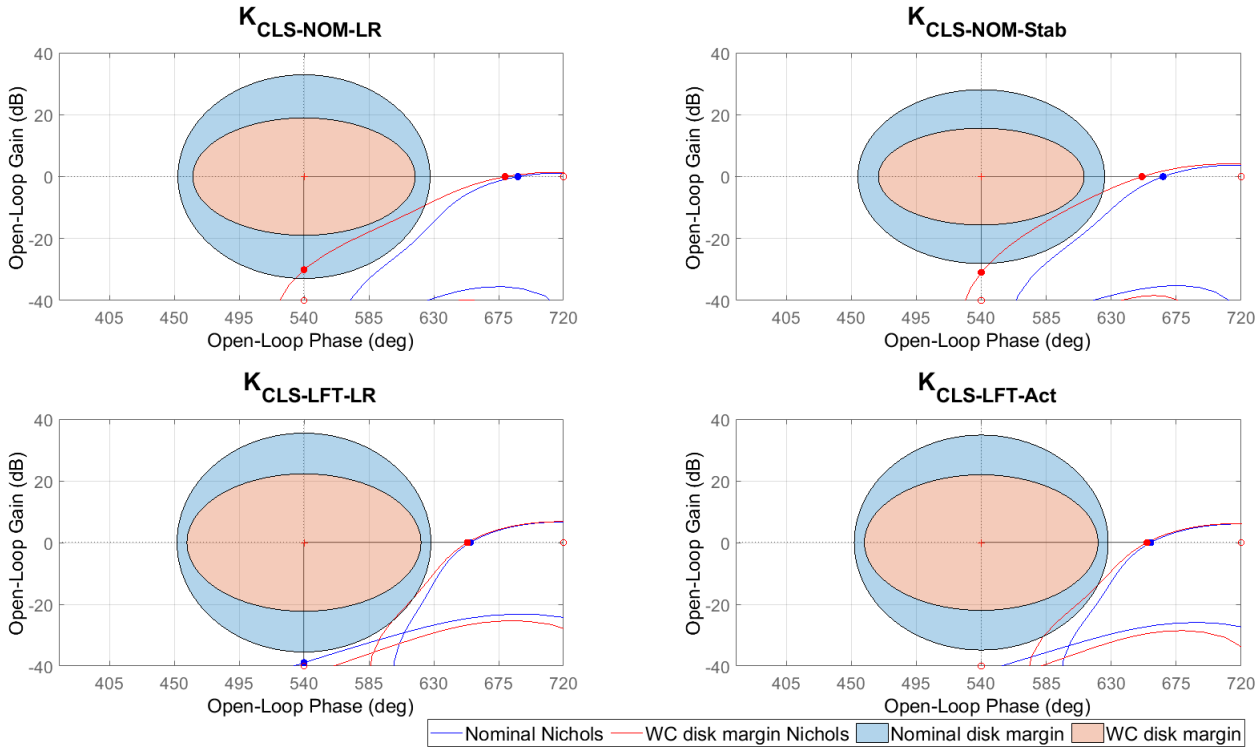

Figure B.5: LAT drift rate Nichols - Nominal Design vs LFT Design $\mathcal{H}_\infty$ CLS controllers.

Stability Margin Summary

Table B.1: Stability Margins - Nominal Design $\mathcal{H}_\infty$ CLS.

		$K_{CLS-NOM-LR}$			$K_{CLS-NOM-Stab}$		
		GM	PM	Freq	GM	PM	Freq
Nominal	Input	6.52	39.47	2.03	9.00	50.92	1.71
Worst-Case	Input	3.56	22.85	0.88	3.79	24.21	0.51
Nominal	LAT drift	10.95	58.35	0.07	12.38	62.95	0.07
	LAT drift rate	32.89	87.40	0.70	28.05	85.47	0.55
	LAT pitch angle	6.85	41.13	1.93	9.37	52.44	1.34
	LAT pitch rate	15.54	71.03	3.82	14.08	67.64	8.94
Worst-Case	LAT drift	6.89	41.34	0.05	9.12	51.44	0.07
	LAT drift rate	18.95	77.12	0.73	15.64	71.23	0.56
	LAT pitch angle	3.59	23.05	0.85	3.71	23.76	0.49
	LAT pitch rate	9.81	54.16	4.83	10.52	56.81	8.85
Nominal	MIMO Output	3.75	23.99	0.07	4.03	25.71	0.08
Worst-Case	MIMO Output	2.33	15.19	0.92	2.24	14.65	0.48
Nominal	I/O	2.46	15.99	2.09	3.20	20.64	1.60
Worst-Case	I/O	1.42	9.30	1.00	1.46	9.57	0.50

Table B.2: Stability Margins - LFT Design $\mathcal{H}_\infty$ CLS.

		$K_{CLS-LFT-LR}$			$K_{CLS-LFT-Act}$		
		GM	PM	Freq	GM	PM	Freq
Nominal	Input	7.01	41.93	5.11	7.37	43.66	4.14
Worst-Case	Input D	5.47	33.91	4.18	5.70	35.18	3.21
Nominal	LAT drift	8.56	49.06	0.10	8.66	49.49	0.09
	LAT drift rate	35.46	88.07	16.53	34.83	87.92	0.72
	LAT pitch angle	8.31	47.96	4.38	8.34	48.12	0.10
	LAT pitch rate	10.90	58.17	8.72	11.98	61.73	8.03
Worst-Case	LAT drift	5.82	35.80	0.07	5.80	35.68	0.07
	LAT drift rate	22.26	81.18	0.94	21.97	80.88	0.90
	LAT pitch angle	5.74	35.38	0.08	5.62	34.71	0.07
	LAT pitch rate	7.78	45.56	9.22	8.85	50.31	8.53
Nominal	MIMO Output	3.21	20.69	0.10	3.20	20.64	0.09
Worst-Case	MIMO Output	2.22	14.47	0.08	2.15	14.01	0.07
Nominal	I/O	2.66	17.30	4.80	2.75	17.88	3.77
Worst-Case	I/O	2.10	13.68	3.97	2.11	13.76	3.17

Time Domain Analysis

For the following analysis, the color coding for the controllers follows the scheme defined in Figure B.6. Additionally, the shaded regions in the plots represent the results of simulations under dispersed conditions.

Figure B.6: Color code for time domain analysis - nominal $\mathcal{H}_\infty$ CLS vs LFT $\mathcal{H}_\infty$ CLS.

Low Frequency, High Amplitude (LFHA) Wind Gust

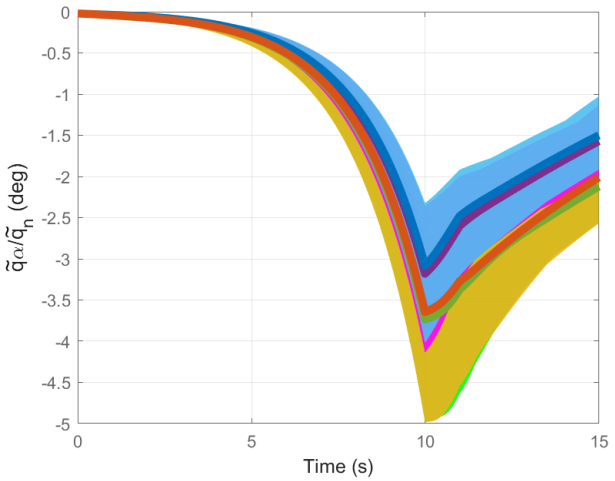

Figure B.7: LFHA Wind - Load performance metric (LPM), normalized by the nominal dynamic pressure.

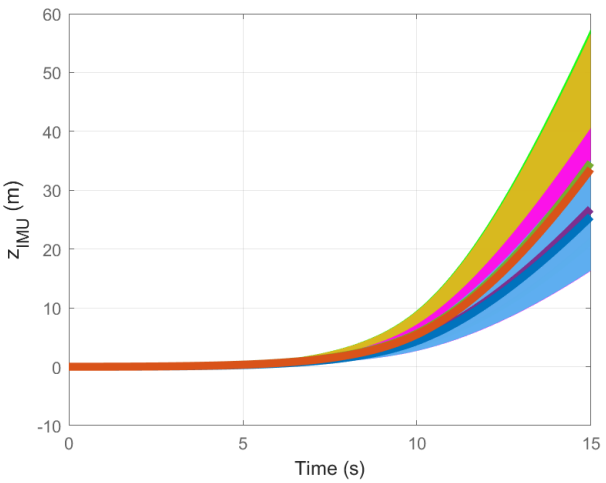

Figure B.8: LFHA Wind - Drift.

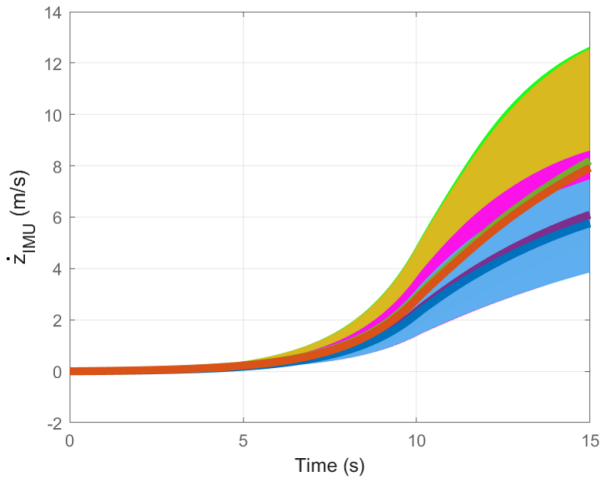

Figure B.9: LFHA Wind - Drift rate.

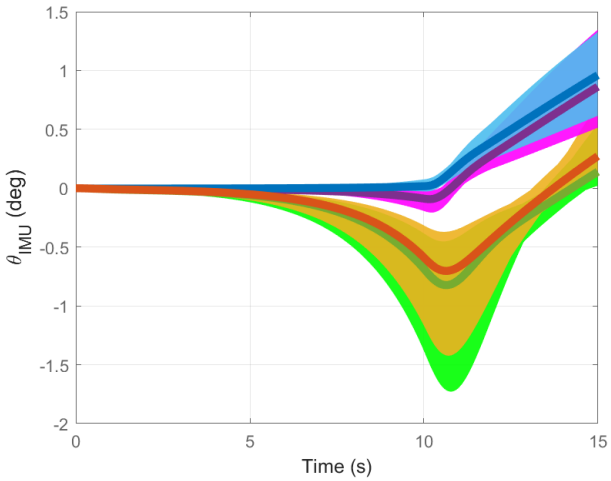

Figure B.10: LFHA Wind - Pitch angle.

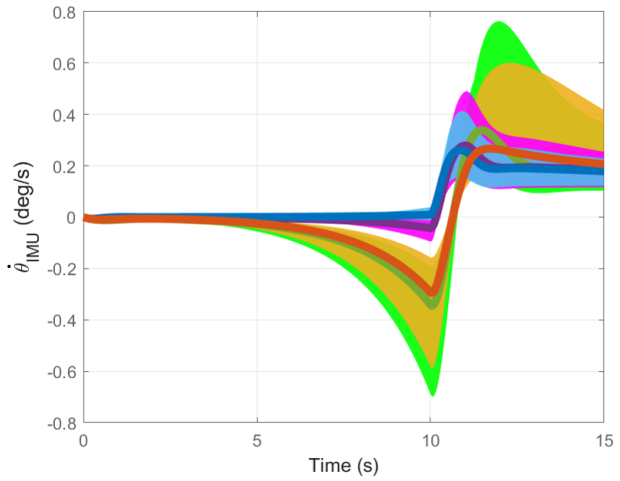

Figure B.11: LFHA Wind - Pitch rate.

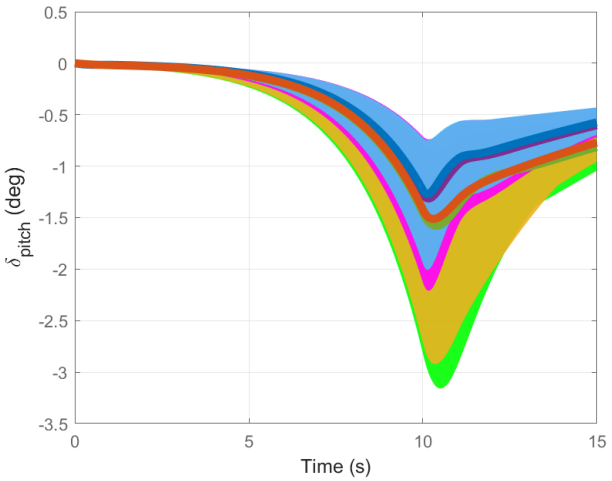

Figure B.12: LFHA Wind - TVC deflection.

Tracking - 0.5 deg step on pitch angle reference

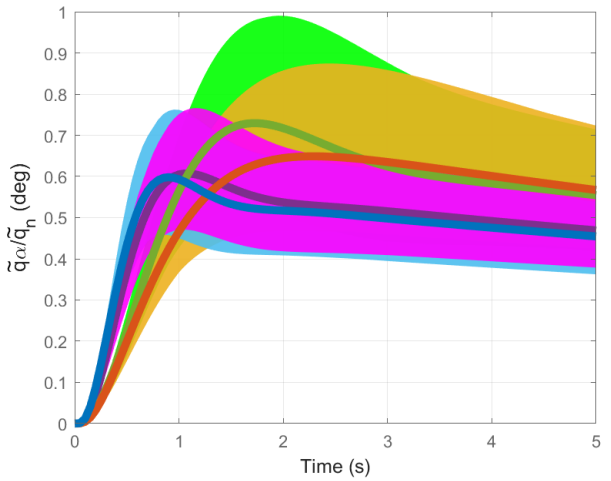

Figure B.13: Pitch Angle Tracking - LPM, normalized by the nominal dynamic pressure.

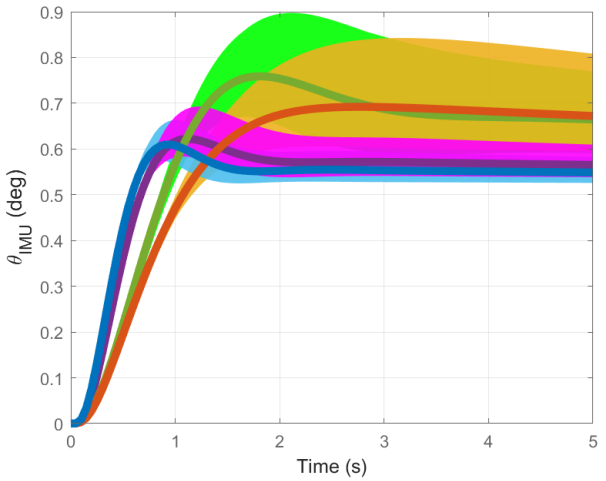

Figure B.14: Pitch Angle Tracking - Pitch angle.

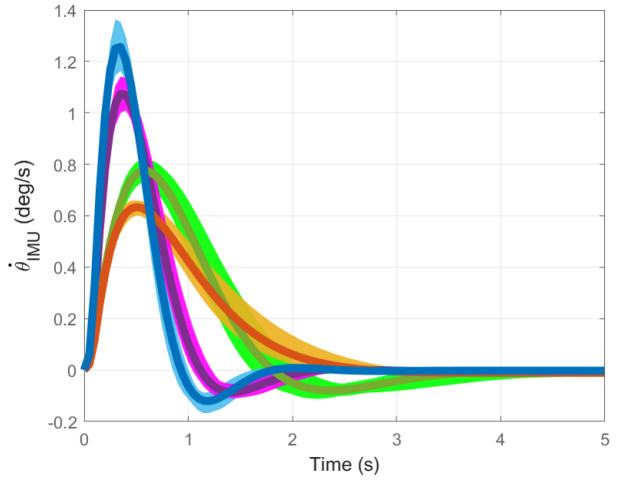

Figure B.15: Pitch Angle Tracking - Pitch rate.

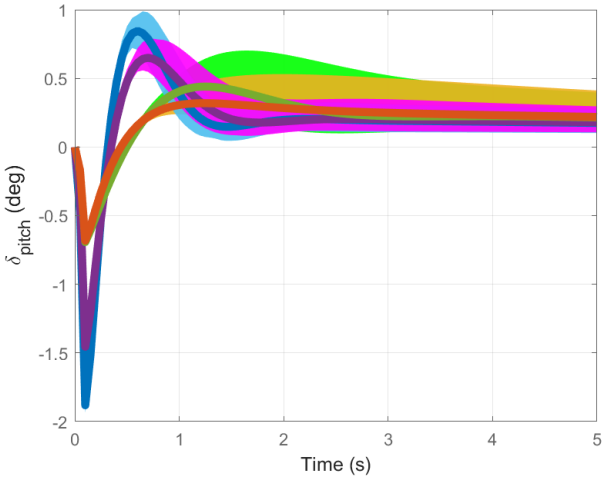

Figure B.16: Pitch Angle Tracking - TVC deflection.

High Frequency, Low to Mid Amplitude (HFLMA) Wind Gust

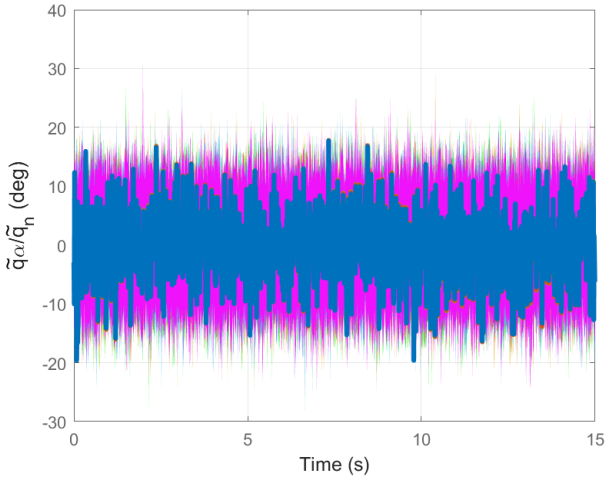

Figure B.17: HFLMA Wind - LPM, normalized by the nominal dynamic pressure.

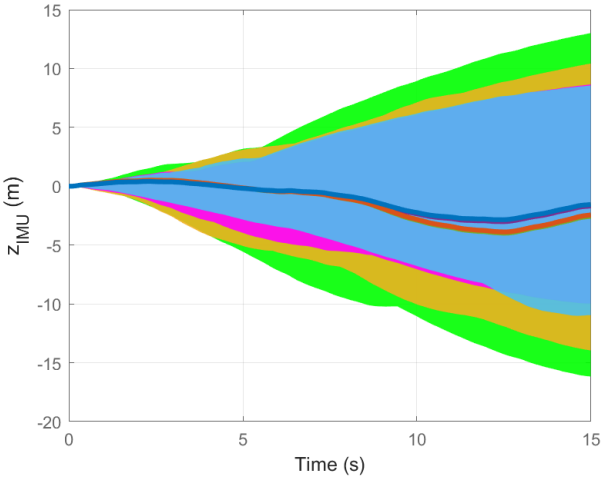

Figure B.18: HFLMA Wind - Drift.

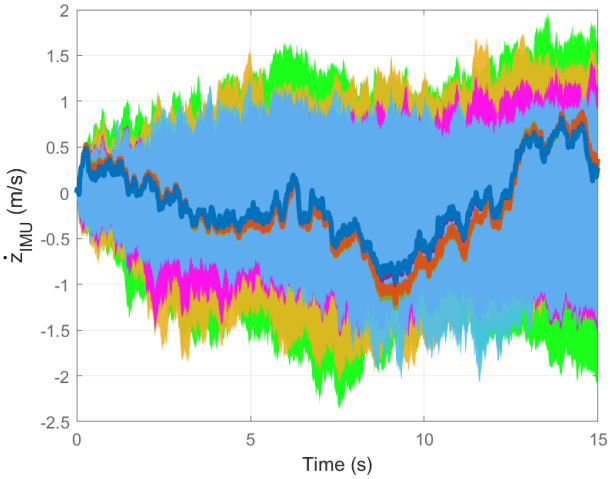

Figure B.19: HFLMA Wind - Drift rate.

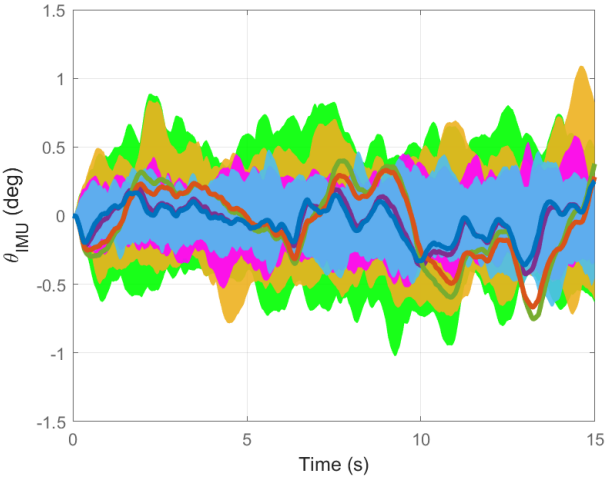

Figure B.20: HFLMA Wind - Pitch angle.

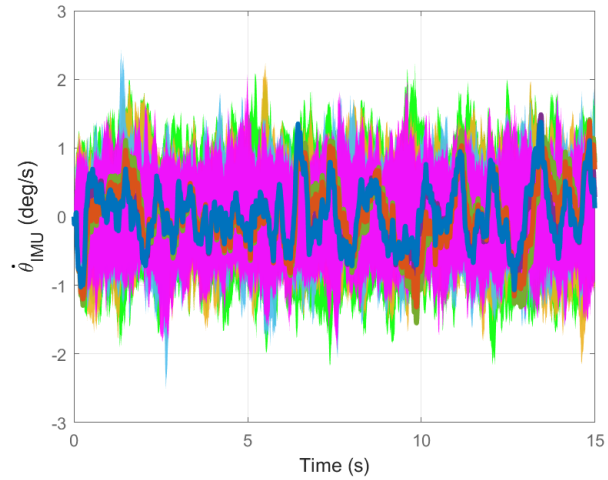

Figure B.21: HFLMA Wind - Pitch rate.

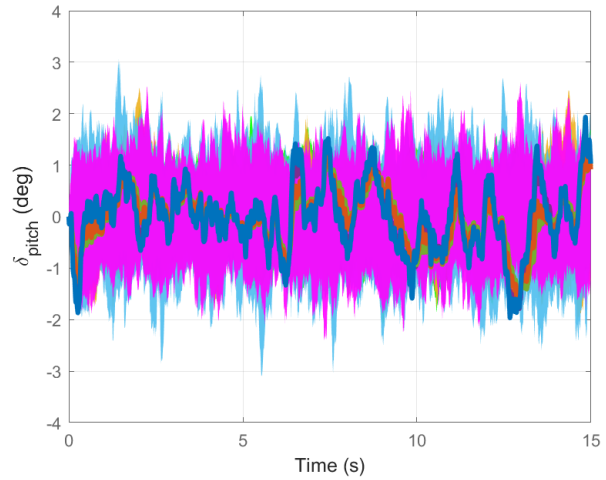

Figure B.22: HFLMA Wind - TVC deflection.

C

Nominal Design $\mathcal{H}_\infty$ CLS vs Nominal Design $\mathcal{H}_\infty$ OLS

Stability Margin Summary

Table C.1: Stability margins - Nominal Design $\mathcal{H}_\infty$ CLS ($K_{CLS-NOM-LR}$) vs Nominal Design $\mathcal{H}_\infty$ OLS ($K_{OLS-NOM}$).

		$K_{CLS-NOM-LR}$			$K_{OLS-NOM}$		
		GM	PM	Freq	GM	PM	Freq
Nominal	Input	6.52	39.47	2.03	5.89	36.19	14.73
Worst-Case	Input	3.56	22.83	0.88	3.03	19.60	17.81
Nominal	LAT drift	10.95	58.35	0.07	12.02	61.86	0.82
	LAT drift rate	32.89	87.40	0.70	8.89	50.48	13.47
	LAT pitch angle	6.85	41.13	1.93	10.39	56.36	3.95
	LAT pitch rate	15.54	71.03	3.82	8.72	49.74	23.70
Worst-Case	LAT drift	6.90	41.36	0.05	9.60	53.35	0.63
	LAT drift rate	19.33	77.67	0.73	5.34	33.20	17.11
	LAT pitch angle	3.54	22.76	0.85	7.28	43.24	3.49
	LAT pitch rate	10.42	56.47	4.83	4.41	27.94	19.06
Nominal	MIMO Output	3.75	23.99	0.07	3.36	21.63	4.15
Worst-Case	MIMO Output	2.33	15.22	0.93	2.41	15.71	3.70
Nominal	I/O	2.46	15.99	2.09	2.43	15.85	10.69
Worst-Case	I/O	1.42	9.32	1.00	1.31	8.62	17.55

Time Domain Analysis

For the following analysis, the color coding for the controllers follows the scheme defined in Figure C.1. Additionally, the shaded regions in the plots represent the results of simulations under dispersed conditions.

Figure C.1: Color code for time domain analysis - $K_{CLS-NOM-LR}$ vs $K_{OLS-NOM}$.

Low Frequency, High Amplitude (LFHA) Wind Gust

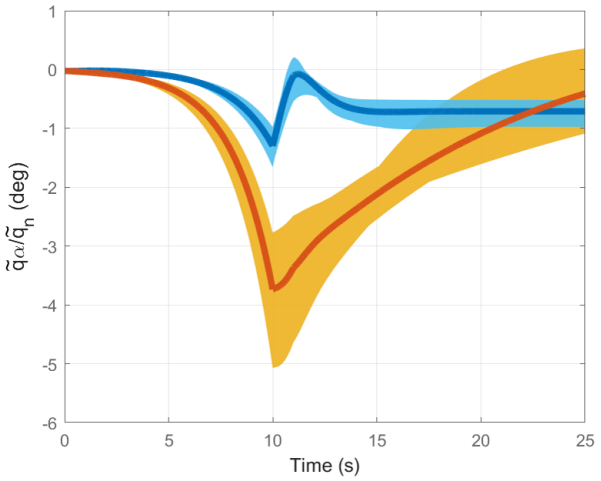

Figure C.2: LFHA Wind - Load performance metric (LPM), normalized by the nominal dynamic pressure.

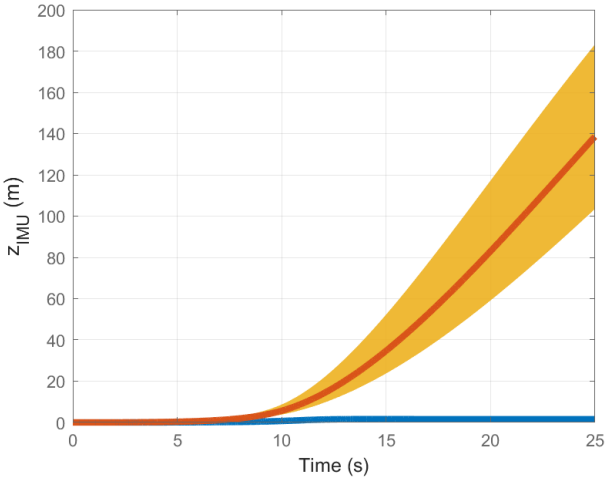

Figure C.3: LFHA Wind - Drift.

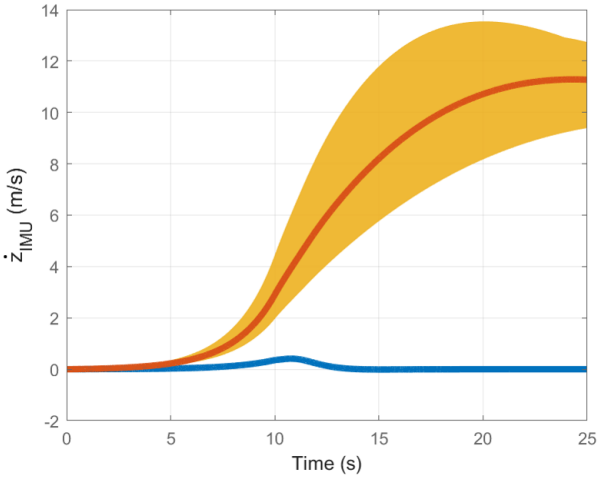

Figure C.4: LFHA Wind - Drift rate.

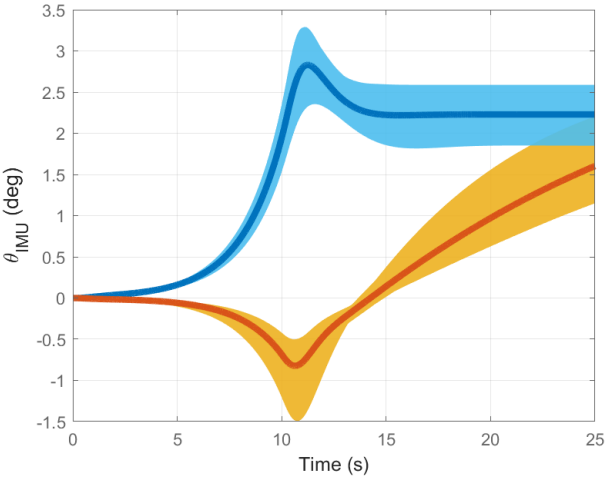

Figure C.5: LFHA Wind - Pitch angle.

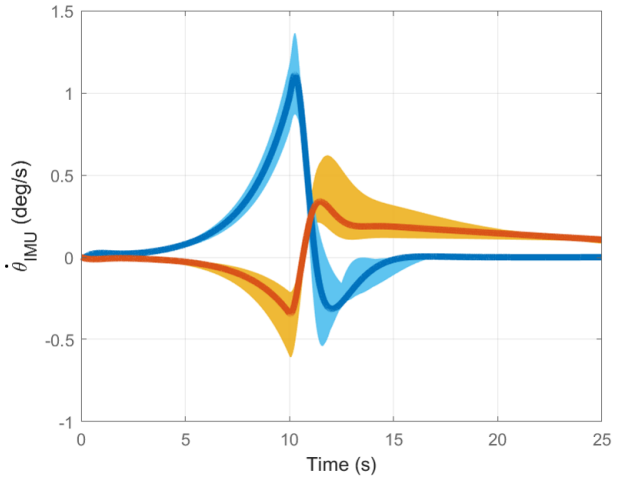

Figure C.6: LFHA Wind - Pitch rate.

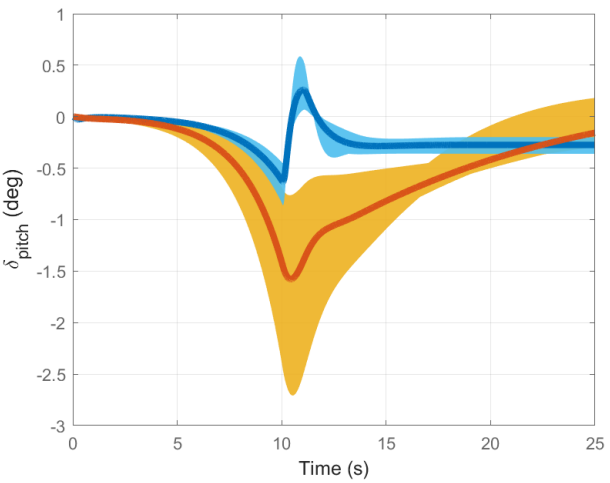

Figure C.7: LFHA Wind - TVC deflection.

Tracking - 0.5 deg step on pitch angle reference

As explained in subsection 6.4.3, the tracking test requires the shutting off of the drift and drift rate feedbacks. However, for the nominal $\mathcal{H}_\infty$ OLS controller, the drift rate feedback cannot be disconnected. This feedback plays a critical role in providing damping, which is essential due to the higher bandwidth and higher loop gain achieved by the controller. If the drift rate feedback is disconnected, the system becomes unstable. Therefore, the question arises: how can tracking be tested on the nominal $\mathcal{H}_\infty$ OLS controller?

The solution found was to apply a high-pass filter to the drift rate output. This filter reduces the influence of drift rate feedback for very low-frequency commands, preventing it from dominating the system's response. At the same time, it allows the drift rate feedback to still provide damping at higher frequencies, maintaining system stability. However, its important to keep in mind that the the high-pass filter will affect the results. On one hand, the tracking response will be slower than in reality, as the high-pass filter only attenuates some of the feedback's influence at low frequencies. On the other hand, the system's response will exhibit more overshoot and oscillations, as the high-pass filter reduces the damping contribution of the drift rate feedback in the mid frequencies.

While this approach allows for evaluating the tracking capabilities of the nominal $\mathcal{H}_\infty$ OLS controller, it is important to note that the results reflect a conservative scenario. The controller's actual performance should be no worse than the results presented here. With this in mind, Figures C.8 through C.11, illustrate the results of the tracking tests.

While the $\mathcal{H}_\infty$ OLS controller exhibits a higher actuation peak value and rate (Fig. C.11), it achieves a smaller load performance metric (LPM) peak and reduced pitch overshoot (Figs. C.8 and C.9, respectively). Additionally, the pitch tracking response (Fig. C.9) presents smaller rising time and settling time. The high actuation rate and the oscillations observed in the $\mathcal{H}_\infty$ OLS response can be attributed to the limitations of the testing methodology. Therefore, despite these limitations, it can be concluded that the $\mathcal{H}_\infty$ OLS controller achieves results comparable to those of the nominal $\mathcal{H}_\infty$ CLS controller.

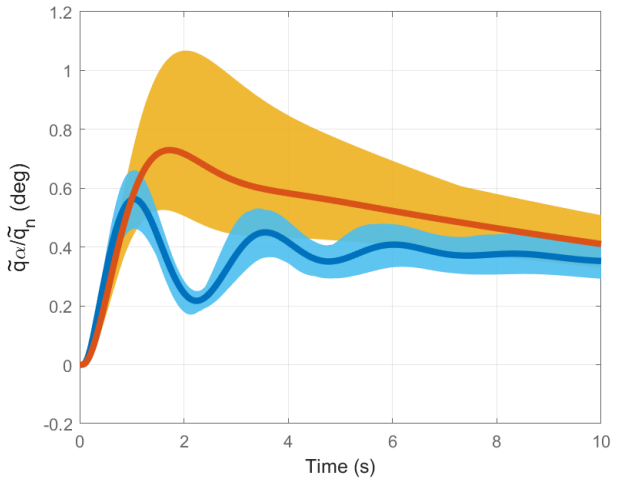

Figure C.8: Pitch Angle Tracking - LPM, normalized by the nominal dynamic pressure.

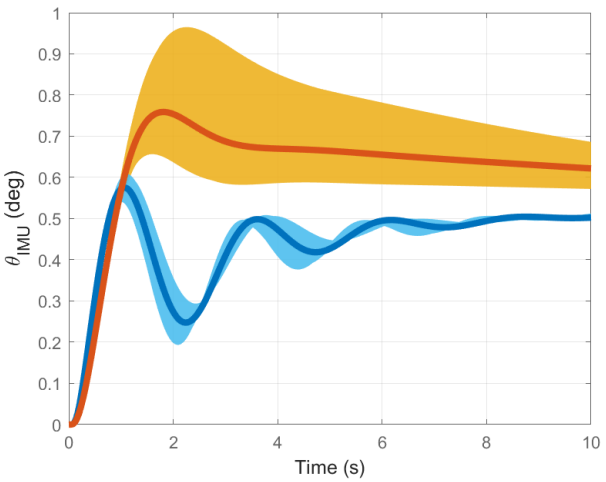

Figure C.9: Pitch Angle Tracking - Pitch angle.

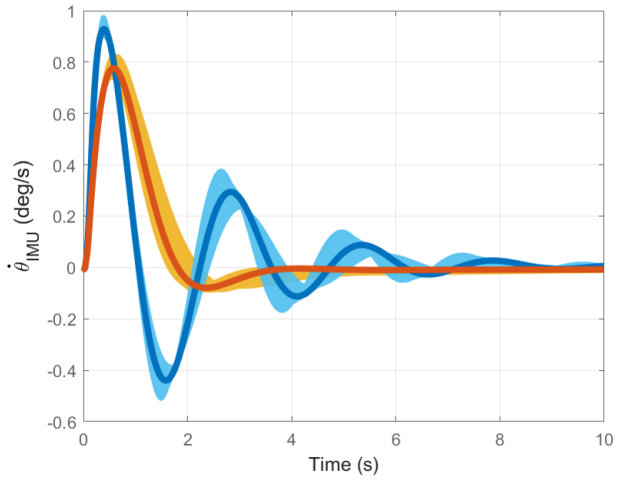

Figure C.10: Pitch Angle Tracking - Pitch rate.

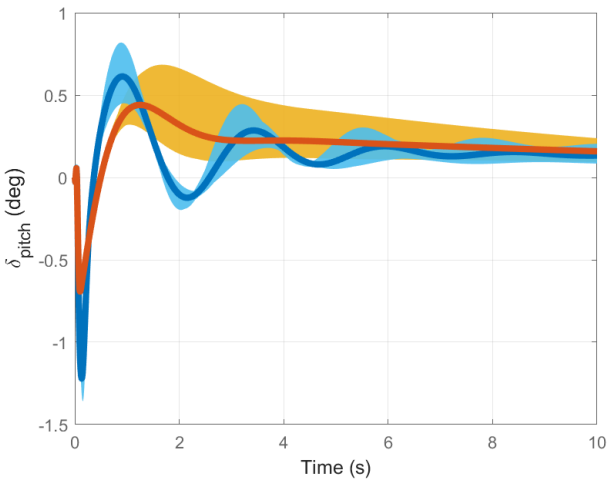

Figure C.11: Pitch Angle Tracking - TVC deflection.

High Frequency, Low to Mid Amplitude (HFLMA) Wind Gust

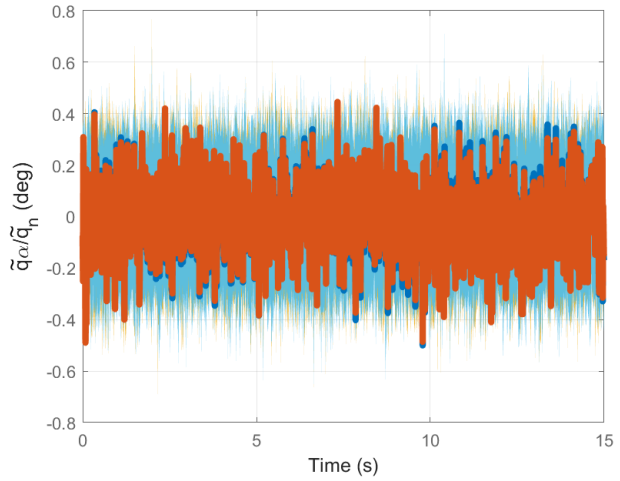

Figure C.12: HFLMA Wind - LPM, normalized by the nominal dynamic pressure.

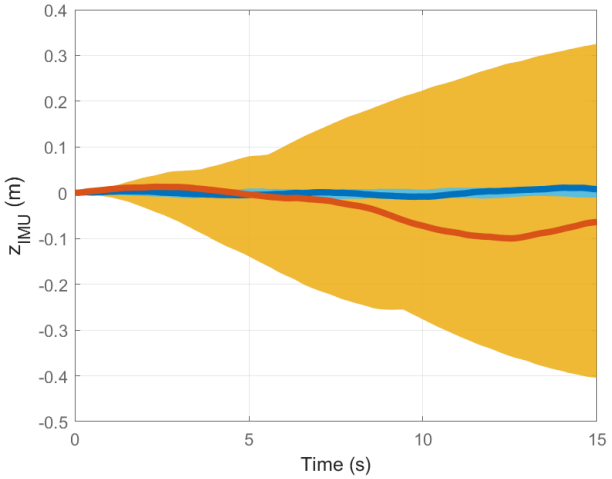

Figure C.13: HFLMA Wind - Drift.

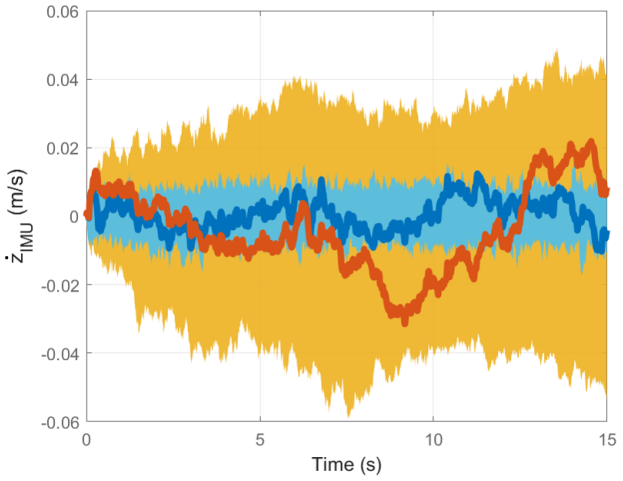

Figure C.14: HFLMA Wind - Drift rate.

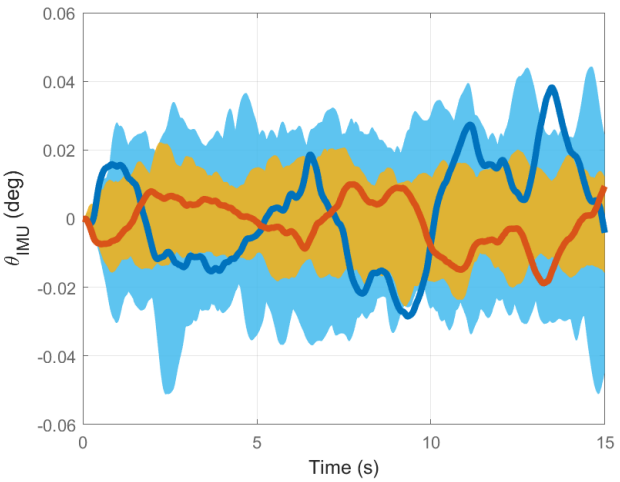

Figure C.15: HFLMA Wind - Pitch angle.

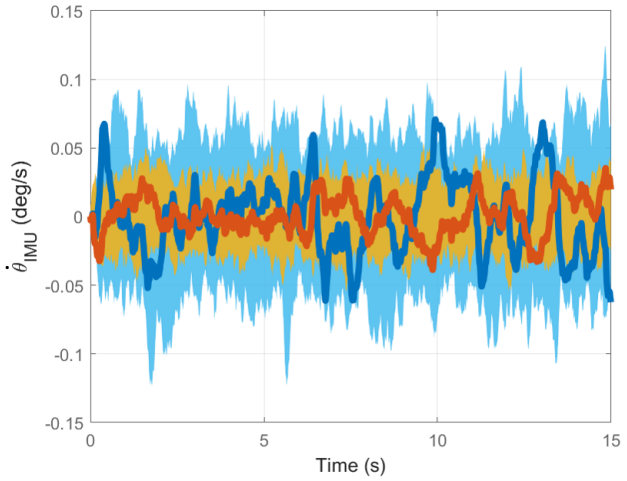

Figure C.16: HFLMA Wind - Pitch rate.

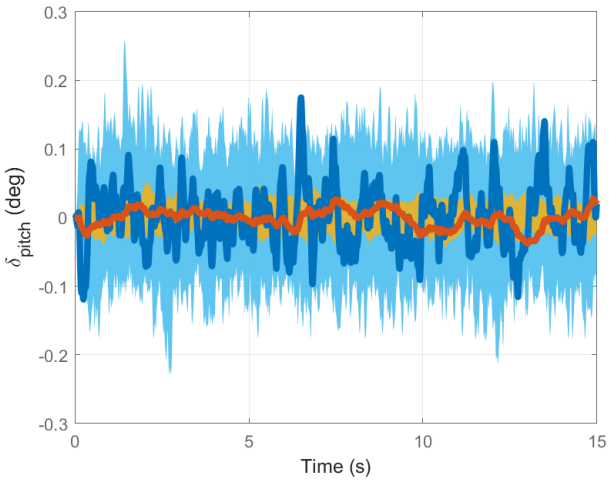

Figure C.17: HFLMA Wind - TVC deflection.

Worst-Case Analysis

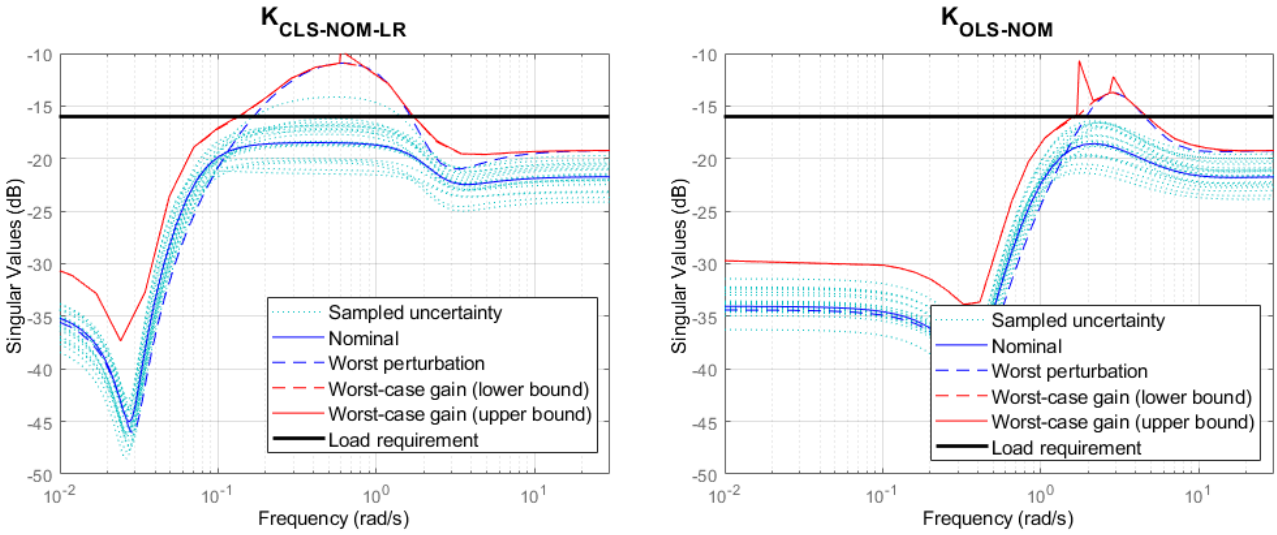

Figure C.18: WC gain of $T_{v_w \to \tilde{q}\alpha/\tilde{q}_n}$ - $K_{OLS-NOM}$ vs $K_{CLS-NOM-LR}$.

D

PP Vs $\mathcal{H}_\infty$ OLS Vs $\mathcal{H}_\infty$ CLS

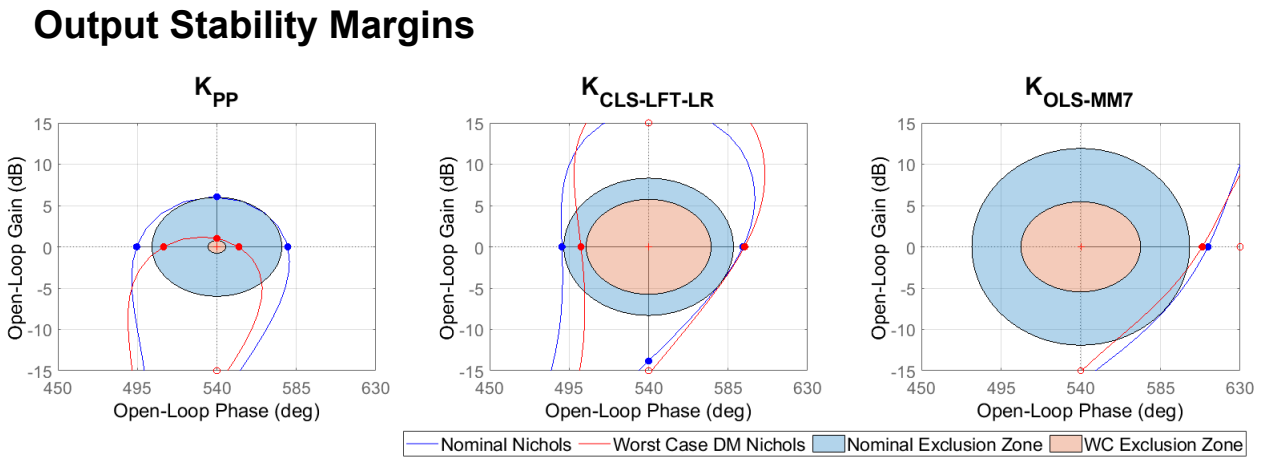

Figure D.1: Loop-at-a-time (LAT) pitch angle Nichols - PP (K_{PP}) vs $\mathcal{H}_\infty$ CLS ($K_{CLS-LFT-LR}$) vs $\mathcal{H}_\infty$ OLS ($K_{CLS-MM7}$).

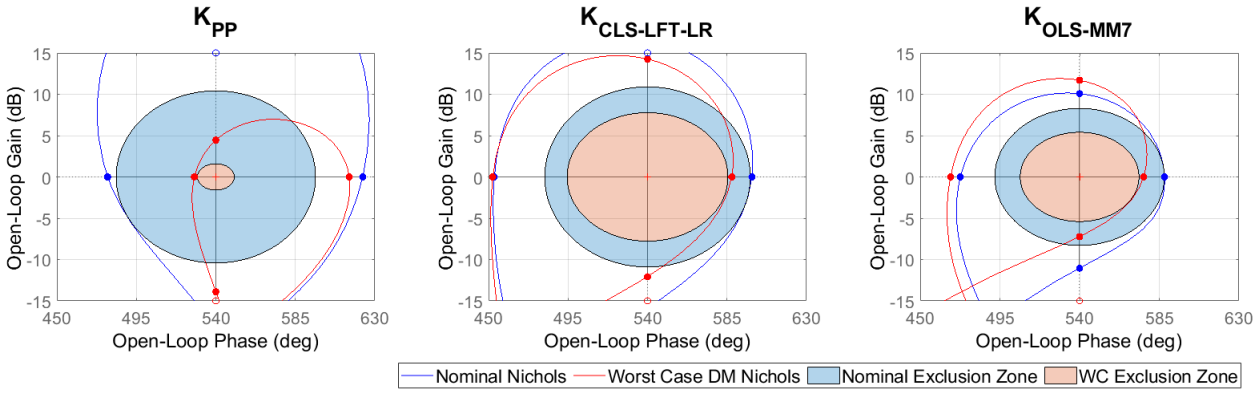

Figure D.2: LAT pitch rate Nichols - K_{PP} vs $K_{CLS-LFT-LR}$ vs $K_{CLS-MM7}$.

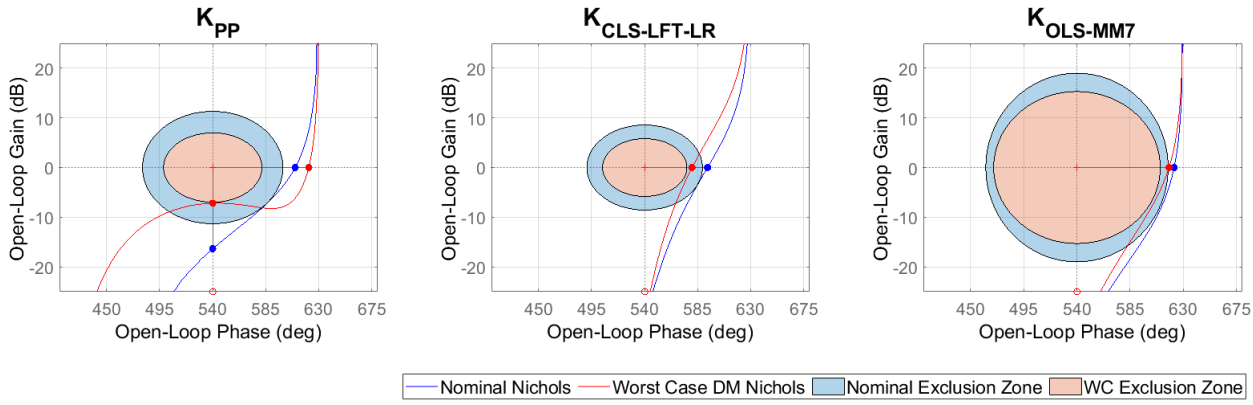

Figure D.3: LAT drift Nichols - K_{PP} vs $K_{CLS-LFT-LR}$ vs $K_{CLS-MM7}$.

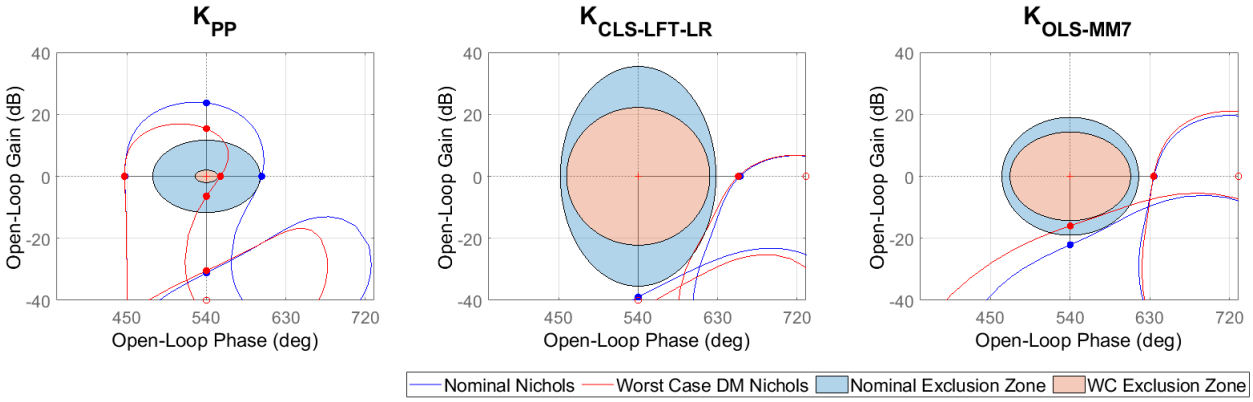

Figure D.4: LAT drift rate Nichols - K_{PP} vs $K_{CLS-LFT-LR}$ vs $K_{CLS-MM7}$.

Structured Singular Value

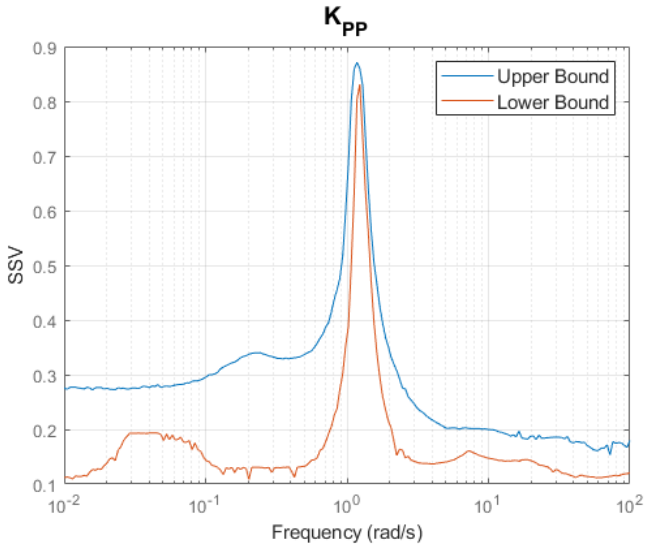

Figure D.5: Structured Singular Value - K_{PP}.

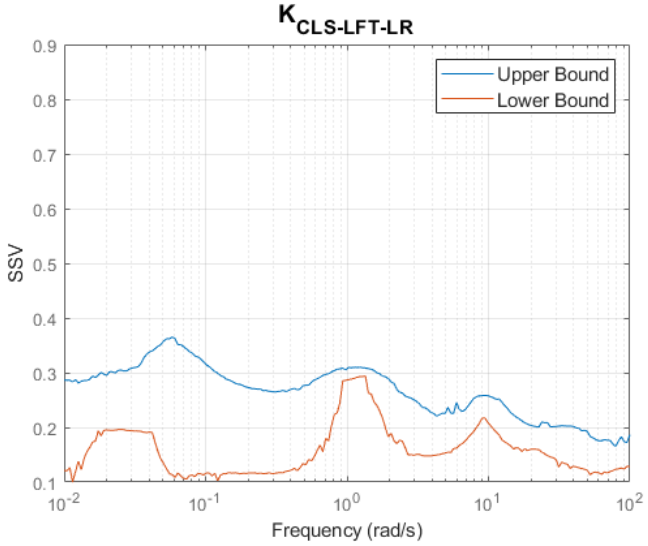

Figure D.6: Structured Singular Value - $K_{CLS-LFT-LR}$.

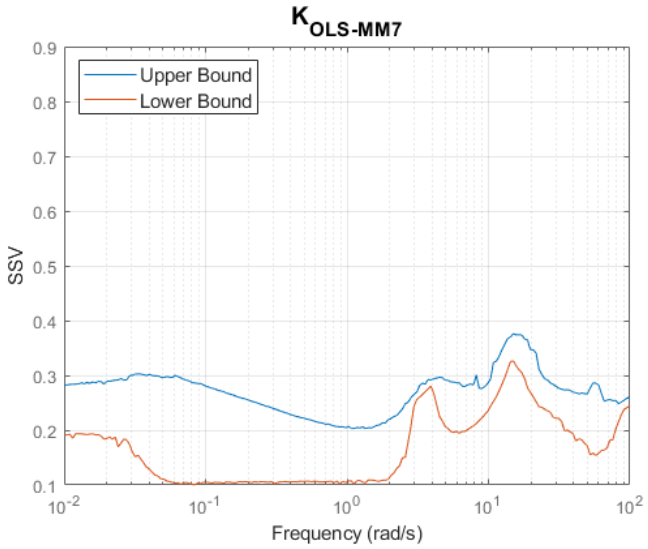

Figure D.7: Structured Singular Value - $K_{CLS-MM7}$.