

Dissertation presented to the Instituto Tecnológico de Aeronáutica, in partial fulfillment of the requirements for the degree of Master in Engineering in the Professional Master's Program of Aeronautics and Mechanics Engineering.

Pedro Henrique Oliveira de Paula

**DYNAMIC STIFFNESS OPTIMIZATION OF A
FLIGHT CONTROL ACTUATION SYSTEM
USING CONTROL TECHNIQUES**

Dissertation approved in its final version by signatories below:

Ph.D. Alberto Adade Filho

Advisor

M.Sc. Raphael das Neves Calvo

Co-advisor

Ph.D. Pedro Teixeira Lacava

Prorector of Graduate Studies and Research

Campo Montenegro
São José dos Campos, SP - Brazil
2018

**Cataloging-in Publication Data
Documentation and Information Division**

Paula, Pedro Henrique Oliveira de
Dynamic stiffness optimization of a flight control actuation system using control techniques /
Pedro Henrique Oliveira de Paula.
São José dos Campos, 2018.
120f.

Dissertation of Master in Engineering – Course of Aeronautics and Mechanics Engineering. Area of Aerospace Systems and Mechatronics – Instituto Tecnológico de Aeronáutica, 2018. Advisor: Ph.D. Alberto Adade Filho. Co-advisor: M.Sc. Raphael das Neves Calvo.

1. Actuators. 2. Optimization. 3. Closed-loop system. 4. Flight control. 5. Aerodynamics.
6. Aeronautical engineering. I. Instituto Tecnológico de Aeronáutica. II. Title.

BIBLIOGRAPHIC REFERENCE

PAULA, Pedro Henrique Oliveira de. **Dynamic stiffness optimization of a flight control actuation system using control techniques**. 2018. 120f. Dissertation of Master in Engineering – Instituto Tecnológico de Aeronáutica, São José dos Campos.

CESSION OF RIGHTS

AUTHOR'S NAME: Pedro Henrique Oliveira de Paula

PUBLICATION TITLE: Dynamic stiffness optimization of a flight control actuation system using control techniques.

PUBLICATION KIND/YEAR: Dissertation / 2018

It is granted to Instituto Tecnológico de Aeronáutica permission to reproduce copies of this dissertation and to only loan or to sell copies for academic and scientific purposes. The author reserves other publication rights and no part of this dissertation can be reproduced without the authorization of the author.

Pedro Henrique Oliveira de Paula
R. Jorge Barbosa Moreira, 361, Apt. 5
12.243-070 – São José dos Campos–SP

DYNAMIC STIFFNESS OPTIMIZATION OF A FLIGHT CONTROL ACTUATION SYSTEM USING CONTROL TECHNIQUES

Pedro Henrique Oliveira de Paula

Thesis Committee Composition:

Ph.D.	Luiz Carlos Sandoval Góes	President	-	ITA
Ph.D.	Alberto Adade Filho	Advisor	-	ITA
M.Sc.	Raphael das Neves Calvo	Co-advisor	-	EMBRAER
M.Sc.	Carlos Augusto Constantino	External member	-	EMBRAER

ITA

Para Larissa, Anelisa, Maria Elisa e José
Antônio.

Acknowledgments

My sincere acknowledgments and gratitude to my advisor, Dr. Alberto Adade Filho and my co-advisor, M.Sc. Raphael das Neves Calvo, for providing this opportunity and also for trusting and supporting me along this journey.

I would like to thank Embraer, above all Ricardo Robusti and Paulo Armanini, for the support and the time allowed for my dedication to this work.

I also would like to thank the Casemiro Montenegro Filho Foundation, for the financial support through this research work.

My special acknowledgment to Larissa, for all her love, dedication, patience and encouragement along this road.

I also thank my mother, Anelisa, who always inspired me with her dedication and excellence, my family, that even without understanding my challenges always believed in my potential, and my friends, whose joy and care I could not live without.

Finally, I thank God for this existence in wonderful planet Earth and for being with me in every step of this road.

“O homem há de voar.”
— ALBERTO SANTOS DUMONT

Resumo

Sistemas de Comandos de Voo *fly-by-wire* tem auxiliado inovações na indústria aeronáutica através de proteções de envelope, leis de controle customizadas e manutenibilidade. Para efetuar o controle de posição de uma superfície, a maioria dos sistemas *fly-by-wire* utilizam servo atuadores eletro-hidráulicos e um dos requisitos de projeto desses atuadores é a sua rigidez dinâmica. Tradicionalmente, requisitos de rigidez dinâmica são cumpridos com o balanceamento de massa da superfície de controle ou aumentando a área do pistão de atuadores hidráulicos mas ambas as soluções aumentam o peso da aeronave. Investigações anteriores sobre alternativas para essas estratégias demonstraram que com um projeto adequado do sistema de malha-fechada é possível aumentar a rigidez de um atuador em um sistema de posicionamento de superfície aerodinâmica de controle melhorando as características de rejeição de *flutter* do sistema sem prejuízo do atendimento de requisitos de desempenho usualmente demandados pela aplicação, permitindo revisão das características de projeto do atuador em benefício do sistema global. Entretanto, o projeto da malha de controle para aumento da rigidez dinâmica exige recursos valiosos. Para contornar isto, técnicas de otimização podem ser usadas já que elas permitem a investigação de um grande número de soluções em muito menos tempo do que processos convencionais. Além disso, elas auxiliam engenheiros no entendimento dos efeitos dos parâmetros e restrições de modelos e a usar essas informações para tomar decisões com mais conhecimento. Os objetivos deste trabalho são utilizar técnicas de otimização no projeto de atuadores hidráulicos para reduzir o ciclo de projeto da malha de controle, maximizar a resposta rigidez dinâmica e cumprir com requisitos de desempenho no domínio do tempo e da frequência.

Abstract

Fly-by-wire flight control systems have been supporting innovations in the aircraft industry by providing envelope protection, tailored control laws and maintainability. To perform surface position control, most fly-by-wire systems rely on electro-hydraulic servo actuators and one of their design requirements is the dynamic stiffness. Traditionally, dynamic stiffness requirements are met by either mass balancing of the control surface or by increasing the piston area of hydraulic actuators but both solutions increase aircraft weight. Previous investigations on alternatives to these approaches demonstrated that with an adequate closed-loop system design it is possible to increase the dynamic stiffness of an aerodynamic control surface actuator positioning system, improving the systems' flutter rejection characteristics without jeopardizing the performance requirements usually demanded in this application, allowing a revision of the design characteristics in favor of the global system. However, the design of control loop to increase dynamic stiffness demands valuable resources. To overcome this, optimization techniques can be used since they allow the investigation of a vast number of solutions in much less time than conventional processes. Also, they help engineers understand the effects of parameters and constraints of models and use this information to make better informed decisions. The objectives of this work are to use optimization techniques in hydraulic actuator design to reduce control loop design cycles, maximize dynamic stiffness response and comply with time and frequency domain performance requirements.

List of Figures

FIGURE 2.1 – Notation for Aircraft Body Axes. Source: Etkin e Reid (1995)	25
FIGURE 2.2 – Surface stiffness components. Adapted from Constantino (2010)	26
FIGURE 2.3 – Step response of a control system. Source: Dorf and Bishop (2008)	27
FIGURE 2.4 – Block diagram of a one degree-of-freedom feedback control system. Source: Skogestad and Postlethwaite (2001)	28
FIGURE 2.5 – Example of magnitude response of $T(j\omega)$. Adapted from Skogestad and Postlethwaite (2001)	32
FIGURE 2.6 – Closed-loop frequency response droop and magnitude peak. Adapted from Stevens and Lewis (1992)	32
FIGURE 2.7 – ARX Model Structure. Source: MathWorks (2015b)	33
FIGURE 2.8 – One dimension multi-modal function. Source: Venter (2010)	42
FIGURE 3.1 – Actuator hydraulic schematic. Source: Ballesteros (2015)	46
FIGURE 3.2 – Two-stage EHSV schematic. Source: THAYER (1965)	47
FIGURE 3.3 – Actuation System and Interfaces Simulink Model	48
FIGURE 3.4 – Actuation System Simulink Model	49
FIGURE 3.5 – Electronic System Simulink Model	50
FIGURE 3.6 – Digital PID Controller Model	50
FIGURE 3.7 – Actuator Simulink Model	51
FIGURE 3.8 – Actuator Valves Simulink Model	52
FIGURE 3.9 – Actuator Cylinder Simulink Model	52
FIGURE 3.10 –Actuator LVDT Sensors Simulink Model	53
FIGURE 4.1 – Program Flowchart	55

FIGURE 4.2 – Step Response Test Flowchart	58
FIGURE 4.3 – Fluid Density and Bulk Modulus Contribution to Step Response Actuation Rate	59
FIGURE 4.4 – Flowchart of Baseline Frequency Response Test	61
FIGURE 4.5 – 20 Hz Sine Input System Response	63
FIGURE 4.6 – 20 Hz Sine Input System Response	63
FIGURE 4.7 – Baseline and Modified Frequency Response Test Results Comparison	65
FIGURE 4.8 – Dynamic Stiffness Test Flowchart	66
FIGURE 4.9 – Baseline and Modified Dynamic Stiffness Test Comparison	68
FIGURE 5.1 – Dynamic Stiffness Comparison for Proportional Controller.	72
FIGURE 5.2 – Time Response Comparison for Proportional Controller.	73
FIGURE 5.3 – Frequency Response Comparison for Proportional Controller.	74
FIGURE 5.4 – Dynamic Stiffness Optimization Result for the PI Controller.	75
FIGURE 5.5 – Time Response Optimization Result for the PI Controller.	76
FIGURE 5.6 – Frequency Response Baseline for the PI Controller.	77
FIGURE 5.7 – Dynamic Stiffness Optimization Result for the PD Controller.	78
FIGURE 5.8 – Time Response Optimization Result for the PD Controller.	79
FIGURE 5.9 – Frequency Response Baseline for the PD Controller.	80
FIGURE 5.10 –Dynamic Stiffness Optimization Result for the PID Controller.	81
FIGURE 5.11 –Time Response Optimization Result for the PID Controller.	82
FIGURE 5.12 –Frequency Response Baseline for the PID Controller.	83
FIGURE 5.13 –Dynamic Stiffness Comparison for Proportional Controller.	85
FIGURE 5.14 –Time Response Comparison for Proportional Controller.	86
FIGURE 5.15 –Frequency Response Comparison for Proportional Controller.	87
FIGURE 5.16 –Dynamic Stiffness Optimization Result for the PI Controller.	88
FIGURE 5.17 –Time Response Optimization Result for the PI Controller.	89
FIGURE 5.18 –Frequency Response Baseline for the PI Controller.	90
FIGURE 5.19 –Dynamic Stiffness Optimization Result for the PD Controller.	91
FIGURE 5.20 –Time Response Optimization Result for the PD Controller.	92

FIGURE 5.21 –Frequency Response Baseline for the PD Controller.	93
FIGURE 5.22 –Dynamic Stiffness Optimization Result for the PID Controller.	95
FIGURE 5.23 –Time Response Optimization Result for the PID Controller.	96
FIGURE 5.24 –Frequency Response Baseline for the PID Controller.	97
FIGURE 5.25 –Optimized Dynamic Stiffness of Proportional Controller for Different Settling Time Requirements	99
FIGURE A.1 –Chirp Signal	106
FIGURE A.2 –Flowchart of Frequency Response Test with Chirp Signal	107
FIGURE A.3 –Baseline and Chirp Frequency Response Test Results Comparison .	108
FIGURE A.4 –Flowchart of Frequency Response Test with ARX Identification . . .	109
FIGURE A.5 –Identification Signal	110
FIGURE A.6 –Simulink Model for Baseline Frequency Response Test	110
FIGURE A.7 –Baseline and ARX Model Identification Frequency Response Test Results Comparison	111
FIGURE A.8 –Non-Linear and Linearized Model Frequency Response Test Results Comparison	113
FIGURE A.9 –Baseline and Modified Frequency Response Test Results Comparison	115
FIGURE B.1 –Actuation System and Interfaces Simulink Model	117
FIGURE B.2 –System Output for a 20Hz sine input signal	118
FIGURE B.3 –System Output for a 20Hz sine input signal	119

List of Tables

TABLE 4.1 – Classic Controller Gain Boundaries	56
TABLE 4.2 – Stopping Criteria Parameters	56
TABLE 4.3 – Constraints Parameters	57
TABLE 4.4 – Baseline and Modified Frequency Response Test Results Comparison	65
TABLE 5.1 – Initial Solution for Classic Controllers	71
TABLE 5.2 – P Controller Optimization Execution Results	72
TABLE 5.3 – P Controller Final Solution Cost Function for Each Evaluated Frequency	73
TABLE 5.4 – Requirement Compliance for Proportional Controller	74
TABLE 5.5 – PI Controller Optimization Execution Results	75
TABLE 5.6 – PI Controller Final Solution Cost Function for Each Evaluated Frequency	76
TABLE 5.7 – Requirement Compliance for PI Controller	77
TABLE 5.8 – PD Controller Optimization Execution Results	78
TABLE 5.9 – PD Controller Final Solution Cost Function for Each Evaluated Frequency	79
TABLE 5.10 – Requirement Compliance for PD Controller	80
TABLE 5.11 – PID Controller Optimization Execution Results	81
TABLE 5.12 – PID Controller Final Solution Cost Function for Each Evaluated Frequency	82
TABLE 5.13 – Requirement Compliance for PID Controller	83
TABLE 5.14 – P Controller Optimization Execution Results	85

TABLE 5.15 – P Controller Final Solution Cost Function for Each Evaluated Frequency	86
TABLE 5.16 – Requirement Compliance for Proportional Controller	87
TABLE 5.17 – PI Controller Optimization Execution Results	88
TABLE 5.18 – PI Controller Final Solution Cost Function for Each Evaluated Frequency	89
TABLE 5.19 – Requirement Compliance for PI Controller	90
TABLE 5.20 – PD Controller Optimization Execution Results	91
TABLE 5.21 – PD Controller Final Solution Cost Function for Each Evaluated Frequency	92
TABLE 5.22 – Requirement Compliance for PD Controller	94
TABLE 5.23 – PID Controller Optimization Execution Results	94
TABLE 5.24 – PID Controller Final Solution Cost Function for Each Evaluated Frequency	95
TABLE 5.25 – Requirement Compliance for PID Controller	97
TABLE 5.26 – Requirement Compliance of Proportional Controller for Different Settling Time Requirements	100
TABLE A.1 – Chirp Frequency Response Test Parameters	107
TABLE A.2 – Baseline and Chirp Frequency Response Test Comparison	109
TABLE A.3 – Baseline and ARX Model Identification Frequency Response Test Results Comparison	111
TABLE A.4 – Non-Linear and Linearized Model Frequency Response Comparison	114
TABLE A.5 – Baseline and Modified Frequency Response Test Results Comparison	115
TABLE A.6 – Baseline and Modified Frequency Response Test Execution Time Comparison	116
TABLE B.1 – THD values of a PD controller frequency response	119

List of Abbreviations and Acronyms

AFCS	Automatic Flight Control System
ARX	Auto-Regressive with Exogenous Input
BFGS	Broyden-Fletcher-Goldfarb-Shanno
CLGA	Closed-Loop Gain Allowance
CLPA	Closed-Loop Phase Allowance
CSAS	Control and Stability Augmentation System
EHA	Electro-Hydrostatic Actuator
EHSA	Electro-Hydraulic Servo Actuator
EHSV	Electro-Hydraulic Servo Valve
EMA	Electro-Mechanical Actuator
FCS	Flight Control System
FFT	Fast Fourier Transform
GA	Genetic Algorithm
GM	Gain Margin
KKT	Karush-Kuhn-Tucker
P	Proportional
PD	Proportional Derivative
PFCS	Primary Flight Control System
PI	Proportional Integral
PID	Proportional Integral Derivative
PM	Phase Margin
LVDT	Linear Variable Differential Transformer
SLP	Sequential Linear Programming
SQP	Sequential Quadratic Programming
THD	Total Harmonic Distortion

List of Symbols

x_v	EHSV 1st stage spool position
w_{ne}	EHSV 1st stage natural frequency
ξ_e	EHSV 1st stage damping ratio
A_s	EHSV spool end area
K_r	EHSV flapper-armature gain
K_l	EHSV 1st stage torque motor gain
x_v	EHSV spool displacement
i	EHSV current
K_2	Hydraulic amplifier flow gain
K_w	Feedback wire stiffness
P_1	Pressure in piston chamber 1
P_2	Pressure in piston chamber 2
x_p	Piston position
A_p	Piston area
V_{01}	Chamber 1 initial volume
V_{02}	Chamber 2 initial volume
β_c	Constant bulk modulus
K_{CD}	Constant based on the discharge coefficient and fluid density
P_s	Supply pressure
P_t	Return pressure
A_{int}	Internal leakage equivalent area
A_{ext}	External leakage equivalent area
x_w	EHSV second-stage orifice area width
A_{leak}	EHSV second-stage leakage
m_p	Piston mass
B_p	Viscous damping coefficient

Contents

1	INTRODUCTION	19
1.1	Motivation	19
1.2	Bibliographic Review	20
1.3	Objective	22
1.4	Methodology	22
1.5	Organization	22
2	THEORY BACKGROUND	24
2.1	Actuator Dynamic Stiffness	24
2.2	Control Systems Performance	26
2.2.1	System Step Input Response	26
2.2.2	System Frequency Response	28
2.2.3	Actuation System Frequency Domain Requirements	34
2.3	Optimization Problems	36
2.3.1	Constrained vs. Unconstrained	37
2.3.2	Linear vs. Nonlinear	37
2.3.3	Discrete vs. Continuous	38
2.3.4	Deterministic vs. Nondeterministic	38
2.3.5	Single vs. Multi-objective	38
2.3.6	Single vs. Multiple Minima	38
2.3.7	Simple vs. Complex	38
2.4	Optimization Techniques	39
2.4.1	Local Optimization Algorithms	39

2.4.2	Global Optimization Algorithms	42
2.4.3	Selected Algorithm	43
3	ACTUATION SYSTEM MODEL	46
3.1	Simulink Model	48
3.1.1	Electronic System	49
3.1.2	Actuator	51
3.1.3	Surface Kinematics	53
4	DYNAMIC STIFFNESS OPTIMIZATION	54
4.1	Problem Definition	54
4.2	Optimization Settings	55
4.3	Optimization Constraints	56
4.3.1	Step Response Test	58
4.3.2	Frequency Response Test	60
4.4	Objective Function	66
4.4.1	Dynamic Stiffness Test	66
4.4.2	Cost Function	68
5	OPTIMIZATION RESULTS	71
5.1	Classic Controllers and Simple Cost Function	71
5.1.1	Proportional Controller (P)	71
5.1.2	Proportional Integral Controller (PI)	75
5.1.3	Proportional Derivative Controller (PD)	77
5.1.4	Proportional Integral Derivative Controller (PID)	81
5.2	Classic Controllers and Frequency Weighted Cost Function	84
5.2.1	Proportional Controller (P)	84
5.2.2	Proportional Integral Controller (PI)	88
5.2.3	Proportional Derivative Controller (PD)	90
5.2.4	Proportional Integral Derivative Controller (PID)	94
5.3	Optimization Constraints Sensitivity	98

5.3.1	P and PI Controllers	98
5.3.2	PD and PID Controllers	100
6	CONCLUSION	101
6.1	Future Works	102
	BIBLIOGRAPHY	104
	APPENDIX A – FREQUENCY RESPONSE TEST	106
A.1	Chirp Input Frequency Response Test	106
A.2	ARX Identification Frequency Response Test	109
A.3	Linear Model Frequency Response Test	112
A.4	Modified Baseline Frequency Response Test	114
	APPENDIX B – FREQUENCY RESPONSE OF NON-LINEAR SYSTEMS	117

1 Introduction

This work explores the optimization of hydraulic actuator dynamic stiffness through position control loop design for classical controllers. The motivation, objective, bibliographic review and organization of this document are presented in the following sections.

1.1 Motivation

Technological innovations in aircraft design have transformed the industry in many aspects from aerodynamics, materials and propulsion to human factors and handling. Fly-by-wire flight control systems supported many of these innovations by providing envelope protection, tailored control laws and maintainability. Most fly-by-wire systems rely on electro-hydraulic servo actuators to perform surface position control.

Control surfaces are subject to aeroelastic phenomena as well as other aircraft structures, thus they need to comply with flutter requirements as to prevent conditions that may lead to a catastrophic event. Usually, flutter requirements are met by either mass balancing of the control surface or by increasing the piston area of hydraulic actuators to increase hydraulic dynamic stiffness. These traditional solutions penalize aircraft performance: surface mass balancing increases weight and larger actuator piston area increases overall actuator size and weight which implies in thicker aerodynamic profiles on stabilizers and wing.

In addition, complex products, cost reduction and short development cycles are a reality in a great number of engineering projects and to overcome these challenges it is necessary to explore alternatives that allow greater efficiency in project development. One alternative is the use of optimization algorithms that have been spreading over the industry in the last years and can be developed with a range of softwares available in the market.

Design optimization allows the improvement of products because it can investigate a vast number of solutions in much less time than conventional processes. Also, engineers are able to better understand the effects of parameters and constraints of models and use

this information to make better informed decisions.

By using optimization techniques in hydraulic actuator design, it is possible to improve the dynamic stiffness response of an actuator design that previously did not satisfy safety requirements without degrading time and frequency domain performance, allowing actuator size reduction. This leads to reduced actuator weight, reduced hydraulic consumption and therefore overall weight reduction on the aircraft.

1.2 Bibliographic Review

This section presents a quick review of important publications regarding optimization, aircraft control and actuator design that investigated different aspects of these subjects and somehow contributed to this work. In addition, other references such as books and manuals will be presented.

An algorithmic approach for robust controller design was presented by Almeida and Adade Filho (2011). The authors used search algorithms to find H-infinity controllers that would comply with the control problem requirements. Their approach not only achieved this goal but also yielded a lower order controller compared to previous works.

Silva, Paiva and Galvão (2009) investigated the use of simplex algorithm in aircraft control law design such as control and stability augmentation system (CSAS) and automatic flight control system (AFCS). The authors proposed a simple cost function that considered performance and robustness criteria. The cost function format allowed the definition of targets for each criteria in a way that when one target was reached the further improvement of this target did not reduce the cost function considerably anymore. This format proved to be useful for optimization of multiple objectives while guaranteeing a minimum acceptable value for each of them.

An optimization of electrical flight control actuators was studied by Chakraborty *et al.* (2013). The aim was to minimize weight of two types of electric actuators: electro-hydrostatic (EHA) and electromechanical (EMA). The authors managed to use a reduced number of design variables to allow the use of gradient-based optimization. The problem constraints were actuator operational parameters such as motor stall current and winding temperature and were selected based on the most demanding flight conditions. The study presented a weight reduction in the set of actuators used in a more electric aircraft.

An investigation in multi-objective optimization in engineering design was conducted by Andersson (2001). The author proposes a cost function formulation that considers multiple objectives and their relative importance and a method for selecting the design variables. Also, the study integrates different disciplines and proposes the use of the

House of Quality method to identify the importance of each optimization objective. This method was applied successfully in the optimization of a landing gear system.

Venter (2010) presented a broad review of optimization techniques evaluating the advantages and disadvantages of their strategies to solve optimization problems as well as explaining these strategies. The review covered techniques for global and local optimization as well as unconstrained and constrained problems.

Constantino (2010) developed a model of primary flight control actuation system. The model represents an active-active configuration considering two hydraulic actuators and their servo valves besides a position control loop and an aircraft control surface. The study also evaluated the behavior of oscillatory failures and indicated the need of a monitor to identify this failure modes in order to prevent the fatigue of the aircraft structure.

The actuation system model was further developed by Ballesteros (2015), who introduced the variation of hydraulic fluid bulk modulus as a function of fluid temperature, pressure and percentage of entrained-air. Also, the dynamic response of the LVDT sensors was included as well as its associated measurement error. Electro-hydraulic servo valve and inlet check valve models were updated to allow use of industry catalog parameters. Ballesteros (2015) also performed, a broad parametric study in order to understand the influence of classic and modern control loop topologies in the actuator performance.

The parametric study performed by Ballesteros (2015) is an important source for understanding the relationship between design parameters and dynamic stiffness and will be used as the starting point for the development of the work present herein.

Other important references are mentioned below. These are books and manuals that were mainly consulted for support and clarification of the concepts involved in this work.

An overview in classic and modern control systems is presented by Dorf e Bishop (2008). Covered topics range from model representations, performance and stability of feedback systems and frequency response methods. Merritt (1967) presents concepts in hydraulic systems, such as design, components and control.

In Ljung (1987) important concepts in system identification are detailed including underlying theory. Time and frequency domain identification methods are presented as well as convergence, consistency and model validation concepts.

An approach in engineering optimization is presented by Rao (2009). The book introduces basic optimization concepts and methods for solving simple problems. Additionally, advanced techniques for solving nonlinear, geometric, discrete and stochastic problems are also presented.

Messac (2015) introduces optimization concepts side by side with practice using MATLAB. Nevertheless, the book provides an overview in many advanced techniques including

nonlinear programming and evolutionary algorithms.

1.3 Objective

The main objective of this work is to reduce the development cycle and the number of trials and rework associated with control loop tuning while complying with their performance requirements. Therefore, it is proposed an optimization approach as the solution for the presented problem.

Hence, an specific objective is to develop an optimization algorithm to maximize a rudder flight control electro-hydraulic servo actuator (EHSA) dynamic stiffness response while maintaining time and frequency domain performance through the tuning of controller gains for classical control techniques.

Dynamic stiffness improvement for a given actuation system compliant with stiffness requirements will increase margins for flutter suppression. Additionally, for an actuation system that is not compliant with flutter requirements, it may be possible to reach the required dynamic stiffness with the optimization algorithm's support.

That is, controller gains of actuators with reduced piston area may be optimized to increase dynamic stiffness response up to required levels, therefore, smaller actuators can meet safety and performance requirements, delivering the benefits discussed in Section 1.1.

1.4 Methodology

The development of the dynamic stiffness optimization algorithm as well as the algorithmic design program was performed with MATLAB, a software environment able to support modeling and simulation of dynamic systems as well as executing programmed functions and scripts. The rudder actuation system model used in this work is the high fidelity model developed by Constantino (2010) and further improved by Ballesteros (2015), who also developed routines to evaluate time, frequency and dynamic stiffness response.

1.5 Organization

Chapter 1 introduces the motivation, bibliographic review, objectives and methodology of this work.

Chapter 2 presents the concepts used in this work. First, the actuator dynamic stiffness

is introduced as well as the reasons why it is an important specification. Next, a review of time and frequency domain performance criteria is presented along with the definition of parameters used in this work to evaluate frequency domain performance. Additionally, an identification methodology for nonlinear system is presented. The chapter also presents concepts on optimization problems, its categories and available techniques for solving them.

Chapter 3 presents the architecture of the rudder surface actuation system which is the object of this study. Additionally, it provides an overview of the computational model of the rudder actuation system used to obtain the behavior of the system for each controller evaluated by the optimization algorithm.

Chapter 4 provides a definition of the optimization problem solved in this work, comprising the design variables the constraints and the objective function. Further sections of this chapter detail the settings used in the optimization, the constraints of the problem and how their values are obtained by the program and finally the formulation of the objective function.

Chapter 5 contains the results obtained with the developed optimization program for the evaluated classical control strategies. Furthermore, it is presented a constraint sensitivity analysis in which it is evaluated the impact of changing the constraints on the optimization results.

Chapter 6 closes with conclusions and suggestions of further development.

2 Theory Background

This chapter presents a review in important concepts required in this development. The relationship between Flight Control Systems (FCS) and aeroelasticity will be briefly introduced since a detailed review can be found in Ballesteros (2015).

Next, the parameters used to evaluate actuation system performance are presented. In this section, non-linear system identification techniques such as Autoregressive with Exogenous Input (ARX) are also discussed because of their potential use in an alternative frequency response test.

Finally, an overview on optimization problems and programming techniques is presented. For other specific details, the indicated references can be consulted.

2.1 Actuator Dynamic Stiffness

In an aircraft, the capability to turn, roll and climb (N , L and M in Figure 2.1) is provided by the Primary Flight Control System (PFCS). This system is responsible of acquiring movement commands from the pilots and executing them in a predictable and smooth fashion. Currently, most PFCS architectures rely on fly-by-wire technology to deliver the expected performance.

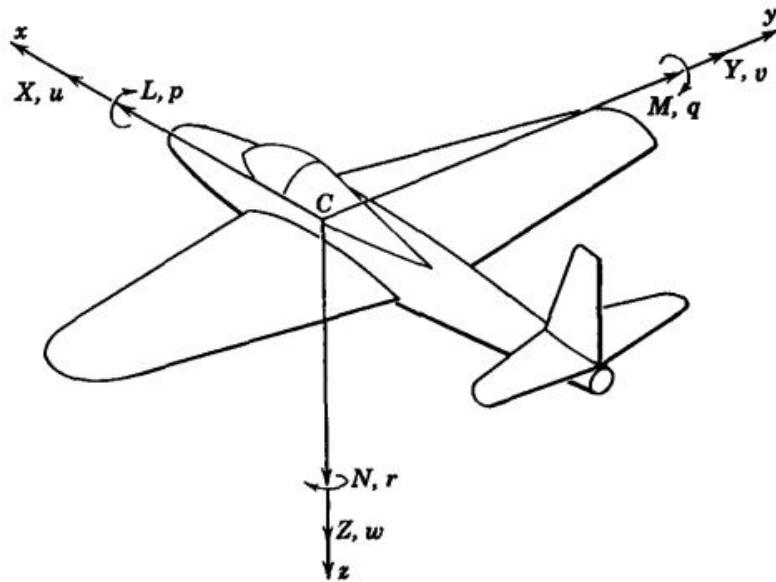


FIGURE 2.1 – Notation for Aircraft Body Axes. Source: Etkin e Reid (1995)

In fly-by-wire architectures, an electronic computer receives electrical signals from sensors measuring the pilots inputs over cockpit controls. After processing this data, the computer electrically sends a position command to each primary surface actuator which is responsible to move the piston and maintain the commanded position. Usually, this task is performed by EHSAs which receive an electric position command signal and uses hydraulic power to convert the surface position command and achieve the commanded piston position.

Aircraft control surfaces are subject to aeroelastic phenomena, such as flutter, as well as other structure parts. Wright and Cooper 2007 define flutter as:

“(...) an unstable self-excited vibration in which the structure extracts energy from the air stream and often results in catastrophic structural failure. Classical binary flutter occurs when the aerodynamic forces associated with motion in two modes of vibration cause the modes to couple in an unfavourable manner.”

Flutter is avoided by designing structures and other parts with appropriate stiffness at critical frequencies i.e. a surface vibration movement induced by the air stream will be small enough to avoid coupling with other vibration modes.

The overall stiffness of a flight control surface for flutter suppression is broken down in budgets for each part of the assembly. Figure 2.2 shows an example of this distribution.

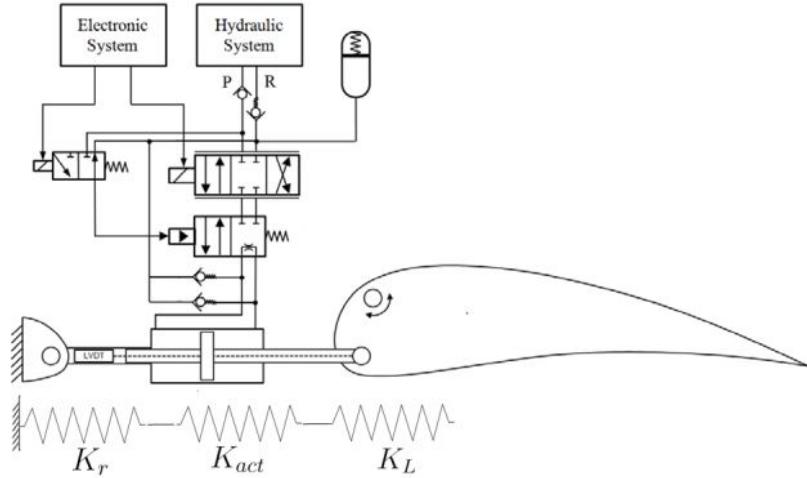


FIGURE 2.2 – Surface stiffness components. Adapted from Constantino (2010)

The flight control surface equivalent linear dynamic stiffness (K_{lin}) is defined as:

$$\frac{1}{K_{lin}} = \frac{1}{K_r} + \frac{1}{K_L} + \frac{1}{K_{act}} \quad (2.1)$$

Where K_r and K_L refer to structural stiffness components and K_{act} is the actuator contribution to the surface dynamic stiffness.

Since flight control actuators contribute to control surface stiffness, one of the EHSA design requirements is its dynamic stiffness.

The subject of this work is the maximization of actuator dynamic stiffness using optimization algorithms to ascertain control loop gains while maintaining compliance with time and frequency domain performance requirements.

2.2 Control Systems Performance

In Flight Control Systems design, actuator control performance specifications contain time and frequency domain parameters. This section will define the parameters used in this work to assess the feasibility of each design.

2.2.1 System Step Input Response

In general, time domain performance of a system is defined based on its step response as shown in Figure 2.3 (DORF; BISHOP, 2008).

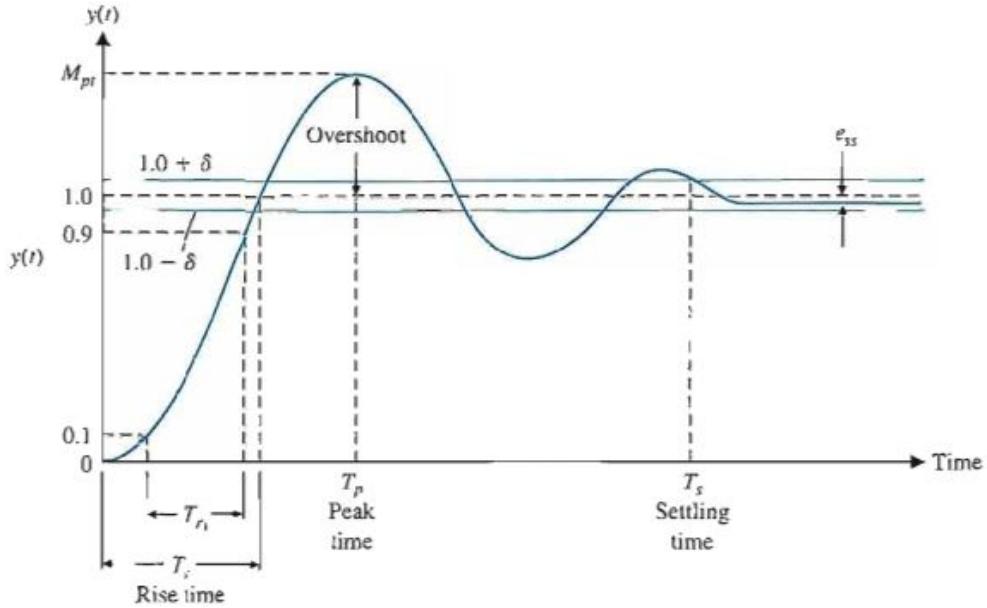


FIGURE 2.3 – Step response of a control system. Source: Dorf and Bishop (2008)

The quickness of the response is measured by the rise time (T_r) and the peak time (T_p). For over damped systems the rise time is the difference between the instant where the response reaches 90% of its final value and the instant when it reaches 10%. The peak time is the instant when the response reaches its maximum value but this measure is not defined for over damped systems. For under damped systems the peak time is well defined and the rise time is usually taken between 0% and 100% of the reference input value.

The closeness of the response to the step command is determined by the overshoot, settling time and steady state error.

The response overshoot is the percentage of the final value that exceeds it in the first peak of an under damped system response. Formally, it is given by Equation 2.2:

$$O = \frac{M_p - y_s}{y_s} \times 100\% \quad (2.2)$$

Where M_p is the maximum value of the system step response and y_s is the final value of the response.

The settling time (T_s) is defined as the time when model response reaches and remains within a certain percentage of its steady-state value. This parameter indicates the actual time it takes the system to reach the commanded position.

The steady state error determines the response closeness to the commanded value and

it is given by Equation 2.3:

$$e_{ss} = \frac{y_s - r}{r} \times 100\% \quad (2.3)$$

Where y_s is the final value of the response and r is the commanded value.

2.2.2 System Frequency Response

Figure 2.4 shows a general one degree-of-freedom feedback control system.

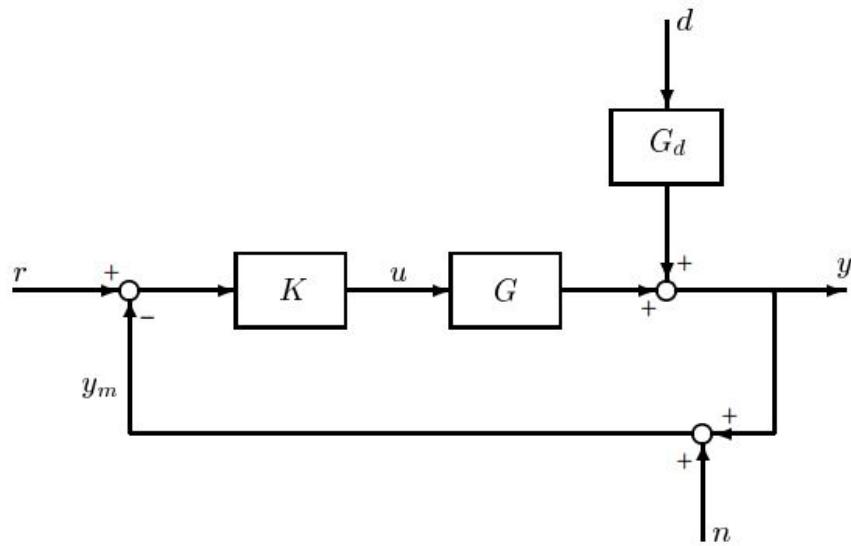


FIGURE 2.4 – Block diagram of a one degree-of-freedom feedback control system. Source: Skogestad and Postlethwaite (2001)

Skogestad and Postlethwaite (2001) deduce for this general system the following relationship between reference, disturbance and noise inputs and the system output in Equation 2.4.

$$y = (I + GK)^{-1}GKr + (I + GK)^{-1}G_d d + (I + GK)^{-1}GKn \quad (2.4)$$

Where:

r : reference input signals;

u : control signals;

- d : disturbance input signals;
- y : system output signals;
- n : noise input in measured signals;
- ym : system feedback signals;
- K : controller transfer function;
- G : plant transfer function;
- G_d : disturbance plant transfer function;

Additionally, it is defined the following terminology for transfer functions that are useful for frequency domain analysis, shown in Equations 2.5, 2.6 and 2.7.

$$L = GK \quad (2.5)$$

$$S = (I + GK)^{-1} = (I + L)^{-1} \quad (2.6)$$

$$T = (I + GK)^{-1}GK = (I + L)^{-1}L \quad (2.7)$$

Where:

- L : open-loop function;
- S : sensitivity function;
- T : complementary sensitivity function;

The frequency response of L , S and T may be used to establish a relationship between open-loop frequency response and system performance parameters such as gain and phase margins.

Gain margin, usually given in Decibels (dB), is the factor by which the loop gain $|L(jw)|$ may be increased before the closed-loop system becomes unstable (SKOGESTAD; POSTLETHWAITE, 2001). It is defined by Equation 2.8.

$$GM = 20\log \left(\frac{1}{|L(j\omega_{180})|} \right) \quad (2.8)$$

Where the phase crossover frequency w_{180} is the frequency whose the phase of $|L(jw)|$ is 180 degrees.

The gain margin is important because of the real system parameters uncertainty that may cause a difference between real and modeled transfer function gains. If this difference is higher than the gain margin the system will be unstable.

Phase margin, usually given in Degrees ($^{\circ}$), is the phase delay at the gain crossover frequency w_{0dB} that can be introduced in the feedback loop before the closed-loop system becomes unstable. It is defined by Equation 2.9 (SKOGESTAD; POSTLETHWAITE, 2001).

$$PM = \angle L(j\omega_{0dB}) + 180^{\circ} \quad (2.9)$$

It is also important to understand and define proper phase margin requirements to avoid unexpected instability due to system parameters uncertainty.

Sometimes is not possible to obtain these margins via Equations 2.8 and 2.9, for instance, when $L(j\omega)$ is unknown. An alternative method is to use a computer model to perform simulations and obtain the system response for a sine wave input at each frequency of interest.

Considering the block diagram of Figure 2.4 as this computer model, the input would be r and the output of transfer function $L(j\omega)$ would be y_m .

In this case, the magnitude and phase delay at each frequency can be obtained by processing the input and output of these simulations. Signals r and y_m Fourier Transform (R and Y_m) provide their spectrum, where the component at the frequency of interest ω_i can be obtained.

In this case, magnitude and phase of the frequency response are given by Equations 2.10 and 2.11.

$$|L(j\omega_i)| = \frac{|Y_m(j\omega_i)|}{|R(j\omega_i)|} \quad (2.10)$$

$$\angle L(j\omega_i) = \angle Y_m(j\omega_i) - \angle R(j\omega_i) \quad (2.11)$$

After obtaining magnitude and phase of $L(j\omega)$ for an adequate set of frequencies, the gain and phase margins of the control system can be obtained with Equations 2.8 and 2.9.

Another alternative method to obtain gain and phase margins of a control system is to use Equations 2.12, 2.13 and 2.14 (SKOGESTAD; POSTLETHWAITE, 2001). This alternative will provide a minimum value for these margins.

If the complementary sensitivity function T is unknown, its frequency response can also be obtained through simulations as mentioned above using r as the input and y as the output.

$$M_T = \max_{\omega} |T(j\omega)| \quad (2.12)$$

$$GM \geq 1 + \frac{1}{M_T} \quad (2.13)$$

$$PM \geq 2 \arcsin \left(\frac{1}{2M_T} \right) \quad (2.14)$$

Other useful parameters to assess frequency domain performance are the bandwidth, droop and magnitude peak. These are all obtained from the complementary sensitivity function as detailed below.

The bandwidth provides the frequency range in which control is effective. Dorf e Bishop (2008) defines that, in systems where the low-frequency magnitude is 0 dB, the bandwidth (ω_{BT}) is the frequency in which the closed-loop magnitude response is -3 dB as per Equation 2.15. A visual representation of the bandwidth is shown in Figure 2.5.

$$|T(j\omega_{BT})| = -3dB \quad (2.15)$$

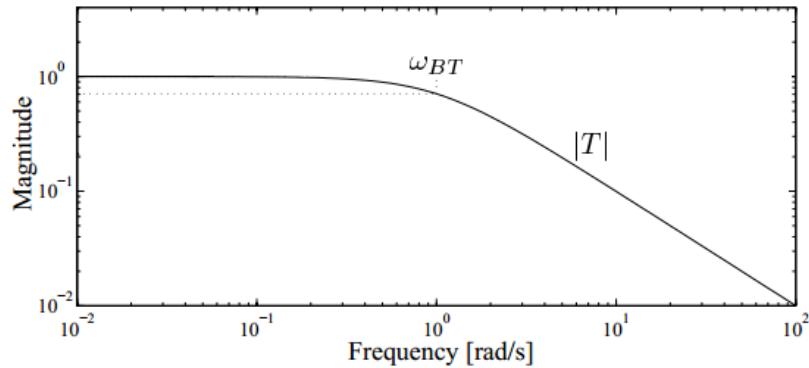


FIGURE 2.5 – Example of magnitude response of $T(j\omega)$. Adapted from Skogestad and Postlethwaite (2001)

The magnitude peak is defined by equation 2.12 and can be seen in Figure 2.6 which also shows the droop. The droop is the minimum value of magnitude of the frequency response between the zero and the peak frequency. Droop requirements are often defined to avoid a reduction of the bandwidth.

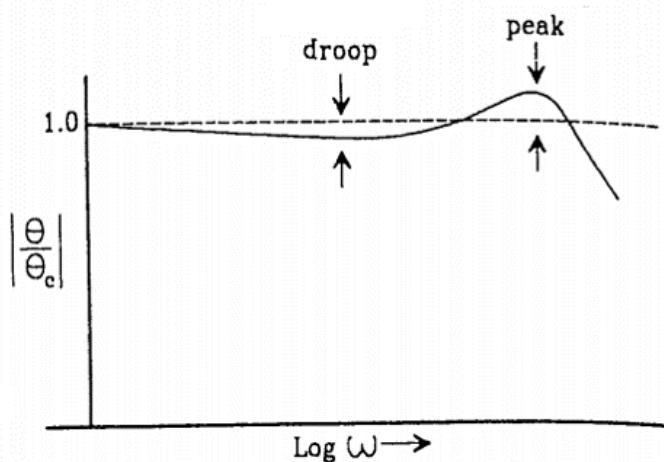


FIGURE 2.6 – Closed-loop frequency response droop and magnitude peak. Adapted from Stevens and Lewis (1992)

2.2.2.1 Nonlinear ARX Models

Another method for obtaining the frequency response is through the identification of the system model. System Identification is a discipline dedicated to build mathematical models of dynamic systems based on observed data from these systems (LJUNG, 1987). Many applications of this discipline focus on finding a model that describes well a system of interest so this model can be easily simulated to support the controller design process.

However, there are different uses for these techniques such as finding the frequency

response of a system. Computational models can be very complex and time consuming to simulate, what burdens their use as a design tool. An option to overcome this challenge is to find a mathematical model that describes well the complex computational model and is easier to manipulate.

A mathematical model that can describe nonlinear systems is the ARX. These model consists in a set of regressors and a nonlinearity estimator, as shown in Figure 2.7.

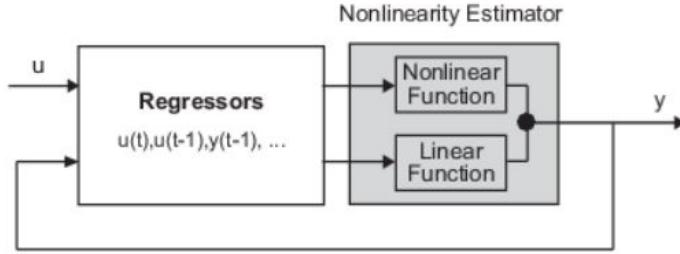


FIGURE 2.7 – ARX Model Structure. Source: MathWorks (2015b)

The nonlinearity estimator contains linear and nonlinear functions that act on the model regressors to give the model output (MATHWORKS, 2015b). Equation 2.16 shows the mathematical structure of an ARX model for a single-input single-output (SISO) system.

$$y(t) = \mathbf{F}(y(t-1), y(t-2), \dots, y(t-n_o), u(t), u(t-1), \dots, u(t-(n_i-1))) \quad (2.16)$$

Where:

$y(t)$: model output at instant t ;

\mathbf{F} : nonlinearity estimator function whose inputs are model regressors;

n_o : regressor order that specifies the number of regressors from the output to predict the output;

n_i : regressor order that specifies the number of regressors from the input to predict the output.

The regressor stage introduces delays in the inputs and outputs of the model and creates an array of inputs used by the non linear estimator to generate a model output. \mathbf{F} is a nonlinear function that is defined by the type of non linear estimator, such as *wavelet network*, *sigmoid network* and *binary tree*.

The *wavelet network* estimator has a mathematical structure in which the wavelet function is applied to each regressor and is the default estimator for nonlinear ARX model identification in MATLAB.

Estimator *sigmoid network* structure is similar to the *wavelet network* but the sigmoid function is applied to each regressor. Finally, *tree partition* is an estimator that uses a binary tree structure to partition the regressor space and apply a linear function to each partition.

2.2.2.2 Total Harmonic Distortion

The Total Harmonic Distortion (THD) is used to quantify the level of harmonics in a waveform. In this work, it has been used to quantify how much non-linear the actuator response is in comparison to its linear component. In others words, the THD quantifies the non-linearity of the response.

The THD is defined by Equation 2.17 (SHMILOVITZ, 2005).

$$THD = \frac{\sqrt{V_2^2 + \dots + V_n^2}}{\sqrt{V_1^2 + V_2^2 + \dots + V_n^2}} \times 100\% \quad (2.17)$$

Where V_n is the root mean square value of the n^{th} harmonic and $n = 1$ is the fundamental frequency. The THD measures the ratio between the power of the harmonics and the total power of the waveform in percentage. Hence, if the THD is low, the harmonics introduced by the system do not have a major influence in the response whereas high values of THD indicate that the system output is highly distorted.

2.2.3 Actuation System Frequency Domain Requirements

Even though gain and phase margin are important parameters in control systems design, sometimes the requirements of a surface actuation system are written in terms of its closed loop frequency response.

The need to characterize the local closed control loop of each surface actuator is due to the aircraft-level control laws that consider them as plants to be controlled. For instance, USAF (2008) provides guidance for the definition of aircraft-level control loop gain and phase margins requirements. These requirements are broken down as frequency response budgets for each component of the aircraft control loop such as the flight control computer, digital data bus delays, actuation control loop and sensors dynamics.

Hence, during early development stages, aircraft manufacturer and supplier work to-

gether to agree a set of closed-loop performance requirements for the actuation systems so that each party can carry-on separately with design activities.

In the perspective of the aircraft manufacturer, regarding frequency domain performance, it is important to know how much delay and gain can be introduced in the aircraft-level control loop so its controllers, filters and other logics have defined restrictions.

Additionally, the verification of gain and phase margins of the local actuation system in the real product is not practical from a development strategy standpoint. The verification of these requirements require a more complex software/firmware and even so there is a risk of the actuator reaching the mechanical stop and consuming fatigue life which is not long for primary actuators, also, it is difficult to guarantee repeatability of this test.

Therefore, it is common in the industry to specify closed-loop frequency response requirements of surface actuation systems. There are many parameters to use in these specifications such as bandwidth, magnitude peak and droop. Despite this, the frequency performance parameters selected for this work were the same used by Ballesteros (2015) in order to allow a fair result comparison, they are the maximum gain at the phase crossover frequency (Closed-Loop Gain Allowance - CLGA) and the maximum phase delay at the gain crossover frequency (Closed-Loop Phase Allowance - CLPA).

For the system in Figure 2.4, CLGA and CLPA can be obtained with Equations 2.18 and 2.19.

$$CLGA = 20 \log \left(\frac{1}{|T(j\omega_{180})|} \right) \quad (2.18)$$

$$CLPA = \angle T(j\omega_{0dB}) + 180^\circ \quad (2.19)$$

Where:

T : complementary sensitivity function;

ω_{180} : phase crossover frequency;

ω_{0dB} : gain crossover frequency;

Using the CLGA as a requirement prevents the occurrence of high gains in high frequencies and as a result large bandwidths are avoided. This is desired since large bandwidths are related to a high noise sensitivity. Additionally, the CLPA is useful for guaranteeing a maximum phase delay at the gain crossover frequency which leads to an overall reduced phase delay.

Even though these are useful metrics, it is suggested for future works to use closed-loop frequency response parameters such as bandwidth, peak magnitude and droop, which are aligned with common industry and academia practice.

2.3 Optimization Problems

In general, single objective optimization problems are formulated as follows (MESSAC, 2015; RAO, 2009; FLETCHER, 2000).

$$\underset{x}{\text{minimize}} f(x) \quad (2.20)$$

Such that:

$$g(x) \leq 0 \quad (2.21)$$

$$h(x) = 0 \quad (2.22)$$

$$x_l \leq x \leq x_u \quad (2.23)$$

Where:

x : array of design variables;

$f(x)$: objective function to be minimized over the x array;

$g(x)$: inequality constraint function;

$h(x)$: equality constraint function;

x_l : lower boundary values for each element of x ;

x_u : upper boundary values for each element of x ;

The problem is to find a solution x_s that minimizes $f(x)$ while respecting constraints and boundaries. The objective function expresses with a scalar value how suitable is a given solution. Hence, it needs to be formulated so that the more suitable the evaluated solution the smaller the objective function value is.

Other than minimizing $f(x)$, x needs to satisfy constraints in which a particular parameter must be greater than, smaller than or even equal to a value. These requirements are captured in Equations 2.21 and 2.22, where $g(x)$ or $h(x)$ are functions that return an array of values that quantify how close each constraint is from its limit.

Additionally, the design variables can be restricted to a certain range (Equation 2.23) in order to capture the boundaries these variables have in real world and avoid evaluation of solutions that cannot be implemented.

The formulation presented above summarizes the problem in a form that can be delivered to most solvers available commercially. It is also useful because it helps to identify the characteristics of the optimization problem at hand. There are seven major categories in which optimization problems can be classified (MESSAC, 2015) as presented in the following sections.

2.3.1 Constrained vs. Unconstrained

An optimization problem is constrained if function $g(x)$ or $h(x)$ exist and define one or more conditions to be satisfied by the solutions x evaluated. In engineering problems, the constraint functions represent real-world restrictions that solutions need to attend. It is different from boundaries x_l and x_u because rather than limiting the values of x it makes a solution for a particular x unfeasible when it does not attend $g(x)$ or $h(x)$.

If functions $g(x)$ or $h(x)$ do not exist, the problem is unconstrained and it is limited only by boundaries of x .

2.3.2 Linear vs. Nonlinear

This category is based on the nature of funtions $f(x)$, $g(x)$ and $h(x)$. If all functions are linear, the optimization is classified as a linear programming problem. On the other hand, if any of those functions are nonlinear, the optimization is a nonlinear programming problem. Rao (2009) states that nonlinear problems may particularly be geometric if objective and constraint functions are a form of geometric progression or quadratic if objective function is quadratic and constraint functions are linear.

Classifying the problem by the nature of its objective and constraint functions is important because there are different methods available for solving efficiently each of the four types of problems discussed above.

2.3.3 Discrete vs. Continuous

Other important category is related to the type of the design variables of the x array. The problem is discrete if one or more variable is either restricted to 0 or 1, called binary programming in this case, to integer values, integer programming, or to a defined set of values, discrete programming. On the other hand, a problem is classified as continuous if all design variables belong to the set of real numbers.

2.3.4 Deterministic vs. Nondeterministic

The design variables can also be classified in terms of their deterministic nature. An optimization problem is non-deterministic or stochastic if one or more parameter, design variables or preassigned parameters, are probabilistic. If all parameter are deterministic, the problem is such as well. Although non-deterministic problems may sometimes require special techniques, some approaches apply to both problem types.

2.3.5 Single vs. Multi-objective

An optimization problem may be single or multi-objective if it has one or more objective functions to be minimized. Multi-objective problems are common in engineering applications which always have trade-offs to be considered. However, the difference between a second objective and a constraint is sometimes subtle and needs to be carefully accessed by the time the problem is stated.

2.3.6 Single vs. Multiple Minima

An optimization problem may have one or multiple optimal solutions depending on the number of local minima of the objective function. For functions with single minimum, local optimization is performed whereas for multiple minima global optimization techniques are required.

2.3.7 Simple vs. Complex

A simple optimization problem is one that can be solved easily because of its characteristics, such as:

1. The model of the system is provided or easily created;
2. There are only continuous variables;

3. It is not strongly nonlinear;
4. It only requires local optimization;
5. The computational model of the system is evaluated in seconds or minutes, not hours;
6. The number of design variables is not large;
7. All models required to assess system behavior can run in a single computer;
8. All design variables are deterministic.

These characteristics do not determine exactly if the problem is complex or simple. Instead, they provide a sense of its complexity and which of them needs to be examined carefully when considering the techniques to solve the problem.

2.4 Optimization Techniques

The number of categories presented in Section 2.3 indicates how vast is the range of optimization problems. Also, there are other classifications that were not mentioned and specific applications that increase the types of optimization problems engineers face.

Naturally, it is unlikely that for such broad range of problems a single technique would be capable of addressing them all. The reality is that many techniques for solving groups of optimization problems are available and it is important to carefully select which one will more effectively solve the task at hand. This section will present, based on the work of Venter (2010), common techniques employed in optimization problems.

The techniques will be presented in two groups: local and global optimization algorithms. Both groups have methods to address nonlinear/linear and constrained/unconstrained problems. The local algorithms generally have poor performance when solving problems with discrete design variables while global algorithm can address these problems very well. Also, global techniques are more suitable for multiple minima objective functions since they generally do not rely on gradient based methods. These and more aspects of both groups will be explored in the following sections.

2.4.1 Local Optimization Algorithms

Most local optimization algorithm are gradient-based (VENTER, 2010). These techniques rely on first and second derivatives of the objective function, and the constraint

functions if applicable, to navigate through the solution space towards the desired minimum.

These techniques are widely used because they can solve problems with large numbers of design variables with a small number of function evaluations. Also, they required little problem-specific parameter tuning, what makes them even more efficient.

On the other hand, gradient-based algorithms can only guarantee local optimum solutions and do not perform well in discrete optimization problems. Moreover, they may be susceptible to numerical noise and difficult to implement efficiently.

The implementation of these techniques usually rely on two basic concepts: the Karush-Kuhn-Tucker (KKT) conditions for optimality and the Newton Method.

The KKT conditions provide the necessary conditions for a local optimum. These can be employed to determine the optimization step direction or evaluate the current step. The conditions are summarized as follows (VENTER, 2010).

1. The optimum design point x^* is feasible;
2. At the optimum design point the Lagrangian gradient is zero

$$\nabla f(x^*) + \sum_{j=1}^m \lambda_j \nabla g_j(x^*) + \sum_{k=1}^p \lambda_{m+k} \nabla h_k(x^*) = 0 \quad (2.24)$$

where the Lagrange multipliers λ_{m+k} are unrestricted in sign and $\lambda_j \geq 0$;

3. For each inequality constraint $\lambda_j g_j(x^*) = 0$, where $j = 1, \dots, m$;

For unconstrained problems, the KKT conditions require only the gradient of the objective function to be zero.

The Newton Method is a classical gradient-based algorithm for unconstrained problems. It is derived from a second order Taylor series expansion of the objective function around the initial design point x_0 :

$$f(x) \approx f(x_0) + \nabla f(x_0)^T (x - x_0) + \frac{1}{2} (x - x_0)^T H(x_0) (x - x_0) \quad (2.25)$$

Where $\nabla f(x_0)$ is the gradient and $H(x_0)$ is the Hessian of the objective function at the initial point. The Hessian is a matrix of second-order partial derivatives that describes the local curvature of a function of many variables. This approximation can be used with the KKT conditions in order to determine the objective step direction.

While most current algorithms employ other strategies for optimization, these two concepts still are present in many of them. For instance, calculating the Hessian can be expensive for problems with many design variables, hence, some methods use approximations of the Hessian to cut down computational cost. These methods are usually referred as Quasi-Newton Methods.

One Quasi-Newton method used in unconstrained optimization is the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method. The BFGS uses an approximation of the Hessian matrix based on the information gained in previous iterations of the optimization. That is, the matrix is updated with the first-order gradient that is calculated at each iteration. Depending on the number of design variables, this method require substantial memory to store the values of the approximate Hessian matrix.

For constrained optimization, two popular algorithms for solving engineering problems are the Sequential Linear programming (SLP) and the Sequential Quadratic Programming (SQP).

The SLP algorithm strategy is to find a linear representation equivalent to the original nonlinear constrained problem. The objective and constraint functions are replaced by linear approximations around the initial design point. Then, the algorithm finds the optimum solution for the approximate problem. Next, the algorithm finds another linear representation of the functions around the obtained optimum solution. These step are repeated until convergence is achieved. This algorithm does not guarantee that a feasible solution will be found.

The SQP algorithm strategy is to use a quadratic approximation of the objective function and a linear one of the constraint functions. The algorithm finds the search direction by solving this approximate functions and uses a penalty function to determine the size of the step. As mentioned before, Quasi-Newton and KKT conditions may be used in this steps.

A new design point is found and the steps are performed iteratively until convergence. Some algorithms that vary from the SQP may use a second form of step based on the conjugate gradient of the objective function.

In summary, local optimization algorithms converge fast but require continuous smooth problems to perform well. Therefore, they are suited for problems with large number of design variables, high computational cost, that have gradient available and that do not require global minimum.

2.4.2 Global Optimization Algorithms

The main advantage of global optimization techniques is the better capacity to avoid local minimum. Despite this, they are not always the best approach because of their high computational cost.

In Figure 2.8 a function with minimums in $x = -1, 0 \text{ and } 1$ is shown. Since all three points satisfy the KKT conditions, local optimization techniques would converge to one of them, depending on each was first encountered.

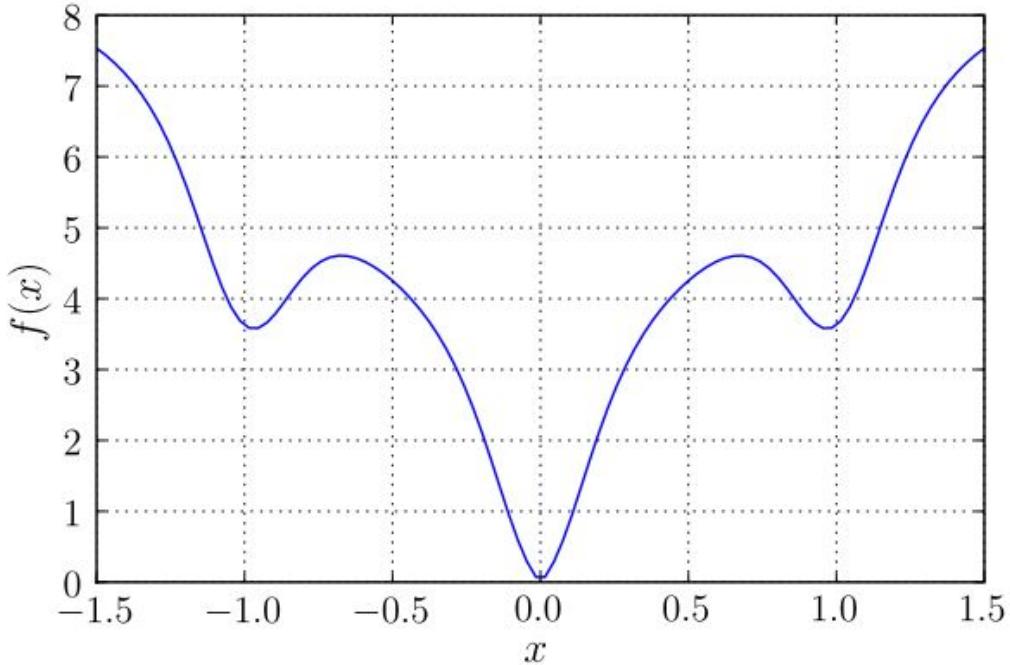


FIGURE 2.8 – One dimension multi-modal function. Source: Venter (2010)

A basic form of global optimization is the multi-start approach where multiple local searches are performed. This strategy requires a previous investigation of the design space in order to avoid computational cost by eliminating initial solutions that are too close and will probably converge to the same local minimum. As mentioned, the computational cost of this strategy is very high, in fact, it will be n times the cost of a local algorithm, where n is the number of initial solutions evaluated.

Another form of global optimization are the evolutionary algorithms. These are inspired in observation of organizations found in nature or society. The most popular evolutionary algorithm is the Genetic Algorithm (GA) which was inspired by Darwin's survival of the fittest principle.

Essentially, GA is a population based algorithm where each individual is a design solution. The population evolves iteratively and the solution set of each iteration is called

a generation.

The first generation individuals are obtained randomly or through an investigation of the design space and selection of feasible solutions. The following generations are obtained by a process that mimics the reproduction of many living beings. Individuals of a generation are ranked by fitness, given by the objective function, and the best fit are grouped in pairs and called parent designs. The parents are mixed and form children designs in a process called cross-over. Since the population size does not change, each pair makes more than one children. The cross-over is the exchange of genetic material that, in this case, is the mix of characteristics of each solution.

Additionally to the cross-over, some GA techniques also perform a mutation step which is the random change of characteristics in the individuals of the next generation (MESSAC, 2015). Mutation increases the possibility of avoiding local minimum because it introduces random changes in population that can place the new individuals in a completely different area of the solution space.

Typical stopping criteria for GA is the number of generations and minimum objective function improvement. The algorithm can yield the best individual of the last generation or its complete set of individuals. The later is useful for multi-objective optimization where a Pareto frontier may be obtained by displaying the best individuals for each objective.

In summary, global optimization techniques are useful for finding global optimum and dealing with discrete design variables. However, it requires the evaluation of a great number of design solutions and this increases its computational cost considerably. It is best suited for problems that do not have a large set of design variables, that are computationally inexpensive, that have discrete variables or problems in which the gradient do not exist.

2.4.3 Selected Algorithm

After studying the categories presented in Section 2.3 and understanding the basic characteristics of the most popular optimization algorithms the method employed in this work was selected.

In this development, the problem is to maximize an actuation system dynamic stiffness through its controller gains while attending several performance requirements. The constraints are the performance requirements and it is a nonlinear problem because the dynamic stiffness and the performance parameters do not have a linear relationship with controller gains.

Also, these gains are continuous and deterministic variables as well as the internal parameters for the later. Finally, it has a single objective which is to maximize dynamic

stiffness but the problem may be stated in a form of multi-objective optimization if there is an objective function for the stiffness at each frequency. Therefore, the problem is constrained, nonlinear, continuous, deterministic and has a single objective.

MathWorks (2015a) presents guidelines for selecting an appropriate algorithm for an optimization problem based on objective and constraint function types. Using these guidelines the interior point method and the *fmincon* solver were selected.

The interior point is a local optimization algorithm for nonlinear constrained problems. Its underlying strategy is to turn the inequality constraints into a barrier function that will penalize the objective function if constraints are not satisfied. Therefore, the minimized function is compound by the objective and the barrier functions.

Additionally, the algorithm performs two types of step at each iteration. The first, called a Newton step, is a direct step that attempts to satisfy the KKT equations for a linear approximation of the updated objective function. If this step is not possible, a conjugated gradient step is performed. In this case, the approach is to minimize a quadratic approximation of the problem subject to linearized constraints (MATHWORKS, 2015a).

The interior point algorithm is executed in MATLAB through the *fmincon* solver. The inputs and outputs parameters of *fmincon* have the following syntax:

$$[x, fval, exitflag, output] = fmincon(fun, x_0, A, b, A_{eq}, b_{eq}, l_b, u_b, nonlcon, options)$$

Where:

x: array of design variables of the optimized solution;

fval: value of the objective function of the optimized solution;

exitflag: parameter that informs triggered stopping criteria;

output: structure with information about the optimization process;

fun: objective function;

x₀: design variables initial values;

A: matrix *A* of $A * x = b$ linear inequality constraint;

b: matrix *b* of $A * x = b$ linear inequality constraint;

A_{eq}: matrix *A_{eq}* of $A_{eq} * x = b_{eq}$ linear equality constraint;

b_{eq} : matrix b_{eq} of $A_{\text{eq}} * x = b_{\text{eq}}$ linear equality constraint;

l_b : lower boundary values for each element of x ;

u_p : upper boundary values for each element of x ;

nonlcon : nonlinear constraint function;

options : structure with information to configure $fmincon$;

The output parameters inform final design variables, objective function and exit flag. Also, output contains details such as number of iterations and number of function evaluations.

Input parameters allow definition of x boundaries as well as linear and nonlinear equality and inequality constraints. The nonlinear constraint function must return each evaluated constraint organized in two arrays: one for quality and other for inequality constraints. Other input parameters are the initial values of x and the objective function, that must return a scalar value. Finally, options allow the configuration of stopping criteria tolerances, optimization algorithm, interface and others.

3 Actuation System Model

This chapter presents an overview of the actuation system model created by Constantino (2010) and improved by Ballesteros (2015) used in this development. Further details can be found in one of the mentioned references.

The architecture of the hydraulic actuator modeled is shown in Figure 3.1. The model considers check valves, an electro-hydraulic servo valve, anti-cavitation valves and the actuator piston.

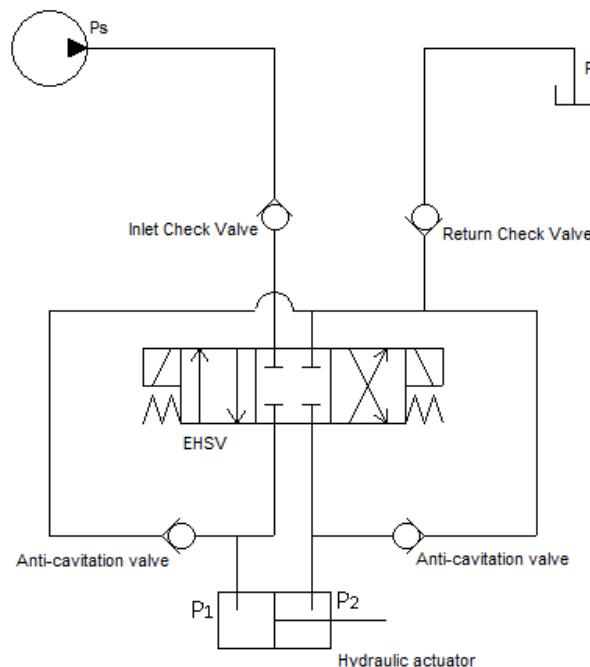


FIGURE 3.1 – Actuator hydraulic schematic. Source: Ballesteros (2015)

The actuator servo valve considered in the model is a two-stage four-way electro-hydraulic servo valve, represented in Figure 3.2.

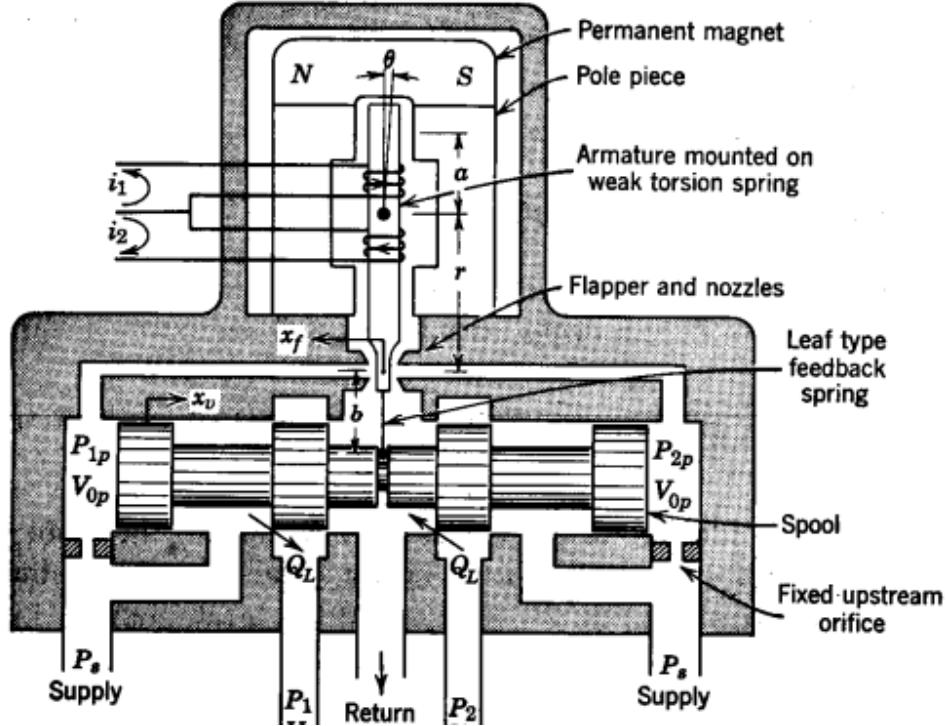


FIGURE 3.2 – Two-stage EHSV schematic. Source: THAYER (1965)

The following equations derived by Ballesteros (2015) represent a simplified version of the non-linear system.

$$\ddot{x}_v = \frac{1}{A_s K_r} (K_l K_2 w_{ne}^2 i - 2\xi_e w_{ne} A_s K_r \dot{x}_v - A_s K_r w_{ne}^2 \dot{x}_v - K_2 K_w w_{ne}^2 x_v) \quad (3.1)$$

$$\dot{P}_1 = \frac{\beta_c}{V_{01} + A_p x_p} (K_{CD}(\sqrt{P_S - P_1}(x_v x_w + A_{leak}) - \sqrt{P_1 - P_T} A_{leak} - A_{ext} \sqrt{P_1} - A_{int} \sqrt{P_1 - P_2}) - A_p \dot{x}_p) \quad (3.2)$$

$$\dot{P}_2 = \frac{\beta_c}{V_{02} - A_p x_p} (A_p \dot{x}_p + K_{CD}(\sqrt{P_S - P_2} A_{leak} - A_{ext} \sqrt{P_2} + A_{int} \sqrt{P_1 - P_2} - \sqrt{P_2 - P_T} (x_v x_w + A_{leak}))) \quad (3.3)$$

$$\ddot{x}_p = \frac{1}{m_p} (A_p (P_1 - P_2) - B_p \dot{x}_p) \quad (3.4)$$

Defining $\Delta P = P_1 - P_2$, the equations above describe the following non-linear state space with state vector X with input U and output Y :

$$\dot{X} = AX + BU \quad (3.5)$$

$$Y = CX + DU \quad (3.6)$$

$$X = \begin{bmatrix} x_v \\ \dot{x}_v \\ \ddot{x}_v \\ \Delta P \\ x_p \\ \dot{x}_p \end{bmatrix}$$

$$U = i$$

$$Y = \begin{bmatrix} x_v \\ \Delta P \\ x_p \end{bmatrix}$$

Where the matrices A, B, C and D are obtained from the terms of the system equations.

3.1 Simulink Model

Figure 3.3 shows the actuation system interfaces and the support blocks necessary to simulate the system.

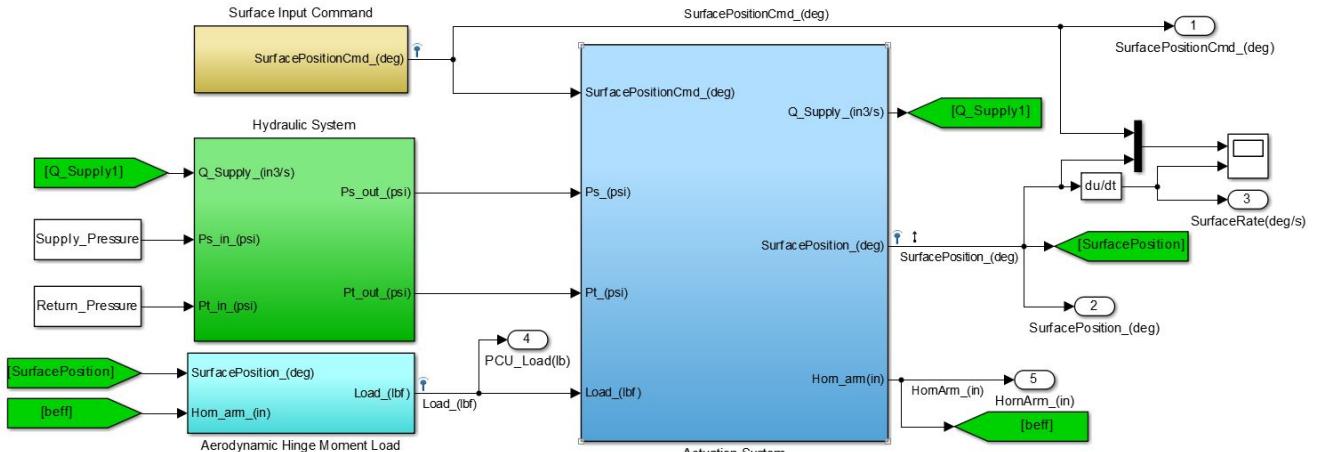


FIGURE 3.3 – Actuation System and Interfaces Simulink Model

The actuation system receives the reference surface position command in degrees. This input can be configured as a null command, a repeating sequence, a sine wave, a step command, a square pulse or a square sine.

Hydraulic power is also provided through the supply and return pressure inputs. The hydraulic system model is configured to provide a nominal supply pressure of 3000psi and 150psi at the return line. Since the fluid inertia has been considered, these values may fluctuate depending on the fluid flow consumed by the actuator.

Another input of the actuation system is the aerodynamic load at the actuator piston. This parameter may be configured as a sine wave, used for the dynamic stiffness test, a linear load as a function of the surface position and a constant value.

The actuation system outputs are the consumed hydraulic fluid flow, the control surface position and the horn arm which is also considered to calculate the linear aerodynamic load.

The actuation system is divided in three parts as shown in Figure 3.4: electronic system, actuator and surface kinematics.

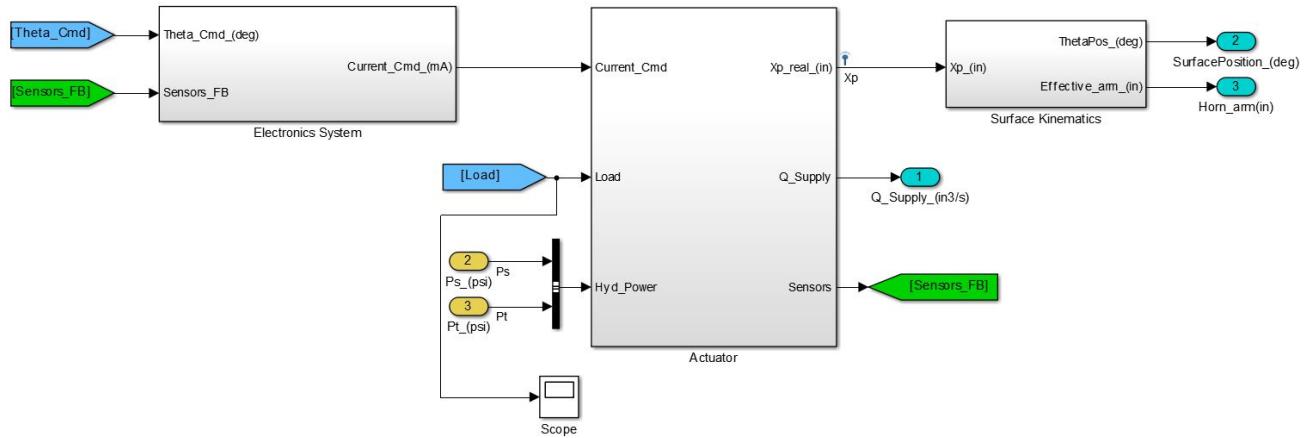


FIGURE 3.4 – Actuation System Simulink Model

The electronics system receives the surface command input and the feedback sensors from the actuator. These sensors are LVDTs that measure the piston position, the differential pressure in the cylinder and the EHSV (Electro-Hydraulic Servo Valve) displacement.

The actuator receives a current command from the electronic system, the hydraulic supply and return pressures and the aerodynamic load at the piston. These inputs are used to calculate piston position and the fluid flow consumed by the actuator. Finally, the control surface position and horn arm are calculated from the piston position.

The following sections detail the characteristics modeled in each of these three parts.

3.1.1 Electronic System

The electronic system model is shown in Figure 3.5.

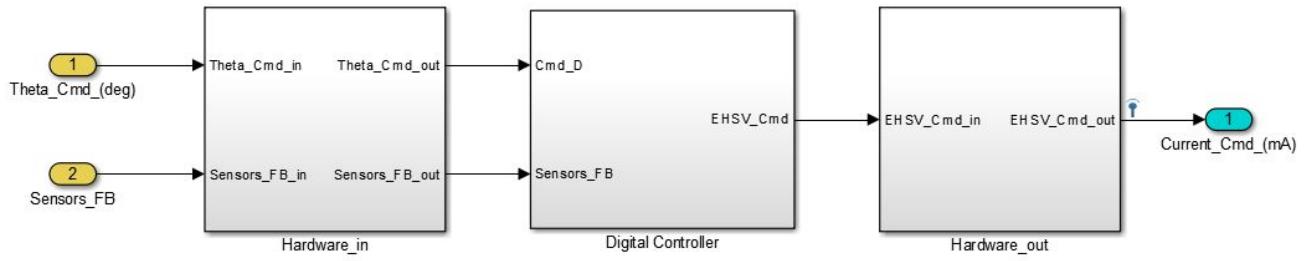


FIGURE 3.5 – Electronic System Simulink Model

The first part of the electronic system is an analog-to-digital converter which is necessary because a digital controller has been modeled and the input is analog. At this step, the signal sampling rate is set and a conversion noise is added.

The position command received by the controller is converted from degrees to millimeters prior to the subtraction of the piston position received from the LVDT sensor. An error signal is generated from this operation and sent to a PID controller that calculates a current command to be sent to the EHSV. This output is limited due to the saturation observed in real controllers.

The digital PID controller is implemented as shown in Figure 3.6

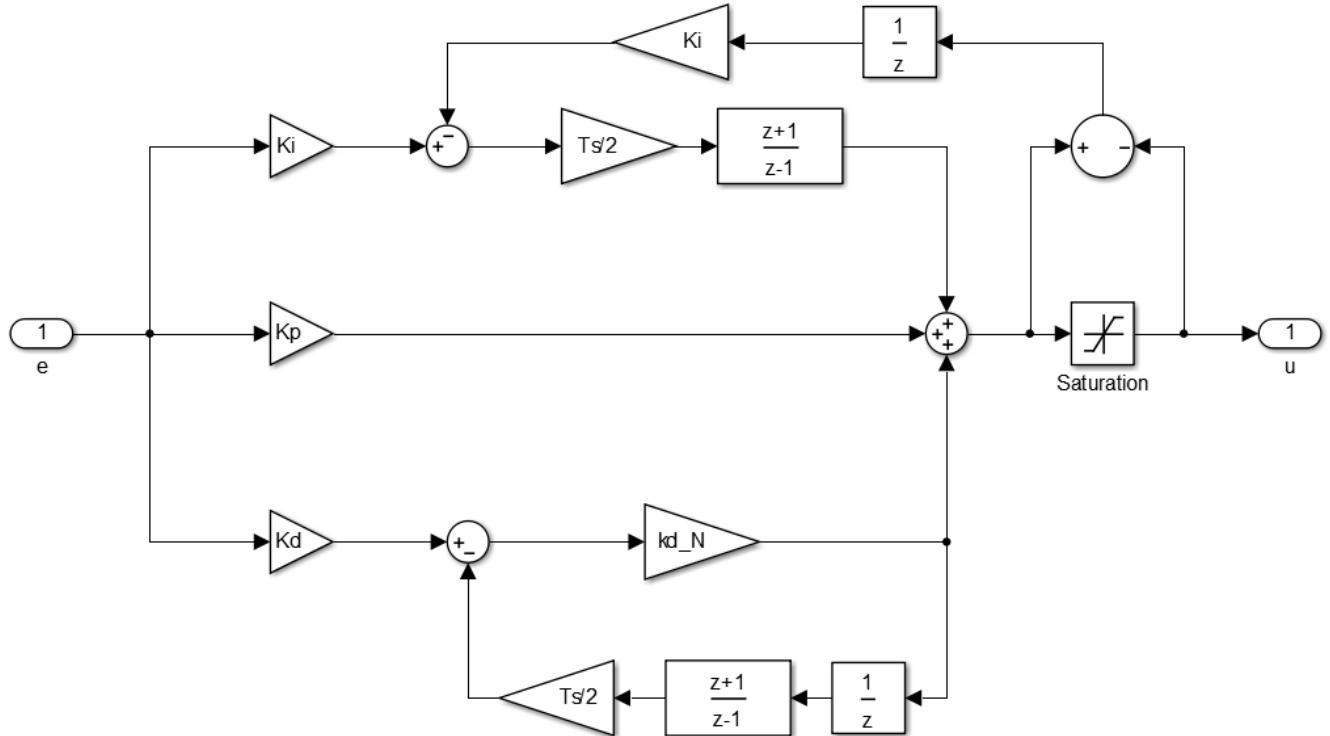


FIGURE 3.6 – Digital PID Controller Model

Finally, the digital current value is converted to an analog signal with power to drive

the first stage of the EHSV.

3.1.2 Actuator

The actuator model is composed by valves, cylinder and sensors dynamics as shown in Figure 3.7.

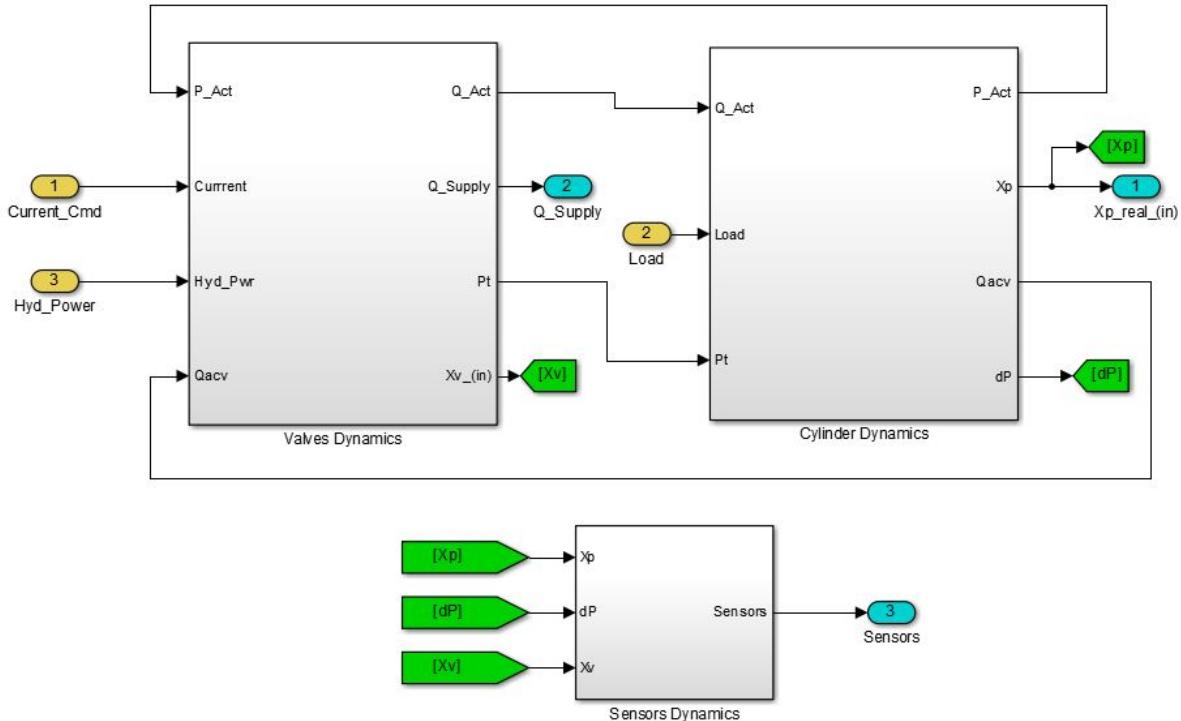


FIGURE 3.7 – Actuator Simulink Model

The first stage of the actuator is a set composed of two inlet check valves and an EHSV as shown in Figure 3.8. The inlet check valves guarantee that no fluid will flow to the supply line and from the return line. The bulk modulus effect is considered at supply and return hydraulic lines.

The EHSV is responsible for delivering fluid flow to the cylinder depending on the current command, hydraulic power available and the pressure at both cylinder chambers. Several non linearities are considered in the EHSV first stage, such as torque motor current saturation and flapper and spool displacement limits.

Based on the spool displacement generated at the EHSV first stage, the second stage calculates the flow between hydraulic lines and cylinder chambers as well as flow resulting from leaks.

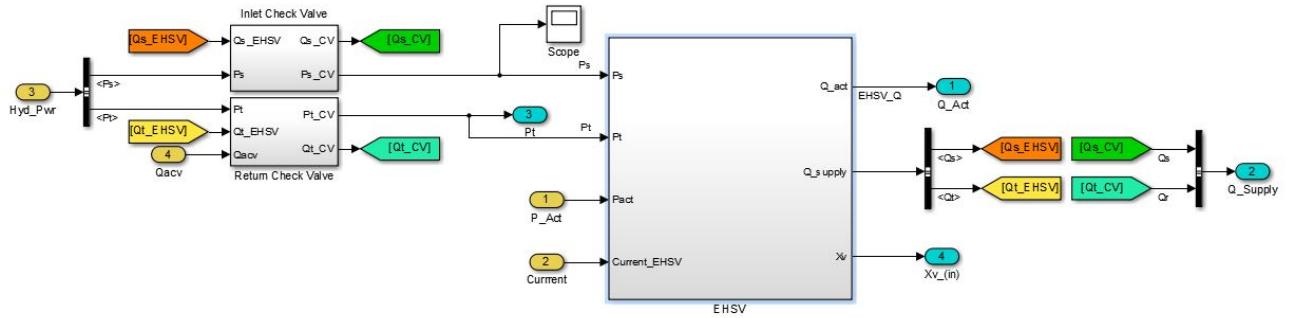


FIGURE 3.8 – Actuator Valves Simulink Model

The second stage of the actuator is composed by fluid and piston dynamics (Figure 3.9). The fluid dynamics stage uses the flow at the cylinder chambers and the piston position to calculate the pressure at each chamber, also considering the bulk modulus effect in these volumes. Anti cavitation valves are also implemented and the flow from them is considered when calculating the flow to the return line.

The position of the piston is obtained at the piston dynamics stage based on the forces acting on it, that are: hydraulic force due to the pressure difference between cylinder chambers, damping force due to fluid viscosity, friction force at the seal and the force due to the aerodynamic load at the control surface.

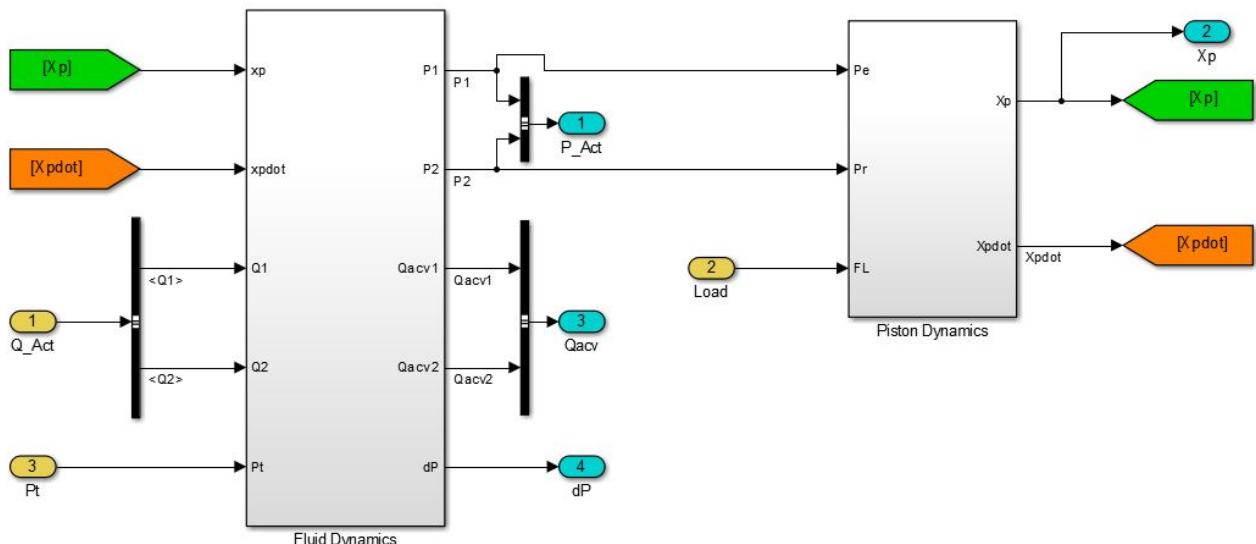


FIGURE 3.9 – Actuator Cylinder Simulink Model

The piston position, spool position and delta pressure at the cylinder are measured by LVDTs (Figure 3.10). The dynamics of these sensors are also modeled with the

actuator considering the sensor's bandwidth, range, error and hysteresis. The measured signals are fed back to the digital controller.

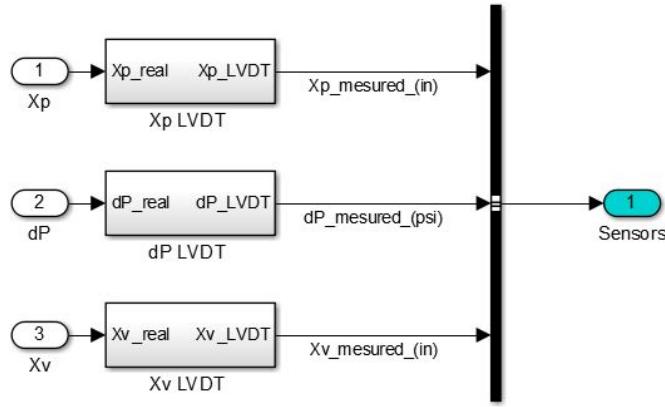


FIGURE 3.10 – Actuator LVDT Sensors Simulink Model

3.1.3 Surface Kinematics

The surface kinematics block only performs the conversion between linear piston movement, from the actuator, to angular control surface movement. The effective arm is also calculated to be provided to the aerodynamic load block.

Originally, the surface kinematics also considered a dynamic rotational model (CONSTANTINO, 2010) but this feature was removed by Ballesteros (2015) since it was not required for actuator dynamic stiffness performance evaluation.

However, the surface dynamics influence other performance characteristics such as step and frequency response. For example, when considering the control surface inertia overshoot tends to increase in the step response. Despite this, these dynamics were not considered in order to allow a fair comparison between the results achieved by Ballesteros (2015) and in this development.

4 Dynamic Stiffness Optimization

This chapter presents the optimization development. Topics such as algorithm configuration, choice of design variables and definition of constraints and objective function will be addressed.

4.1 Problem Definition

The problem definition is to find a controller design that provides increased actuation system dynamic stiffness when compared to an initial provided controller while respecting the performance requirements of the system. In this work, four classical controllers were considered, hence, the optimization was performed for each of them.

As explained in Section 2.4.3 this optimization is constrained, nonlinear, continuous, deterministic and single objective. Despite the small number of design variables, this is a complex problem because of its high computational cost which will be discussed in the following sections.

The problem is described by Equations 2.20 to 2.23. The objective function $f(x)$ is described in section 4.4, the constraint function $g(x)$ is presented in Section 4.3 and the design variables x , its boundaries and other optimization parameters are presented in Section 4.2. The equality constraint function $h(x)$ does not apply to this problem, hence it will not be considered.

An algorithmic program was developed to structure the optimization problem solution in an organized and user friendly manner. A high level flowchart of the program is presented in Figure 4.1.

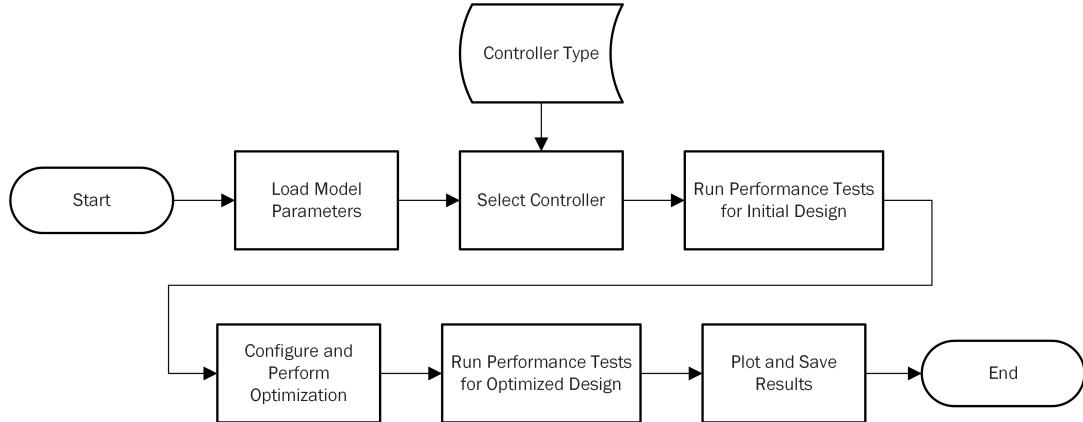


FIGURE 4.1 – Program Flowchart

The program starts by clearing the workspace and initializing parameters of the model presented in Chapter 3. These include the physical dimensions valves, cylinder and rod as well as hydraulic fluid characteristics and simulation parameters.

After that, the user selects the controller type to be evaluated using a dialog box. Next, the initial solution performance is evaluated with dynamic stiffness, step response and frequency response tests and the results saved for future comparison with optimized design.

The next step is configuring and performing the optimization itself, where the chosen solver, *fmincon*, iteratively updates controller gains while evaluating objective and constraint functions. After optimization, dynamic stiffness, step response and frequency response tests are performed with the optimized controller and finally the results are plotted into graphs and saved in the hard drive.

The following sections will present in detail the optimization configuration, the constraints and the objective function.

4.2 Optimization Settings

This section will present the settings used to solve the optimization problem.

One of them is the optimization method employed. In this case, as discussed in Section 2.4.3, the Interior Point algorithm was selected. This method needs few function evaluations to converge and is therefore suitable for problems with long execution time.

The design variables x defined for this problem are the gains of the control architecture being evaluated. This is a natural definition that comes from the parametric study performed by Ballesteros (2015) which observed the influence of such gains in the actuation system dynamic stiffness response.

The range of these design variables that will be considered as possible solutions to the problem can be bounded as per Equation 2.23. Through the correct definition of these boundaries it is possible to improve algorithm convergence and also to reduce execution time since the space of possible solutions will be smaller.

These boundaries have been defined considering the mentioned study performed by Ballesteros (2015). It evaluated the performance of the actuation system for a broad range of proportional, integral and derivative gain values. The gain values that yielded unstable actuation systems or ones that did not comply with performance requirements were discarded. The lower and upper boundaries selected are shown in Table 4.1.

TABLE 4.1 – Classic Controller Gain Boundaries

Parameter	Lower Boundary (x_l)	Upper Boundary (x_u)
K_p	10	100
K_i	0	5
K_d	0	5

Also, stopping criteria tolerances were defined for the design variables, constraints and objective function. Based on these definitions, the optimization program stops when the changes in these parameters are smaller than the tolerance values. These values were defined empirically during development because they depend on the magnitude of these parameters. Table 4.2 shows their final values.

TABLE 4.2 – Stopping Criteria Parameters

Parameter	Tolerance Value
Objective Function	$1e - 6$
Design Variable	$1e - 4$
Constraint	$1e - 4$

4.3 Optimization Constraints

The constraint function $g(x)$ is designed to incorporate the time and frequency domain performance requirements that are usually applied to rudder surface actuation systems. Table 4.3 shows the considered performance parameter as well as their respective requirements.

TABLE 4.3 – Constraints Parameters

Performance Parameter	Requirement
Settling Time (ms)	< 1100
Steady State Error (%)	< 1
Overshoot (%)	< 10
Minimum Average Rate ($^{\circ}/s$)	> 32
Maximum Average Rate ($^{\circ}/s$)	< 36
Closed-Loop Gain Allowance (dB)	> 10
Closed-Loop Phase Allowance ($^{\circ}$)	> 45
Closed-Loop Maximum Peak (dB)	< 0.5

The requirements used in this work are similar to those proposed by Ballesteros (2015). However, there are two differences: the maximum magnitude requirement was introduced and in this work, the settling time requirement is 1100 ms, instead of 850 ms.

The maximum magnitude requirement was introduced to limit eventual peaks in the magnitude frequency response. High peaks are not desired because they are related to poor dynamic stiffness at the frequency of the peak (BALLESTEROS, 2015).

To obtain the settling time, the step time instant is subtracted from the instant when model response reaches and remains within $\pm 2\%$ of its steady-state value. The settling times presented in the results of Ballesteros (2015) subtracted twice the step time instant instead of one. This reduced all settling times by 250ms what made the 850ms requirement of that work feasible.

To enable a fair comparison with previous results, the settling time requirement considered in this development is 1100 ms, which is the previous requirement plus 250ms of the step time instant.

The step response is obtained for a maximum deflection scenario, in which the actuator initial position is neutral and the final position is the nominal mechanical stop of the piston. This condition is more conservative because the actuator load is always against the movement and thus requiring more effort from the actuation system. The numerical definition of the step will be presented in the following sections.

The frequency domain performance parameters are all obtained from the closed-loop frequency response which is the response of the complementary sensitivity function (Equation 2.7). As explained in Section 2.4.3, it is not practical to specify the loop transfer function parameters and therefore the industry approach will be followed in this development.

The frequency response is obtained through model simulations with a sinusoidal input. The response is obtained from the surface command in degrees to the actuator piston posi-

tion converted from millimeters to degrees. Also, the surface degrees command amplitude must be one that does not lead to a saturation of the servo valve current command.

The constraint parameters are obtained by simulating step and frequency response tests in the actuation system Simulink model. These tests are described in the following sections.

4.3.1 Step Response Test

The step response test is simple and its execution sequence is described in Figure 4.2.

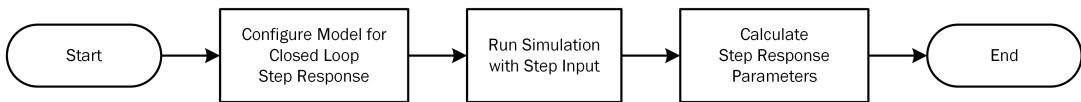


FIGURE 4.2 – Step Response Test Flowchart

Initially, the model is configured to match the following test conditions:

- a) Single actuator connected to rudder surface;
- b) Initial position = 0° ;
- c) Step amplitude = 30° ;
- d) Opposing hinge moment load varying linearly from 0 N.m to 3050 N.m;
- e) Maximum hydraulic supply pressure of 3000 psi;
- f) Maximum hydraulic return pressure of 150 psi;
- g) At $-15^\circ C$ ($5^\circ F$) fluid temperature.

The test is performed with one actuator because the surface is required to maintain performance even with the other actuators failed. Also, the opposing hinge moment load profile refers to equivalent aerodynamic forces in rudder surfaces for the actuator sizing of this study.

Additionally, the step amplitude is 30° because that is the common deflection range of a rudder surface. It is important to evaluate time domain performance in a maximum deflection scenario due to critical events that may occur that require prompt surface response such as the loss of one engine or other thrust asymmetry scenario.

Typical fluid temperature operational envelope of the actuation system is between $-15^\circ C$ and $100^\circ C$. Despite this, the step response is evaluated only at $-15^\circ C$ because

the actuation rate decreases when temperature falls what makes this condition the most conservative to evaluate settling time, steady state error and the actuation average rate.

At higher temperatures, fluid density and bulk modulus decrease and each one has a different effect in actuation rate. Lower fluid density leads to higher flow and as a consequence higher pressure at cylinder chambers, what increases actuation rate. On the other hand, lower bulk modulus increases fluid compressibility, leading to less pressure for the same volume and flow, what decreases actuation rate.

However, the effect of decreasing fluid density is greater than the effect of a lower bulk modulus and, therefore, actuation rate will increase with fluid temperature. Figure 4.3 illustrates this behavior.

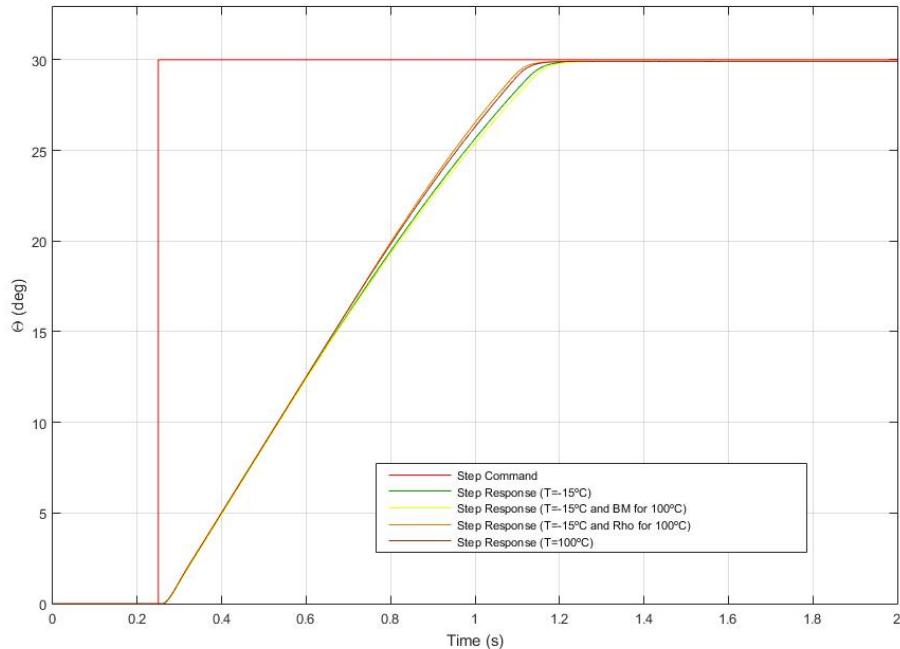


FIGURE 4.3 – Fluid Density and Bulk Modulus Contribution to Step Response Actuation Rate

Because higher actuation rates lead to overshoot, it is more conservative to evaluate this parameter at higher fluid temperatures. However, overshoot is associated with dynamic stiffness reduction (BALLESTEROS, 2015) and, since the optimization will focus in increasing dynamic stiffness, it is expected that solutions that increase overshoot are avoided by the optimization algorithm. Also, since the actuation system model does not consider the control surface inertia only small overshoots are expected.

Therefore, evaluation at 100°C is not performed to save execution time. Nevertheless, the final design step response will be obtained for both envelope corner temperatures, assuring average actuation rate and overshoot requirement compliance.

Additionally, it is advised to evaluate these constraints at $100^{\circ}C$ in a future work if the control surface dynamics is considered. In this case, the fluid temperature and the inertia will lead to an overshoot increase that will need to be avoided during optimization.

After parameter configuration, the test is performed and the following performance parameters are calculated:

1. Settling time $t_{ss} < 1100\text{ms}$;
2. Steady state error $< 1\%$ of actuator full stroke;
3. Overshoot $< 10\%$ of actuator full stroke;
4. Average rate $> 32^{\circ}/s$;
5. Maximum rate $< 36^{\circ}/s$.

4.3.2 Frequency Response Test

The frequency response is obtained through a series of model simulations, one for each evaluated frequency, and further processing of the data generated by these simulations. Figure 4.4 shows the flowchart of this test.

First, the model is configured to obtain a closed-loop frequency response in the following conditions:

- a) Closed-loop control configuration;
- b) Measured from the surface angular position command in degrees provided at the actuator controller digital input to the Actuator Ram Position Feedback converted to surface angular position in degrees;
- c) Single actuator connected to rudder surface;
- d) Sinusoidal commands of 0.5° amplitude;
- e) Without the presence of aerodynamic load and backlash;
- f) For an actuator within nominal tolerances;
- g) At $100^{\circ}C$ ($212^{\circ}F$) fluid temperature.

The fluid temperature is set to 100°C , which is the highest envelope temperature. Simulations performed during this development show that at this temperature the gain

and phase allowances are smaller than these allowances at -15°C , thus evaluating this condition is a conservative approach for actuator design.

Next, the sine input frequency to be evaluated is selected and the simulation duration period is calculated. For frequencies up to 1 Hz, the duration spans for 3 complete input cycles and, for frequencies between 1 Hz and 35 Hz, the number of cycles covered by the simulation is 3 times the evaluated frequency. For instance, when evaluating the actuation system at 20 Hz, the simulation spans for 60 complete cycles.

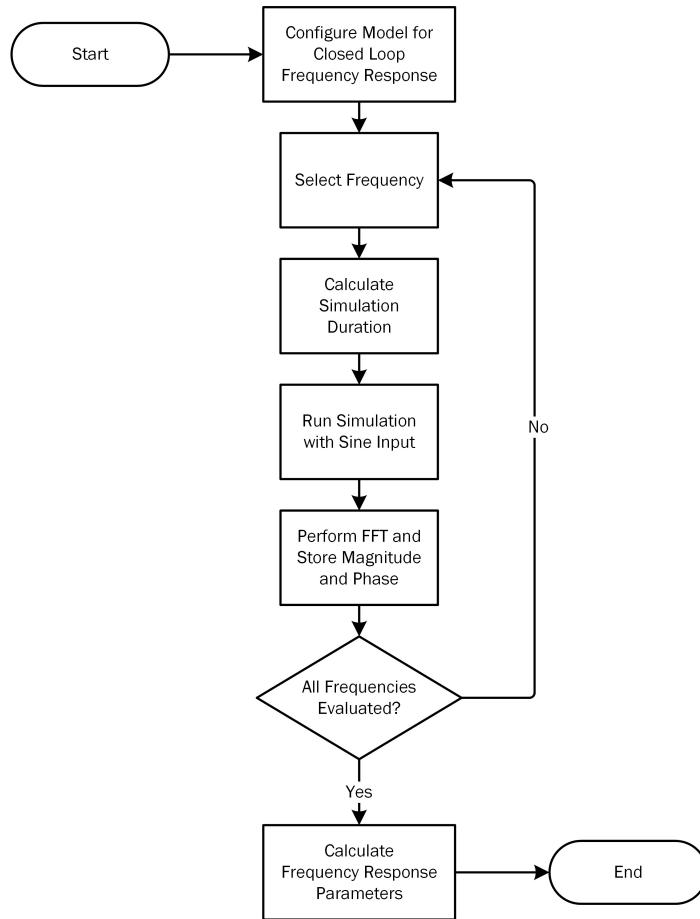


FIGURE 4.4 – Flowchart of Baseline Frequency Response Test

After that, the simulation is performed and the sine input and surface position values are stored. This information is processed in the next step, when a Fast Fourier Transform (FFT) is performed for both simulation input and output considering all simulation time. The transformed vectors are manipulated and at this stage the magnitude and phase at the excitation frequency are stored. The last steps are repeated for all frequencies of interest until the magnitude and phase responses are complete. Finally, the performance parameters are obtained from these responses through Equations 2.12, 2.18 and 2.19.

The execution time of this test is approximately 12 minutes but it depends on many factors such as the computer configuration, applications in execution and the model itself.

During this development, when the frequency response constraints were implemented, it was noticed that this computational cost would lead to days of optimization execution what would eventually reduce the advantages of performing the optimization.

Execution time is critical in optimization algorithms because of their iterative nature. The frequency response test described above, referred later on as *baseline*, is executed several times in a single run of the optimization program and its execution time can represent more than two hours altogether. If global optimization techniques are used, the frequency response algorithm can represent days of execution time.

To minimize this computational cost, other methods for obtaining the frequency response were studied as presented in Appendix A. The investigation on alternative methods resulted in a modification of the *baseline* frequency response test that is presented below.

4.3.2.1 Modified Frequency Response Test

Since the execution time of the baseline test is mostly the time to execute all simulations, their duration was revisited. Instead of calculating simulation's length only from input signal cycles, a non variable time parameter was introduced with the aim to capture model dynamics settling time. The duration of each simulation was calculated with Equation 4.1.

$$T_{\text{sim}} = T_{\text{dyn}} + K_{\text{int}} \times T_p \quad (4.1)$$

T_{sim} is the simulation duration for each frequency whereas T_{dyn} is the model dynamics settling time. This constant parameter was obtained empirically from measuring the amplitude of actuator surface position in simulations with sine input for several frequencies which yielded a 0.5 seconds dynamics settling time.

Parameter K_{int} is the number of integration cycles for the respective frequency and T_p is the frequency's period. K_{int} was selected to minimize simulation time but in a way that did not jeopardize the frequency response. Frequencies 0.1, 0.5 and 1 Hz were integrated for only one cycle while three cycles were considered for 2 and 3 Hz and nine cycles for frequencies from 4 to 35 Hz.

Figure 4.5 shows an example of the actuation system response for a 20 Hz sine input in which the orange curve shows approximately how the output amplitude behaves. It can be observed that dynamics of the model in the beginning of the simulation disappear after approximately 0.2 seconds, when the actuator response reaches a steady value. Even though the dynamics settling time is less than 0.2 seconds, a generous margin was allowed

and a settling time of 0.5 seconds was considered.

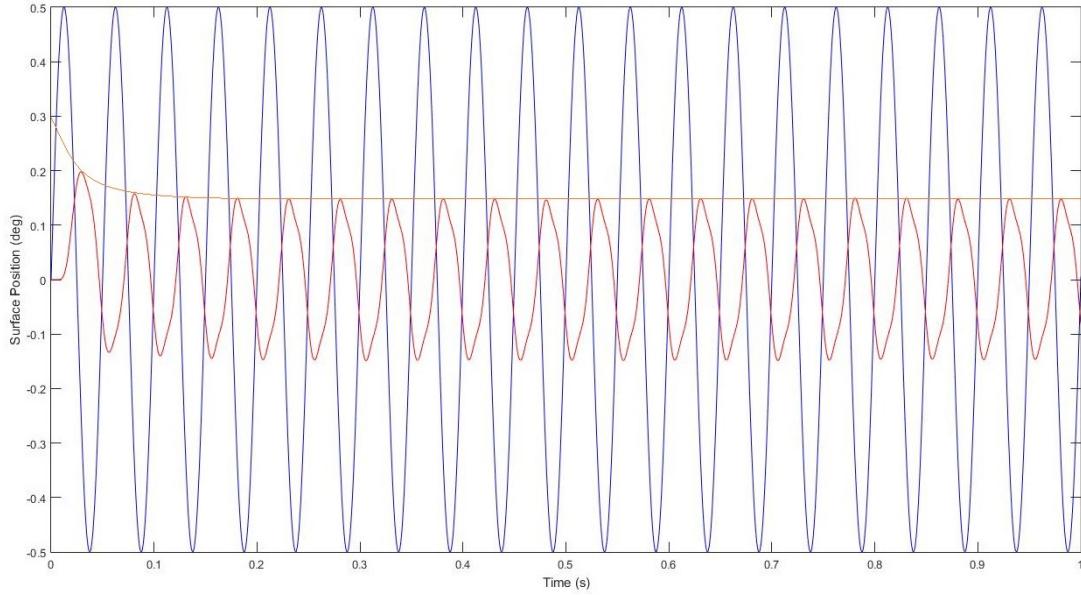


FIGURE 4.5 – 20 Hz Sine Input System Response

Figure 4.5 also shows that for a pure sinusoidal input the output of the model is not a pure sinusoidal wave and therefore the system output is distorted and has harmonic components. This distortion is clearly observed in Figure 4.6 which shows three cycles of the input after the dynamics have settled.

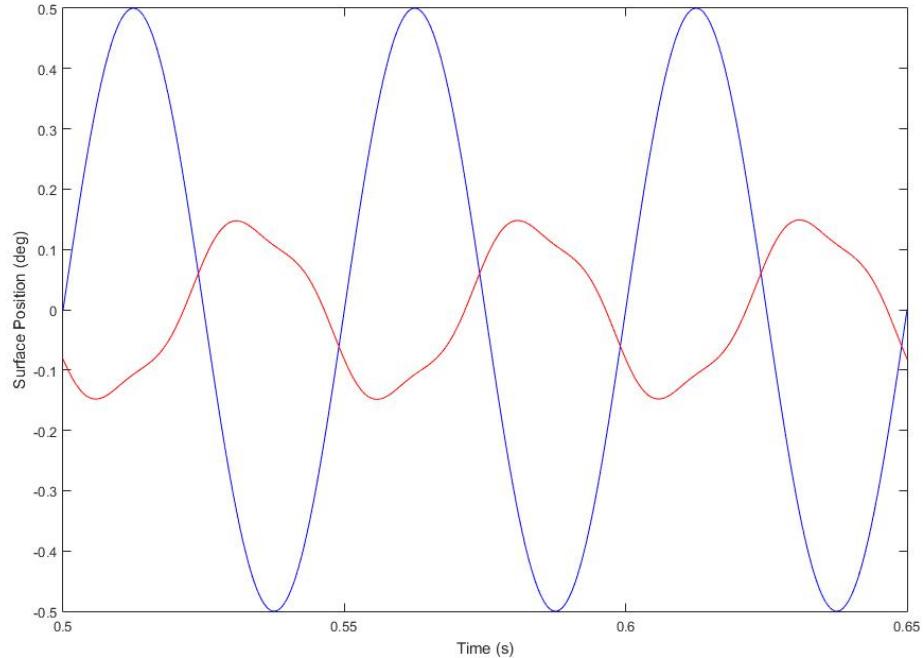


FIGURE 4.6 – 20 Hz Sine Input System Response

Since the system output is distorted and therefore have harmonic components in its spectrum, it is required an analysis to assess if it is possible to consider only the component of the fundamental frequency in the calculation of the magnitude and phase of the output. The non-linearity at each frequency was evaluated by observing the respective THD as shown in Appendix B.

The FFT performed in the inputs and output signals did not consider the model dynamics settling time. Instead, it was performed for K_{int} complete cycles from instant T_{dyn} to the end of the simulation. A possible cause for the difference observed between the baseline and modified tests is that the baseline test considers all frequency cycles simulated, including those in the dynamics settling period whereas the modified test considers only the cycles after the dynamics have settled. When the first cycles are contained in the FFT data the result is affected by the different amplitude value observed in them. Both tests consider whole cycles for FFT data selection.

A comparison between the baseline and modified tests is presented in Figure 4.7. The figure shows that magnitude and phase responses for both tests match in most frequencies. Even though there is a difference between the curves at around 13 Hz, the overall response of the modified test is suitable for constraint evaluation purposes. It is important to remark that the baseline test is still used to evaluate initial and optimized controllers prior and after optimization.

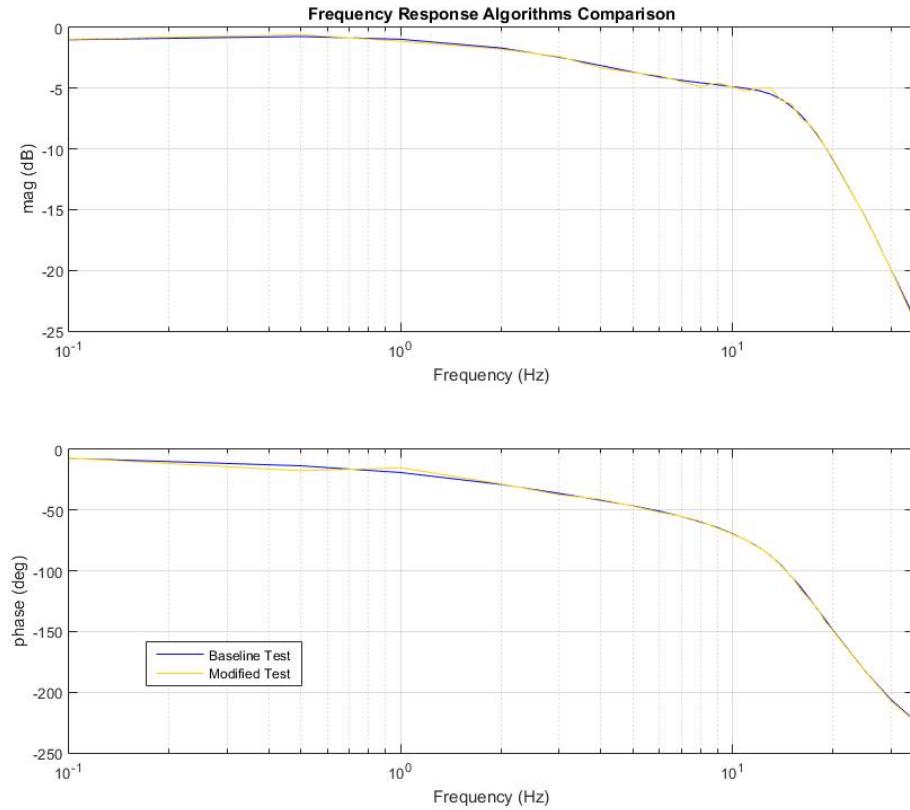


FIGURE 4.7 – Baseline and Modified Frequency Response Test Results Comparison

Table 4.4 shows that the closed-loop gain and phase allowances for both tests are the same and that bandwidth slightly increases. Also, execution time reduced by approximately 60%, falling from 12.6 to 4.9 minutes.

TABLE 4.4 – Baseline and Modified Frequency Response Test Results Comparison

Constraint	Baseline	Modified
Closed-Loop Gain Allowance (dB)	13.0	13.0
Closed-Loop Phase Allowance (°)	<i>Inf</i>	<i>Inf</i>
Closed-Loop Bandwidth (Hz)	5.8	6.0
Execution Time (min)	12.6	4.9

Therefore, the modified test is a valid alternative to obtain the closed-loop frequency response of the actuation system since it yields a fairly similar response in much less time. This alternative test was used solely to evaluate optimization constraints. To evaluate the frequency performance of initial and optimized controllers the baseline test is still employed.

4.4 Objective Function

The objective function $f(x)$ is responsible for conveying the fitness of a solution to a scalar value that will be minimized by the optimization algorithm. The output of the objective function is obtained by performing a dynamic stiffness test and calculating the cost function. These steps are described in the following sections.

4.4.1 Dynamic Stiffness Test

The dynamic stiffness is obtained similarly to the frequency response, through simulation of the model for each evaluated frequency and further processing of the data generated by these simulations. Figure 4.8 shows the flowchart of this test.

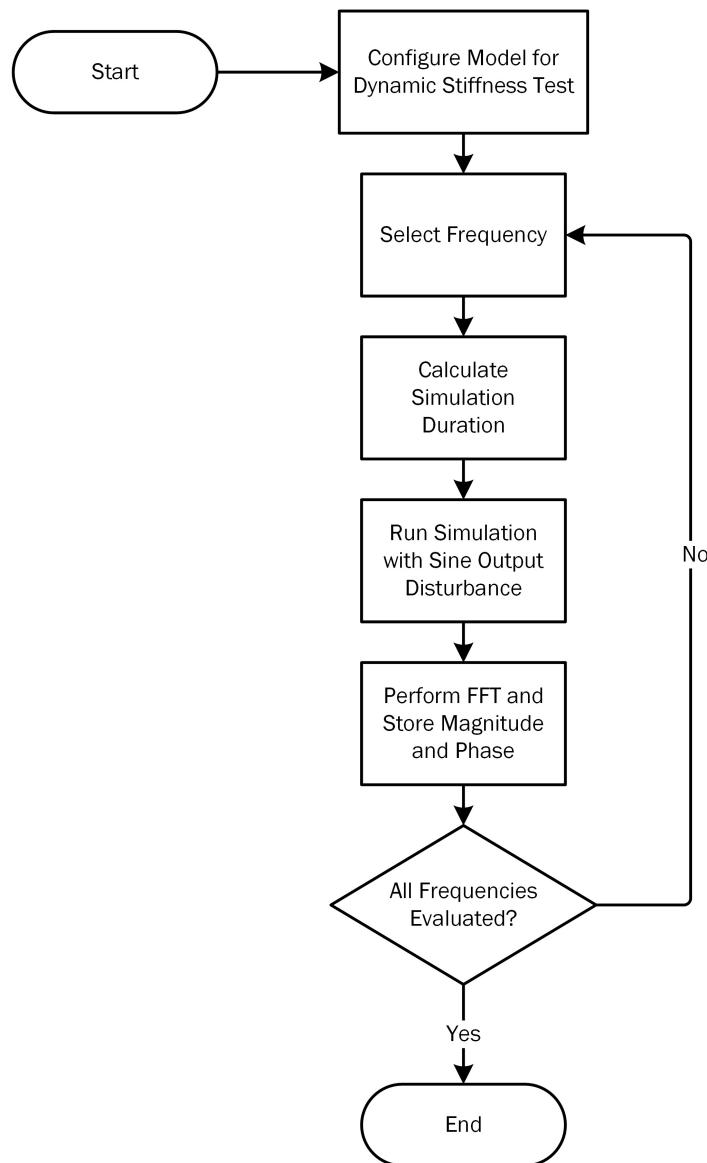


FIGURE 4.8 – Dynamic Stiffness Test Flowchart

First, the model is configured to the following test condition:

- a) Single actuator connected to rudder surface;
- b) Initial position = 0° ;
- c) Position command = 0° ;
- d) Sinusoidal aerodynamic load disturbance between 0.1 and 0.5 of actuator stall load;
- e) Maximum hydraulic supply pressure of 3000 psi;
- f) Maximum hydraulic return pressure of 150 psi;
- g) At $100^\circ C$ ($212^\circ F$) fluid temperature.

The test is performed with the highest fluid temperature of the operational envelope because, since the bulk modulus decreases when temperature rises, the fluid compressibility increases and that leads to lower dynamic stiffness. Since there is no surface command, the flow in the cylinder chamber is not substantial and therefore the fluid density does not have a major influence in the response. Therefore, it is more conservative to evaluate actuator dynamic stiffness at $100^\circ C$. This behavior will be evident in chapter 5.

Next, the aerodynamic load disturbance frequency to be evaluated is selected and the simulation duration period is calculated with the same method used in the baseline frequency response test. For frequencies up to 1 Hz, the duration spans for 3 complete sine cycles and, for frequencies between 1 Hz and 35 Hz, the number of cycles covered by the simulation is 3 times the evaluated frequency.

After that, the simulation is performed and the sinusoidal load disturbance and the ram position values are stored. This information is processed in the next step, when a FFT is performed for both signals. The transformed vectors are divided by each other and the real part of the resulting value at the excitation frequency yields the dynamic stiffness. The last steps are repeated until the dynamic stiffness is obtained for all frequencies of interest.

The execution time of the presented dynamic stiffness test is approximately 10 minutes which is a considerable computational cost. However, because this test is similar to the frequency response test, it is possible to implement the same modifications of the frequency response test in the dynamic stiffness test.

Hence, the dynamic stiffness test was modified and the duration of each simulation is now calculated with equation 4.1. This modification also excludes the effect of the model dynamics settling time in the actuator dynamic stiffness response. The same values of K_{int} for each frequencies were used.

A comparison between both tests is shown in Figure 4.9. The yellow curve shows the dynamic stiffness obtained from the modified test which is fairly similar to the one from the original test, in blue.

Therefore, the modified test can be used to evaluate the dynamic stiffness during optimization in order to reduce execution time which was reduced by approximately 43%. Despite this, the original dynamic stiffness test is still employed to obtain this characteristic of the initial and final solutions of the optimization.

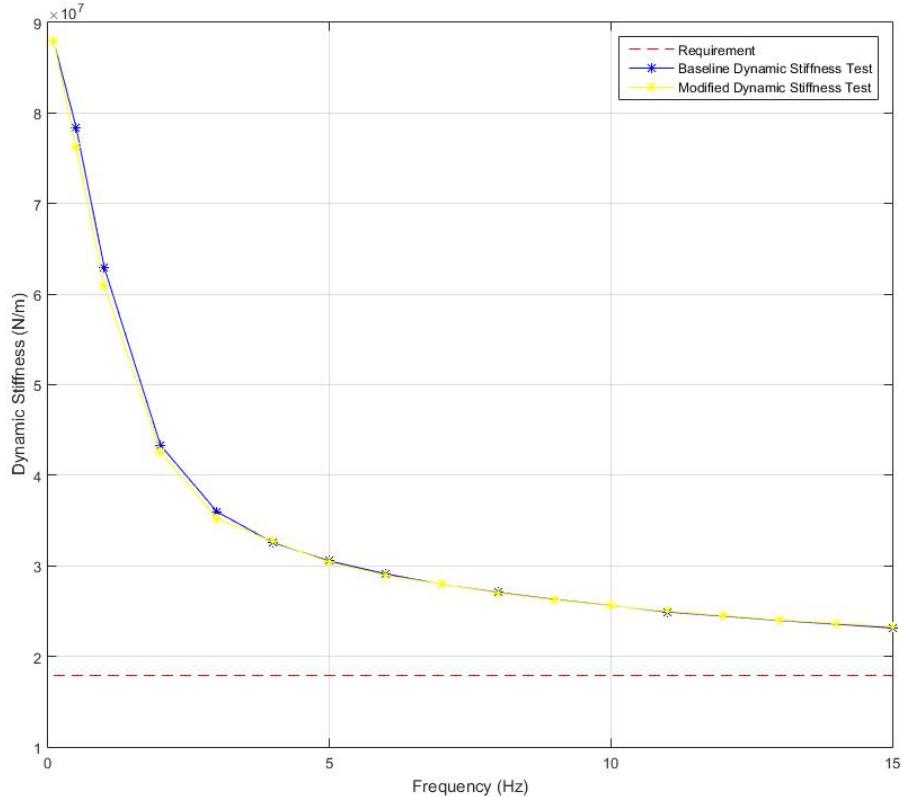


FIGURE 4.9 – Baseline and Modified Dynamic Stiffness Test Comparison

4.4.2 Cost Function

After obtaining the dynamic stiffness response it is possible to calculate the cost function. Initially, a simple approach was considered as described by equations 4.2 and 4.3.

$$f_1(x) = -(J_1(x) + J_2(x) + \dots + J_i(x)) \quad (4.2)$$

$$J_i(x) = K_{\text{act}}(x, w_i) - K_{\text{req}} \quad (4.3)$$

Where:

w_i : i^{th} frequency evaluated;

$J_i(x)$: partial cost function related to the i^{th} frequency evaluated;

K_{act} : actuation system dynamic stiffness for a given frequency;

K_{req} : dynamic stiffness requirement.

Function $f_1(x)$ is a sum of several partial values related to each evaluated frequency in the system. These partial values are the difference between the dynamic stiffness value and the requirement at each frequency. Because the objective function is minimized during optimization, it must return a negative value in order to be maximized, hence the minus signal before the sum.

A second, more complex, cost function was considered in this work. Function K , proposed by Andersson (2001) for multi-objective optimization in engineering design is represented as:

$$K = \left[\left(\frac{k_1}{k_{i0}} \right)^{\gamma_1} + \left(\frac{k_2}{k_{i0}} \right)^{\gamma_2} + \dots + \left(\frac{k_i}{k_{i0}} \right)^{\gamma_i} \right] \times [(1 + c_1)^{\alpha_1} + (1 + c_2)^{\alpha_2} + \dots + (1 + c_j)^{\alpha_j}] \quad (4.4)$$

Where:

k_i : sub function of G that expresses a characteristic of the system dependent on the optimization parameters;

k_{i0} : sub function value for an acceptable solution;

γ_i : weight that represents the relative importance of each sub function;

c_j : constraint parameter equal to zero if j^{th} constraint is not violated or considerably greater than one otherwise;

α_j : weight that represents the strength of each constraint.

This function provides flexibility to prioritize sub functions and better guide the optimization process. It also provides the possibility to incorporate penalization for constraint violation in the cost function value. Initially, it was implemented without the terms related to the problem constraints as shown in Equation 4.5.

$$f_2(x) = - \left[\left(\frac{J_1(x)}{J_{10}} \right)^{\gamma_1} + \left(\frac{J_2(x)}{J_{20}} \right)^{\gamma_2} + \dots + \left(\frac{J_i(x)}{J_{i0}} \right)^{\gamma_i} \right] \quad (4.5)$$

In this case, $J_i(x)$ remains defined by Equation 4.3, J_{i0} is the partial value for a dynamic stiffness marginally above the requirement and γ_i is a weight dependent on the frequency. A γ_i of 5 was considered for the infinite frequency and 1 otherwise.

However, this implementation was not robust against constraint violation. The results obtained using this approach did not satisfy the performance requirements implemented using the constraint function described in section 4.3.

To overcome this issue, function $f_2(x)$ was modified to include penalties for not meeting required performance. The updated function is shown in Equation 4.6.

$$f_2(x) = - \left[\left(\frac{J_1(x)}{J_{10}} \right)^{\gamma_1} + \left(\frac{J_2(x)}{J_{20}} \right)^{\gamma_2} + \dots + \left(\frac{J_i(x)}{J_{i0}} \right)^{\gamma_i} \right] + [(1 + r_1)^{\alpha_1} + (1 + r_2)^{\alpha_2} + \dots + (1 + r_j)^{\alpha_j}] \quad (4.6)$$

Where r_j is the result of the evaluation of the j^{th} constraint. If the constraint complies with the requirement r_j equals zero otherwise r_j is the difference between the constraint and the requirement. Also, α_j is the weight that represents the strength of this constraint. In this case, α_j was arbitrated as 5 for all constraints. Finally, the constraint polynomial is summed to the objective polynomial instead of multiplied. This modification was necessary because multiplying would benefit instead of penalizing the inadequate solution.

Finally, both cost functions $f_1(x)$ and $f_2(x)$ have another component intended to avoid evaluation of solutions that do not meet the dynamic stiffness requirement: if the requirement is not met at any evaluated frequency, a component substantially greater than the cost function is added turning $f_1(x)$ and $f_2(x)$ into large positive values.

The next chapter presents the optimization results obtained with each function.

5 Optimization Results

This chapter presents the results obtained by the optimization algorithm for classical control strategies. An optimized controller design was obtained for each control strategy and its dynamic stiffness, step and frequency responses were evaluated.

Each response is shown in a separate figure that contains four curves which comprise initial and optimized design evaluation in two fluid temperatures each. The tests with two temperatures were performed to cover the entire actuator operation range, from -15°C to 100°C.

5.1 Classic Controllers and Simple Cost Function

The classic controllers P, PI, PD and PID were optimized. The initial solutions selected as an input to the optimization algorithm are the controller designs proposed by Ballesteros (2015) and are listed in table 5.1.

TABLE 5.1 – Initial Solution for Classic Controllers

Parameter	P	PI	PD	PID
K_p	40	40	40	40
K_i	0.0	$1e - 6$	0.0	$1e - 6$
K_d	0.0	0.0	0.7	0.7

The results presented in this section were obtained using the simple cost function described by Equations 4.2 and 4.3.

5.1.1 Proportional Controller (P)

The optimization results for the proportional controller are presented in this section. As shown in table 5.2, the execution time of the optimization was 5.6 hours and 15 iterations were performed until stopping criteria was reached. Also, the proportional gain

after optimization was 84.3. This value is in the range evaluated by Ballesteros (2015) and therefore similar behavior as the one observed by the author is expected.

TABLE 5.2 – P Controller Optimization Execution Results

Optimized Proportional Gain	84.3
Optimization Execution Time (h)	5.6
Number of Iterations	15
Number of Obj. Function Evaluations	32

The dynamic stiffness response for each design is shown in Figure 5.1. The dynamic stiffness for high fluid temperature is considerably lower than for low temperature which reinforces the use of high temperature response to calculate the cost function as discussed in section 4.4.1.

An increase in dynamic stiffness is observed below 4Hz but, for frequencies up to 15Hz a decrease occurred. The dynamic stiffness is more critical at 15Hz because for greater frequencies, more stiffness is required to avoid coupling between control surface rotational mode and aircraft aeroelastic modes (BALLESTEROS, 2015). Thus, because of the dynamic stiffness decrease at 15Hz, the actuation system with the optimized controller has a worse flutter suppression characteristic than with the initial controller.

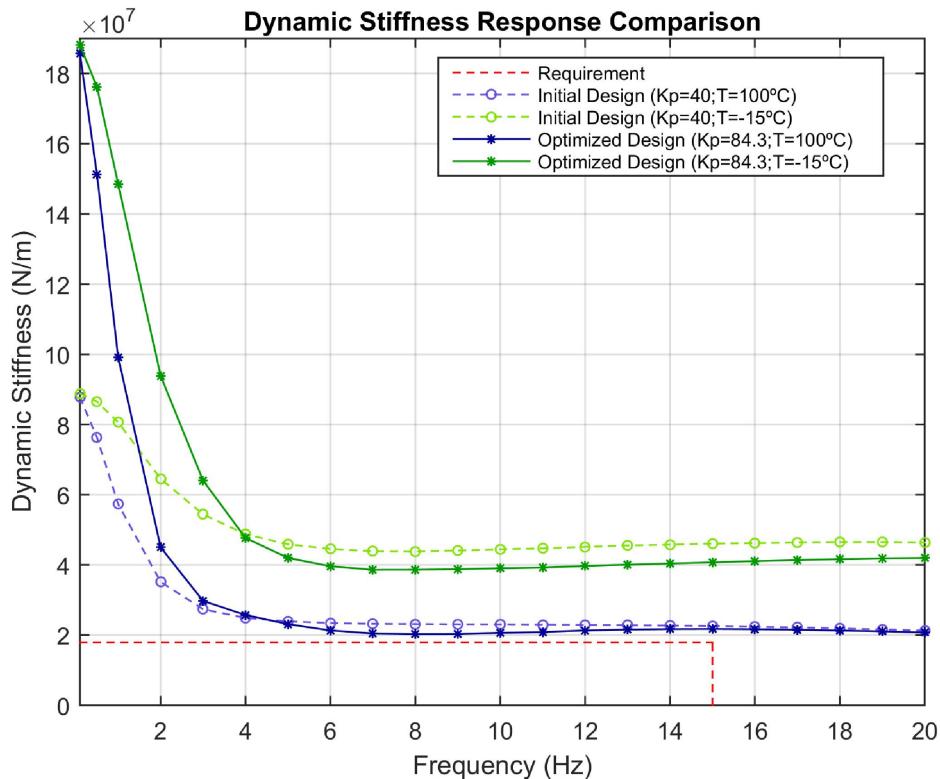


FIGURE 5.1 – Dynamic Stiffness Comparison for Proportional Controller.

Table 5.3 show the partial values of the cost function for each frequency evaluated by the dynamic stiffness test. The values at the lower frequencies are considerably larger since stiffness increase occurred in this range.

TABLE 5.3 – P Controller Final Solution Cost Function for Each Evaluated Frequency

Frequency (Hz)	J_i	Frequency (Hz)	J_i	Frequency (Hz)	J_i
0.1	$1.68e + 08$	5	$5.22e + 06$	11	$2.96e + 06$
0.5	$1.33e + 08$	6	$3.42e + 06$	12	$3.42e + 06$
1	$8.12e + 07$	7	$2.56e + 06$	13	$3.66e + 06$
2	$2.71e + 07$	8	$2.4e + 06$	14	$3.8e + 06$
3	$1.18e + 07$	9	$2.43e + 06$	15	$3.86e + 06$
4	$7.83e + 06$	10	$2.72e + 06$		

The time response comparison is presented in Figure 5.2 and the performance requirements are shown in Table 5.4. The settling time decreased by 68ms and the steady state error decreased approximately 50%. Also, it is possible to observe that for higher fluid temperature the surface position rises faster than for the lower temperature, because of the lower fluid density, as discussed in section 4.3.1.

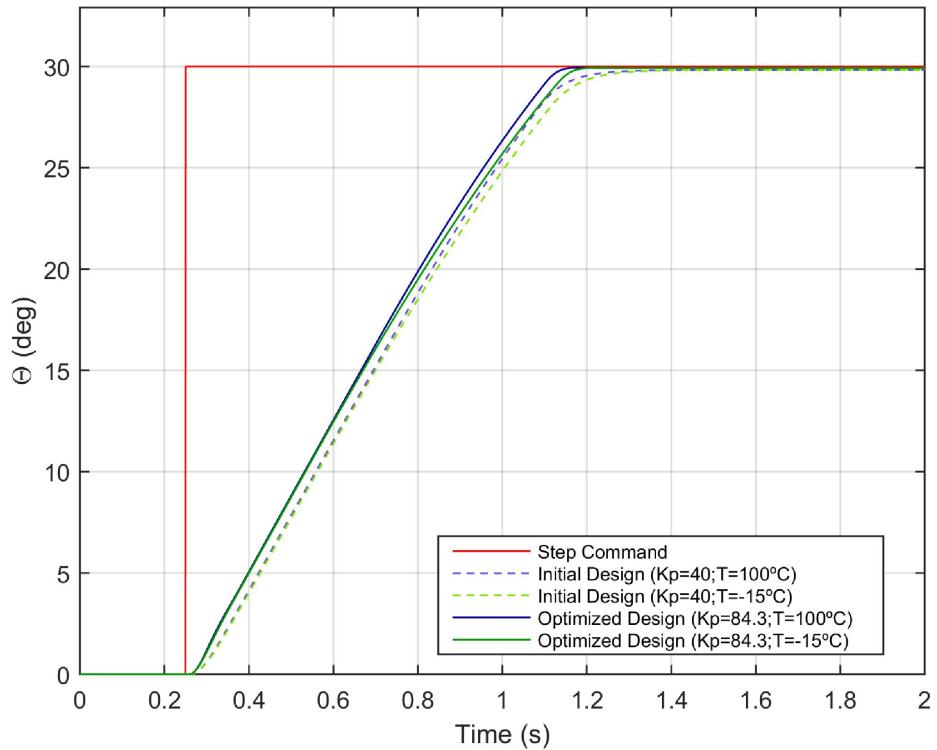


FIGURE 5.2 – Time Response Comparison for Proportional Controller.

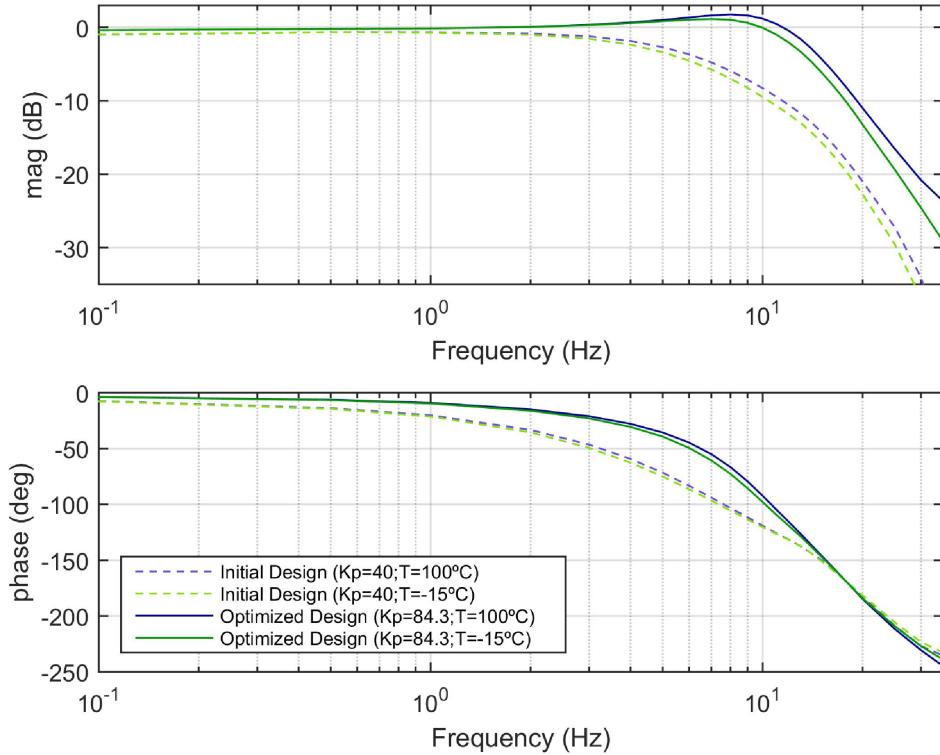


FIGURE 5.3 – Frequency Response Comparison for Proportional Controller.

The closed-loop frequency response comparison is presented in Figure 5.3 and the performance requirements are also shown in Table 5.4. The closed-loop gain and phase allowances were fairly reduced but still comply with requirements. The magnitude response surpassed 0dB and peaked at 1.75dB, above optimization constraint. Positive magnitude peaks are related to time response overshoot and dynamic stiffness reduction but in this case it did not result in any of these concerns.

TABLE 5.4 – Requirement Compliance for Proportional Controller

Design Parameter	Requirement	Baseline	Optimized Controller	Difference (%)
Settling Time (ms)	< 1100	959	891	-7.1
Steady State Error (%)	< 1	0.27	0.13	-51.9
Overshoot (%)	< 10	0.0	0.0	N/A
Minimum Average Rate ($^{\circ}/s$)	> 32	34.45	34.62	0.5
Maximum Average Rate ($^{\circ}/s$)	< 36	34.45	34.62	0.5
Closed-Loop Gain Allowance (dB)	≥ 10	20.35	10.13	-50.2
Closed-Loop Phase Allowance ($^{\circ}$)	≥ 45	inf	69.78	N/A
Closed-Loop Maximum Peak (dB)	< 0.5	-0.65	1.75	369.2
Closed-Loop Initial Magnitude (dB)	None	-1.0	-0.39	61.0
Closed-Loop Bandwidth (Hz)	None	6.30	14.4	128.6

5.1.2 Proportional Integral Controller (PI)

The optimization of the proportional integral controller was executed in 11.2 hours and required 19 iterations, as shown in Table 5.5. Also, the optimization yielded a proportional gain of 66.88 and an integral gain of 0.001. These values also were evaluated by Ballesteros (2015) and similar behavior in this case is expected as well.

TABLE 5.5 – PI Controller Optimization Execution Results

Optimized Proportional Gain	66.88
Optimized Integral Gain	0.001
Optimization Execution Time (h)	11.2
Number of Iterations	19
Number of Obj. Function Evaluations	68

The dynamic stiffness response for each PI design is shown in Figure 5.4. Similarly to the P controller, increase in dynamic stiffness was observed only in frequencies below 4Hz and a decrease occurred for frequencies between 4Hz and 15Hz.

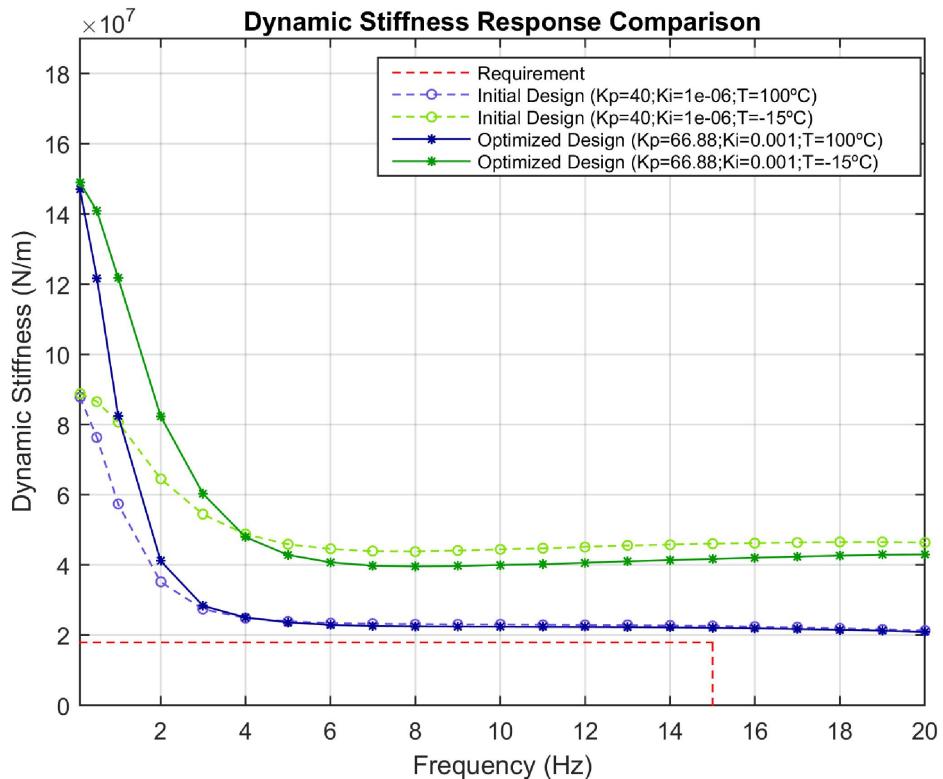


FIGURE 5.4 – Dynamic Stiffness Optimization Result for the PI Controller.

Table 5.6 show the partial values of the cost function for each frequency evaluated by the dynamic stiffness test. The lower frequencies values are considerably larger since stiffness increase occurred in this range.

TABLE 5.6 – PI Controller Final Solution Cost Function for Each Evaluated Frequency

Frequency (Hz)	J_i	Frequency (Hz)	J_i	Frequency (Hz)	J_i
0.1	$1.29e + 08$	5	$5.72e + 06$	11	$4.46e + 06$
0.5	$1.04e + 08$	6	$5.01e + 06$	12	$4.44e + 06$
1	$6.46e + 07$	7	$4.67e + 06$	13	$4.38e + 06$
2	$2.32e + 07$	8	$4.58e + 06$	14	$4.29e + 06$
3	$1.04e + 07$	9	$4.54e + 06$	15	$4.15e + 06$
4	$7.12e + 06$	10	$4.5e + 06$		

The time response comparison is presented in Figure 5.5 and the performance requirements are shown in Table 5.7. The settling time decreased by 57ms and the steady state error decreased almost 40%.

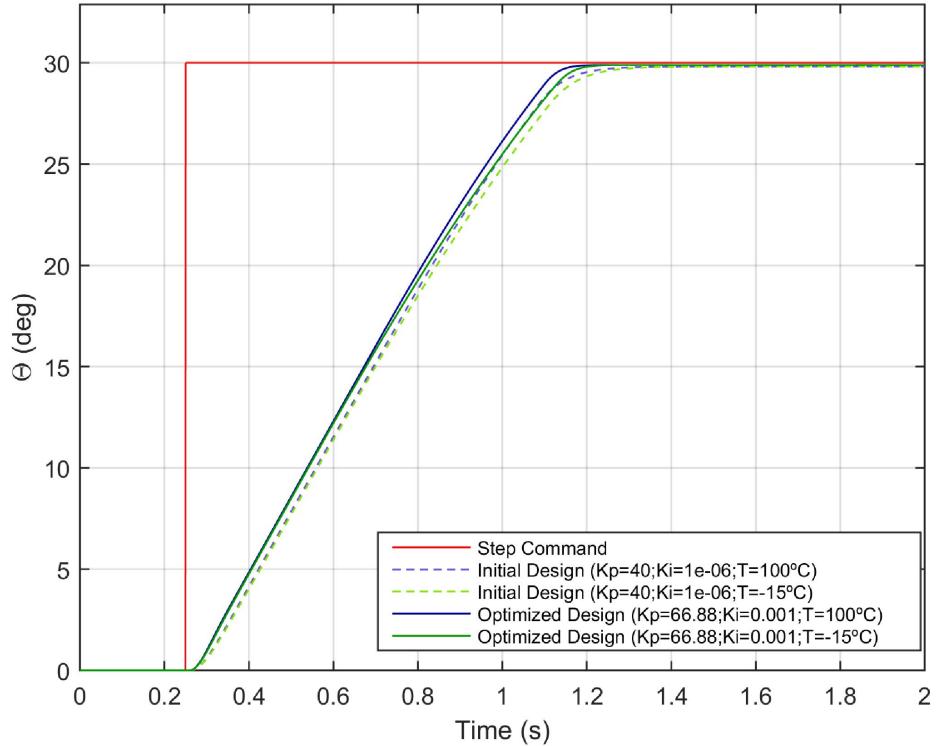


FIGURE 5.5 – Time Response Optimization Result for the PI Controller.

The closed-loop frequency response comparison is presented in Figure 5.6 and the performance requirements are also shown in Table 5.7. The closed-loop gain and phase allowances were fairly reduced but still comply with requirements. The magnitude response surpassed 0 dB and peaked at 0.41 dB, just below the 0.5dB requirement.

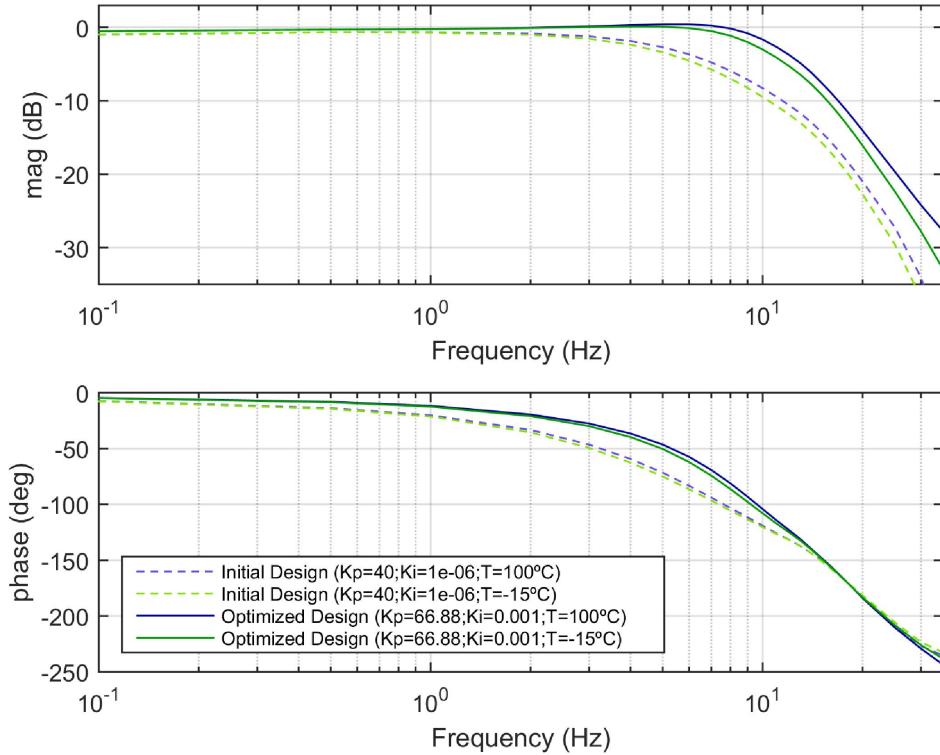


FIGURE 5.6 – Frequency Response Baseline for the PI Controller.

TABLE 5.7 – Requirement Compliance for PI Controller

Design Parameter	Requirement	Baseline	Optimized Controller	Difference (%)
Settling Time (ms)	< 1100	959	902	-5.9.
Steady State Error (%)	< 1	0.27	0.16	-40.7
Overshoot (%)	< 10	0.0	0.0	N/A
Minimum Average Rate ($^{\circ}/s$)	> 32	34.45	34.58	0.4
Maximum Average Rate ($^{\circ}/s$)	< 36	34.45	34.58	0.4
Closed-Loop Gain Allowance (dB)	≥ 10	20.34	13.31	-34.6.
Closed-Loop Phase Allowance ($^{\circ}$)	≥ 45	<i>inf</i>	103.6	N/A
Closed-Loop Maximum Peak (dB)	< 0.5	-0.65	0.41	163.1
Closed-Loop Initial Magnitude (dB)	<i>None</i>	-1.0	-0.52	48.0
Closed-Loop Bandwidth (Hz)	<i>None</i>	6.30	11.80	87.3

5.1.3 Proportional Derivative Controller (PD)

The proportional derivative controller optimization execution time was 18 hours and required 32 iterations, as per Table 5.8. Also, the optimization yielded a proportional gain of 79.59 and a derivative gain of 0.3339. These values also were evaluated by Ballesteros (2015) and similar behavior in this case is expected as well.

TABLE 5.8 – PD Controller Optimization Execution Results

Optimized Proportional Gain	79.59
Optimized Derivative Gain	0.3339
Optimization Execution Time (h)	18.0
Number of Iterations	32
Number of Obj. Function Evaluations	121

The dynamic stiffness response for each PD design is shown in Figure 5.7. Similarly to previous controllers, the dynamic stiffness increase was restricted to frequencies below 3Hz but, for the PD, a fair reduction was observed in higher frequencies. This behavior is coherent with K_d parametric study which demonstrated dynamic stiffness reduction for frequencies below 12Hz (BALLESTEROS, 2015).

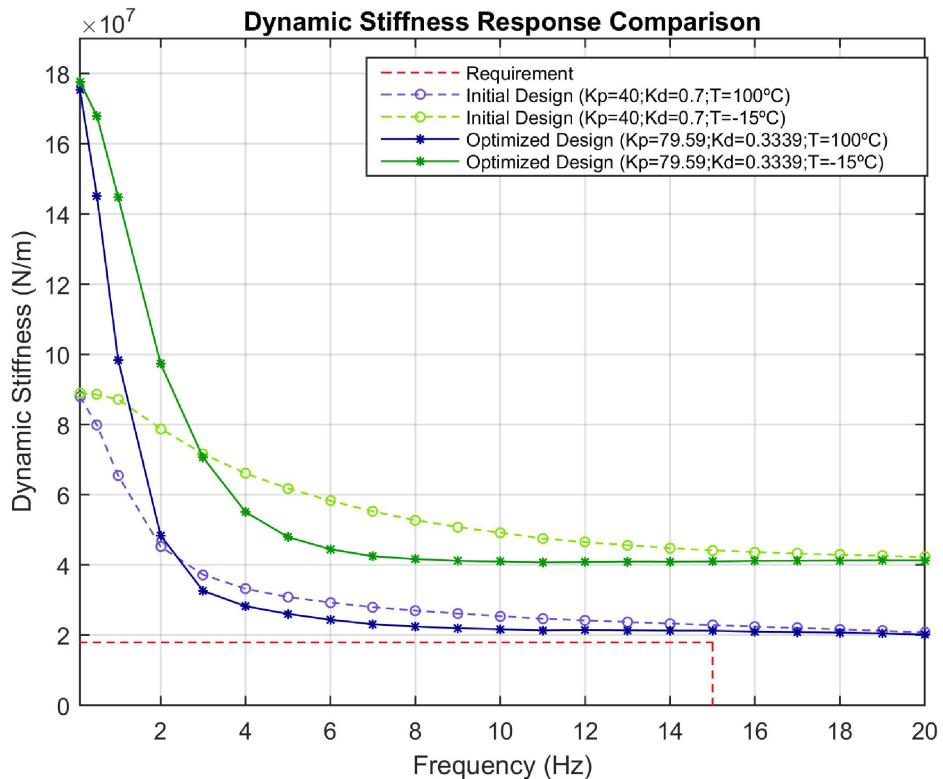


FIGURE 5.7 – Dynamic Stiffness Optimization Result for the PD Controller.

Table 5.9 show the partial values of the cost function for each frequency evaluated by the dynamic stiffness test. The lower frequencies values are considerably larger since stiffness increase occurred in this range.

TABLE 5.9 – PD Controller Final Solution Cost Function for Each Evaluated Frequency

Frequency (Hz)	J_i	Frequency (Hz)	J_i	Frequency (Hz)	J_i
0.1	$1.57e + 08$	5	$8.14e + 06$	11	$3.45e + 06$
0.5	$1.27e + 08$	6	$6.47e + 06$	12	$3.55e + 06$
1	$8.05e + 07$	7	$5.19e + 06$	13	$3.46e + 06$
2	$3.05e + 07$	8	$4.54e + 06$	14	$3.36e + 06$
3	$1.47e + 07$	9	$4.06e + 06$	15	$3.33e + 06$
4	$1.03e + 07$	10	$3.77e + 06$		

The time response comparison is presented in Figure 5.8 and the performance requirements are shown in Table 5.10. The settling time decreased by 97ms and the steady state error decreased almost 50%.

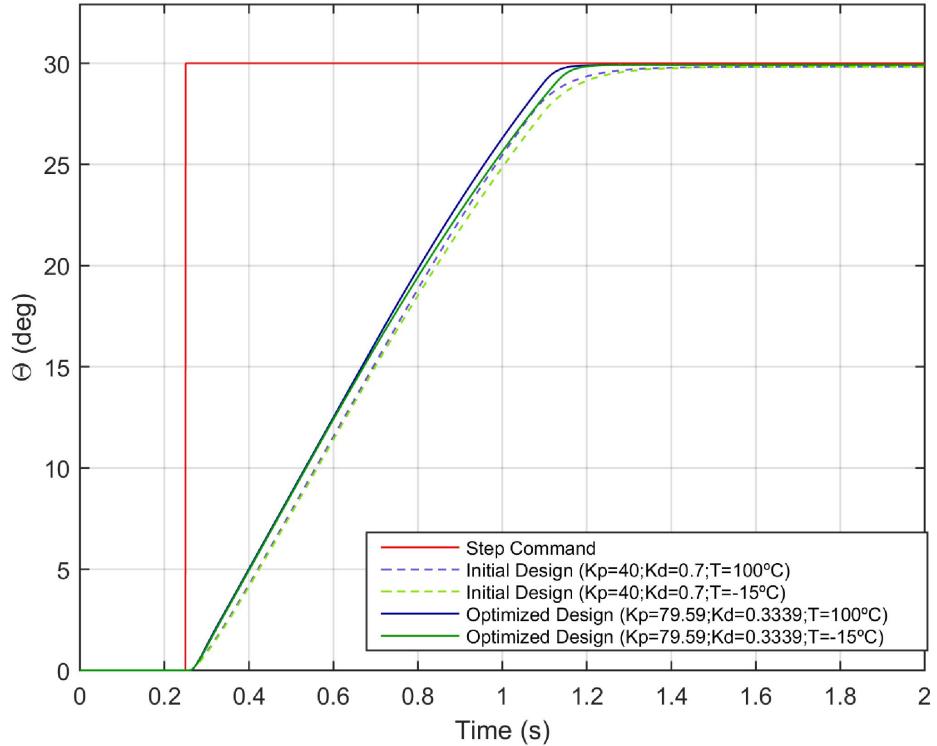


FIGURE 5.8 – Time Response Optimization Result for the PD Controller.

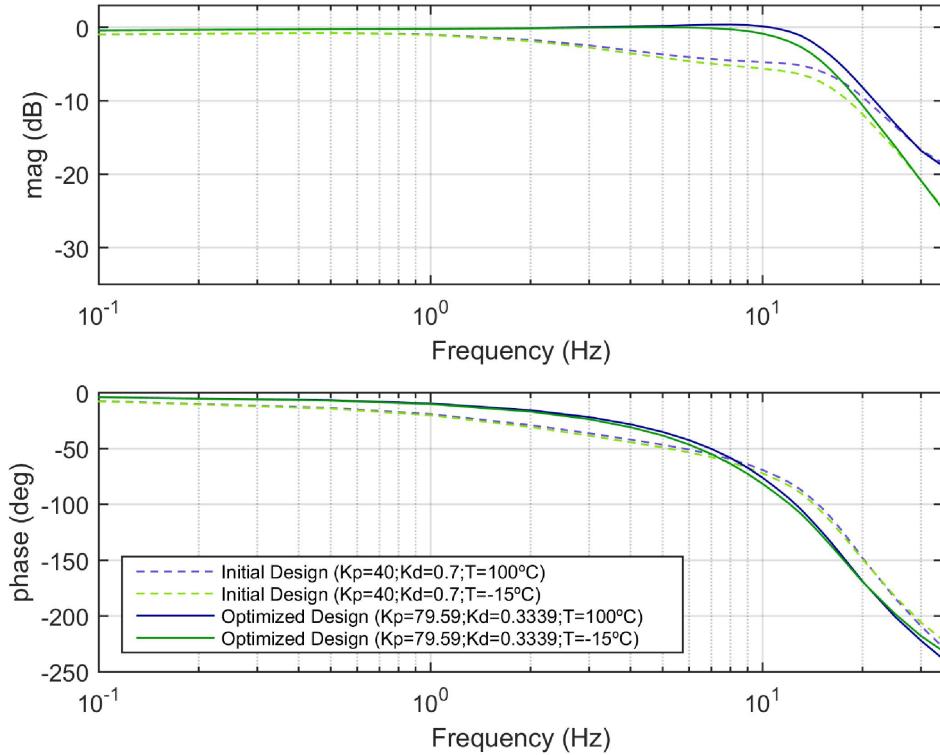


FIGURE 5.9 – Frequency Response Baseline for the PD Controller.

The closed-loop frequency response comparison is presented in Figure 5.9 and the performance requirements are also shown in Table 5.10. The closed-loop gain allowance was reduced and was just below requirement and the closed-loop phase allowance went from infinite to 99.2° . The magnitude response surpassed 0 dB and peaked at 0.38 dB, just below the 0.5dB requirement and the bandwidth increased considerably.

TABLE 5.10 – Requirement Compliance for PD Controller

Design Parameter	Requirement	Baseline	Optimized Controller	Difference (%)
Settling Time (ms)	< 1100	992	895	-9.8
Steady State Error (%)	< 1	0.27	0.14	-48.1
Overshoot (%)	< 10	0.0	0.0	N/A
Minimum Average Rate ($^\circ/s$)	> 32	34.32	34.62	0.9
Maximum Average Rate ($^\circ/s$)	< 36	34.32	34.62	0.9
Closed-Loop Gain Allowance (dB)	≥ 10	13.00	9.90	-23.8
Closed-Loop Phase Allowance ($^\circ$)	≥ 45	<i>inf</i>	99.20	N/A
Closed-Loop Maximum Peak (dB)	< 0.5	-0.74	0.38	151.4
Closed-Loop Initial Magnitude (dB)	<i>None</i>	-0.99	-0.41	58.6
Closed-Loop Bandwidth (Hz)	<i>None</i>	5.83	15.59	167.4

5.1.4 Proportional Integral Derivative Controller (PID)

The proportional integral derivative controller optimization execution time was 16.4 hours and required 20 iterations, as per Table 5.11. Also, the optimization yielded a proportional gain of 76.05, integral gain of 3.285e-9 and a derivative gain of 0.4318. These values were evaluated separately by Ballesteros (2015) but were not considered together as a PID design.

TABLE 5.11 – PID Controller Optimization Execution Results

Optimized Proportional Gain	76.05
Optimized Integral Gain	3.285e-9
Optimized Derivative Gain	0.4318
Optimization Execution Time (h)	16.4
Number of Iterations	20
Number of Obj. Function Evaluations	111

The dynamic stiffness response for each PID design is shown in Figure 5.10. Similarly to the PD controller, increase in dynamic stiffness was observed in frequencies below 3Hz and also a fair reduction was observed in higher frequencies.

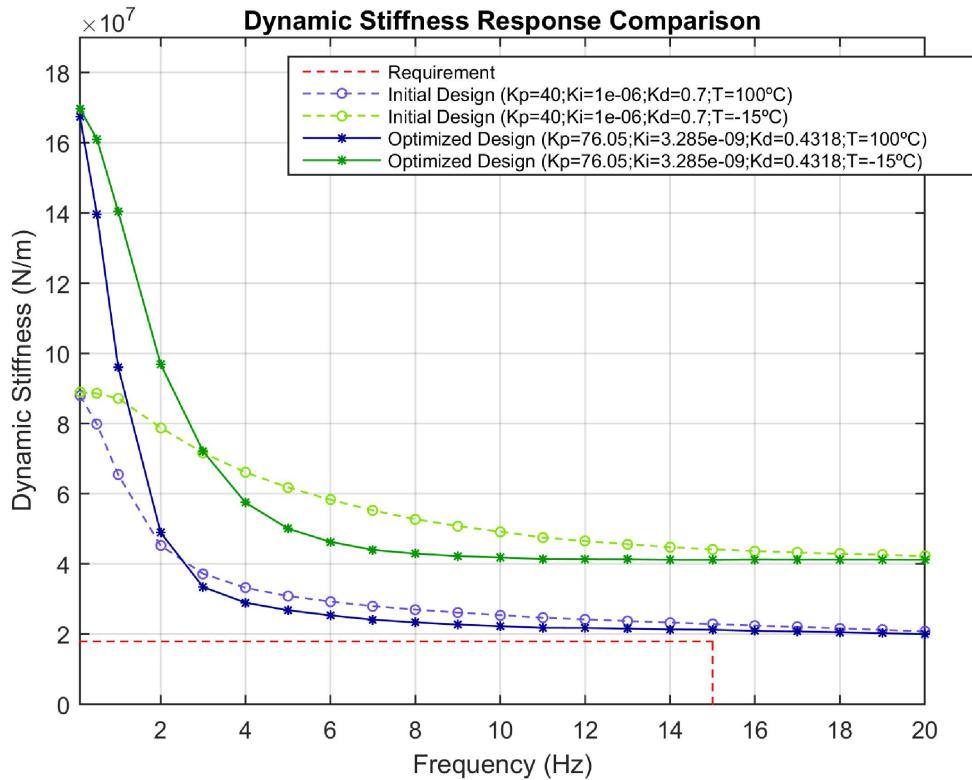


FIGURE 5.10 – Dynamic Stiffness Optimization Result for the PID Controller.

Table 5.12 show the partial values of the cost function for each frequency evaluated

by the dynamic stiffness test. The lower frequencies values are considerably larger since stiffness increase occurred in this range.

TABLE 5.12 – PID Controller Final Solution Cost Function for Each Evaluated Frequency

Frequency (Hz)	J_i	Frequency (Hz)	J_i	Frequency (Hz)	J_i
0.1	$1.5e + 08$	5	$8.9e + 06$	11	$3.89e + 06$
0.5	$1.22e + 08$	6	$7.42e + 06$	12	$3.87e + 06$
1	$7.82e + 07$	7	$6.2e + 06$	13	$3.66e + 06$
2	$3.1e + 07$	8	$5.43e + 06$	14	$3.44e + 06$
3	$1.55e + 07$	9	$4.81e + 06$	15	$3.33e + 06$
4	$1.1e + 07$	10	$4.36e + 06$	0	0

The time response comparison is presented in Figure 5.11 and the performance requirements are shown in Table 5.13. The settling time was reduced by 94ms and the steady state error decreased by almost 50%.

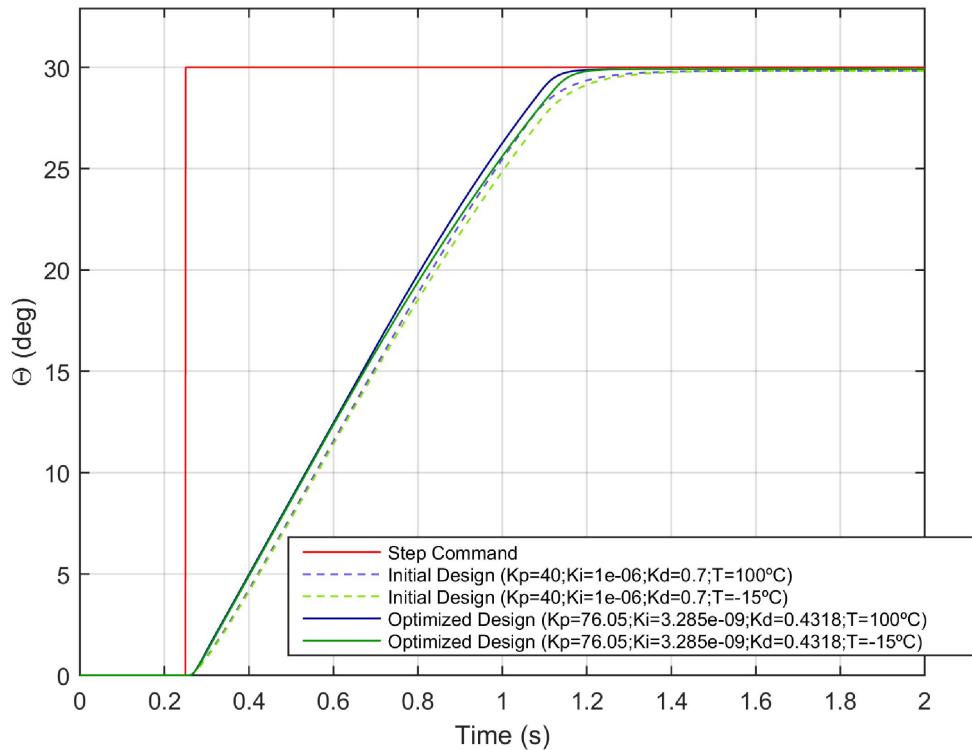


FIGURE 5.11 – Time Response Optimization Result for the PID Controller.

The closed-loop frequency response comparison is presented in Figure 5.12 and the performance requirements are also shown in Table 5.13. The closed-loop gain allowance was reduced and was slightly lower than requirement and the closed-loop phase allowance remained infinite.

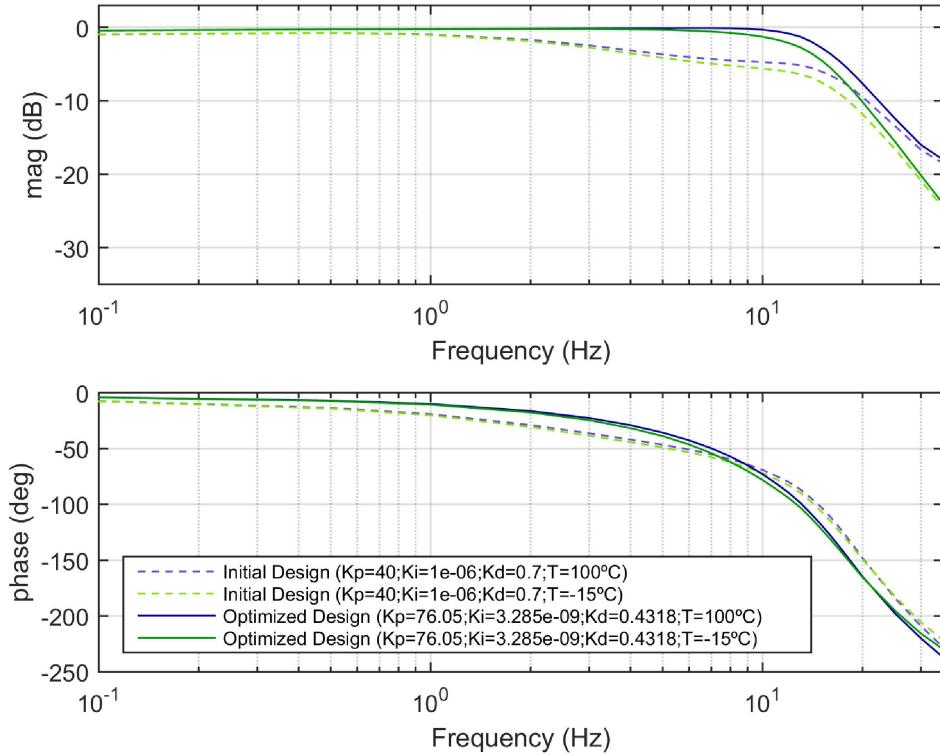


FIGURE 5.12 – Frequency Response Baseline for the PID Controller.

TABLE 5.13 – Requirement Compliance for PID Controller

Design Parameter	Requirement	Baseline	Optimized Controller	Difference (%)
Settling Time (ms)	< 1100	992	898	-9.5.
Steady State Error (%)	< 1	0.27	0.14	-48.1
Overshoot (%)	< 10	0.0	0.0	N/A
Minimum Average Rate ($^\circ/s$)	> 32	34.32	34.61	0.8
Maximum Average Rate ($^\circ/s$)	< 36	34.32	34.61	0.8
Closed-Loop Gain Allowance (dB)	≥ 10	13.00	9.87	-24.1
Closed-Loop Phase Allowance ($^\circ$)	≥ 45	inf	inf	N/A
Closed-Loop Maximum Peak (dB)	< 0.5	-0.74	-0.10	86.5
Closed-Loop Initial Magnitude (dB)	None	-0.99	-0.44	55.6
Closed-Loop Bandwidth (Hz)	None	5.83	15.8	171.0

The results obtained for P and PI controllers are very similar what is coherent with previous findings which showed that integral gain does not influence dynamic stiffness. The same occurred between PD and PID controllers results that were also fairly similar.

Optimization for all controllers yielded improvement in overall dynamic stiffness as expected, however, a tendency for increases in lower frequencies is clear. This can be explained by the fact that the control loop has a high influence in lower frequency dynamic stiffness because is where it can respond faster to external disturbances.

These results also show a relationship between time response performance improve-

ment (mainly settling time and steady state error) and dynamic stiffness reduction in high frequencies. Therefore, this suggests that to increase dynamic stiffness a reduction in time domain performance would be required.

A deterioration of frequency domain performance was observed for all controllers, including a requirement violation for proportional controller (closed-loop peak magnitude) and for PD and PID controllers (closed-loop gain allowance). These findings suggest that improvement in time domain performance also jeopardize frequency domain parameters.

In summary, this section has shown that the developed algorithm delivers overall increase in dynamic stiffness. However, its cost function considers improvements in both low and high frequencies equally what has led to dynamic stiffness improvements only in low frequencies and inferior values at the infinite frequency.

Hence, the solutions yielded by the optimization diminish flutter suppression characteristics when compared to the initial solutions. This result did not satisfy the main goal of the optimization and, therefore, a change in the cost function was necessary. The next section presents a new set of results obtained by the optimization considering a frequency weighted cost function.

5.2 Classic Controllers and Frequency Weighted Cost Function

This section presents the results obtained through the optimization considering a more complex cost function, described in Equation 4.6. The initial solutions in this case were the same used to obtain the previous results and are shown in Table 5.1.

5.2.1 Proportional Controller (P)

The results obtained with the weighted cost function for the proportional controller are presented in this section. As shown in table 5.14, the execution time of the optimization was 2.5 hours and 7 iterations were performed until stopping criteria was reached. Also, the proportional gain after optimization is 24.74. This value is in the range evaluated by Ballesteros (2015) and therefore similar behavior as the one observed by the author is expected.

TABLE 5.14 – P Controller Optimization Execution Results

Optimized Proportional Gain	24.74
Optimization Execution Time (h)	6.4
Number of Iterations	4
Number of Obj. Function Evaluations	42

The dynamic stiffness response for each design is shown in Figure 5.13. Despite a decrease in lower frequencies, an increase in dynamic stiffness is observed in all frequencies above 4Hz, including the infinite frequency.

The result is coherent with previous studies findings that for proportional controllers the dynamic stiffness at the infinite frequency are higher when the gain is smaller (BALLES-TEROS, 2015). Also, comparing Figures 5.13 and 5.1 is possible to note the impact of changing the cost function on the final optimization result.

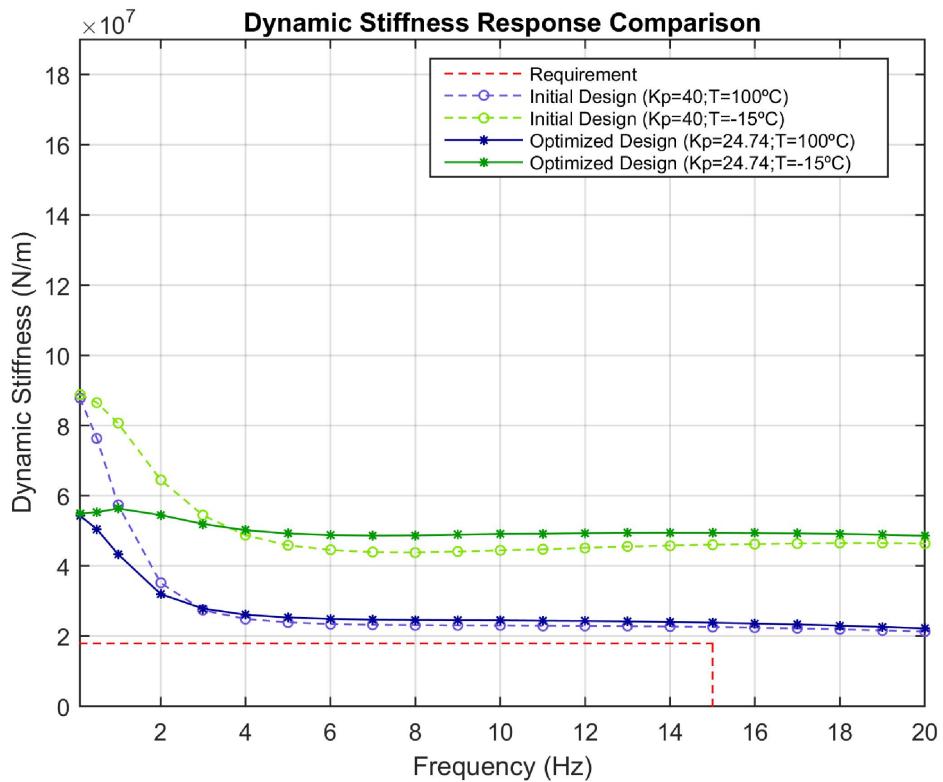


FIGURE 5.13 – Dynamic Stiffness Comparison for Proportional Controller.

Table 5.15 show the partial values of the cost function for each frequency evaluated by the dynamic stiffness test. The values at the lower frequencies are still larger than in higher frequencies but the difference is not as great as in Table 5.3. This shows that even with smaller partial cost functions at 15Hz the algorithm preserved this solutions because of the high weight this partial has on the global cost function.

TABLE 5.15 – P Controller Final Solution Cost Function for Each Evaluated Frequency

Frequency (Hz)	J_i	Frequency (Hz)	J_i	Frequency (Hz)	J_i
0.1	$1.51e + 07$	5	$9.5e + 06$	11	$8.25e + 06$
0.5	$1.56e + 07$	6	$9.2e + 06$	12	$8.05e + 06$
1	$1.65e + 07$	7	$8.96e + 06$	13	$7.81e + 06$
2	$1.29e + 07$	8	$8.79e + 06$	14	$7.53e + 06$
3	$1.1e + 07$	9	$8.63e + 06$	15	$7.2e + 06$
4	$1e + 07$	10	$8.47e + 06$		

The time response comparison is presented in Figure 5.14 and the performance requirements are shown in Table 5.16. The settling time increased by 142ms and was slightly higher than requirement. The steady state error increased approximately 60% and the average rate of actuation slightly decreased.

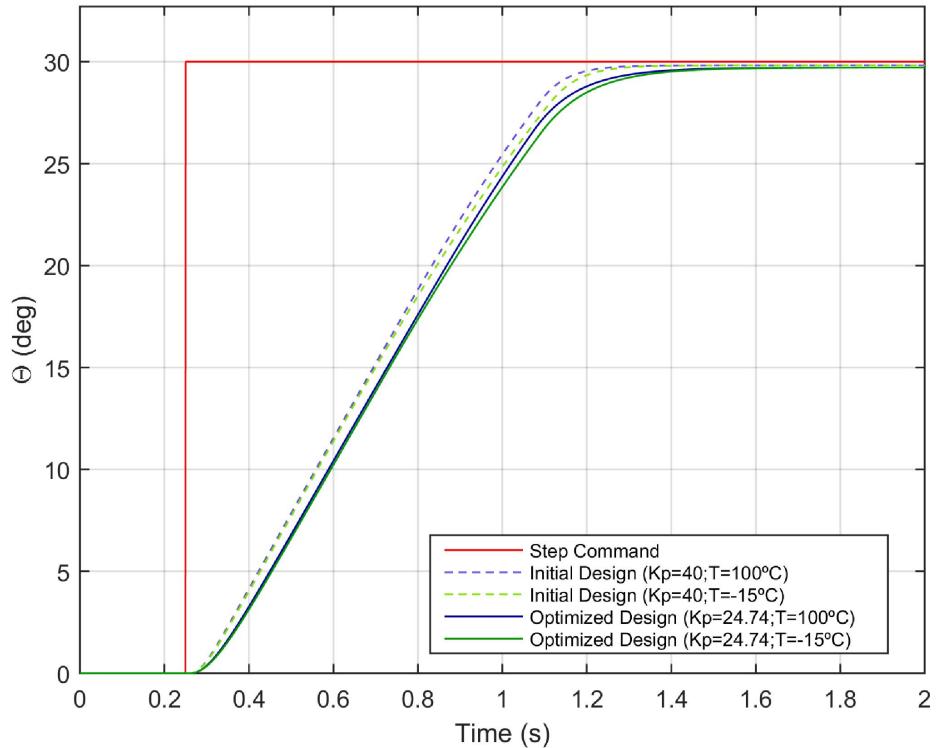


FIGURE 5.14 – Time Response Comparison for Proportional Controller.

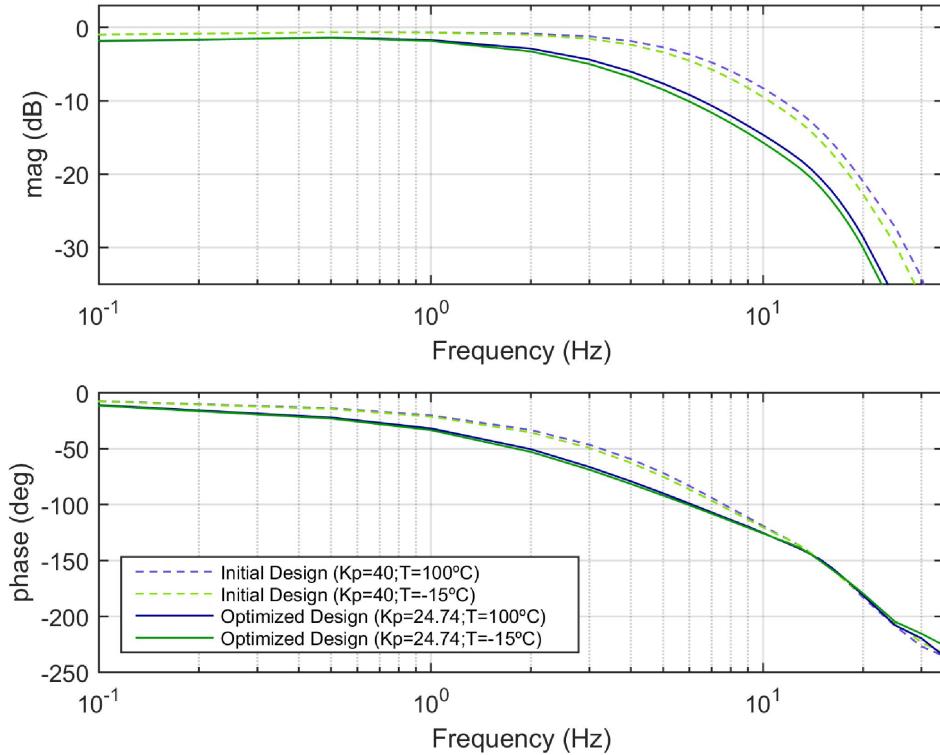


FIGURE 5.15 – Frequency Response Comparison for Proportional Controller.

The closed-loop frequency response comparison is presented in Figure 5.15 and the performance requirements are also shown in Table 5.16. The closed-loop gain allowance increased and the closed-loop phase allowance remained infinite.

Both frequency and time performances of the optimized controller behaved in an opposite way to the ones from the controller obtained with the simple cost function since the optimization with the previous function yielded a K_p higher than the initial and a lower K_p was obtained with the current cost function.

TABLE 5.16 – Requirement Compliance for Proportional Controller

Design Parameter	Requirement	Baseline	Optimized Controller	Difference (%)
Settling Time (ms)	< 1100	959	1101	14.8
Steady State Error (%)	< 1	0.27	0.43	59.3
Overshoot (%)	< 10	0	0	N/A
Minimum Average Rate (°/s)	> 32	34.45	34.09	-1.0
Maximum Average Rate (°/s)	< 36	34.45	34.09	-1.0
Closed-Loop Gain Allowance (dB)	> 10	20.35	28.21	38.6
Closed-Loop Phase Allowance (°)	> 45	<i>inf</i>	<i>inf</i>	N/A
Closed-Loop Maximum Peak (dB)	< 0.5	-0.65	-1.38	-112.3
Closed-Loop Initial Magnitude(dB)	<i>None</i>	-1.0	-1.9	-90.0
Closed-Loop Bandwidth (Hz)	<i>None</i>	6.3	3.3	-47.6

5.2.2 Proportional Integral Controller (PI)

The optimization of the proportional integral controller was executed in 7.4 hours and required 6 iterations, as shown in Table 5.17. Also, the optimization yielded a proportional gain of 24.76 and an integral gain of 1.467e-3. These values also were evaluated by Ballesteros (2015) and similar behavior in this case is expected as well.

TABLE 5.17 – PI Controller Optimization Execution Results

Optimized Proportional Gain	24.76
Optimized Integral Gain	1.467e-3
Optimization Execution Time (h)	7.4
Number of Iterations	6
Number of Obj. Function Evaluations	49

The dynamic stiffness response for each PI design is shown in Figure 5.4. Similarly to the P controller, increase in dynamic stiffness was observed in all frequencies above 4Hz.

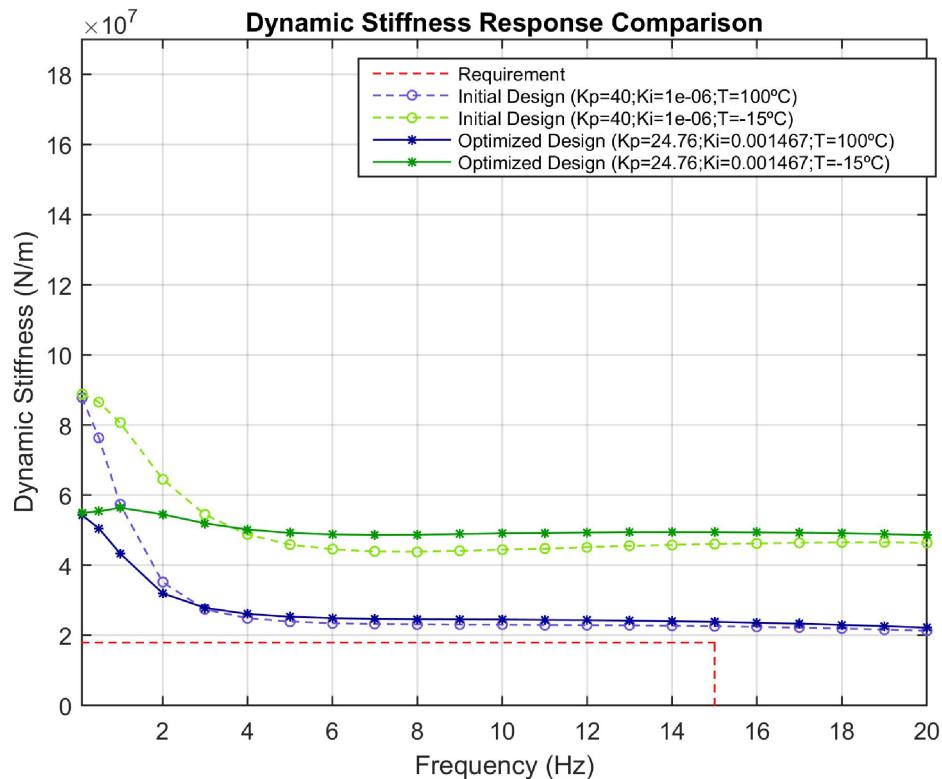


FIGURE 5.16 – Dynamic Stiffness Optimization Result for the PI Controller.

Table 5.18 show the partial values of the cost function for each frequency evaluated by the dynamic stiffness test. As for the P controller, the lower frequencies values are still larger than the higher frequency values.

TABLE 5.18 – PI Controller Final Solution Cost Function for Each Evaluated Frequency

Frequency (Hz)	J_i	Frequency (Hz)	J_i	Frequency (Hz)	J_i
0.1	$3.64e + 07$	5	$7.39e + 06$	11	$6.48e + 06$
0.5	$3.25e + 07$	6	$7.01e + 06$	12	$6.42e + 06$
1	$2.54e + 07$	7	$6.78e + 06$	13	$6.3e + 06$
2	$1.41e + 07$	8	$6.68e + 06$	14	$6.13e + 06$
3	$9.93e + 06$	9	$6.63e + 06$	15	$5.91e + 06$
4	$8.19e + 06$	10	$6.59e + 06$		

The time response comparison is presented in Figure 5.17 and the performance requirements are shown in Table 5.19. As observed for proportional controller, the settling time increased by 141ms and the steady state error increased approximately 60%.

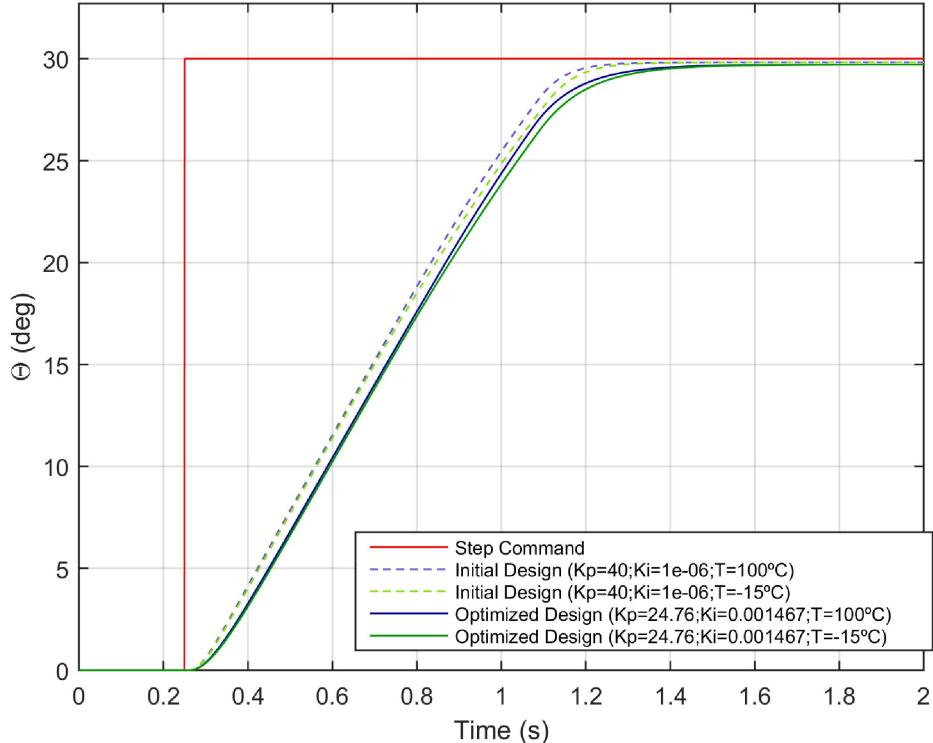


FIGURE 5.17 – Time Response Optimization Result for the PI Controller.

The closed-loop frequency response comparison is presented in Figure 5.18 and the performance requirements are also shown in Table 5.19. The closed-loop gain allowance increased to 28.19dB and closed-loop phase allowance remained infinite. Peak magnitude, initial magnitude and bandwidth decreased as expected.

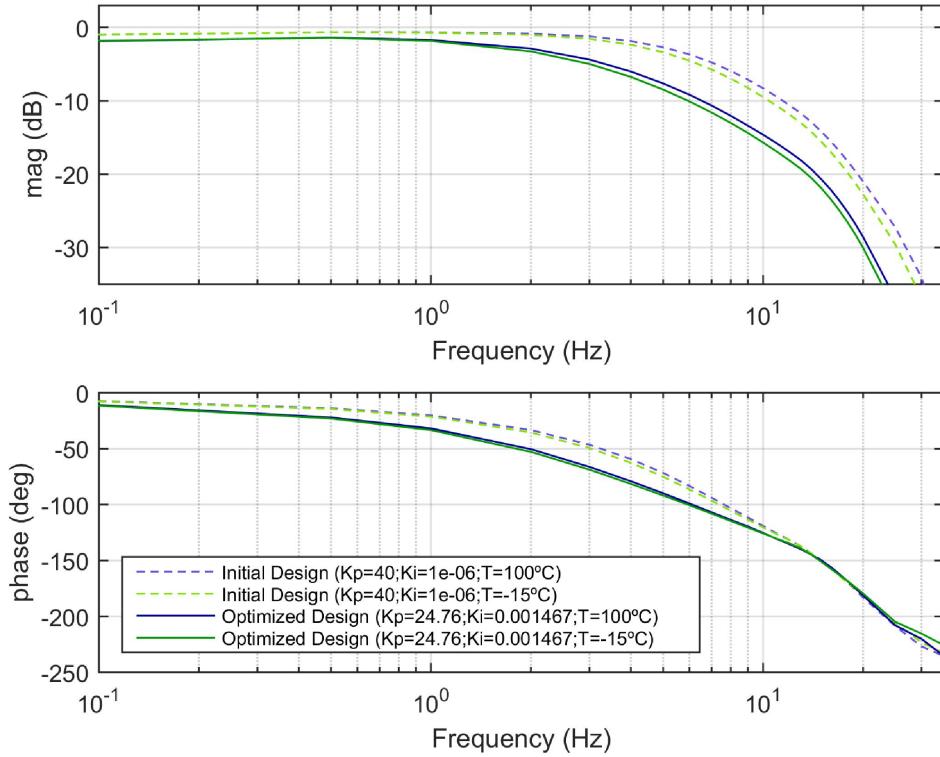


FIGURE 5.18 – Frequency Response Baseline for the PI Controller.

TABLE 5.19 – Requirement Compliance for PI Controller

Design Parameter	Requirement	Baseline	Optimized Controller	Difference (%)
Settling Time (ms)	< 1100	959	1100	14.7
Steady State Error (%)	< 1	0.27	0.43	59.3
Overshoot (%)	< 10	0	0	N/A
Minimum Average Rate (°/s)	> 32	34.45	34.09	-1.0
Maximum Average Rate (°/s)	< 36	34.45	34.09	-1.0
Closed-Loop Gain Allowance (dB)	> 10	20.35	28.19	38.6
Closed-Loop Phase Allowance (°)	> 45	inf	inf	N/A
Closed-Loop Maximum Peak (dB)	< 0.5	-0.65	-1.38	-112.3
Closed-Loop Initial Magnitude(dB)	None	-1.0	-1.9	-90.0
Closed-Loop Bandwidth (Hz)	None	6.3	3.3	-47.6

5.2.3 Proportional Derivative Controller (PD)

The proportional derivative controller optimization execution time was 10.6 hours and required 9 iterations, as per Table 5.20. Also, the optimization yielded a proportional gain of 38.75 and a derivative gain of 0.8134.

TABLE 5.20 – PD Controller Optimization Execution Results

Optimized Proportional Gain	38.75
Optimized Derivative Gain	0.8134
Optimization Execution Time (h)	10.6
Number of Iterations	9
Number of Obj. Function Evaluations	71

The dynamic stiffness response for each PD design is shown in Figure 5.19. Similarly to P and PI controllers, dynamic stiffness increase occurred in frequencies above 1Hz but in this case the gains were smaller, specially at the infinite frequency.

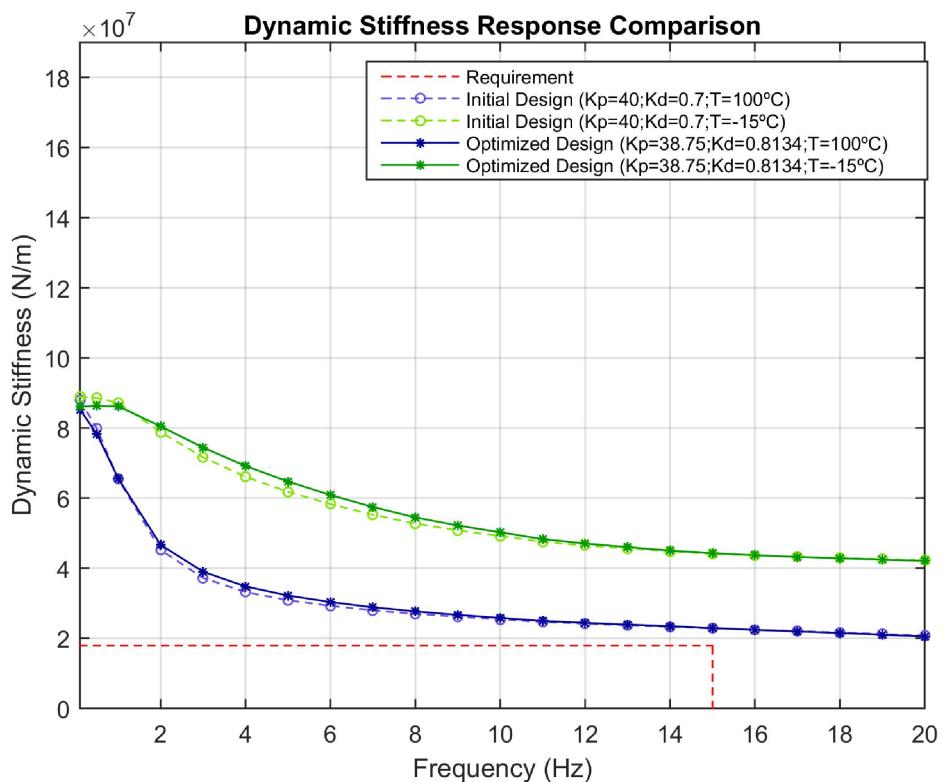


FIGURE 5.19 – Dynamic Stiffness Optimization Result for the PD Controller.

Table 5.21 show the partial values of the cost function for each frequency evaluated by the dynamic stiffness test. The lower frequencies values are larger since the dynamic stiffness is still greater in this range.

TABLE 5.21 – PD Controller Final Solution Cost Function for Each Evaluated Frequency

Frequency (Hz)	J_i	Frequency (Hz)	J_i	Frequency (Hz)	J_i
0.1	$6.73e + 07$	5	$1.43e + 07$	11	$7.05e + 06$
0.5	$6.03e + 07$	6	$1.24e + 07$	12	$6.52e + 06$
1	$4.77e + 07$	7	$1.09e + 07$	13	$6.01e + 06$
2	$2.87e + 07$	8	$9.79e + 06$	14	$5.55e + 06$
3	$2.11e + 07$	9	$8.79e + 06$	15	$5.02e + 06$
4	$1.69e + 07$	10	$7.92e + 06$		

The time response comparison is presented in Figure 5.20 and the performance requirements are shown in Table 5.22. The settling time increased by 14ms and the steady state error did not change.

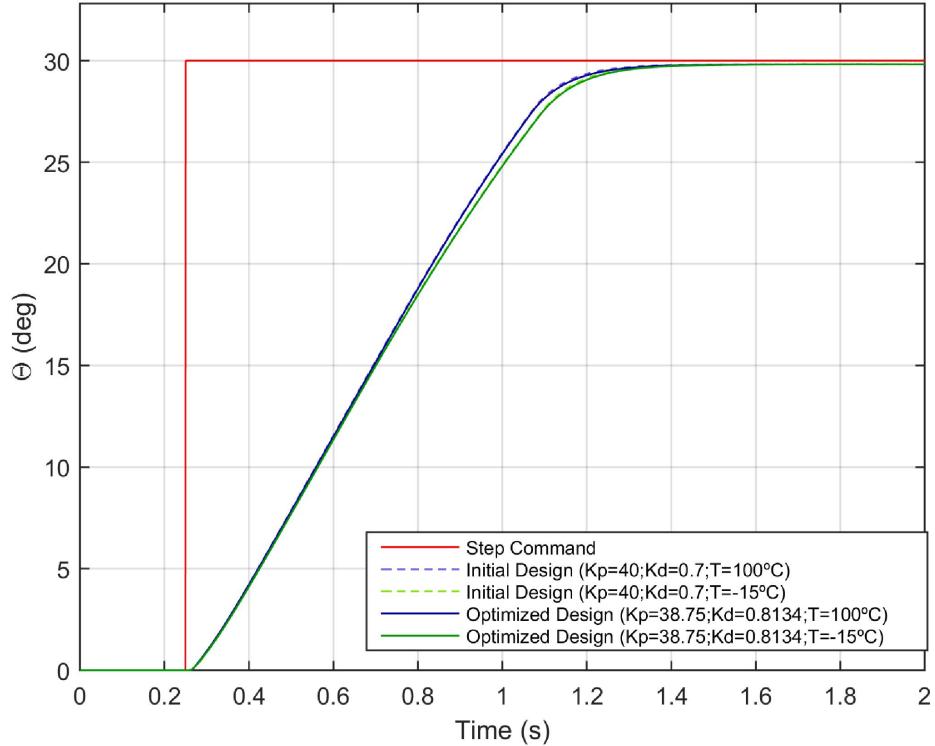


FIGURE 5.20 – Time Response Optimization Result for the PD Controller.

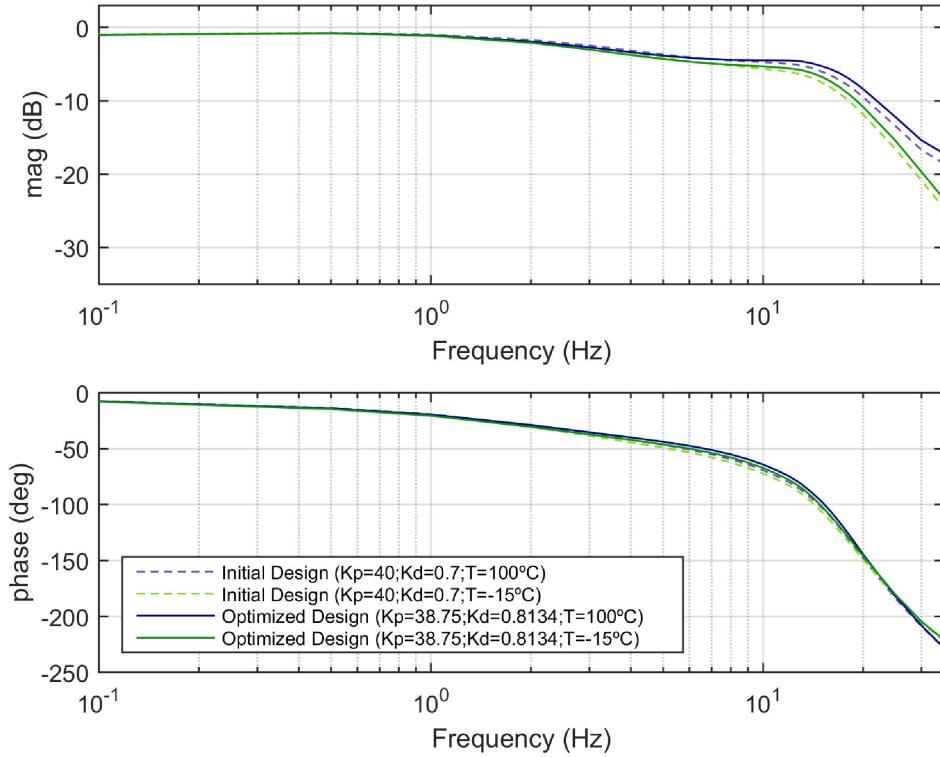


FIGURE 5.21 – Frequency Response Baseline for the PD Controller.

The closed-loop frequency response comparison is presented in Figure 5.21 and the performance requirements are also shown in Table 5.22. The closed-loop gain allowance was slightly reduced but the closed-loop phase allowance remained infinite. Other performance parameters did not change significantly.

The optimization result in this case is different from all previously obtained. None of the parameters in Table 5.22 were close to the requirement, unlike the previous results in which at least one parameter either violated or was equal to the requirement. This suggests that in this case the optimization algorithm found a local minimum, a solution in which all of its neighbors have a greater objective function than itself.

TABLE 5.22 – Requirement Compliance for PD Controller

Design Parameter	Requirement	Baseline	Optimized Controller	Difference (%)
Settling Time (ms)	< 1100	992	1006	1.4
Steady State Error (%)	< 1	0.27	0.28	3.7
Overshoot (%)	< 10	0	0	N/A
Minimum Average Rate ($^{\circ}/s$)	> 32	34.32	34.27	-0.1
Maximum Average Rate ($^{\circ}/s$)	< 36	34.32	34.27	-0.1
Closed-Loop Gain Allowance (dB)	> 10	13	12	-7.7
Closed-Loop Phase Allowance ($^{\circ}$)	> 45	<i>inf</i>	<i>inf</i>	N/A
Closed-Loop Maximum Peak (dB)	< 0.5	-0.74	-0.79	-6.8
Closed-Loop Initial Magnitude(dB)	<i>None</i>	-0.99	-1.03	-4.0
Closed-Loop Bandwidth (Hz)	<i>None</i>	5.8	5.6	-3.9

5.2.4 Proportional Integral Derivative Controller (PID)

The proportional integral derivative controller optimization execution time was 9.9 hours and required 9 iterations, as per Table 5.23. Also, the optimization yielded a proportional gain of 39.21, integral gain of 7.005e-4 and a derivative gain of 0.8067. These values were evaluated separately by Ballesteros (2015) but were not considered together as a PID design.

TABLE 5.23 – PID Controller Optimization Execution Results

Optimized Proportional Gain	39.21
Optimized Integral Gain	7.005e-4
Optimized Derivative Gain	0.8067
Optimization Execution Time (h)	9.9
Number of Iterations	9
Number of Obj. Function Evaluations	64

The dynamic stiffness response for each PID design is shown in Figure 5.22. Similarly to the PD controller, increase in dynamic stiffness was observed in frequencies above 1Hz.

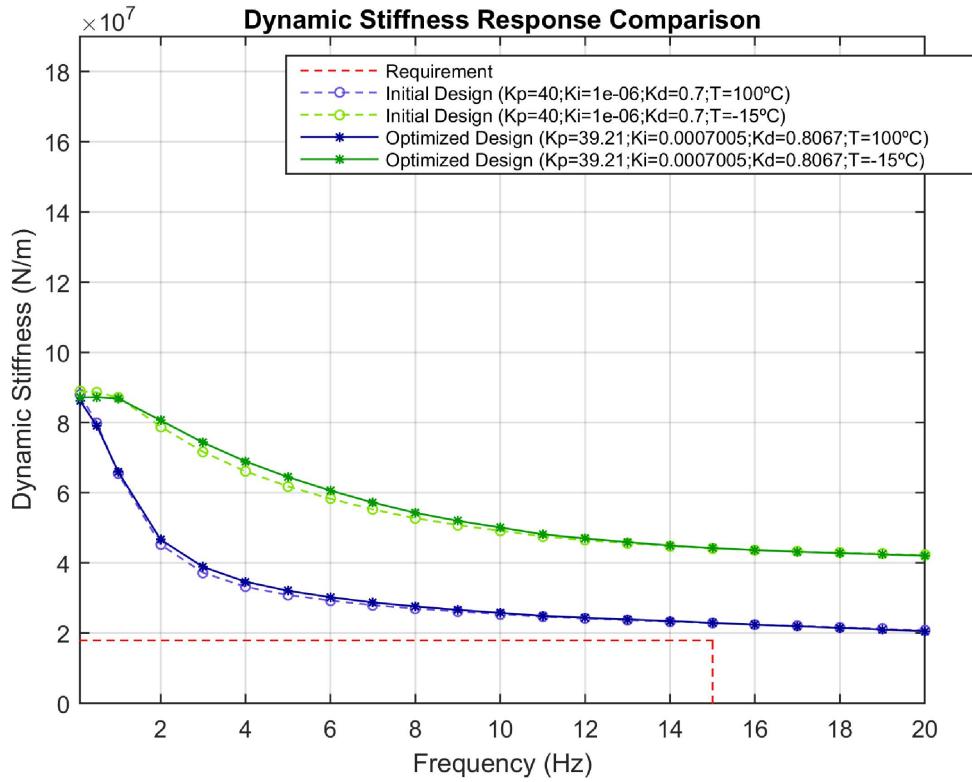


FIGURE 5.22 – Dynamic Stiffness Optimization Result for the PID Controller.

Table 5.24 show the partial values of the cost function for each frequency evaluated by the dynamic stiffness test. The lower frequencies values are also larger than the higher frequency ones since stiffness is still greater in this range.

TABLE 5.24 – PID Controller Final Solution Cost Function for Each Evaluated Frequency

Frequency (Hz)	J_i	Frequency (Hz)	J_i	Frequency (Hz)	J_i
0.1	$6.83e + 07$	5	$1.42e + 07$	11	$7.02e + 06$
0.5	$6.11e + 07$	6	$1.23e + 07$	12	$6.49e + 06$
1	$4.81e + 07$	7	$1.08e + 07$	13	$6e + 06$
2	$2.87e + 07$	8	$9.7e + 06$	14	$5.54e + 06$
3	$2.1e + 07$	9	$8.72e + 06$	15	$5.01e + 06$
4	$1.67e + 07$	10	$7.86e + 06$	0	0

The time response comparison is presented in Figure 5.23 and the performance requirements are shown in Table 5.25. The settling time was increased by 10ms and the steady state error slightly increased.

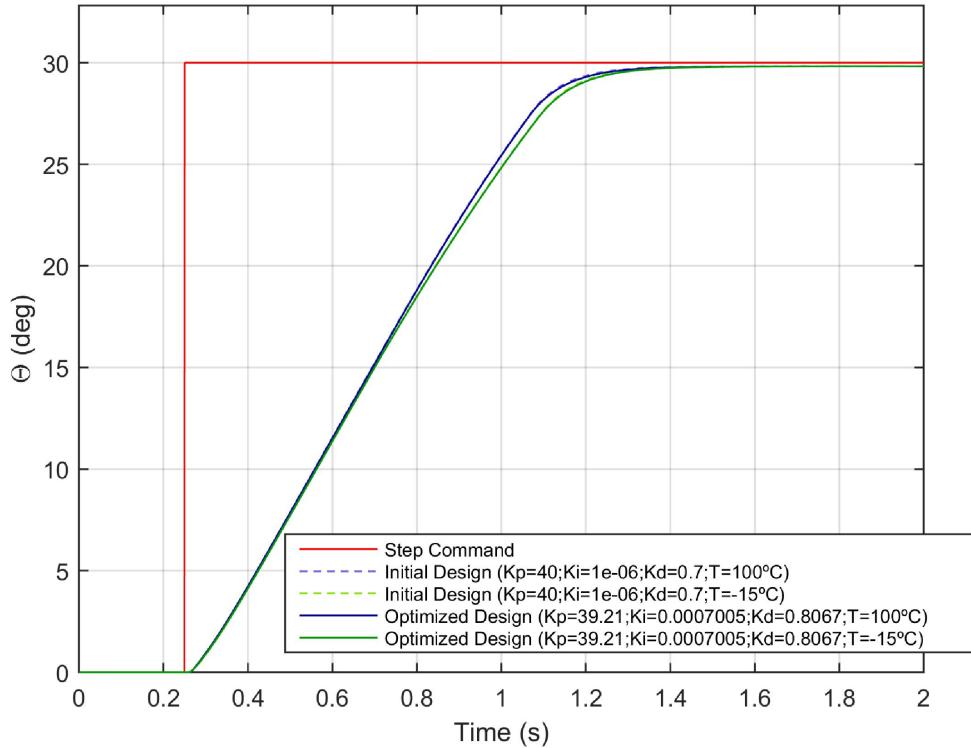


FIGURE 5.23 – Time Response Optimization Result for the PID Controller.

The closed-loop frequency response comparison is presented in Figure 5.24 and the performance requirements are also shown in Table 5.25. The closed-loop gain allowance was reduced to 12dB and the closed-loop phase allowance remained infinite. As for the PD controller, the performance parameters for the optimized PID did not reach or violate the requirements, suggesting that the solution is also a local minimum.

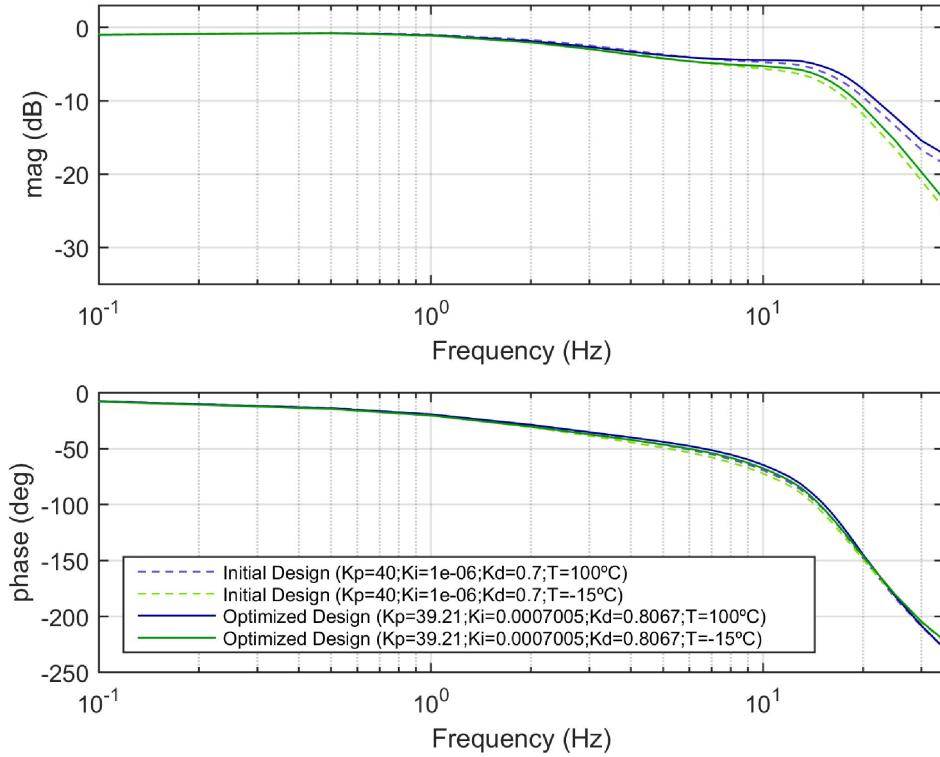


FIGURE 5.24 – Frequency Response Baseline for the PID Controller.

TABLE 5.25 – Requirement Compliance for PID Controller

Design Parameter	Requirement	Baseline	Optimized Controller	Difference (%)
Settling Time (ms)	< 1100	992	1002	1.0
Steady State Error (%)	< 1	0.27	0.28	3.7
Overshoot (%)	< 10	0	0	N/A
Minimum Average Rate (°/s)	> 32	34.32	34.28	-0.1
Maximum Average Rate (°/s)	< 36	34.32	34.28	-0.1
Closed-Loop Gain Allowance (dB)	> 10	13	12	-7.7
Closed-Loop Phase Allowance (°)	> 45	<i>inf</i>	<i>inf</i>	N/A
Closed-Loop Maximum Peak (dB)	< 0.5	-0.74	-0.77	-4.1
Closed-Loop Initial Magnitude(dB)	<i>None</i>	-0.99	-1.01	-2.0
Closed-Loop Bandwidth (Hz)	<i>None</i>	5.83	5.75	-1.4

The results obtained with the frequency weighted cost function follow the same pattern observed previously: P and PI results are fairly similar as well as PD and PID results. In addition, optimization for all controllers yielded improvement in overall dynamic stiffness as expected. Also, it was possible to improve stiffness in higher frequencies, therefore, achieving the goal of the weighted cost function. Additionally, almost all optimizations were performed in less iterations and objective function executions, resulting in reduced execution times.

The results were aligned with the hypothesis that to increase dynamic stiffness in

high frequencies a reduction in time domain performance would be required. P and PI controllers results show this clearly whereas PD and PID time performance were not affected substantially.

Other interesting observation is that P and PI controllers basically improved dynamic stiffness by having their proportional gain reduced until a constraint was violated. This happened because for these controllers only one design variable (K_p) effectively change dynamic stiffness, what naturally makes the optimization tend to one or the other bound of this gain.

On the other hand, PD and PID controllers gains were fine tuned around the initial solution and probably converged to local minimums. In this case, *fmincon* had two variables to use and many combinations of them to explore.

In summary, this section has shown that optimization with the frequency weighted cost function not only yielded controllers with better flutter suppression characteristics but also were executed faster than the optimizations with the previous cost function.

5.3 Optimization Constraints Sensitivity

The optimization results are directly affected by the constraints imposed in the evaluated solutions. This naturally leads to the question: what would be the result if one or more constraints were slightly different? This section will present a constraint sensitivity analysis that aims to answer this question. The results obtained with the simple cost function will not be considered in this analysis as their results were not satisfactory.

5.3.1 P and PI Controllers

As discussed previously, the behavior of P and PI controllers optimization was very similar and therefore their sensitivity analysis is the same.

The optimization of these controllers was effectively performed by tuning the proportional gain which has a decreasing monotonic relationship with dynamic stiffness at 15Hz i.e. stiffness at this frequency tends to increase with proportional gain reduction.

Therefore, the optimization reduced K_p until a constraint was violated, in this case the settling time, as shown in Tables 5.16 and 5.19. If the settling time requirement was greater, K_p would be smaller and the dynamic stiffness at 15Hz would be higher. In this case, closed-loop gain allowance would increase and closed-loop phase allowance would remain infinite hence, their requirements would not be violated. On the other hand, steady state error would increase and average rate would decrease and depending on how

greater is the settling time this requirements could not be achieved.

The opposite would happen if the settling time requirement was smaller. In this case, K_p would be greater, steady state error would be smaller and average rate would be higher. The closed-loop frequency response parameters would be jeopardized but would hardly violate their requirements.

For these controllers, the performance parameters did not reach the other constraints. Hence, any slight change of these requirements would not impact the obtained results.

The optimization with the weighted cost function was performed for the proportional controller with changes in the settling time constraint to confirm the expected results discussed in this section. Figure 5.25 shows the behavior of the dynamic stiffness with a variation of $\pm 10\%$ in the settling time requirement.

As expected, the dynamic stiffness at 15 Hz increased when the settling time requirement was greater while the proportional gain decreased. Also, when the settling time requirement was reduced, the dynamic stiffness decreased and the proportional gain increased.

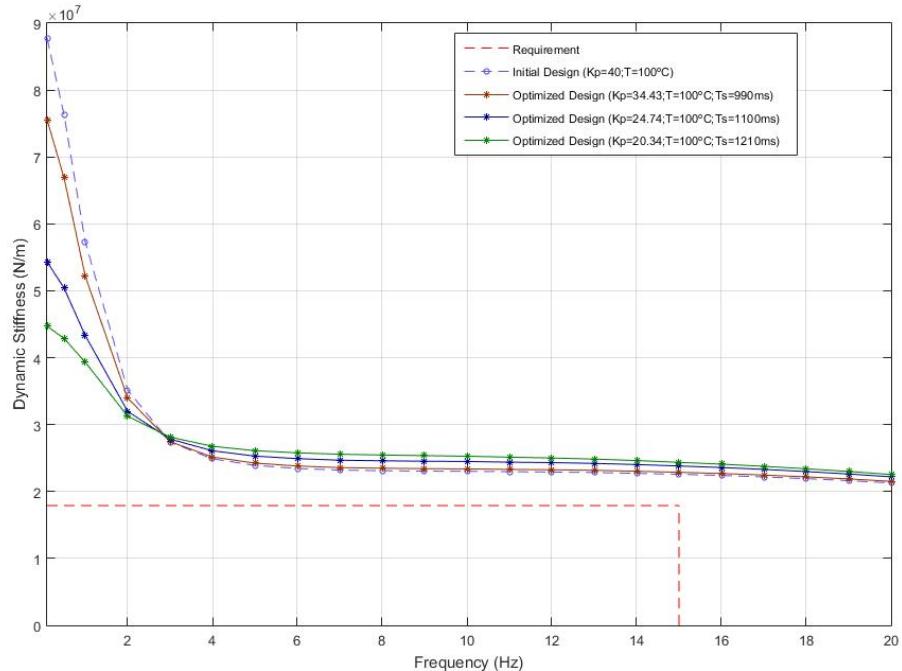


FIGURE 5.25 – Optimized Dynamic Stiffness of Proportional Controller for Different Settling Time Requirements

Table 5.26 presents the behavior of the constraints in this analysis for each Optimized Controller (OC).

TABLE 5.26 – Requirement Compliance of Proportional Controller for Different Settling Time Requirements

Design Parameter	Requirement	Baseline	OC (Ts-10%)	OC (Ts=1100ms)	OC (Ts+10%)
Settling Time (ms)	< 990/ < 1100/ < 1210	959	990	1101	1211
Steady State Error (%)	< 1	0.27	0.31	0.43	0.53
Overshoot (%)	< 10	0	0	0	0
Minimum Average Rate (°/s)	> 32	34.45	34.36	34.09	33.78
Maximum Average Rate (°/s)	< 36	34.45	34.36	34.09	33.78
Closed-Loop Gain Allowance (dB)	> 10	20.35	22.56	28.21	32.70
Closed-Loop Phase Allowance (°)	> 45	<i>inf</i>	<i>inf</i>	<i>inf</i>	<i>inf</i>
Closed-Loop Maximum Peak (dB)	< 0.5	-0.65	-0.82	-1.38	-1.90
Closed-Loop Initial Magnitude(dB)	<i>None</i>	-1.0	-1.2	-1.9	-2.5
Closed-Loop Bandwidth (Hz)	<i>None</i>	6.3	5.1	3.3	2.5

As expected, the response is faster and with less error when the settling time requirement is reduced and vice-versa. Additionally, the closed-loop gain allowance increased even more with a greater settling time requirement while the bandwidth was fairly reduced.

5.3.2 PD and PID Controllers

The behavior of PD and PID controllers optimization was also very similar. In this case, two variables, K_p and K_d , were effectively used to achieve greater dynamic stiffness. Thus, the relationship between the design variables and the objective function was more complex and no longer monotonic.

This explains why no constraint was violated or reached by the performance parameters. As shown in Tables 5.22 and 5.25, a slight change in requirements ($\pm 5\%$) would not make the obtained solutions infeasible. Even a greater change ($\pm 15\%$) would not invalidate the solutions unless it is applied to the settling time requirement. However, in this case not even the initial solution would be feasible therefore the optimization would most likely decrease dynamic stiffness at the infinite frequency.

6 Conclusion

In this work, an optimization program was developed to maximize actuator dynamic stiffness while complying with closed loop performance requirements. The problem was classified as a constrained non linear optimization and solved with the Interior Point algorithm.

A nonlinear constraint function was defined based on time and frequency performance requirements. Due to the long execution time of the frequency response test, an investigation was conducted to find an alternative method to obtain the frequency response of the model and it is presented in the Appendix A.

The first alternative was to excite the model with a chirp signal and obtain the frequency response using function *spafdr*. This option reduced the execution time by 82% but the frequency response did not match the expected for frequencies above 7Hz. The second alternative was to identify an ARX non linear model of the actuation system and run the frequency response test in this model. In this case, the execution time was reduced by 87% but the response matching was worse than the achieved with the chirp signal strategy.

Finally, a modification was implemented in the frequency response test developed by Ballesteros (2015) in which the time length of the simulations was fairly reduced. The modified test execution time is 60% faster and the frequency response matched well the expected. This modification was also implemented in the dynamic stiffness test and similar results were achieved. The test execution time dropped 43% while the response also matched the expected. Together, these modifications saved approximately 10 minutes of execution time per solution evaluation, resulting in total execution time savings of up to 10 hours.

Two cost function were considered: the first consisted in only adding the partial cost functions at each frequency while the second considered weights for each partial function.

The optimization results obtained with the simple cost function validated the developed algorithm since overall dynamic stiffness increase was observed. However, this increase was restricted to frequencies below 4Hz while at higher frequencies the dynamic stiffness actually decreased. This result was not satisfactory since the solutions yielded by

the optimization diminish flutter suppression characteristics when compared to the initial solutions.

The frequency weighted cost function improved the dynamic stiffness in higher frequencies. In this case, dynamic stiffness increase at 15Hz was observed for all controllers but this increase was higher for P and PI than for PD and PID controllers. This higher increase is related to the worse time domain performance since the settling time for these controllers was marginally violated.

The optimization execution time ranged from 5 to 10 hours but could change depending on the initial solution. Despite this, the program provides a huge productivity increase since it can achieve in hours what used to spend several days of work.

Therefore, the developed algorithm was capable of delivering controller designs with higher dynamic stiffness and compliant with performance requirements. The algorithm proved to be a valuable tool to assist actuator control loop design by evaluating an extensive number of solutions in much less time than an engineering team would spend.

6.1 Future Works

Future works of this development are either model or optimization improvements as discussed in this section.

The rudder control surface dynamics can be introduced and considered in the step and frequency response tests. This modification would provide a more complete behavior of the actuation action. This dynamics does not need to be considered to obtain the dynamic stiffness since this test aims at acquiring only the actuator contribution to the surface stiffness.

Additionally, the temperature effect on the hydraulic fluid viscosity can be introduced as well as the possibility to use different hydraulic fluids in the system. This feature would allow analysis on the actuation system robustness as well as simulations for specific applications.

Regarding optimization improvements, the first suggestion is to expand the range of controllers the program can optimize, such as partial and full state feedback controllers. Even though these controllers are not usually required in commercial aircraft design, their cost can be justified in other applications, such as fighter aircraft, where performance and weight are much more critical.

The introduction of filters in the control architecture may be considered as well. These can be feed-forward, low-pass, high-pass, band-pass or notch filters and would enable dynamic stiffness increase in specific frequencies.

Global optimization techniques may be explored for control architectures with more than one design variable, such as proportional derivative and modern state feedback controllers. In this case, the solution space is vast and by starting at multiple initial solutions the program will more likely find the optimal solution.

Using global optimization will greatly increase execution time, therefore, computing performance improvements should also be pursued. One way to reduce computational cost is to use alternative techniques to find frequency and dynamic stiffness response. The investigation conducted in this development showed outstanding reductions in execution time, therefore, it is worth to keep studying these alternative techniques to find one that will deliver both time savings and appropriate responses. Additionally, another way of reducing computational cost is by using parallel computing. The tests mentioned above are performed with a number of independent simulations that can run in parallel without any results drawback.

Furthermore, the optimization can be improved to contemplate not only the controller design but also the actuator design. Piston seal size can be included as an optimization variable in order to evaluate different actuator sizes. In this case, dynamic stiffness may be considered a constraint and the objective function would be the actuator hydraulic consumption. Additionally, a multi-objective optimization can be performed if the program attempts to minimize both hydraulic consumption and actuator weight.

Another possibility is to modify the objective function to penalize solutions that saturate the controller output. This kind of saturation may be undesirable as it reduces control loop authority outside nominal operation. The objective function can also be modified to adjust the exponents based on the dynamic stiffness behavior when the actuator is in hydraulic shim.

Finally, a different method for obtaining the dynamic stiffness can be studied. An interesting approach is to define a methodology to use the actuator frequency response to obtain its dynamic stiffness performance or a variation of it from a baseline stiffness response.

Bibliography

- ALMEIDA, A. X. C. de; FILHO, A. A. Algorithmic design for a robust control benchmark problem. **21st Brazilian Congress of Mechanical Engineering**, 2011.
- ANDERSSON, J. **Multiobjective Optimization in Engineering Design**. Tese (Ph.D. in Engineering) — Linköpings Universitet, Linköping, 2001.
- BALLESTEROS, H. M. S. **Dynamic Stiffness Enhancement of a Flight Control Actuator using control techniques**. 2015. 164 p. Thesis (M.Sc. in Engineering) — ITA, São José dos Campos, 2015.
- CHAKRABORTY DAVID TRAWICK, D. J. I.; MAVRI, D. N. Electric control surface actuator design optimization and allocation for the more electric aircraft. **Aviation Technology, Integration, and Operations Conference**, 2013.
- CONSTANTINO, C. A. **Hydraulic Actuation System Modeling: An Analysis of High Frequency Modelling**. 2010. 121 p. Thesis (M.Sc. in Engineering) — ITA, São José dos Campos, 2010.
- DORF, R. C.; BISHOP, R. H. **Modern Control Systems**. 11th. ed. Singapore: Prentice Hall, 2008.
- ETKIN, B.; REID, L. D. **Dynamics of Flight: Stability and Control**. 3rd. ed. United States: John Wiley & Sons, 1995.
- FLETCHER, R. **Practical Methods of Optimization**. 2nd. ed. United Kingdom: John Wiley & Sons, 2000.
- LJUNG, L. **System Identification: Theory for the User**. 1st. ed. United States: Prentice Hall, 1987.
- MATHWORKS. **MATLAB and Optimization Toolbox Release 2017b**. Massachusetts, United States, 2015. Disponível em:
<<http://www.mathworks.com/help/optim> - visited in Apr/2017>.
- MATHWORKS. **MATLAB and System Identification Toolbox Release 2017b**. Massachusetts, United States, 2015. Disponível em:
<<http://www.mathworks.com/help/ident> - visited in Aug/2017>.
- MERRITT, H. E. **Hydraulic Control Systems**. [S.l.]: John Wiley & Sons, 1967.

- MESSAC, A. **Optimization in Practice with MATLAB®**. 1st. ed. [S.l.]: Cambridge University Press, 2015.
- RAO, S. S. **Engineering Optimization: Theory and Practice**. 4th. ed. United States: John Wiley & Sons, 2009.
- SHMILOVITZ, D. On the definition of total harmonic distortion and its effect on measurement interpretation. **IEEE Transactions on Power Delivery**, 2005.
- SILVA, H. M. P. Alex S. F. da; AO, R. K. H. G. An optimization approach to design feedback controllers for flight control systems. **20th International Congress of Mechanical Engineering**, 2009.
- SKOGESTAD, S.; POSTLETHWAITE, I. **Multivariable Feedback Control: Analysis and Design**. 2nd. ed. United Kingdom: John Wiley & Sons, 2001.
- STEVENS, B. L.; LEWIS, F. L. **Aircraft Control and Simulation**. 1st. ed. United States: John Wiley & Sons, 1992.
- THAYER, W. J. **Transfer Functions for Moog Servovalves**. East Aurora, 1965. (RR 95-02).
- USAF. **Flight Control Systems - Design, Installation and Test of Piloted Aircraft, General Specification for**. Military specification MIL-F-9490E. [S.l.], 2008.
- VENTER, G. Review of optimization techniques. **Encyclopedia of Aerospace Engineering**, 2010.
- WRIGHT, J.; COOPER, J. E. **Introduction to Aircraft Aeroelasticity and Loads**. [S.l.]: John Wiley & Sons, 2007.

Appendix A - Frequency Response Test

This appendix will present the investigation performed during this development to find an alternative frequency response test to improve optimization execution time.

The test developed by (BALLESTEROS, 2015), referred as the *baseline* test, is executed in 12.6 minutes. The goal of this investigation is to find another method that yields a similar response in less execution time. All tests presented in this section were performed in the conditions described in section 4.3.2.

A.1 Chirp Input Frequency Response Test

The baseline test consists in obtaining the steady state response of the actuation system for each evaluated frequency and calculating the gain and phase inserted by the system in these frequencies. This is achieved through simulation of the Simulink model for each frequency and post processing of each result.

An alternative to this method is running only one simulation with a chirp signal input. The chirp signal (Figure A.1) is sinusoidal wave with time variable frequency, i. e. the signal starts at one frequency and finishes in a greater one. Hence, the signal stimulates all frequencies of interest and thus only one simulation is required.

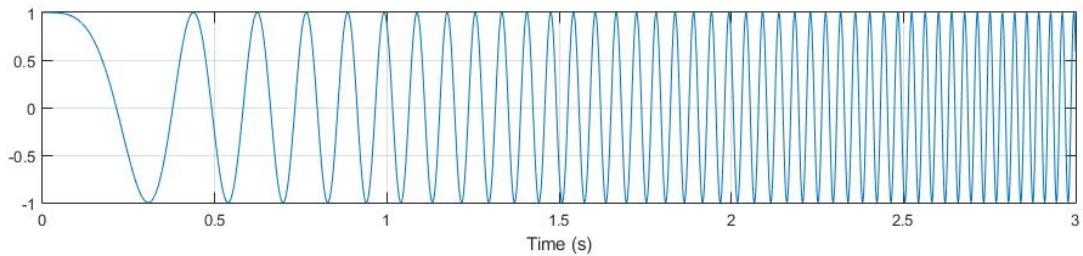


FIGURE A.1 – Chirp Signal

To obtain the system frequency response, the chirp input and the surface position

output must be recorded. Then, the frequency response is obtained through manipulation of these signals.

According to (MATHWORKS, 2015b), “if the data is given in the time domain [$u(t)$ and $y(t)$], it is first converted to the frequency domain. Then averages of $Y(w)Conj(U(w))$ and $U(w)Conj(Y(w))$ are formed over the frequency ranges w , corresponding to the desired resolution around the frequency in question. The ratio of these averages is then formed for the frequency-function estimate”.

A flowchart for a test using a chirp input is shown in Figure A.2.

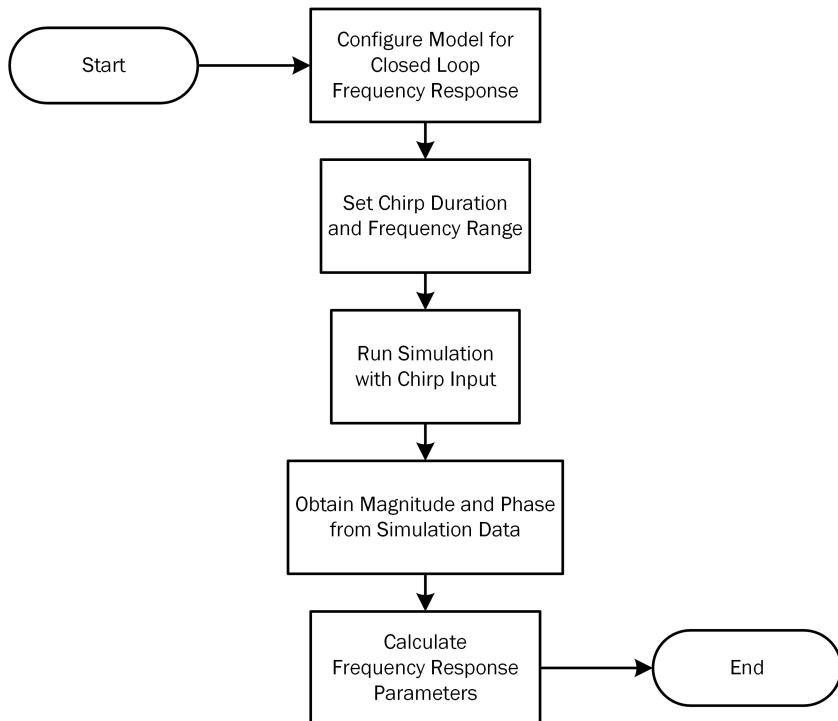


FIGURE A.2 – Flowchart of Frequency Response Test with Chirp Signal

The first step, model configuration, is the same performed in the baseline test. Then, the following step is to customize the chirp input with its initial and final frequency as well as its length. The chirp frequencies were selected to match those evaluated by the baseline test and the chirp length was tested with three different values, as shown in Table A.1.

TABLE A.1 – Chirp Frequency Response Test Parameters

Parameter	Value
Chirp Initial Frequency (Hz)	0.1
Chirp Final Frequency (Hz)	35
Chirp Signal Length (s)	20, 80 and 100

After that, the simulation is executed and the results stored. The next step is to obtain magnitude and phase from the simulation data, what was evaluated with the MATLAB function *spafdr*. This function manipulates data stored after simulation to produce a frequency-response model that can be used to obtain magnitude and phase information, as mentioned. The last step, the same for the baseline test, is to calculate the frequency constraint parameters.

Figure A.3 shows a comparison between magnitude and phase obtained with this test for different chirp signal lengths and the results obtained with baseline test.

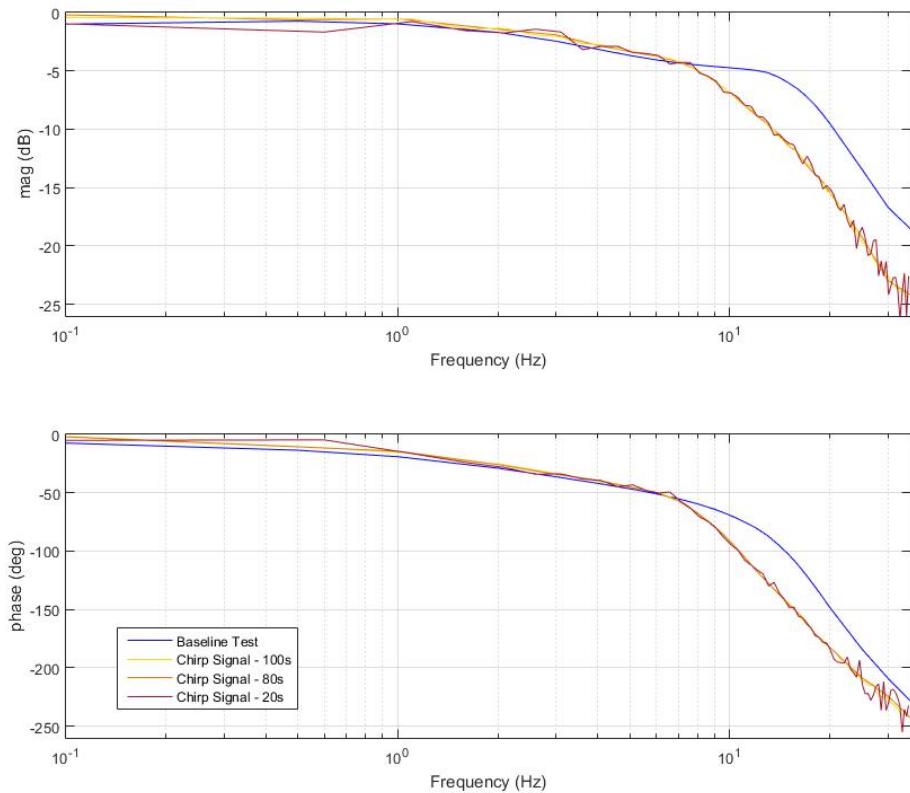


FIGURE A.3 – Baseline and Chirp Frequency Response Test Results Comparison

The figure shows that for frequencies above 7 Hz the frequency responses given by *spafdr* diverge from the baseline response, independently of the chirp signal length. This discrepancy has been attributed to model non-linearities that are not sufficiently captured either by the chirp signal itself or the *spafdr* algorithm.

Table A.2 presents a performance comparison between the different input durations. The execution time for 80 and 100 seconds chirp signals is approximately 5 times the time to execute the baseline frequency response which is not desired. On the other hand, the 20 seconds chirp input is executed in 2.3 minutes, 82% less time than the baseline response, which is a remarkable improvement in performance.

TABLE A.2 – Baseline and Chirp Frequency Response Test Comparison

Constraint	Baseline	Chirp		
		100	80	20
Chirp Signal Length (s)	-	100	80	20
Closed-Loop Gain Allowance (dB)	13.0	15.2	15.0	14.7
Closed-Loop Phase Allowance ($^{\circ}$)	<i>Inf</i>	<i>Inf</i>	<i>Inf</i>	<i>Inf</i>
Closed-Loop Bandwidth (Hz)	5.8	5.3	4.7	6.6
Execution Time (min)	12.6	60	55	2.3

Even though the execution time was reduced by 82%, the chirp input test was discarded since its result does not match the baseline test.

A.2 ARX Identification Frequency Response Test

Another alternative method considered was to use model identification techniques to find an equivalent non-linear model and then use the baseline test to obtain the frequency response of this model.

This strategy differs from the one presented previously because, instead of reducing the number of simulations, it adds one simulation and a model identification process.

Despite this, the simulations performed by the baseline test on the identified model will be faster since the simulated model will be significantly simpler. Figure A.4 shows the flowchart of this approach.

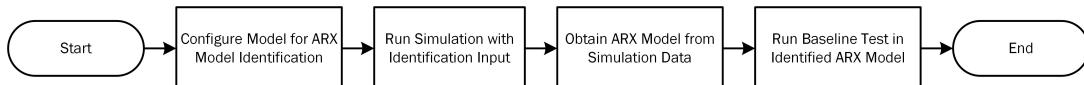


FIGURE A.4 – Flowchart of Frequency Response Test with ARX Identification

The first step is to configure the model to receive an identification signal (Figure A.5) as input. Then, the model is simulated and data is generated for the identification process.

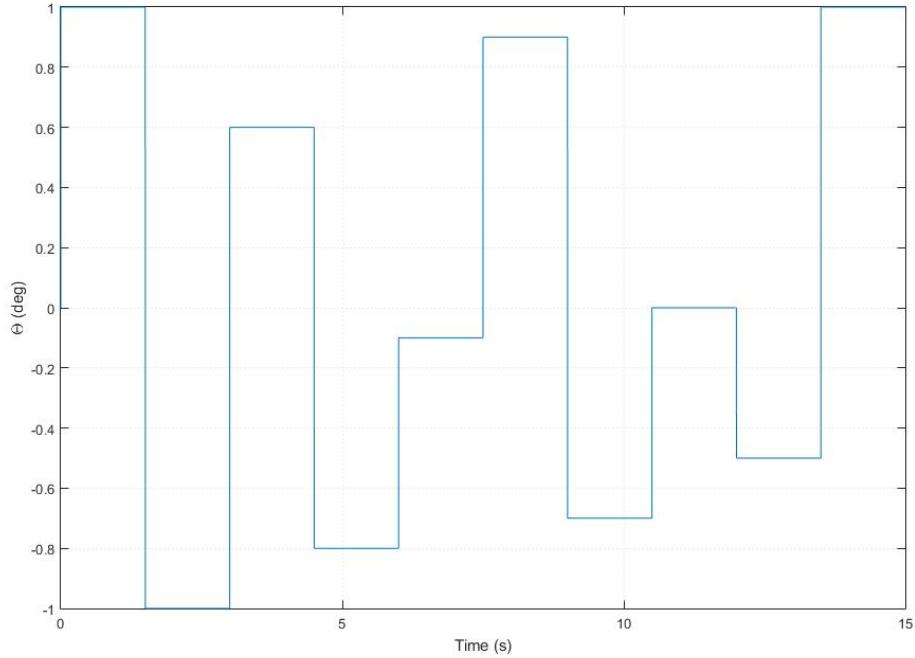


FIGURE A.5 – Identification Signal

Next, the ARX model is obtained using *nlarx*. Three types of nonlinearity estimators were used to evaluate this strategy: *wavenet*, *sigmoidnet* and *treepartition*. Default configurations of these estimators were used but several model orders were evaluated for each nonlinearity estimator.

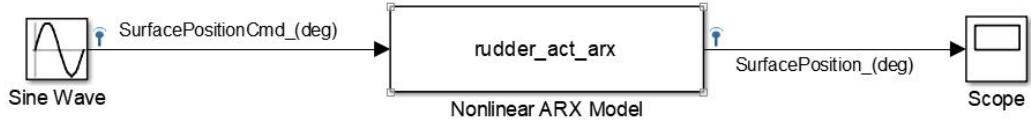


FIGURE A.6 – Simulink Model for Baseline Frequency Response Test

After obtaining the ARX model, the baseline frequency response test is performed in the simplified system model shown in Figure A.6.

Figure A.7 shows the obtained responses for the model orders that responded most closely to the baseline test response.

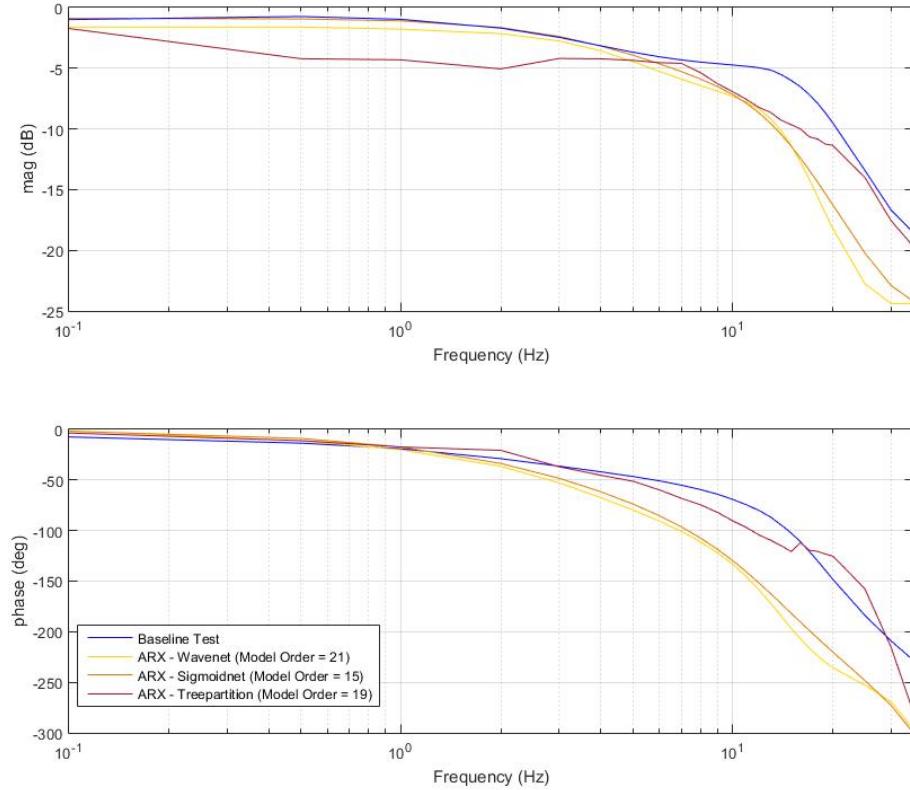


FIGURE A.7 – Baseline and ARX Model Identification Frequency Response Test Results Comparison

Figure A.7 shows that with estimators *wavenet* and *sigmoidnet* the magnitude frequency response of the identified ARX model matched the baseline test for frequencies up to 5Hz. However, above 5Hz, these responses diverged from the expected and behaved similarly to the chirp test response. The result obtained with the *treepartition* ARX model matched the baseline test only for frequencies around 7Hz and 30Hz. The phase responses matched the expected only until 1Hz, after that results for all estimators diverged.

TABLE A.3 – Baseline and ARX Model Identification Frequency Response Test Results Comparison

Constraint	Baseline	ARX Model		
		wavenet	sigmoidnet	treepartition
Nonlinearity estimator	-	wavenet	sigmoidnet	treepartition
Closed-Loop Gain Allowance (dB)	13.0	9.8	11.2	15.4
Closed-Loop Phase Allowance (°)	<i>Inf</i>	<i>Inf</i>	<i>Inf</i>	<i>Inf</i>
Closed-Loop Bandwidth (Hz)	5.8	5.2	5.0	1.6
Execution Time (min)	12.6	1.66	1.95	5.15

Table A.3 presents the performance comparison between the baseline test and the ARX model for each estimator. Despite the significant execution time reduction, up to

87%, the responses for these alternatives did not match closed-loop gain allowance and bandwidth at all. Therefore, this alternative was also discarded.

A.3 Linear Model Frequency Response Test

The next alternative evaluated was to use the baseline test to obtain the frequency response of a linearized model of the actuation system. This strategy was assessed with the linearization presented by Ballesteros (2015):

$$\begin{bmatrix} \dot{x}_v \\ \ddot{x}_v \\ \ddot{\Delta P} \\ \dot{x}_p \\ \ddot{x}_p \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ -1.77 \cdot 10^{10} & -2.11 \cdot 10^7 & -3.64 \cdot 10^3 & 0 & 0 \\ 2.30 \cdot 10^7 & 0 & 0 & -9.36 \cdot 10^{-2} & 8.16 \cdot 10^1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_v \\ \dot{x}_v \\ \ddot{x}_v \\ \Delta P \\ x_p \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 3.28 \cdot 10^7 \\ 0 \\ 0 \end{bmatrix} i$$

$$\begin{bmatrix} x_v \\ \dot{x}_v \\ \ddot{x}_v \\ \Delta P \\ x_p \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_v \\ \dot{x}_v \\ \ddot{x}_v \\ \Delta P \\ x_p \\ \dot{x}_p \end{bmatrix}$$

A comparison between the baseline strategy and this alternative is presented in Figure A.8.

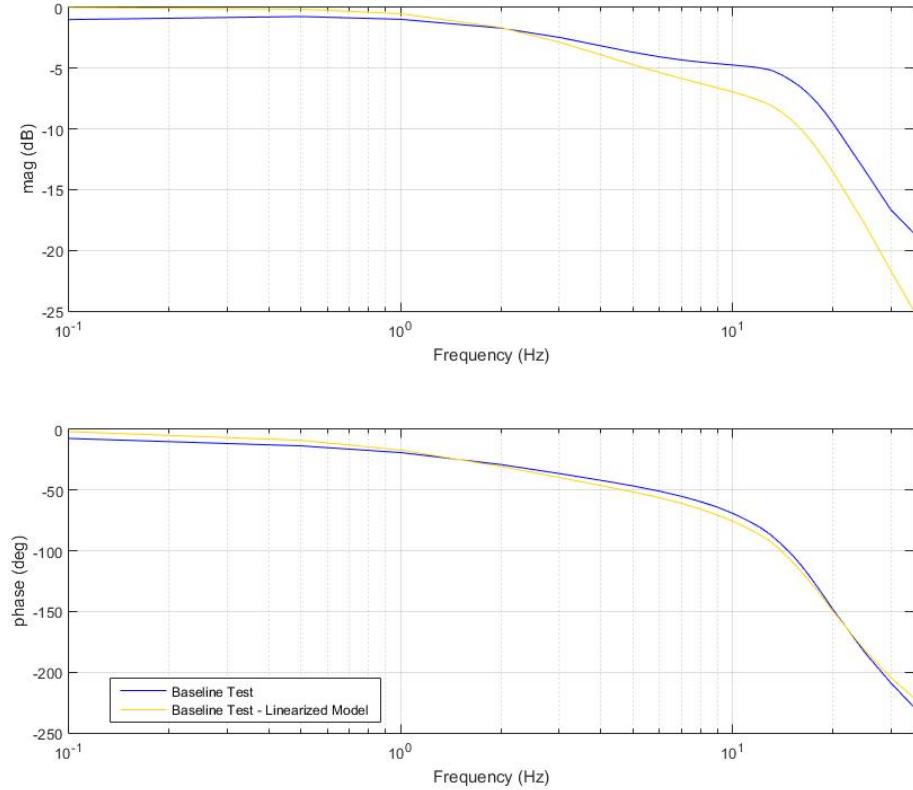


FIGURE A.8 – Non-Linear and Linearized Model Frequency Response Test Results Comparison

Figure A.8 shows that the frequency response of the linearized model does not match the non-linear model response, specially in magnitude. This indicates that even for small inputs, of 0.5 degrees, the linear model is not able to replicate the non-linear model frequency response.

Further investigation showed that for a 0.5 degree sine amplitude the EHSV current reached almost 4mA. These value is distant from the point where the system was linearized (0mA). EHSV current step input simulations were performed and confirmed that some dynamics of the model are not well represented in the linear model for this current.

Table A.4 shows the impact of this behavior in closed-loop gain allowance and bandwidth of the linearized model that were fairly different from the ones obtained by the baseline test. Despite this, the execution time of this alternative was approximately 60% lower than the baseline, as expected.

TABLE A.4 – Non-Linear and Linearized Model Frequency Response Comparison

Constraint	Baseline	Linearized Model
Closed-Loop Gain Allowance (dB)	13.0	17.6
Closed-Loop Phase Allowance (°)	<i>Inf</i>	<i>Inf</i>
Bandwidth (Hz)	5.8	3.1
Execution Time (min)	12.6	4.28

At this point, it was clear that finding alternative methods that could both save execution time and replicate the baseline's response was very challenging. The next step then was in the direction to improve the baseline test's execution time, rather than to find an alternative strategy.

A.4 Modified Baseline Frequency Response Test

Since the execution time of the baseline test is mostly the time to execute all simulations, their duration was revisited. Instead of calculating simulation's length only from input signal cycles, a non variable time parameter was introduced with the aim to capture model dynamics settling time. The duration of each simulation was calculated with Equation A.1.

$$T_{\text{sim}} = T_{\text{dyn}} + K_{\text{int}} \times T_p \quad (\text{A.1})$$

T_{sim} is the simulation duration for each frequency whereas T_{dyn} is the model dynamics settling time. This constant parameter was obtained empirically from measuring the amplitude of actuator surface position in simulations with sine input for several frequencies which yielded a 0.5 seconds settling time.

Parameter K_{int} is the number of integration cycles for the respective frequency and T_p is the frequency's period. K_{int} was selected to minimize simulation time but in a way that did not jeopardize the frequency response. Frequencies 0.1, 0.5 and 1 Hz were integrated for only one cycle while three cycles were considered for 2 and 3 Hz and nine cycles for frequencies from 4 to 35 Hz. A comparison between the baseline and modified tests is presented in Figure A.9.

The figure shows that magnitude and phase responses for both tests match in most frequencies. Even though there is a difference between the curves at around 13 Hz, the overall response of the modified test is suitable for constraint evaluation purposes. It is important to remark that the baseline test will be used to evaluate initial and optimized controllers prior and after optimization.

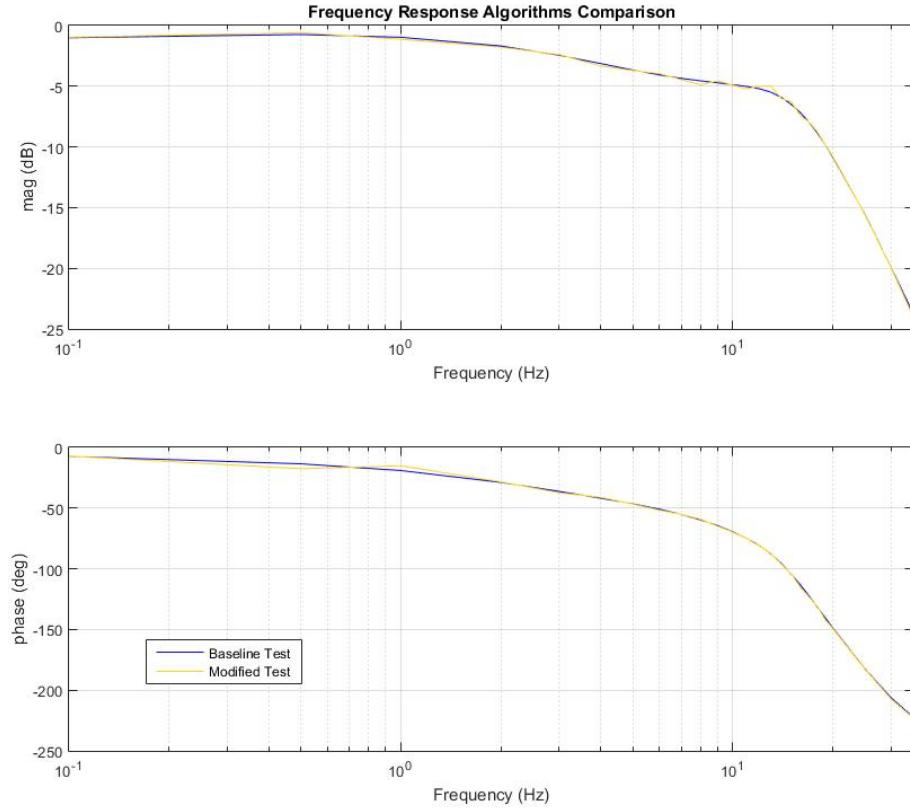


FIGURE A.9 – Baseline and Modified Frequency Response Test Results Comparison

Table A.5 shows the closed-loop gain and phase allowances for both tests are the same and that bandwidth slightly increases. Also, execution time reduced considerably, falling from 12.6 to 4.85 minutes.

TABLE A.5 – Baseline and Modified Frequency Response Test Results Comparison

Constraint	Baseline	Modified
Closed-Loop Gain Allowance (dB)	13.0	13.0
Closed-Loop Phase Allowance (°)	<i>Inf</i>	<i>Inf</i>
Bandwidth (Hz)	5.8	6.0
Execution Time (min)	12.6	4.85

The modified test is executed in approximately 60% less time than the baseline. Table A.6 presents an execution time breakdown for both tests and the difference between them.

Simulations in lower frequencies have the longest durations due to their large periods, even though they have few integration cycles. Despite this, time savings were observed in all simulations and most of them were reduced by more than 50%.

TABLE A.6 – Baseline and Modified Frequency Response Test Execution Time Comparison

Frequency (Hz)	Baseline Exec. Time (s)	Modified Exec. Time (s)	Time Reduction (s)	Percentual Red. (%)
0.1	171	59.8	-111	-65.1
0.5	36.0	15.3	-20.7	-57.5
1	18.9	9.76	-9.10	-48.3
2	20.3	12.5	-7.86	-38.7
3	18.6	9.71	-8.84	-47.7
4	18.3	13.7	-4.59	-25.1
5	18.2	11.9	-6.29	-34.6
6	18.7	10.6	-8.11	-43.4
7	18.6	9.70	-8.90	-47.9
8	20.8	9.01	-11.8	-56.6
9	18.1	8.49	-9.59	-53.0
10	18.0	8.03	-9.97	-55.4
11	18.1	7.75	-10.3	-57.1
12	18.1	7.45	-10.7	-58.9
13	18.2	7.11	-11.1	-61.0
14	18.3	7.09	-11.2	-61.2
15	18.0	6.68	-11.3	-62.8
16	18.2	6.52	-11.7	-64.1
17	18.0	6.46	-11.6	-64.2
18	18.6	6.38	-12.2	-65.7
19	18.2	6.16	-12.0	-66.1
20	18.0	6.08	-11.9	-66.2
25	18.0	5.80	-12.2	-67.8
30	18.0	5.44	-12.6	-69.8
35	18.3	5.27	-13.0	-71.2
Total	10.5 min	4.38 min	-6.15 min	-58.4%

In summary, several strategies to obtain the frequency response of the actuation system model were investigated and none of them were suitable to substitute the baseline test. Despite this, considerable execution time reductions were observed also in all strategies what indicates that it is still worth to study these strategies.

Nevertheless, an alternative solution was adopted by reassessing the length of simulations in the baseline frequency response test. This alternative test was used solely to evaluate optimization constraints while the frequency performance of initial and optimized controllers was still evaluated with the baseline test.

Appendix B - Frequency Response of Non-Linear Systems

This appendix will present the method used in this work to obtain the magnitude and phase response of the actuation system for a given sinusoidal input.

The model block diagram is shown in Figure B.1. In the frequency response test, the sinusoidal surface command is inserted in signal *SurfacePositionCmd_(deg)* and the output is the surface position in degrees represented by the signal *SurfacePosition_(deg)*.

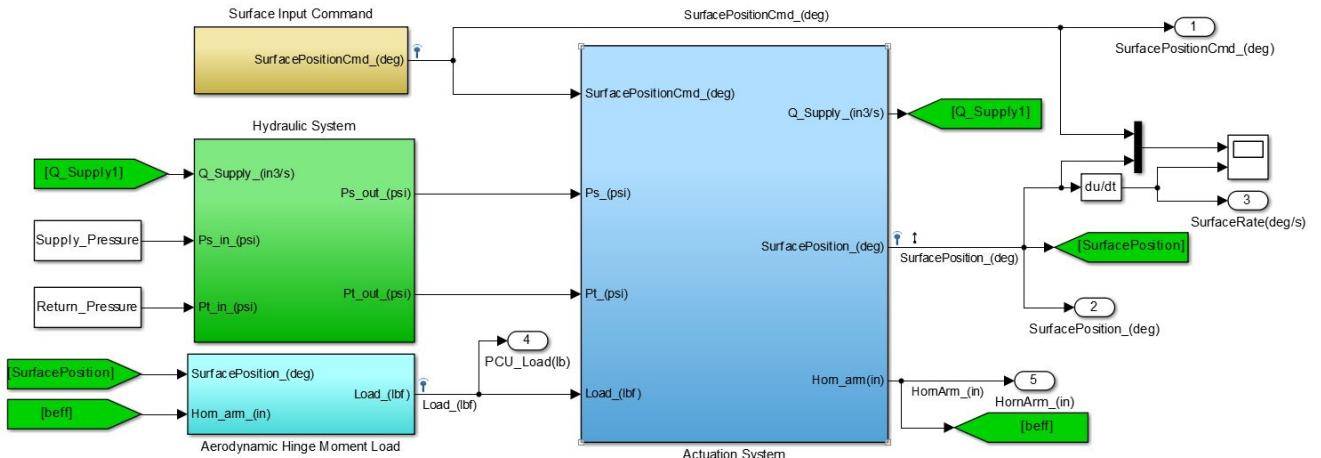


FIGURE B.1 – Actuation System and Interfaces Simulink Model

Since the actuation system is non-linear, the output signal may contain components in frequencies different from the one of the input. Figure shows an example of a 20 Hz sine surface command with an amplitude of 0.5 degrees. The figure also presents the correspondent output signal and it is possible to observe that the output is not a pure sine and therefore was distorted by the system.

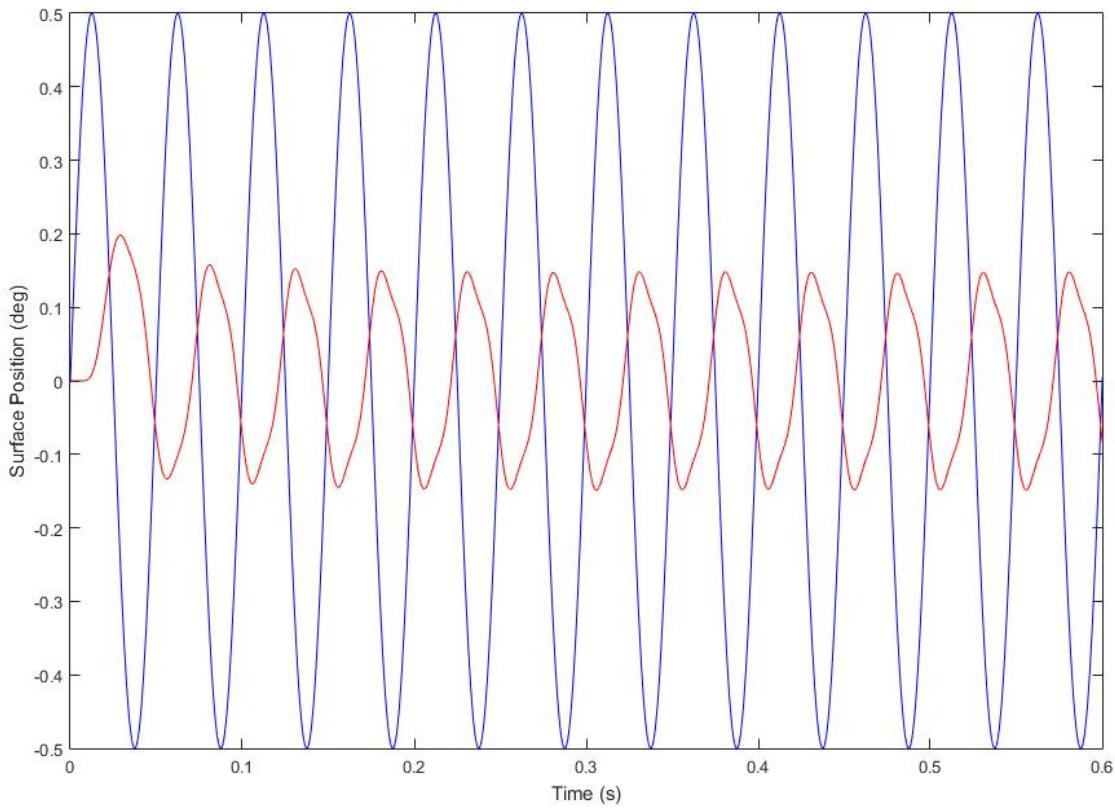


FIGURE B.2 – System Output for a 20Hz sine input signal

Hence, the system output can be a linear combination of sine signals of different frequencies and for this reason a method to find the magnitude and phase of the system transfer function needs to be defined. The Total Harmonic distortion (THD) of a signals quantifies the relative power between the fundamental frequency and the harmonics and it can be used as a measure of how non-linear is the system.

Figure B.3 shows the frequency power spectrum of the system output presented previously. In this case, the power of the fundamental frequency is -18.54 dB and the Total Harmonic Distortion (THD) is -31.8 dB or 2.5%.

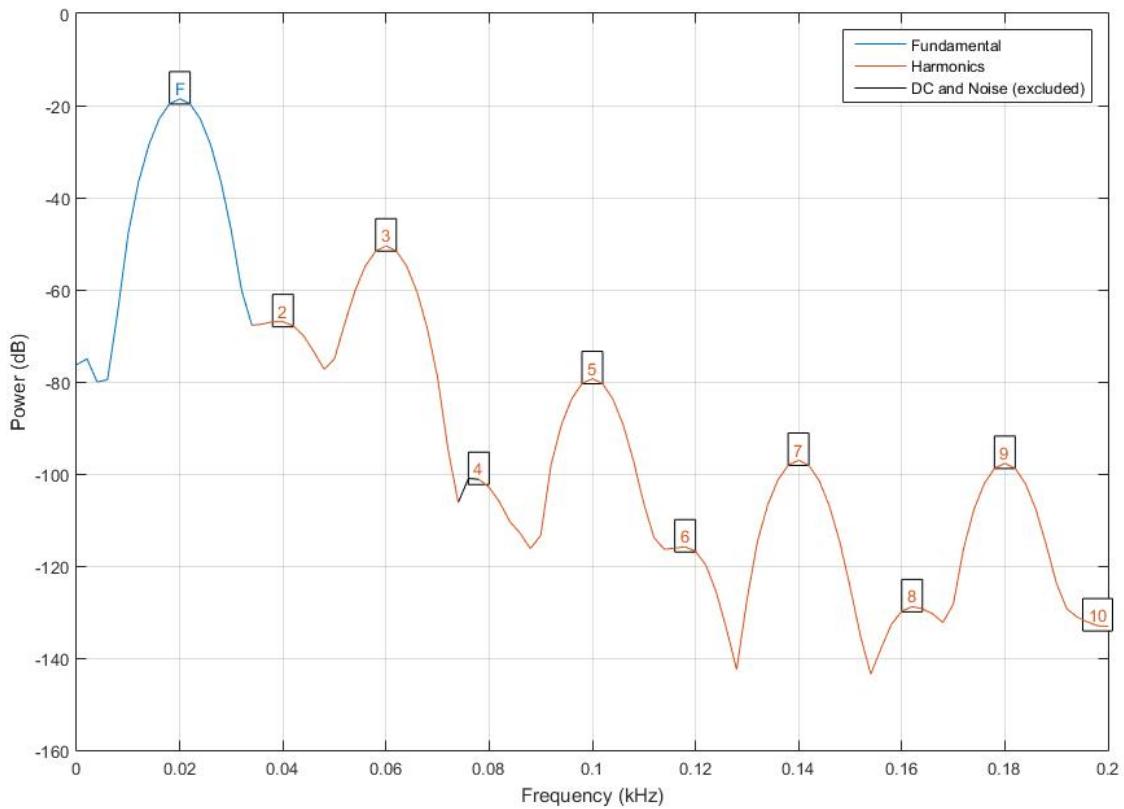


FIGURE B.3 – System Output for a 20Hz sine input signal

In this work it was considered an acceptable THD of 10% or lower. In this cases, the system magnitude and phase responses were calculated considering only the component of the fundamental frequency in the output signal of the system.

Table B.1 below shows the THD in each evaluated frequency for a PD controller with $K_P = 40$ and $K_D = 0.7$.

TABLE B.1 – THD values of a PD controller frequency response

Frequency (Hz)	THD (%)	Frequency (Hz)	THD (%)	Frequency (Hz)	THD (%)
0.1	6.21	8	3.18	17	5.29
0.5	4.61	9	3.51	18	4.11
1	4.05	10	2.72	19	3.15
2	4.90	11	2.38	20	2.46
3	2.98	12	2.58	25	0.99
4	2.83	13	3.63	30	0.42
5	2.76	14	6.09	35	0.20
6	2.76	15	7.83		
7	2.89	16	6.82		

The table shows that the maximum value of the THD is 7.83% at 15 Hz, therefore the proposed methodology was applied for all frequencies. This was also considered for the four types of controllers evaluated in this work (P, PI, PD and PID) since the controllers have linear transfer functions and should not interfere in the spectrum of the output signal.

FOLHA DE REGISTRO DO DOCUMENTO

1. CLASSIFICAÇÃO/TIPO DP	2. DATA 20 de março de 2018	3. DOCUMENTO Nº DCTA/ITA/DP-012/2018	4. Nº DE PÁGINAS 120
5. TÍTULO E SUBTÍTULO: Dynamic stiffness optimization of a flight control actuation system using control techniques			
6. AUTOR(ES): Pedro Henrique Oliveira de Paula			
7. INSTITUIÇÃO(ÓES)/ÓRGÃO(S) INTERNO(S)/DIVISÃO(ÓES): Instituto Tecnológico de Aeronáutica – ITA			
8. PALAVRAS-CHAVE SUGERIDAS PELO AUTOR: Atuador; Rigidez Dinâmica; Otimização			
9. PALAVRAS-CHAVE RESULTANTES DE INDEXAÇÃO: Atuadores; Otimização; Sistema de circuito fechado; Controle de voo; Aerodinâmica; Engenharia aeronáutica.			
10. APRESENTAÇÃO: <input type="checkbox"/> Nacional <input checked="" type="checkbox"/> Internacional ITA, São José dos Campos. Curso de Mestrado Profissional em Engenharia Aeronáutica. Programa de Pós-Graduação em Engenharia Aeronáutica e Mecânica. Área de Sistemas Aeroespaciais e Mecatrônica. Orientador: Prof. Dr. Alberto Adade Filho; coorientador: MSc. Raphael das Neves Calvo. Defesa em 06/03/2018. Publicada em 2018.			
11. RESUMO: <p>Fly-by-wire flight control systems have been supporting innovations in the aircraft industry by providing envelope protection, tailored control laws and maintainability. To perform surface position control, most fly-by-wire systems rely on electro-hydraulic servo actuators and one of their design requirements is the dynamic stiffness. Traditionally, dynamic stiffness requirements are met by either mass balancing of the control surface or by increasing the piston area of hydraulic actuators but both solutions increase aircraft weight. Previous investigations on alternatives to these approaches demonstrated that with an adequate closed-loop system design it is possible to increase the dynamic stiffness of an aerodynamic control surface actuator positioning system, improving the systems' flutter rejection characteristics without jeopardizing the performance requirements usually demanded in this application, allowing a revision of the design characteristics in favor of the global system. However, the design of control loop to increase dynamic stiffness demands valuable resources. To overcome this, optimization techniques can be used since they allow the investigation of a vast number of solutions in much less time than conventional processes. Also, they help engineers understand the effects of parameters and constraints of models and use this information to make better informed decisions. The objectives of this work are to use optimization techniques in hydraulic actuator design to reduce control loop design cycles, maximize dynamic stiffness response and comply with time and frequency domain performance requirements.</p>			
12. GRAU DE SIGILO: (X) OSTENSIVO <input type="checkbox"/> RESERVADO <input type="checkbox"/> CONFIDENCIAL <input type="checkbox"/> SECRETO			