MIXED INTEGER QUADRATIC PROGRAMMING AUTOPILOT DESIGN BASED ON MODEL PREDICTIVE CONTROL FOR AIR DEFENCE MISSILE

MODEL ÖNGÖRÜLÜ KONTROLE DAYALI HAVA SAVUNMA FÜZESİ İÇİN KARIŞIK TAMSAYILI KARESEL PROGRAMLAMALI OTOPİLOT TASARIMI

MEHMET ALUÇ

PROF. DR. S.ÇAĞLAR BAŞLAMIŞLI
Supervisor

Submitted to
Graduate School of Science and Engineering of Hacettepe University
as a Partial Fulfillment to the Requirements
for the Award of the Degree of Master of Science
in Mechanical Engineering

June 2024

ETHICS

In this thesis study, prepared in accordance with the spelling rules of Institute of Graduate Studies in Science of Hacettepe University,

I declare that

- all the information and documents have been obtained in the base of the academic rules.
- all audio-visual and written information and results have been presented according to the rules of scientific ethics
- in case of using others works, related studies have been cited in accordance with the scientific standards
- all cited studies have been fully referenced
- I did not do any distortion in the data set
- and any part of this thesis has not been presented as another thesis study at this or any other university.

... / ... / .....

MEHMET ALUÇ

YAYINLAMA FİKRİ MÜLKİYET HAKLARI BEYANI

Enstitü tarafından onaylanan lisansüstü tezimin tamamını veya herhangi bir kısmını, basılı (kağıt) ve elektronik formatta arşivleme ve aşağıda verilen koşullarla kullanıma açma iznini Hacettepe üniversitesine verdiğimi bildiririm. Bu izinle Üniversiteye verilen kullanım hakları dışındaki tüm fikri mülkiyet haklarım bende kalacak, tezimin tamamının ya da bir bölümünün gelecekteki çalışmalarda (makale, kitap, lisans ve patent vb.) kullanım hakları bana ait olacaktır.

Tezin kendi orijinal çalışmam olduğunu, başkalarının haklarını ihlal etmediğimi ve tezimin tek yetkili sahibi olduğumu beyan ve taahhüt ederim. Tezimde yer alan telif hakkı bulunan ve sahiplerinden yazılı izin alınarak kullanması zorunlu metinlerin yazılı izin alarak kullandığımı ve istenildiğinde suretlerini Üniversiteye teslim etmeyi taahhüt ederim.

Yükseköğretim Kurulu tarafından yayınlanan **"Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge"** kapsamında tezim aşağıda belirtilen koşullar haricince YÖK Ulusal Tez Merkezi / H. Ü. Kütüphaneleri Açık Erişim Sisteminde erişime açılır.

☐ Enstitü yönetim kurulu kararı ile tezimin erişime açılması mezuniyet tarihimden itibaren 2 yıl ertelenmiştir.

☐ Enstitü yönetim kurulu gerekçeli kararı ile tezimin erişime açılması mezuniyet tarihimden itibaren .... ay ertelenmiştir.

☐ Tezim ile ilgili gizlilik kararı verilmiştir.

... / ... / .....

MEHMET ALUÇ

ABSTRACT

MIXED INTEGER QUADRATIC PROGRAMMING AUTOPILOT DESIGN BASED ON MODEL PREDICTIVE CONTROL FOR AIR DEFENCE MISSILE

MEHMET ALUÇ
Master of Science, Mechanical Engineering
Supervisor: Prof. Dr. S.Çağlar Başlamışlı
June 2024, 94 pages

In modern combat scenarios, the effectiveness of tactical missiles relies heavily on their maneuverability and control systems. Enhancing missile performance, considering the challenges posed by highly maneuverable or high-speed ballistic missile targets and the need for faster response times, depends on actuator capability. To address the demand for enhanced maneuverability, in addition to aerodynamic surfaces, secondary actuators such as lateral thruster systems are explored. Unlike traditional control mechanisms, these systems utilize reaction forces from ejected high-velocity gases, offering consistent control efficacy across all flight phases, especially in low dynamic pressure conditions where aerodynamic surfaces may be less effective. However, the on-off nature and limited number of propulsion systems necessitate defining constraints in controller design. In addition to lateral thruster constraints, aerodynamic surfaces have limitations of angle and angular rate.

Within the scope of the thesis, aiming to provide an optimal solution under defining constraints, an approach for Mixed Integer Quadratic Programming (MIQP) based on Model Predictive Control (MPC) is proposed. In MPC, a dynamic model of the system is used to forecast its future behavior by considering its current state and the control inputs applied.

MIQP is an optimization problem characterized by a quadratic objective function and decision variables that must be integers. MIQP is employed because the lateral thruster is treated as an integer variable. In the study, the aerodynamic fin angle constraint, which is another system constraint, has been defined as an inequality constraint. However, the system struggles with stability when the fin constraint is active while there is already an equality constraint in the system. Therefore, one of the aims of this thesis is to update the value of the fin in the penalty function to ensure that the fin constraint does not become active. This is attempted using an approach called the ”MPC-Based Adaptive Weight Estimation Algorithm” method. The method simply aims to reduce the effect of the fin by increasing the weight if the fin limit is active, checking narrow windows from the MPC solution to see whether the fin limit is triggered.

The main target of the autopilot (MIQP-MPC) is to track the acceleration command that comes from guidance. The autopilot is designed for only the pitch channel of a missile in the linear domain, and its performance is analyzed for different scenarios. Moreover, the robustness of the autopilot against the jet interaction effect between lateral thrusters and the air stream is also examined. This interaction in missile systems is a critical aspect of aerodynamic control and creates complex aerodynamic forces and moments, which can alter the missile’s flight path. The expelled gas jets disrupt the smooth flow of air around the missile, potentially causing turbulence and affecting overall stability and control. Finally, to compare the performance of the autopilot integrated with the lateral thruster with the conventional fin-controlled autopilot structure and to evaluate the designed autopilot in the guidance loop, a ballistic missile target scenario was created and the results were evaluated.

Keywords: Missile, Optimal Control, MPC, Constraint MPC, MIQP, Autopilot, Lateral Thruster,

ÖZET

MODEL ÖNGÖRÜLÜ KONTROLE DAYALI HAVA SAVUNMA FÜZESİ İÇİN KARIŞIK TAMSAYILI KARESEL PROGRAMLAMALI OTOPİLOT TASARIMI

MEHMET ALUÇ
Yüksek Lisans, Makina Mühendisliği
Danışman: Prof. Dr. S.Çağlar Başlamışlı
Haziran 2024, 94 sayfa

Modern muharebe senaryolarında, taktik füzelerin etkinliği büyük ölçüde manevra kabiliyeti ve kontrol sistemlerine bağlıdır. Yüksek manevra kabiliyetli veya hızlı balistik füze hedeflerinin yarattığı zorluklar ve daha hızlı tepki süreleri ihtiyacı göz önüne alındığında, füze performansını artırmak, eyleyici kapasitesine dayanır. Manevra kabiliyetini artırmak için aerodinamik yüzeylerin yanı sıra, yanal itici sistemler gibi ikincil eyleyiciler de incelenmektedir. Geleneksel kontrol mekanizmalarından farklı olarak, bu sistemler fırlatılan yüksek hızlı gazlardan gelen reaksiyon kuvvetlerini kullanarak, özellikle aerodinamik yüzeylerin düşük dinamik basınç koşullarında daha az etkili olabileceği tüm uçuş aşamalarında tutarlı bir kontrol verimliliği sunar. Ancak açma-kapama niteliğinde olmaları ve sınırlı sayıda itki sistemi bulunması, kontrolör tasarımında kısıtların tanımlanmasını gerektirir. Yanal itici kısıtlarına ek olarak, aerodinamik yüzeylerin açı ve açısal hız sınırlamaları vardır.

Bu çalışma tanımlanan kısıtlar altında optimal çözüm sağlamak için MÖK (Model Öngörülü Kontrol) tabanlı ”Karışık Tamsayılı Karesel Programlama (KTKP)” yaklaşımını önermektedir. MÖK'te, kontrol edilecek sistemin dinamik modeli, mevcut durumu ve uygulanan kontrol girdilerine dayalı olarak gelecekteki davranışını tahmin etmek için kullanılır. MÖK, amaç fonksiyonunun kuadratik olduğu ve karar değişkenlerinin bazılarının veya tamamının tamsayı olması gereken bir optimizasyon problemidir. MIQP'nin kullanılmasının nedeni, yanal iticinin tamsayı bir değişken olarak kabul edilmesidir. Çalışmada, bir diğer sistem kısıtı olan aerodinamik kanatçık açı kısıtı, eşitsizlik kısıtı olarak tanımlanmıştır. Ancak, zaten sistemde bir eşitlik kısıtı varken kanatçık kısıtının aktif olduğu durumlarda sistem kararlılık açısından zorlanır. Bu nedenle, tezin amaçlarından biri, kanatçık kısıtının aktif olmamasını sağlamak için ceza fonksiyonundaki kanatçığın değerini güncellemektir. Bu, MÖK tabanlı ağırlık tahmin yöntemi olan yenilikçi bir yaklaşımla denenmektedir. Yöntem basitçe MÖK çözümünden dar pencereler kontrol ederek kanatçık sınırının tetiklenip tetiklenmediğini görmek ve kanatçık sınırı aktifse ağırlığı artırarak kanatçığın etkisini azaltmayı amaçlamaktadır. otopilot ana hedefi, güdümden gelen ivme komutunu takip etmektir. otopilot, bir füzenin yalnızca pitch kanalında doğrusal alanda tasarlanmış ve performansı farklı senaryolar için analiz edilmiştir. Ayrıca, otopilot yan iticiler ile hava akışı arasındaki jet etkileşimi etkisine karşı gürbüzlüğü de incelenmiştir. Bu etkileşim, aerodinamik kontrolün kritik bir yönüdür ve füzenin uçuş yolunu değiştirebilecek karmaşık aerodinamik kuvvetler ve momentler yaratır. Fırlatılan gaz jetleri, füzenin etrafındaki hava akışının düzgünlüğünü bozarak potansiyel olarak türbülans yaratabilir ve genel kararlılığı ve kontrolü etkileyebilir. Son olarak, yan itici ile entegre edilen otopilot performansını geleneksel kanatçık kontrollü otopilot yapısı ile karşılaştırmak ve tasarlanan otopilotu güdüm döngüsünde değerlendirmek için bir balistik füze hedef senaryosu kurgulanmış ve sonuçlar değerlendirilmiştir.

Keywords: Fuze, Optimal Kontrol, MPC, Kısıt altında MPC, MIQP, Otopilot, Yan İtici, Adaptive Ağırlık,

ACKNOWLEDGEMENTS

I would like to convey my appreciation to my advisor, Prof. Dr. S. Çağlar Başlamışlı, for his guidance and keen interest throughout this study. I also wish to express my sincere thanks to my committee members, Prof. Dr. Yiğit Yazıcıoğlu, Prof. Dr. Barış Sabuncuoğlu, Assist. Prof. Dr. Selçuk Himmetoğlu, and Assist. Prof. Dr. Kutluk Bilge Arıkan, for their valuable contributions and insightful comments on this study.

I would also like to thank all my colleagues for the insightful technical discussions that broadened my perspective and advanced my technical knowledge. In particular, I would like to extend special thanks to Suzan Kale Güvenç for her understanding, dedication, and encouragement.

I would like to thank my father Ayhan Aluç and my mother Dilek Aluç, for their love and the significant effort they have put into my education. I would like also to express my special thanks to my siblings Kemal, Mert, and Melek Aluç, whose love and support I always feel.

Last but not the least important, I am deeply grateful to my dear wife, Tuğçe Aluç, for her endless support, love and patience.

CONTENTS

Page

ABSTRACT i

ÖZET iii

ACKNOWLEDGEMENTS v

CONTENTS vi

TABLES viii

FIGURES ix

ABBREVIATIONS xii

1 INTRODUCTION 1

  1.1 Scope Of The Thesis 2

  1.2 Contributions 3

  1.3 Organization 3

2 Literature Survey 4

3 SIX DEGREE OF FREEDOM EQUATIONS OF MOTION 10

  3.1 Reference Coordinate Frames 10

  3.2 Kinematics 12

  3.3 Translational Motion 14

  3.4 Rotational Motion 15

  3.5 Incidence Angle Dynamics 16

  3.6 Forces and Moments 17

4 Linear Domain Analysis 20

  4.1 Linearization 20

  4.2 Actuator Dynamics 21

  4.3 System Model 24

5 Model Predictive Control 27

  5.1 Discrete State Space Representation 27

  5.2 Optimization Problem Definition 28

  5.3 Receeding Horizon Control 31

5.4 Constrained Optimization 32
6 Design of MPC 34
6.1 Discretization 34
6.2 Tuning Parameters 37
6.3 Unconstrained Solution 38
6.4 Lateral Thruster Constraint Solution with Mixed Integer Quadratic Programming 42
6.5 Lateral Thruster and Elevator Deflection Constraints Solution MPC based Weight Adaptation 48
6.6 Robustness Analysis 57
6.7 Dual vs Aero Only Autopilot Comparison 62
6.8 Guidance Performance 64
7 CONCLUSION 71

TABLES

Page

Table 6.1 Missile's overall parameters 35

Table 6.2 Aerodynamic Coefficient 35

Table 6.3 Scenario Tunable Parameters 68

Table 6.4 Dual vs Aero-Only Performance Comparison 70

FIGURES

Page

Figure 2.1 Structure of Variable Structure Control 6
Figure 2.2 Structure of Feedback Linearization 6
Figure 2.3 Structure of Neural Network Based Adaptive Control 7
Figure 2.4 Structure of Output feedback Control Allocation 8
Figure 2.5 Structure of Finite Time Sliding Mode Control Allocation 9
Figure 3.1 Reference Coordinate Frames [1] 11
Figure 3.2 Reference Coordinate Frames and Transformation Angles 11
Figure 3.3 ZYX Euler Sequence [2] 12
Figure 4.1 Lateral Thruster Model [3] 23
Figure 4.2 Lateral Thruster Model 23
Figure 6.1 Rise Time Contour Comprasion for two different actuator weight wrt N_p, Nc 39
Figure 6.2 Settling Time Contour Comprasion for two different actuator weight wrt N_p, N_c 39
Figure 6.3 OS Contour Comprasion for two different actuator weight wrt N_p, N_c 39
Figure 6.4 Trim δ_e under 5g acceleration command Contour Comprasion for two different actuator weight wrt N_p, Nc 40
Figure 6.5 Trim Thruster Thrust under 5g acceleration command Contour Comprasion for two different actuator weight wrt N_p, N_c 40
Figure 6.6 Acceleration and Angle of Attack Comparison under 5g acceleration command for two different actuator weight 41
Figure 6.7 δ_e and Thruster Comparison under 5g acceleration command for two different actuator weight 41
Figure 6.8 Force and Moment under 5g acceleration command for two different actuator weight 42

Figure 6.9 Acceleration and Angle of Attack Response under 5g acceleration command and lateral thruster constraint 45

Figure 6.10 δ_e and Thruster Response under 5g acceleration command and lateral thruster constraint 46

Figure 6.11 Force and Moment Response under 5g acceleration command and lateral thruster constraint 46

Figure 6.12 Acceleration and Angle of Attack Response under 5g acceleration command and lateral thruster constraint for limited thruster 47

Figure 6.13 δ_e and Thruster Response under 5g acceleration command and lateral thruster constraint for limited thruster 47

Figure 6.14 Force and Moment Response under 5g acceleration command and lateral thruster constraint for limited thruster 48

Figure 6.15 δ_e limit comparison, acc vs time 49

Figure 6.16 δ_e limit comparison, δ_e vs time 50

Figure 6.17 δ_e limit comparison, thr vs time 50

Figure 6.18 Online MPC based Adaptive Weight Flow Chart [4] 51

Figure 6.19 Acceleration and Angle of Attack Response under 7g acceleration command with adaptive weight algorithm 52

Figure 6.20 δ_e and Thruster Response under 7g acceleration command with adaptive weight algorithm 53

Figure 6.21 Force and Moment Response under 7g acceleration command with adaptive weight algorithm 53

Figure 6.22 Weight and Iteration Response under 7g acceleration command with adaptive weight algorithm 54

Figure 6.23 Acceleration and Angle of Attack Response under different IC's acceleration command scenario with adaptive weight algorithm 55

Figure 6.24 δ_e and Thruster Response under different IC's acceleration command scenario with adaptive weight algorithm 55

Figure 6.25 Force and Moment Response under different IC's acceleration command scenario with adaptive weight algorithm 56

Figure 6.26 Weight and Iteration Response under different IC's acceleration command scenario with adaptive weight algorithm 56
Figure 6.27 CPU Time Response under different IC's acceleration command scenario with adaptive weight algorithm 57
Figure 6.28 Jet interaction 58
Figure 6.29 MPC with disturbance Blok Diagram 58
Figure 6.30 Normal Distribution of Thrust Amplification Factor 59
Figure 6.31 Effect of Thrust amplification factor to acceleration tracking performance 60
Figure 6.32 Effect of Thrust amplification factor to δ_e performance 60
Figure 6.33 Effect of Thrust amplification factor to thruster performance 61
Figure 6.34 Multiple Run solution of Thrust Amplification effect 61
Figure 6.35 Multiple Run Disturbance Rejection characteristic of Thrust Amplification effect for different σ 62
Figure 6.36 Acceleration and Angle of Attack Response Comparison for Dual and Aero-Only Autopilots 64
Figure 6.37 Total Force and Moment Response Comparison for Dual and Aero-Only Autopilots 64
Figure 6.38 PPN Engagement Kinematics 66
Figure 6.39 Midcourse to Terminal Phase Handover Engagement Kinematics 66
Figure 6.40 Guidance Loop Block Diagram 67
Figure 6.41 Missile Trajectory at Scenario 1 69
Figure 6.42 Guidance acceleration command and autopilot response at Scenario 1 69
Figure 6.43 Angle of attack and Acceleration Effort at Scenario 1 70
Figure 6.44 Actuators Performance at Scenario 1 70

ABBREVIATIONS

MPC : **M**odel Predictive Control
MIQP : **M**ixed Integer Quadratic Programming
DOF : **D**egree of Freedom
TVC : **T**hrust Vector Control
RCS : **R**eaction Control System
CG : **C**enter of Gravity
PPN : **P**ure Proportinal Navigation
LOS : **L**ine of Sight
LOR : **L**ock on Range
PIP : **P**redicted Intercept Point
PD : **P**ositive Definite
SISO : **S**ingle Input Single Output
MIMO : **M**ulti Input Multi Output
ZOH : **Z**ero Order Hold
OS : **O**vershoot
N_p : **P**rediction Horizon
N_c : **C**ontrol Horizon

1 INTRODUCTION

Evolving technologies have enhanced the capability of fighter aircraft to perform high-altitude evasion maneuvers. Additionally, ballistic missiles pose a significant threat to air defense missiles due to their high speeds and destructive capabilities. The ability to intercept targets at high altitude has become an important requirement for air defense missiles to reduce or prevent the level of damage to specified targets. Interception at high altitude, or in other words, the tracking of guidance-induced acceleration commands, depends not only on the missile's acceleration capacity at that altitude but also on its agility[5–8]. Missiles that rely solely on aerodynamic surfaces for control cannot achieve the desired agility at high altitude due to the decrease in dynamic pressure. To address this problem, alternative actuator solutions have been suggested to ensure stable control regardless of altitude or flight regime. These include thrust vector control (TVC) and diverters, also known as Reaction Control Systems (RCS) [9, 10].

Thrust vector control (TVC) or reaction control system (RCS) technologies are employed in real missile applications based on various mission requirements[11], such as the initial turn of a vertical launch missile[12] and terminal phase end-game scenarios[3]. These technologies are even found in applications where they are used together, such as in the Terminal High Altitude Area Defense (THAAD) system produced by Lockheed Martin[11, 13]. Due to their ability to provide very fast lateral movements, RCS surpasses TVC in terms of missile agility[9]. Especially for ballistic missile defense scenarios, the high closing velocity and low seeker lock-on range significantly reduce terminal phase times, thus increasing the probability of rapid lateral movement and improving the likelihood of intercepting targets. In TVC, the long duration of thrust, the ability to direct thrust in many directions within certain limits, and the continuity and precision make it superior to RCS for continuous and precise control[11]. The limited number and on-off operation of RCS hinder the establishment of a continuous control structure. In this thesis, considering rapid ascent times as the main factor, a tail-controlled missile configuration with RCS control has been addressed.

RCS on missiles can be utilized in two ways: 'divert' and 'moment' types. In the 'divert' type, thrusters are positioned near the center of gravity, resulting in translational movement without rotation. For the 'moment' type, the thrusters are mounted either in front of or behind the center of gravity. When interacting with moment-type thrusters alongside aerodynamic surface control, the missile experiences rotational movement, allowing for the separation of lateral RCS and aero-surface control[5]. In tail-controlled missiles, acceleration control is in a "non-minimum phase" state, and a configuration where RCS are moment-type and positioned in front of the center of gravity is thought to compensate for this and has been examined in this study to prevent this situation.

The control method is a key contributor in maximizing missile performance by ensuring the harmonious operation of the two actuators. Being a multi-input system and having the thruster in an on-off mode with limited time and number of activations has led to the need for a different control method compared to traditional tail-controlled missiles[3, 14]. The Model Predictive Control (MPC) method has been selected for this study due to its effectiveness under actuator constraints. MPC is a control technique that, at each sampling instant, utilizes the current system states as initial conditions to produce an optimal control sequence over a finite time horizon. It executes the first control step of the control trajectory, repeating the process at every iteration. Fundamentally, MPC seeks to minimize a specified penalty function, similar to other optimal control methods. However, unlike traditional pre-computed and embedded control methods, MPC distinguishes itself by generating a control sequence using current state values. Another difference is that standard optimal control problems use the Hamilton-Jacobi-Bellman equation (Dynamic Programming) as a solution, which significantly increases computational load for online usage. On the other hand, MPC can provide a solution within an acceptable computational load[15–17].

1.1 Scope Of The Thesis

This thesis focuses on examining the acceleration tracking performance of a missile with both lateral thruster and tail control under constraints of lateral thrust and aerodynamic surface. The Model Predictive Control (MPC) method will be used as the autopilot method, and the autopilot performance will be evaluated only for the pitch channel of the missile. The derivation and linearization of motion equations required for autopilot design will also be examined within the scope of the thesis. Since actuator dynamics will be included in the autopilot design, actuator modeling and dynamics will also be studied. The work will attempt to explain the process of how the MPC method is used in the model when there are actuator constraints and when there are none. The design results will be tested by creating different scenarios, and the results will be interpreted. Thrust uncertainty due to the lateral thruster jet interaction has been modeled, and the robustness of the designed autopilot under this uncertainty has been examined. Finally, an engagement scenario against a ballistic missile is created, and the autopilot's behaviors are evaluated.

1.2 Contributions

This study introduces a novel, straightforward, and efficient approach. The key contributions of this paper are as follows:

- A blended autopilot structure that tracks the reference acceleration command in the linear domain under the on-off behavior of the lateral thruster and the angle constraints of the aerodynamic surfaces.
- Unlike most previous works, we included all actuator dynamics as state variables and all actuator constraints. Moreover, the system's performance is tested under the critical uncertainty of jet interaction.
- Our simulation results show how the lateral thruster enhances system performance compared to traditional aero-surface control in ballistic missile target guidance scenarios.

1.3 Organization

The structure of the thesis is as follows:

- Section 1 details our inspiration, key contributions, and the overall scope of this thesis.
- Section 2 offers a review of the existing literature.
- Section 3 presents a derivation of the 6-DOF equations of motion.
- Section 4 introduces linear domain analysis.
- Section 5 provides the method of MPC.
- Section 6 explains MPC design steps under both constrained and unconstrained actuators. Robustness and guidance analysis are also provided.
- Section 7 presents the conclusion and summary of the thesis.

2 Literature Survey

The evolution of missile technology has necessitated the development of sophisticated autopilot systems capable of handling complex, dynamic environments and maneuvering scenarios. This survey delves into various methodologies and enhancements in autopilot systems, particularly those integrating lateral thrusters or Reaction Control Systems (RCS), drawing from significant contributions in the field.

The paper ”Agile Missile Dynamics and Control” by Wise and Broy (1998)[9] explores the challenges and strategies in the dynamic control of agile missiles, focusing on flight behavior and guidance in high-angle-of-attack (AOA) scenarios. The authors highlight the importance of high-AOA agility, which enables agile missiles to conduct sharp maneuvers for effective engagement of targets across a broad spectrum, including those in the rear hemisphere. They highlight the essential role of thrust vector control (TVC) and reaction control system (RCS) thrusters when aerodynamic surfaces become ineffective at extreme angles. A key point of the discussion is the interaction between thrust vectoring, reaction control, and aerodynamic surfaces in enhancing maneuverability. The authors elaborate on challenges like jet interaction effects, which can cause problems with stability and control effectiveness.

TVC and RCS, in particular, are presented as vital control effectors for agile maneuvers, especially in near-stall or post-stall flight regimes where conventional aerodynamic controls falter. In addressing these challenges, the paper describes various autopilot designs tailored for different flight phases, each incorporating nonlinear dynamics suited to their specific demands. For instance, autopilots manage launch separation, agile turning, midcourse flight, and terminal homing, each requiring distinct algorithms to handle unique aerodynamic phenomena. Simulation results presented by the authors underscore the viability of nonlinear control algorithms using RCS thrusters to maintain agile orientation. This approach enables a missile to precisely intercept a target with minimal roll disturbances and efficiently transition control authority between thrusters and aerodynamic surfaces. Overall, the paper provides valuable insights into high-AOA missile dynamics and control, laying the groundwork for future research in agile missile design.

The paper ”Variable Structure Control for the Autopilot of Aerodynamic/Reaction-Jet Multiple-Control Missiles”[18] presents an approach for enhancing missile autopilot. The paper outlines a missile model equipped with an array of micro pulse engines, which are arranged in circles on the front side of the missile to assist with lateral thrust. The author introduces a control system that integrates aerodynamic and reaction-jet systems, recognizing the dynamic coupling between them. The autopilot is designed using Variable Structure Control (VSC) to enhance robustness and response speed, especially during high-G maneuvers. By incorporating this method, the missile can smoothly switch between aerodynamic and reaction-jet control, achieving high performance and stability, as shown in Figure 2.1. Simulations demonstrate that this multiple-control system effectively manages the missile’s complex dynamics, leading to accurate and agile maneuvering. The study also mentions that jet interaction is a critical parameter and incorporates this parameter into the analysis.

Early works, such as those by Zhou and Shao (2009), laid foundational concepts for feedback linearization techniques that handle the nonlinear dynamics typical of missiles with dual control systems, such as aerodynamic fins and RCS[19]. The structure of input-output feedback linearization is shown in Figure 2.2. This approach simplifies the

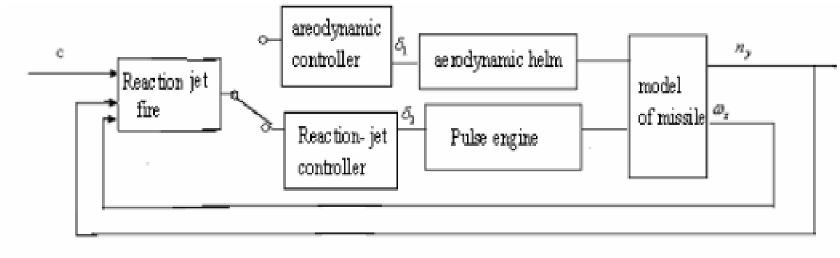


Figure 2.1 Structure of Variable Structure Control

control problem by decoupling the dynamics into linearizable components, thus enabling the more straightforward application of traditional control techniques such as LQR. Although the work considers the nonlinear dynamics of the system, including actuator dynamics, there is no constraint analysis, and the lateral thruster is treated as a continuous behavior. This study concludes that incorporating feedback linearization in the autopilot structure for missiles equipped with hybrid control (tail fins, RCS) effectively manages internal dynamics when the output variables are divert accelerations and rotational rates. The fundamental concept of blending RCS and tail fins is essential, mainly depending on aerodynamic lift, while the forces from the lateral thruster and tail fins counterbalance each other. This principle allows for stable and agile missile control, essential for engaging fast-moving targets at high altitudes.

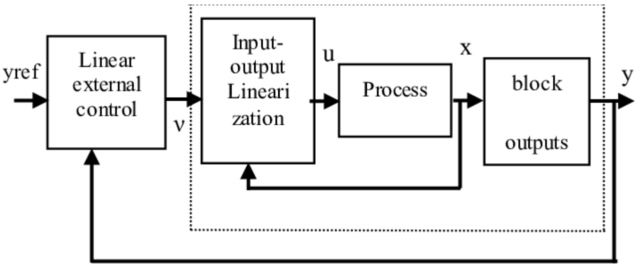


Figure 2.2 Structure of Feedback Linearization

McFarland and Calise (2000) explored the use of neural networks to adaptively cancel linearization errors in real-time, a method that enhances robustness against dynamic uncertainties inherent in agile missiles[20]. The autopilot structure is shown in Figure 2.3. The article presents an innovative approach to controlling nonlinear dynamic systems through neural networks. This study presents an adaptive control method that utilizes neural networks' learning capabilities to approximate system dynamics and create a robust controller. By applying nonlinear adaptive control, it tackles the challenges of uncertainties and external disturbances in missile control systems. The neural network is trained online, allowing continuous adaptation and performance maintenance. This method shows promising results, providing superior control and stability compared to traditional controllers, particularly in dynamic and unpredictable environments, marking a significant advancement in nonlinear missile control. However, actuator constraints and dynamics were not considered in this study.

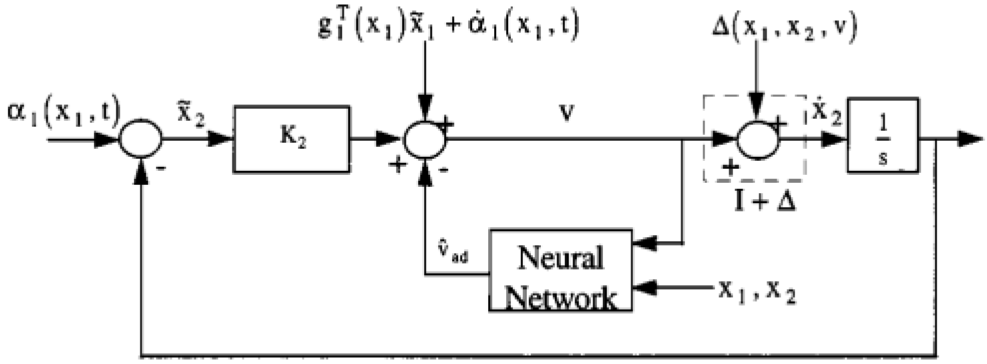


Figure 2.3 Structure of Neural Network Based Adaptive Control

In the paper on the topic of "Optimal Control and Control Allocation" [21, 22], the authors tackle the problem of optimal control for a dual-control missile system utilizing both fin and jet reaction controls. They formulate an optimal control problem aimed at minimizing control energy while ensuring accurate trajectory tracking, considering the dynamic constraints of the missile system. By implementing dynamic control allocation, the solution effectively distributes commands across actuators based on their capabilities and system limitations. The methodology includes the concept of virtual controls, which helps translate optimal force and moment commands into actuator inputs, and allows a thorough evaluation of the control strategies. The results indicate significant improvements in maneuverability and responsiveness, as the optimal control solution ensures precise trajectory tracking with reduced response time. The comparative analysis shows that the combination of optimal control with a comprehensive allocation scheme significantly enhances the performance of the missile over traditional linear methods.

The 2010 paper published by Lidan, Ke'nan, Wanchun, and Xingliang [7] outlines the evolution of a hybrid control autopilot for missiles utilizing both aero-fin and RCS considering output feedback. The proposed autopilot combines an optimal controller with a control allocation module as depicted in Figure 2.4. The optimal controller employs a hybrid optimal/classical method to determine virtual controls, while the control allocation module distributes these virtual controls among redundant effectors. This design ensures robustness, effective tracking performance, and simplicity. The paper notes that using total acceleration as feedback can result in instability due to transient effects. Therefore, an improved method eliminates these effects by using acceleration caused solely by the angle of attack as feedback. Additionally, techniques to obtain precise acceleration feedback are discussed, indicating that filtering total acceleration through a low-pass filter can provide effective feedback. The paper concludes by emphasizing the potential for future research to enhance control allocation and address hybrid-input issues.

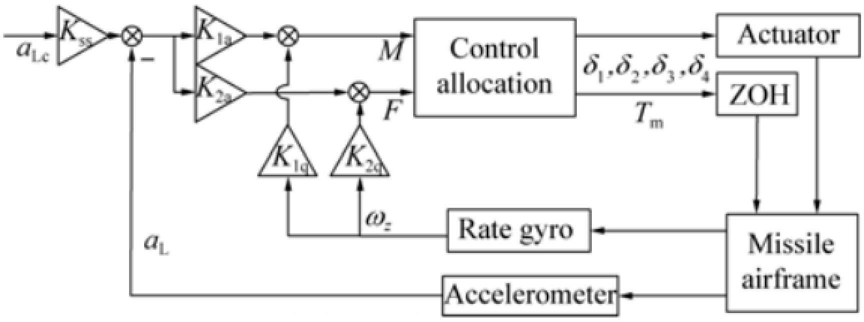


Figure 2.4 Structure of Output feedback Control Allocation

The 2019 paper by Yong Guo et al. introduces a robust finite-time controller for air defence missiles equipped with both RCS and aerodynamic fins [14]. The authors developed an integral sliding mode manifold to address input constraints using a saturation function. This finite-time controller manages input saturation and external disturbances through dynamic control allocation, as illustrated in Figure 2.5. A constrained quadratic programming approach is utilized to distribute control commands between aerodynamic surfaces and lateral thrusters. Simulations indicate that the proposed control strategy enhances response speed, tracking accuracy, and control efficiency compared to PID and conventional sliding mode controllers. The findings confirm the proposed scheme's effectiveness in real-time missile control applications.

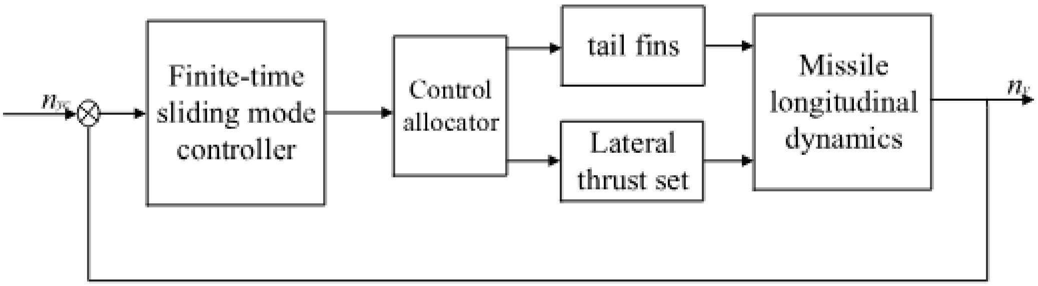

n_{yc}
Finite-time sliding mode controller
Control allocator
tail fins
Lateral thrust set
Missile longitudinal dynamics
n_y

Figure 2.5 Structure of Finite Time Sliding Mode Control Allocation

Another study, which also served as the starting point for this thesis, was examined by Baoqing Yang and Yuyu Zhao using the Model Predictive Control approach[3]. As is known, missile system dynamics change significantly according to different angles of attack; in this study, system models for angles of attack regions are treated as 'piece-wise affine' and an adaptation for varying system states has been attempted. The reference input addresses a tracking problem with angles of attack. Furthermore, constraints on both actuators are included in this study. Moreover, an optimal control problem that better reflects the on-off nature of the actuators, such as Mixed Integer Quadratic Programming, was worked on. Instead of online optimization, 'explicit' MPC has been preferred. 'Explicit MPC' indicates that control parameters for the defined sub-regions are obtained simultaneously while solving the optimization problem, distinguishing it from the 'gain-scheduling' method because 'gain-scheduling' calculates control parameters separately for each region. The autopilot performance with different reference inputs is examined in detail.

3 SIX DEGREE OF FREEDOM EQUATIONS OF MOTION

To model missile movement and create controller equations, equations of motion are essential. This chapter presents reference coordinate frames along with the equations for translational and rotational motion under the following assumptions:

A1) Missile is considered as rigid body which means there is no structural deflections.

A2) Rotation of earth is neglected and assumed it is flat.

A3) There is no wind effect.

3.1 Reference Coordinate Frames

To describe the motion of the missile, three reference coordinate frames are utilized and shown in Figure 3.1. First is fixed Earth Frame $(\mathbf{F}_E)$ that is defined inertial reference frame located center of earth. It is widely used in navigation, and mapping systems to keep a consistent frame for measurements that stay fixed relative to the Earth's surface and denoted as which aligns with the axes: $\mathbf{X}_E$ points north, $\mathbf{Y}_E$ points east, $\mathbf{Z}_E$ points east, pointing down, aligns with the direction of the gravitational force.
The second is the body-fixed frame $(\mathbf{F}_B)$ which is the reference frame moves and rotates with the vehicle, and its origin is typically located at the vehicle's center of gravity. The body frame is crucial in dynamics and control analysis since it is used to measure and define the vehicle's orientation (attitude), angular velocities, and accelerations in flight and denoted as: $\mathbf{X}_B$ aligns with the missile's longitudinal centerline, $\mathbf{Y}_B$ points to the right from the rear view, $\mathbf{Z}_B$ points downward.

Last one is wind reference frame $(\mathbf{F}_W)$ that is a coordinate system that describes the motion of a missile relative to the surrounding air (according to A3, it is equal to velocity vector of

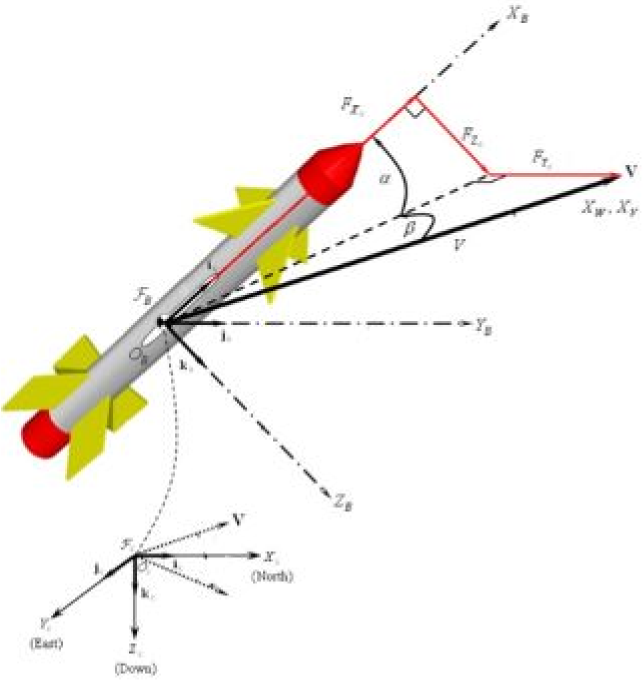


Figure 3.1 Reference Coordinate Frames [1]

missile) and it is denoted as: $\mathbf{X}_w$ aligns with the missile's velocity vector, $\mathbf{Y}_w$ Extends to the right or left side of the vehicle, perpendicular to the X-axis within the plane of symmetry. $\mathbf{Z}_W$ fulfill the right-handed coordinate system. It is possible to transform a vector between the defined coordinate frames by using Euler and incidences angles as shown in Figure 3.2 and details is shown following section.

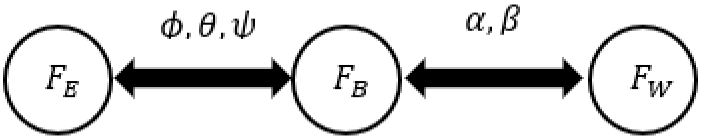


Figure 3.2 Reference Coordinate Frames and Transformation Angles

3.2 Kinematics

Euler angles are commonly used to describe the orientation of a rigid body, shown in Figure 3.3, such as a missile, in three-dimensional space and it is used to transform inertial frame to body frame or vice versa. For the ZYX rotation sequence, the euler angles are defined as follows:

- ψ (yaw) - rotation about the z-axis,
- θ (pitch) - rotation about the y-axis,
- ϕ (roll) - rotation about the x-axis.

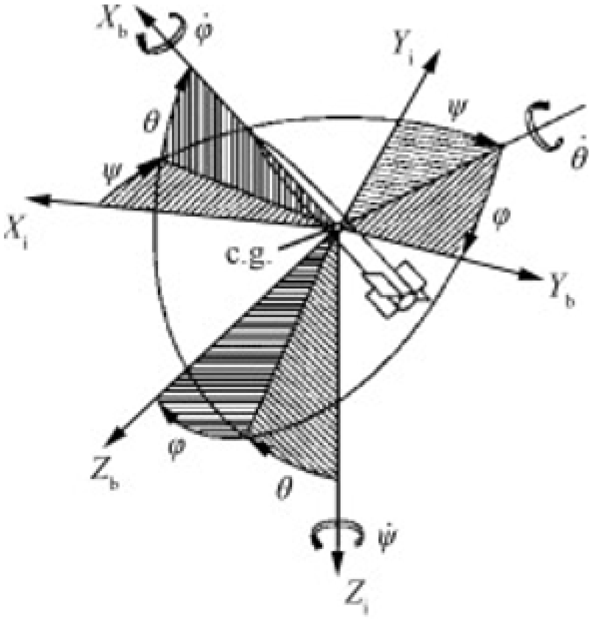


Figure 3.3 ZYX Euler Sequence [2]

The rotation matrix R for the ZYX sequence can be expressed as the product of three basic rotation matrices:

$$R = R_z(\psi)R_y(\theta)R_x(\phi), \tag{3.1}$$

where the rotation matrices are defined as:

$$R_z(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{3.2}$$

$$R_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}, \tag{3.3}$$

$$R_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix}. \tag{3.4}$$

Given the rotation matrix R defined for the ZYX Euler angle sequence, the relationship between a vector resolved in the body frame $\mathbf{V}_{(B)}$ and the same vector resolved in the inertial frame $\mathbf{V}_{(E)}$ can be expressed as:

$$\mathbf{V}_{(B)} = R_{(B,E)}\mathbf{V}_{(E)} \tag{3.5}$$

The spatial relationship between the body-fixed frame and the aerodynamic frame can be understood by considering the angle of attack, denoted as α, and the sideslip angle, denoted as β. If one considers that the aerodynamic frame can be derived from the body frame by initially rotating in the negative β direction followed by a positive rotation around α, then the transformation matrix $T_{(B/W)}$ that describes the body frame in terms of the wind frame can be expressed as:

$$T_{(B/W)} = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ 0 & 1 & 0 \\ \sin\alpha & \cos\alpha & 0 \end{bmatrix} \begin{bmatrix} \cos\beta & 0 & -\sin\beta \\ 0 & 1 & 0 \\ \sin\beta & 0 & \cos\beta \end{bmatrix} \tag{3.6}$$

Given the rotation matrix $T_{(B/W)}$ the relationship between a vector resolved in the body frame $\mathbf{V}_{(B)}$ and the same vector resolved in the wind frame $\mathbf{V}_{(W)}$ can be expressed as:

$$\mathbf{V}_{(B)} = T_{(B/W)}\mathbf{V}_{(W)} \tag{3.7}$$

3.3 Translational Motion

Newton's second law can be applied to describe translational motion. Typically, a missile's path is much slower than orbital speeds and stays relatively close to the Earth's surface. Therefore, assuming a non-rotating and flat Earth, the effects of Earth's rotational accelerations—such as Coriolis and centrifugal forces—as well as changes in gravitational direction due to the missile's motion over the Earth's surface, are negligible. Additionally, the missile's mass is considered constant. This leads to the application of Newton's second law as follows:

$$\begin{gathered} \frac{d_E\mathbf{V}}{dt} = \frac{\sum \mathbf{F}}{m} \\ \frac{d_E\mathbf{V}}{dt} = \frac{d_B\mathbf{V}}{dt} + \boldsymbol{\omega}_{B/E} \times \mathbf{V} \end{gathered} \tag{3.8}$$

where $d_A\mathbf{V}/dt$ denotes that vector V is differentiated in the A frame. In this case, V represent the velocity vector, d_BV/dt and d_EV/dt are expressed velocity vector derivatives in body and earth frame respectively. $\omega_{B/E}$ is defined angular velocity body frame with respect to earth frame. Force vector act on missile in body coordinate frame can be represented as:

$$\vec{F}_{(B)} = \begin{bmatrix} F_{x_B} & F_{y_B} & F_{z_B} \end{bmatrix}^T \tag{3.9}$$

Velocity vector in body coordinate frame can be represented as:

$$\vec{V}_{(B)} = \begin{bmatrix} u & v & w \end{bmatrix}^T \tag{3.10}$$

Angular velocity in body coordinate frame can be represented as:

$$\vec{\omega}_{(B)} = \begin{bmatrix} p & q & r \end{bmatrix}^T \tag{3.11}$$

Substituting equation 3.9-3.11 in equation 3.8.

$$\begin{aligned} \dot{u} &= \frac{F_{x_B}}{m} - qw + rv \\ \dot{v} &= \frac{F_{y_B}}{m} - ru + pw \\ \dot{w} &= \frac{F_{z_B}}{m} - pv + qu \end{aligned} \tag{3.12}$$

3.4 Rotational Motion

Euler's law is required to defined rotational motion and is written as:

$$\frac{d_E\vec{H}}{dt} = \sum \vec{M} \tag{3.13}$$

$$\frac{d_E\mathbf{H}}{dt} = \frac{d_B\mathbf{H}}{dt} + \boldsymbol{\omega}_{B/E} \times \mathbf{H}$$

where H is angular momentum and expressed as:

$$\vec{H} = I\vec{\omega} \tag{3.14}$$

where I is defined as inertia matrix. Substituting equation 3.13 and 3.14:

$$\sum M = I\frac{d_B\omega}{dt} + \boldsymbol{\omega}_{B/E} \times I\omega \tag{3.15}$$

Assumed cross products of inertia is neglected and combining the equation 3.11 and 3.15 results with:

$$
\begin{aligned}
\dot{p} &= \frac{M_x + (I_y - I_z)rq}{I_x} \\
\dot{q} &= \frac{M_y + (I_z - I_x)pr}{I_y} \\
\dot{r} &= \frac{M_z + (I_x - I_y)pq}{I_z}
\end{aligned}
\tag{3.16}
$$

3.5 Incidence Angle Dynamics

The incidence angles of a missile, specifically the angle of attack (α) and side-slip angle (β), are crucial to describe missile dynamics. The angle between missile body longitudinal axis and free stream velocity vector is called angle of attack. It's fundamental in determining aerodynamic behaviour of missile, which affect its pitch, yaw, and roll behavior. Side-slip angle measures the deviation of airflow from the symmetry plane of the missile and is vital for yaw dynamics and control.

In velocity frame, it is assumed that velocity magnitude is align with the x axis of this frame and there is no velocity component on other axis. Missile velocity magnitude can be determined by:

$$V = (u^2 + v^2 + w^2)^{0.5} \tag{3.17}$$

The derivative of airspeed $\dot{V}$ is found by differentiating equation 3.17 and incorporating equation 3.12, resulting in:

$$\dot{V} = \frac{1}{V}(u\dot{u} + v\dot{v} + w\dot{w}) \tag{3.18}$$

which, upon expansion, becomes:

$$\dot{V} = \frac{1}{V}\left[\left(a_{x_b} + rv - qw\right)u + \left(a_{y_b} + pw - ru\right)v + \left(a_{z_b} + qu - pv\right)w\right], \tag{3.19}$$

where $a_{x_b} = \frac{F_{x_b}}{m}$, $a_{y_b} = \frac{F_{y_b}}{m}$, and $a_{z_b} = \frac{F_{z_b}}{m}$ denote the total body-coordinate acceleration components due to gravitational, aerodynamic, and propulsive forces exerted upon the vehicle. Applying transformations 3.6 and 3.7, then substitute in equation 3.19:

$$\dot{V} = a_{x_b}\cos(\alpha)\cos(\beta) + a_{y_b}\sin(\beta) + a_{z_b}\sin(\alpha)\cos(\beta) \tag{3.20}$$

Defining incidence angle can be written as with respect to body frame velocity components and velocity magnitude by following expressions:

$$\alpha = \arctan\left(-\frac{w}{u}\right) \tag{3.21}$$

$$\beta = \arcsin\left(-\frac{v}{V}\right) \tag{3.22}$$

Derivatives of incidence angle by using the equation of 3.21-3.22 as follows:

$$\dot{\alpha} = \frac{u\dot{w} - w\dot{u}}{u^2 + w^2} \tag{3.23}$$

$$\dot{\beta} = \frac{V\dot{v} - v\dot{V}}{V\sqrt{V^2 - v^2}} \tag{3.24}$$

Substituting equation 3.12 into equations 3.23-3.24:

$$\dot{\alpha} = q - (pc(\alpha) + rs(\alpha))t(\beta) + \frac{1}{Vc(\beta)}(a_{z_B}c(\alpha) - a_{x_B}s(\alpha)) \tag{3.25}$$

$$\dot{\beta} = ps(\alpha) - rc(\alpha)) - \frac{1}{V}(a_{x_B}c(\alpha)s(\beta) - a_{y_B}c(\beta) + a_{z_B}s(\alpha)s(\beta) \tag{3.26}$$

where c, s, t abbreviation of cos, sin, tan respectively.

3.6 Forces and Moments

There are 3 main external force and moment sources which are gravity, aerodynamics, and propulsion, acting on a missile. In section 3.3 and 3.4, force and moment equations equation 3.1 and 3.7 are shown in body coordinate frame. Gravity contributes only total forces, whereas propulsion and aerodynamics contributes both total forces and moments.

Gravity force in inertial frame (F_E) has a component only Z_E axis. The gravity force vector resolved in inertial frame is shown as:

$$\vec{g}_{(E)} = \begin{bmatrix} 0 & 0 & g \end{bmatrix}^T \tag{3.27}$$

Using the direction cosine matrix to transform force into body frame is written as:

$$\mathbf{g}_{(B)} = R_{(B,E)}\mathbf{g}_{(E)} \tag{3.28}$$

The force producing by gravity in body frame is expressed as:

$$\mathbf{F}_g = mg_{(B)} \tag{3.29}$$

Forces and moments producing by aerodynamic are represented $\mathbf{F}_a$ and $\mathbf{M}_a$ respectively in the body frame and are derived using the dynamic pressure $\bar{q}$, aerodynamic coefficients along the body axes, C_x, C_y, C_z, and the aerodynamic moment coefficients, C_l, C_m, C_n, with respect to the center of gravity, factoring in the characteristic length d and area S of the missile:

$$\mathbf{F}_a = \bar{q}S[C_x\ C_y\ C_z]^T \tag{3.30}$$

$$\mathbf{M}_a = \bar{q}Sd[C_l\ C_m\ C_n]^T. \tag{3.31}$$

Dynamic pressure $\bar{q}$ is defined as:

$$\bar{q} = 0.5\rho V^2. \tag{3.32}$$

The primary components of the aerodynamic coefficients are largely influenced by mach number M, incidence angles (α,β) and fin deflections (δ_e,δ_r,δ_a). In addition to those, moment coefficients have dependency of angular velocity (p, q, r). General dependency represantation of aerodynamic coefficients are articulates as:

$$\begin{aligned} &C_x(M,\alpha,\beta,\delta_e,\delta_r,\delta_a)\\ &C_y(M,\alpha,\beta,\delta_e,\delta_r,\delta_a)\\ &C_z(M,\alpha,\beta,\delta_e,\delta_r,\delta_a)\\ &C_l(M,p,\alpha,\beta,\delta_e,\delta_r,\delta_a)\\ &C_m(M,q,\alpha,\beta,\delta_e,\delta_r,\delta_a)\\ &C_n(M,r,\alpha,\beta,\delta_e,\delta_r,\delta_a) \end{aligned} \tag{3.33}$$

Under the assumption that the main solid thrust vector $\mathbf{T}_{p_e}$ aligns with the missile's X_b axis and passes through its center of mass (CM), it generates no external moment. Lateral thrusters $\mathbf{T}_{p_y}$ and $\mathbf{T}_{p_z}$ are included missile configuration which generates forces aligns with the missile's Y_b and Z_b, it produces moment around around Y_b and Z_b respectively and distance, location of the thrusters for both direction from the center of gravity of missile is denoted as r_t.

$$\mathbf{F}_p = [T_{p_e}\ T_{p_y}\ T_{p_z}]^T \tag{3.34}$$

$$\mathbf{M}_p = [0\ T_{p_z} r_t\ T_{p_y} r_t]^T. \tag{3.35}$$

Consequently, acting total forces and moments on body are presented component-wise as:

$$\mathbf{F} = \begin{bmatrix} F_{x_b} \\ F_{y_b} \\ F_{z_b} \end{bmatrix} = \begin{bmatrix} -mg\sin\theta + \bar{q}SC_x + T_{p_e} \\ mg\cos\theta\sin\phi + \bar{q}SC_y + T_{p_y} \\ mg\cos\theta\cos\phi + \bar{q}SC_z + T_{p_z} \end{bmatrix}, \tag{3.36}$$

$$\mathbf{M} = \begin{bmatrix} L \\ M \\ N \end{bmatrix} = \begin{bmatrix} \bar{q}SdC_{l_{cg}} \\ \bar{q}SdC_{m_{cg}} + T_{p_z} r_t \\ \bar{q}SdC_{n_{cg}} + T_{p_y} r_t \end{bmatrix}. \tag{3.37}$$

4 Linear Domain Analysis

4.1 Linearization

In scope of this thesis, autopilot is designed for the pitch channel in linear domain. Thus, obtained nonlinear equations in section 3 should be linearized.

Initially, the inherent non-linearities within the equations governing motion are simplified by decoupling the pitch, yaw, and roll dynamics. The studied missile exhibits axisymmetry and performs skid-to-turn maneuvers, maintaining a roll angle that is consistently kept at zero. Thus, we adopt the preliminary conditions that roll rate p is zero. In isolating the pitch and yaw dynamics, it is presumed that pitch motion is absent during yaw maneuvers, and reciprocally, yaw motion is absent during pitch maneuvers. This leads to setting $r = 0$ in the presence of a non-zero pitch rate q, and the reverse is true. For the roll dynamic analysis, both pitch and yaw motions are considered negligible, that is, $p = r = 0$. Recall equation 3.19 and 3.31, which are required to describe pitch motion of missile:

$$\dot{\alpha} = q - (pc(\alpha) + rs(\alpha))t(\beta) + \frac{1}{Vc(\beta)}(a_{z_B}c(\alpha) - a_{x_B}s(\alpha)) \tag{4.1}$$

$$\dot{q} = \frac{M_y + (I_z - I_x)pr}{I_y} \tag{4.2}$$

The following assumptions are implemented to decouple the pitch channel from the others:

- gravity is neglected
- roll rate is zero ($p = 0$)
- yaw motion is absent during pitch maneuvers ($\beta, r \ = 0$)
- cross product of inertia is assumed zero($I_{xz}, I_{yz}, I_{xy} = 0$)
- Assume small angle approximations where $\sin(\alpha) \approx \alpha$ and $\cos(\alpha) \approx 1$.

- main engine thrust is assumed zero ($a_{x_B} = 0$)

$$\dot{\alpha} = q + \frac{a_{z_B}}{V} \tag{4.3}$$

$$\dot{q} = \frac{\sum M_y}{I_{yy}} \tag{4.4}$$

Now, the aerodynamic coefficient should be linearized. According to equation 3.33, with the above assumptions including a single Mach point, the dependency can be reduced as follows:

$$C_z(\alpha, \delta_e) = C_{z_\alpha}\alpha + C_{z_{\delta e}}\delta_e \tag{4.5}$$

$$C_m(q, \alpha, \delta_e) = C_{m_\alpha}\alpha + \frac{d}{2V}C_{m_q}q + C_{m_{\delta e}}\delta_e \tag{4.6}$$

Combining equations from 4.3 to 4.6:

$$\dot{\alpha} = q + \frac{\bar{q}S[C_{z_\alpha}\alpha + C_{z_{\delta e}}\delta_e]}{mV} + \frac{T_{p_z}}{mV} \tag{4.7}$$

$$\dot{q} = \frac{\bar{q}Sd[C_{m_\alpha}\alpha + \frac{d}{2V}C_{m_q}q + C_{m_{\delta e}}\delta_e]}{I_{yy}} + \frac{T_{p_z}r_t}{I_{yy}} \tag{4.8}$$

4.2 Actuator Dynamics

To achieve acceptable performance from the autopilot system, it is also essential to take into account the dynamics of the actuators. Typically, a second-order dynamic model of the aerodynamic fins, characterized by the following transfer function:

$$\frac{\delta_e(s)}{\delta_{e_{\mathrm{com}}}(s)} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{4.9}$$

$$\omega_n = 50\,\mathrm{Hz} \tag{4.10}$$

$$\zeta = 0.7 \tag{4.11}$$

$$-25^o \leq \delta_e^{max} \leq 25^o \tag{4.12}$$

The thruster modeling, inspired by [3], is designed as a structure arranged radially around a ring at a 20-degree angle with 18 lateral thruster on it and, 4 other rings are placed along the x axis. The structure is shown in Figure 4.1 and Figure 4.2. $\mathbf{T}_i^{(j)}$ represents the ith thruster on jth ring. Because the distances between each ring are small, the distance between rings and CG is equal and denoted as r_l. Each thruster is set to produce a force of 2.2kN, with a burn time of 25 ms.

Each thruster is initially set to be fired only once. In order to sustain maneuvering ability during the terminal phase and symmetry, the number of firing is restricted.According to these restrictions, the following rules are defined to fire at the same time:

- maximum 2 thruster on a single ring are available
- maximum 2 rings are available
- thrusters which are out of the symmetry axis should be fired together (i.e 2-18 or 3-17)

After set the firing logic possible available thrust array for positive pitch moment $F^{(+)}$. can be defined as:

$$F^{(+)} = \left[T \; 2T\cos\left(\frac{\pi}{9}\right) \; 2T\cos\left(\frac{\pi}{18}\right) \; 2T \; 4T\cos\left(\frac{\pi}{9}\right) \; 4T\cos\left(\frac{\pi}{18}\right) \right] \tag{4.13}$$

Similarly, thrusters are located symmetrical around y axis, thruster array that produces negative pitch moment can be defined as:

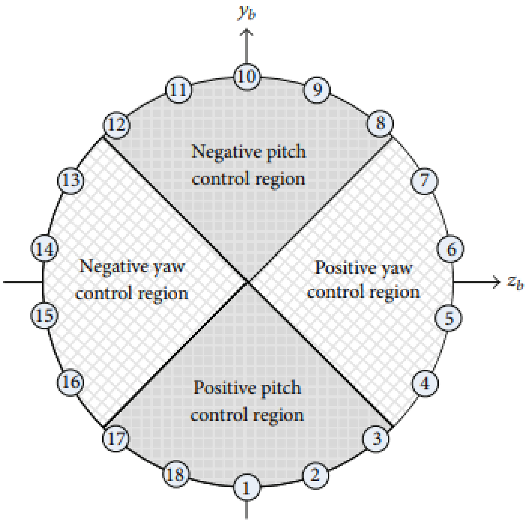
$$F^{(-)} = -F^{(+)} \tag{4.14}$$

Substituting the jet thrust value in thruster array:

$$F^{(+)} = \begin{bmatrix} 2200 & 4135 & 4333 & 4400 & 8269 & 8666 \end{bmatrix} \tag{4.15}$$

including positive, negative and no-fire condition, final thrust array(F) can be expressed as:

$$F = \begin{bmatrix} F^{(+)} & 0 & F^{(-)} \end{bmatrix} \tag{4.16}$$



y_b
z_b
Negative pitch control region
Negative yaw control region
Positive yaw control region
Positive pitch control region
10
9
8
7
6
5
4
3
2
1
18
17
16
15
14
13
12
11

Figure 4.1 Lateral Thruster Model [3]

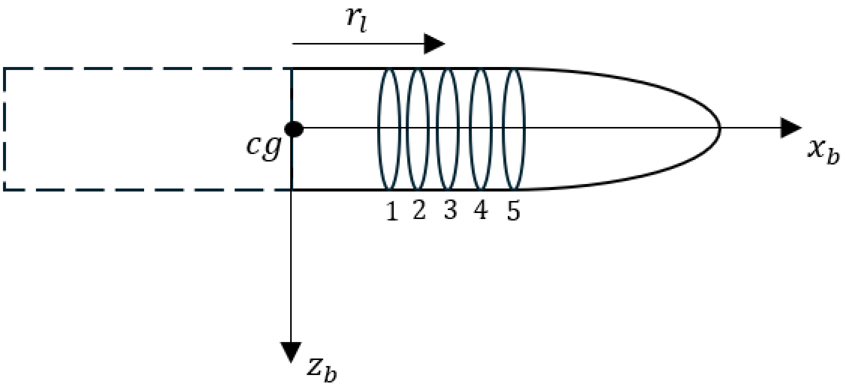


Figure 4.2 Lateral Thruster Model

Since the lateral thruster is quite fast and assumed to have no overshoot, a first-order transfer function is selected for the actuator dynamics. The transfer function for the RCS is written as:

$$G_{\text{thr}} = \frac{\tau}{s + \tau} \tag{4.17}$$

$$\tau = 200 \tag{4.18}$$

4.3 System Model

$$\begin{bmatrix} \dot{\alpha} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} \frac{qSC_{z_\alpha}}{mV} & 1 + \frac{qSdC_{z_q}}{mV^2} \\ \frac{qSdC_{m_\alpha}}{I_{yy}} & \frac{qSd^2C_{m_q}}{2VI_{yy}} \end{bmatrix} \begin{bmatrix} \alpha \\ q \end{bmatrix} + \begin{bmatrix} \frac{qSC_{z_{\delta_e}}}{mV} & \frac{1}{mv} \\ \frac{qSdC_{m_{\delta_e}}}{I_{yy}} & \frac{l_{thr}}{I_{yy}} \end{bmatrix} \begin{bmatrix} \delta_e \\ F_{thr} \end{bmatrix} \tag{4.19}$$

$$Z_\alpha = \frac{qSC_{z_\alpha}}{mV}, \quad Z_q = 1 + \frac{qSdC_{z_q}}{mV^2}, \quad Z_{\delta_e} = \frac{qSC_{z_{\delta e}}}{mV}$$

$$M_\alpha = \frac{qSdC_{m_\alpha}}{I_{yy}}, \quad M_q = \frac{qSd^2C_{m_\alpha}}{2VI_{yy}}, \quad M_{\delta_e} = \frac{qSdC_{m\delta_e}}{I_{yy}}$$

$$\begin{bmatrix} \dot{\alpha} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} Z_\alpha & Z_q \\ M_\alpha & M_q \end{bmatrix} \begin{bmatrix} \alpha \\ q \end{bmatrix} + \begin{bmatrix} Z_{\delta_e} & \frac{1}{mv} \\ M_{\delta_e} & \frac{I_{thr}}{I_{yy}} \end{bmatrix} \begin{bmatrix} \delta_e \\ F_{thr} \end{bmatrix} \tag{4.20}$$

Include actuator dynamics;

$$\begin{bmatrix} \dot{\delta} \\ \ddot{\delta} \\ \dot{F}_{thr} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -\omega_n^2 & -2\zeta\omega_n & 0 \\ 0 & 0 & -\tau \end{bmatrix} \begin{bmatrix} \delta \\ \dot{\delta} \\ F_{thr} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \omega_n^2 & 0 \\ 0 & \tau \end{bmatrix} \begin{bmatrix} \delta_{com} \\ F_{thr_{com}} \end{bmatrix} \tag{4.21}$$

Augmented state space representation of system model is:

$$\begin{bmatrix} \dot{\alpha} \\ \dot{q} \\ \dot{\delta} \\ \ddot{\delta} \\ \dot{F}_{thr} \end{bmatrix} = \begin{bmatrix} Z_\alpha & Z_q & Z_{\delta_e} & 0 & \frac{1}{mv} \\ M_\alpha & M_q & M_{\delta_e} & 0 & \frac{I_{thr}}{I_{yy}} \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & -\omega_n^2 & -2\zeta\omega_n & 0 \\ 0 & 0 & 0 & 0 & -\tau \end{bmatrix} \begin{bmatrix} \alpha \\ q \\ \delta \\ \dot{\delta} \\ F_{thr} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \omega_n^2 & 0 \\ 0 & \tau \end{bmatrix} \begin{bmatrix} \delta_{com} \\ F_{thr_{com}} \end{bmatrix} \tag{4.22}$$

$$a_z = \begin{bmatrix} Z_\alpha V & (1 - Z_q)V & Z_{\delta_e} V & 0 & \frac{1}{m} \end{bmatrix} \begin{bmatrix} \alpha \\ q \\ \delta \\ \dot{\delta} \\ F_{thr} \end{bmatrix} \tag{4.23}$$

$$\dot{x} = Ax + Bu \tag{4.24}$$

$$y = Cx$$

$$A = \begin{bmatrix} Z_\alpha & Z_q & Z_{\delta_e} & 0 & \frac{1}{mv} \\ M_\alpha & M_q & M_{\delta_e} & 0 & \frac{I_{thr}}{I_{yy}} \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & -\omega_n^2 & -2\zeta\omega_n & 0 \\ 0 & 0 & 0 & 0 & -\tau \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 & \omega_n^2 & 0 \\ 0 & 0 & 0 & 0 & \tau \end{bmatrix}^T$$

$$C = \begin{bmatrix} Z_\alpha V & (1 - Z_q)V & Z_{\delta_e} V & 0 & \frac{1}{m} \end{bmatrix}$$

$$x = \begin{bmatrix} \alpha & q & \delta & \dot{\delta} & F_{thr} \end{bmatrix}^T$$

$$u = \begin{bmatrix} \delta_{com} & F_{thr_{com}} \end{bmatrix}^T$$

It is desired to use acceleration as a state variable instead of angle of attack keeping the other state variables. Desired new state vector can be expressed as:

$$z = \begin{bmatrix} a_z & q & \delta & \dot{\delta} & F_{thr} \end{bmatrix}^T$$

In order to do that transformation matrix (T) between x and z is required and defined as:

$$T = \begin{bmatrix} Z_\alpha V & (1 - Z_q)V & Z_{\delta_e} V & 0 & \frac{1}{m} \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.25}$$

The relationship new state vector z and the original state vector x can be defined as:

$$z = Tx \tag{4.26}$$

Equation 4.25 retrieves the original state vector x by multiplying the new state vector z by using following expression:

$$T^{-1}z = x \tag{4.27}$$

Equation 4.25 substitute in equation 4.20 to transform new system dynamics:

$$T^{-1}\dot{z} = AT^{-1}z + Bu \tag{4.28}$$

Multiplying both sides with T:

$$TT^{-1}\dot{z} = TAT^{-1}z + TBu \tag{4.29}$$

Final system dynamics model is obtained as:

$$\dot{z} = TAT^{-1}z + TBu \tag{4.30}$$

For simplicity:

$$\dot{z} = A'z + B'u \tag{4.31}$$

$$y = C'z$$

$$A' = TAT^{-1}$$

$$B' = TB$$

After obtaining new system dynamics output matrix C' redefining as:

$$C' = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.32}$$

5 Model Predictive Control

Model Predictive Control (MPC) is a sophisticated control technique that utilizes a dynamic system model to forecast and optimize future performance. At each time step, MPC addresses an optimization problem within a finite future window to determine the optimal control actions that minimize a specified cost function, typically balancing tracking accuracy and control efforts. This method allows for the explicit management of constraints on both control inputs and system states. By implementing only the initial control action from the calculated sequence and continually updating the process with new measurements, MPC can effectively handle complex, multi-variable systems and adapt to changing conditions and constraints.

In this section, the solution of MPC will be examined in detail and methods and approaches are derived based on [23].

5.1 Discrete State Space Representation

Control systems that utilize model predictive control are constructed using a mathematical representation of the process. The specific model employed in designing the control system is a state-space representation. This state-space framework allows for the encapsulation of the necessary current information within the state variable, which is critical for making future predictions. State-space model obtained in previous section in discrete domain is given by:

$$z(k+1) = A'z(k) + B'u(k) \tag{5.1}$$

$$y(k) = C'z(k) \tag{5.2}$$

Taking the difference equation:

$$\Delta z(k+1) = z(k+1) - z(k) \tag{5.3}$$

$$\Delta u(k+1) = u(k+1) - u(k) \tag{5.4}$$

Difference states are desired as a state variable, so equation 5.3-5.4 can be solved into equation 5.1-5.2 :

$$\Delta z(k+1) = A'\Delta z(k) + B'\Delta u(k) \tag{5.5}$$

$$y(k+1) = C'A'\Delta z(k) + C'B'\Delta u(k) \tag{5.6}$$

Updated state variables of the system is written as:

$$z'(k) = \begin{bmatrix} \Delta z(k) & y(k) \end{bmatrix}^T \tag{5.7}$$

Updated state-space form of the system is written as:

$$\begin{bmatrix} \Delta z(k+1) \\ y(k+1) \end{bmatrix} = \begin{bmatrix} A' & \bar{0} \\ C'A' & 1 \end{bmatrix} \begin{bmatrix} \Delta z(k) \\ y(k) \end{bmatrix} + \begin{bmatrix} B' \\ C'B' \end{bmatrix} \Delta u(k) \tag{5.8}$$

and the output equation becomes:

5.2 Optimization Problem Definition

Given mathematical state and output difference equations, next step is to calculate future behaviour of system states and outputs by adjusting the control variable Δu. Since MPC is a finite time optimization control technique, 2 important parameters should be defined to describe ”finite-time”.One of them is ”Prediction window or horizon”, denoted N_p, which is a prediction duration or number of future steps from the current system. Other is ”Control Horizon”, denoted N_c, which is which is a control duration or number of control steps to reach control trajectory. Starting from at sampling k_0 control, trajectory and state prediction are represent:

$$z(k_0), z(k_0+1) \ldots z(k_0+N_p)$$

$$\Delta u(k_0), \Delta u(k_0+1) \ldots \Delta u(k_0+N_c-1)$$

Note that, it is desired that prediction horizon larger than control horizon($N_p \geq N_c$). The reason for this is practicality and computational efficiency: it’s often neither necessary nor

efficient to compute control actions far into the future when only the near-term actions will be implemented before the model is updated with new data. After the control horizon, the control action is typically held constant or assumed to follow a predefined strategy. This approach balances the need for forward planning with the realities of dynamic environments and the need to make real-time or near-real-time decisions in a control system.
It is possible to estimate future state variables using equation 5.1.

$$z(k_0+1) = A'z(k_0) + B'u(k_0)$$

$$z(k_0+2) = A'z'(k_0+1) + B'u(k_0+1)$$

$$= A'^2 z(k_0) + A'B'\Delta u(k_0) + B'u(k_0+1)$$

$$\vdots$$

$$z(k_0+N_p) = A'^{N_p} z(k_0) + A'^{N_p-1} B'\Delta u(k_0) + \ldots + A'^{N_p-N_c} B'\Delta u(k_0+N_c-1)$$

$$y(k_0+1) = C'A'z(k_0) + C'B'\Delta u(k_0)$$

$$y(k_0+2) = C'A'^2 z(k_0) + C'A'B'\Delta u(k_0) + C'B'\Delta u(k_0+1)\vdots$$

$$y(k_0+N_p) = C'A'^{N_p} z(k_0) + C'A'^{N_p-1} B'\Delta u(k_0) + \ldots + C'A'^{N_p-N_c} B'\Delta u(k_0+N_c-1)$$

$$Y = \begin{bmatrix} y(k_0+1) & y(k_0+2) & \cdots & y(k_0+N_p) \end{bmatrix}^T$$

$$\Delta U = \begin{bmatrix} \Delta u(k_0) & \Delta u(k_0+1) & \cdots \Delta u(k_0+N_c-1) \end{bmatrix}^T$$

$$Y = \Gamma z(k_0) + \Theta \Delta U \tag{5.9}$$

where Γ and Θ are given by:

$$\Gamma = \begin{bmatrix} C'A' \\ C'A'^2 \\ \vdots \\ C'A'^{N_p} \end{bmatrix}, \quad \Theta = \begin{bmatrix} C'B' & 0 & \cdots & 0 \\ C'A'B' & C'B' & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ C'A'^{N_p-1}B' & C'A'^{N_p-2}B' & \cdots & C'A'^{N_p-N_c}B' \end{bmatrix}$$

The goal is basically find the optimal solution by minimizing the cost function using control trajectory (ΔU). Cost function consist of tracking error which is the error between reference input and output and a matrix which adjust the actuators usage. Reference input signal $r(k_0)$ that is desired to track is given at time k_0 and assumed it will not change throughout the optimization window and can be defined as:

$$R_s = \underbrace{[r(k_0)\ r(k_0)\ \ldots\ r(k_0)]}_{N_p}^T$$

The cost function can be defined as:

$$J = (R_s - Y)^T(R_s - Y) + \Delta U^T R \Delta U \tag{5.10}$$

Actuator weight matrix $R = W_{mN_c \times mN_c} \cdot I_{mN_c \times mN_c}$ where $(\cdot)$ is the element wise multiplication operator, I is identity matrix ,m is number of inputs, W is the actuator weights tuning the usage of actuators.

To obtain optimal ΔU that will minimize J it can be expressed by combining equation 5.9 and 5.10.

$$J = (R_s - \Gamma z(k_0))^T(R_s - \Gamma z(k_0)) - 2\Delta U^T \Theta^T (R_s - \Gamma z(k_0)) + \Delta U^T(\Theta^T\Theta + R)\Delta U \tag{5.11}$$

Taking the derivative of J with respect to ΔU and setting it to zero gives us:

$$\frac{\partial J}{\partial \Delta U} = -2\Theta^T(R_s - \Gamma z(k_0)) + 2(\Theta^T\Theta + R)\Delta U = 0 \tag{5.12}$$

The necessary condition for the minimum J is obtained as:

$$\Delta U = (\Theta^T\Theta + R)^{-1}\Theta^T(R_s - \Gamma z(k_0)) \tag{5.13}$$

Assuming that, ”Hessian Matrix”, $(\Theta^T\Theta + R)^{-1}$ exists.

5.3 Receeding Horizon Control

In the receding horizon control strategy, only the initial control action $\Delta u(k)$ from this set is executed at each step. The remaining controls are not used immediately. As a new sampling instant arrives, the latest measurement updates system dynamics and new control trajectory is determined. This process of renewing the control action and advancing the horizon is the essence of the receding horizon control law. It involves the following steps:

1. At a specific point in time, the system predicts future states within a finite horizon using a mathematical model, taking the current system state into account.
2. An optimization problem is solved within this horizon to determine the sequence of control actions that minimizes a cost function. This cost function typically aims to achieve desired system behavior, such as tracking a setpoint or path, while also minimizing energy consumption and respecting various system constraints.
3. Only the first control in the optimized sequence is actually implemented.
4. As time progresses, the horizon shifts forward—hence the term receding and the process is conducted afresh with updated state information.

This control strategy is highly effective for complex systems with constraints, as it allows for dynamic adjustment to disturbances and system changes by continuously updating the predictions and control actions.

5.4 Constrained Optimization

Constraint optimization is essential because, in real-world systems, there are always limits to what can be achieved due to physical, safety etc. In this section, it is introduced how constraint optimization procedure combined with MPC. Equality and inequality constraint concepts and their solution methods in optimization are also explained. In scope of thesis, two constraints are defined on both actuators, one is equality constraint on lateral thruster, other is inequality constraint on aerodynamic fin deflection. For simplicity and desired to get a solution gradually, firstly, equality constraint is defined, optimal solution search for that. Then, inequality constraint is included the optimization problem.

Quadratic optimization problem under constraint consist of decision variable x, computable matrices E, F and constraint matrices M, γ, and it is typically defines as:

$$J = \frac{1}{2}x^T E x + x^T F \tag{5.14}$$

$$Mx \leq \gamma \tag{5.15}$$

Tailor equation 5.11-5.13 in defining new equation 5.14

$$x = \Delta U \tag{5.16}$$

$$E = \Theta^T \Theta + R \tag{5.17}$$

$$F = -\Theta(R_s - Fz(k_0)) \tag{5.18}$$

$$J = \frac{1}{2}\Delta u^T E \Delta u + \Delta u^T F \tag{5.19}$$

$$M\Delta u \leq \gamma \tag{5.20}$$

The Hessian matrix, represented by $E = \Theta^T\Theta + R$, is presumed to be positive definite (PD). When we say that the Hessian matrix is required to be PD, we are stating a condition for the optimization problem to have a unique global minimum. PD Hessian implies that the quadratic function is convex, meaning it curves upwards, and any local minimum is also a global minimum. This property is crucial because it guarantees that the optimization algorithm can reliably find the lowest point of the function, where the optimal solution lies. To solve constraint optimization problem, The Lagrangian function can be used. The Lagrangian enables the use of techniques like the method of Lagrange multipliers (λ) to find the extrema of the objective function subject to the constraints. Lagrangian function under equality constraint can be written as:

$$\mathcal{L}(x,\lambda) = f(x) + \sum_i \lambda_i g(x) \tag{5.21}$$

Tailor equation 5.21 in our problem, it is required to solve it with equation 5.19- 5.20

$$f(x) = J \tag{5.22}$$

$$g(x) = M\Delta u - \gamma \tag{5.23}$$

$$\bar{J} = \mathcal{L} \tag{5.24}$$

Final form of cost function is denoted:

$$\bar{J} = \frac{1}{2}\Delta U^T E \Delta U + \Delta U^T F + \lambda(M\Delta U - \gamma) \tag{5.25}$$

In order to find the minimum point of the cost function, it is required to take partial derivatives of cost function with respect to decision variables x and λ.

$$\frac{\partial \bar{J}}{\partial \Delta U} = E\Delta U + F + M^T\lambda = 0 \tag{5.26}$$

$$\frac{\partial J}{\partial \lambda} = M\Delta u - \gamma = 0. \tag{5.27}$$

$$\frac{\partial \mathcal{L}}{\partial \Delta U} = 0 \tag{5.28}$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = 0 \tag{5.29}$$

Two linear equations are obtained in equation 5.26 and 5.27 and decision variables (ΔU) and λ are unknown parameters. Necessary condition to find the unknowns, dimension of ΔU is m and λ is n, m+n linear independent equations are required. Control trajectory and lagrange multiplier are obtained by calculating equation 5.26 and 5.27:

$$\Delta U = -E^{-1}(M^T\lambda + F) \tag{5.30}$$

$$\lambda = -(ME^{-1}M^T)^{-1}(\gamma + ME^{-1}F) \tag{5.31}$$

It is interesting to note that equation 5.30 can be written as two terms:

$$\Delta U = -E^{-1}F - E^{-1}M^T\lambda \tag{5.32}$$

Notice that first term $-E^{-1}F$ is the same as unconstrained solution. Second term is added because of existing constraint.

6 Design of MPC

6.1 Discretization

The first step of the design process of discrete MPC is to obtain the discrete-time state-space representation of the system. In section 4, parametric continuous state-space equations are obtained. Required parameters in equation 5.8 are defined in Table 6.1 - 6.2. In the analysis, the setup is generated at a point of $h = 20\,\mathrm{km}$ and Mach $= 3$. The "1976 COESA (Committee on Extension to the Standard Atmosphere) Atmosphere Model" in MATLAB is used to calculate air density and the speed of sound. The model is a widely used standard for atmospheric properties. It's often utilized in aerospace engineering for simulations and analysis of flight dynamics, missile trajectories, and other aeronautical applications. MATLAB provides tools and functions to work with this model effectively.

Table 6.1 Missile's overall parameters

Parameter	**Value**
Mass (m) [kg]	255
Missile diameter (d) [m]	0.317
Distance between CG and nose of the missile (x_{cg}) [m]	2.569
Moment of inertia (I_{yy}) [kg·m^2]	306.3
Lateral Thruster Force (F_{thr}) [N]	2200
Distance between Lateral Thruster and center of mass (l_{thr}) [m]	1.26

Table 6.2 Aerodynamic Coefficient

C_{z_α}	$C_{z_{\delta_e}}$	C_{z_q}	C_{m_α}	$C_{m_{\delta_e}}$	C_{m_q}
-21	-4.41	-150	-15	-35	-1076

Defining parameters substitute in equation 4.31 :

$$\dot{z} = A'z + B'u \tag{6.1}$$

$$\begin{bmatrix} \dot{a}_z \\ \dot{q} \\ \dot{\delta} \\ \ddot{\delta} \\ \dot{F}_{thr} \end{bmatrix} = \begin{bmatrix} -0.19 & -172.72 & 0 & -36.27 & -0.82 \\ 0.09 & -0.15 & -34.47 & 0 & -0.0045 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & -24674 & -188.49 & 0 \\ 0 & 0 & 0 & 0 & -150 \end{bmatrix} \begin{bmatrix} a_z \\ q \\ \delta \\ \dot{\delta} \\ F_{thr} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \omega_n^2 & 0 \\ 0 & \tau \end{bmatrix} \begin{bmatrix} \delta_{com} \\ F_{thr_{com}} \end{bmatrix} \tag{6.2}$$

$$a_z = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} a_z \\ q \\ \delta \\ \dot{\delta} \\ F_{thr} \end{bmatrix} \tag{6.3}$$

To discretize the system sample time have to be specified. As a design paremeter T_s, is selected considering the suggestion of "Matlab MPC Toolbox Starting Guide" [24]. It proposes that selecting sample time so that covering rise time of the plant at least 10 to 30 samples. In this proposal is for SISO system but there are two actuator in our system. Thus, rise time of the plant is analyzed for both actuators. Open loop transfer functions of both actuators can be defined as:

$$\frac{a_z}{\delta_e} = \frac{-8.9498e5(s+12.89)(s-12.74)}{(s^2+0.3497s+16.27)(s^2+188.5s+2.467e4)} \tag{6.4}$$

$$\frac{a_z}{F_{thr}} = \frac{0.824(s^2+0.15s+142)}{(s+150)(s^2+0.3497s+16.27)} \tag{6.5}$$

According to the step response of TF's, rise time are measured by 0.25 and 0.36 second respectively. According to results and defined rule in [24], T_s is selected 0.025 second which covers the 10 times of rise time of first TF and approximately 15 step for second TF. This is important because of observing whole transient dynamics of the system. ”Zero-Order Hold (ZOH)” method is used to discretize of continuous system model in equation 6.2. The method assumes that the input to the continuous system is piece-wise constant over each sampling interval. In other words, it holds the input signal constant at its current value until the next sampling instant. In MATLAB, it is used the `c2d` function with the ’zoh’ option to perform this discretization:

$$\begin{bmatrix} a_z(k+1) \\ q(k+1) \\ \delta(k+1) \\ \delta(k+1) \\ F_{thr}(k+1) \end{bmatrix} = \begin{bmatrix} 0.99 & -4.29 & 40.52 & 0.0047 & -0.0052 \\ 0.0023 & 0.99 & -0.23 & -0.0017 & -3.8728\times10^{-5} \\ 0 & 0 & -0.0948 & 0 & 0 \\ 0 & 0 & 0 & -0.0948 & 0 \\ 0 & 0 & 0 & 0 & 0.0235 \end{bmatrix} \begin{bmatrix} a_z(k) \\ q(k) \\ \delta(k) \\ \delta(k) \\ F_{thr}(k) \end{bmatrix} \tag{6.6}$$

$$+ \begin{bmatrix} -38.67 & 0.0055 \\ -0.63 & -0.0001 \\ 1.1 & 0 \\ 0 & 0 \\ 0 & 0.9765 \end{bmatrix} \begin{bmatrix} \delta_{com} \\ F_{thr_{com}} \end{bmatrix}$$

6.2 Tuning Parameters

In this part, three tuning parameters, which are prediction horizon, control horizon, and actuator weight, are selected and the selection process is clarified. These parameters are very crucial because they generate the cost function that is desired to be minimized. For this reason, autopilot performance is highly related to these parameters. Tuning parameters can be defined as:

- The prediction horizon determines how many future time steps the controller predicts the system's behavior. A longer prediction horizon considers long-term effects, improving prediction accuracy. Thus, it offers better stability for systems requiring long-term foresight. However, it leads to slower execution due to higher computational complexity, in addition to causing sluggish performance. In literature, it is typically chosen as ranging from 10 to 20.
- The control horizon defines how many future control actions the controller optimizes. A longer control horizon affects the system performance in the same way as the prediction horizon. A low value results in aggressive performance but might be oscillatory, while a high value results in sluggish but more stable performance. In literature, it is typically chosen as % 10-30 of N_p.
- By calibrating the weights in the cost function, you can tune the MPC to favor either aggressive target tracking through lower actuator weights or smoother, more robust actuation via increased weights on the control effort. In Equation 5.13, the actuator input is found to minimize the cost function and is again expressed as:

$$\Delta U = (\Theta^T \Theta + R)^{-1} \Theta^T (R_s - \Gamma z(k_0)) \tag{6.7}$$

$$R = W_{mN_c \times mN_c} \cdot I_{mN_c \times mN_c} \tag{6.8}$$

It can be seen that actuator weight R highly related to $\Theta^T \Theta$ matrices. Thus, to express better the proportional relation between actuators, W is defined as:

$$W(i,i) = \Theta^T \Theta(1,1) w_1, \quad i = odd \tag{6.9}$$

$$W(i,i) = \Theta^T \Theta(2,2) w_2, \quad i = even \tag{6.10}$$

Rest of the thesis w_1, aerodynamics surface weight , and w_2, lateral thruster weight, are used to describe weight matrix of the actuator.

6.3 Unconstrained Solution

Before moving to solve the constraint solution, it might be helpful to analyze system performance in which actuator constraints are not included. It is considered that exploring the adjustable parameters mentioned in the previous section within a defined space, while considering recommendations from the literature, would provide valuable insights into understanding system performance and validating the established guidelines. The actuator weights are expected to be lower for the side thruster, meaning that the thruster is intended to be used more aggressively than the fin. Therefore, a scan space was examined as follows:

$$\begin{gathered} 5 \leq N_p \leq 30 \\ 1 \leq N_c \leq 5 \\ 1 \leq w_1 \leq 0.0001 \\ 10 \leq w_1/w_2 \leq 10000 \end{gathered}$$

Due to the abundance of parameters, instead of showing the entire space, contour graphs in Figure 6.1-6.3 of the critical system performance parameters such as rise time, settling time, and overshoot have been displayed with two different actuator weights for N_p and N_c to provide more clarity and comparability. When examining the figures, an increase in N_p significantly slows down the system response, while an increase in N_c, although not as effective as N_p, also slows down the system response. As expected, reducing the actuator weights accelerates the system response. The amount of overshoot also increases as N_p increases. Moreover, trim δ_e and thruster thrust in a 5g acceleration command are shown in Figure 6.4-6.5. These plots are added because actuators have limits, and they help in selecting tuning parameters while taking into consideration the limits. The reason for selecting 5g is that it is thought to be sufficient to defend aircraft at high altitude (20km). Here, because the acceleration response decreases with the larger value of N_p and N_c, the trim actuator values also decrease accordingly. As expected, the reduction in actuator weights drives the system towards more aggressive actuator usage, which increases the trim actuator values.

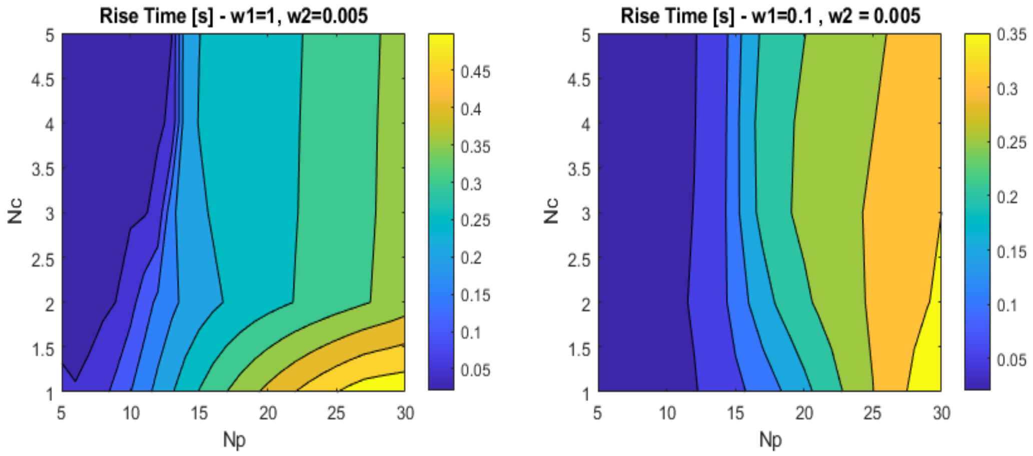


Figure 6.1 Rise Time Contour Comprasion for two different actuator weight wrt N_p, Nc

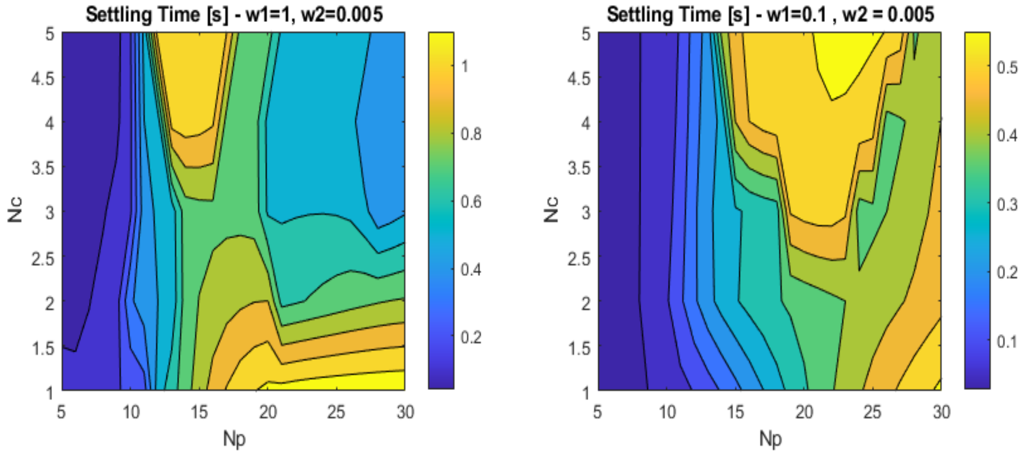


Figure 6.2 Settling Time Contour Comprasion for two different actuator weight wrt N_p, N_c

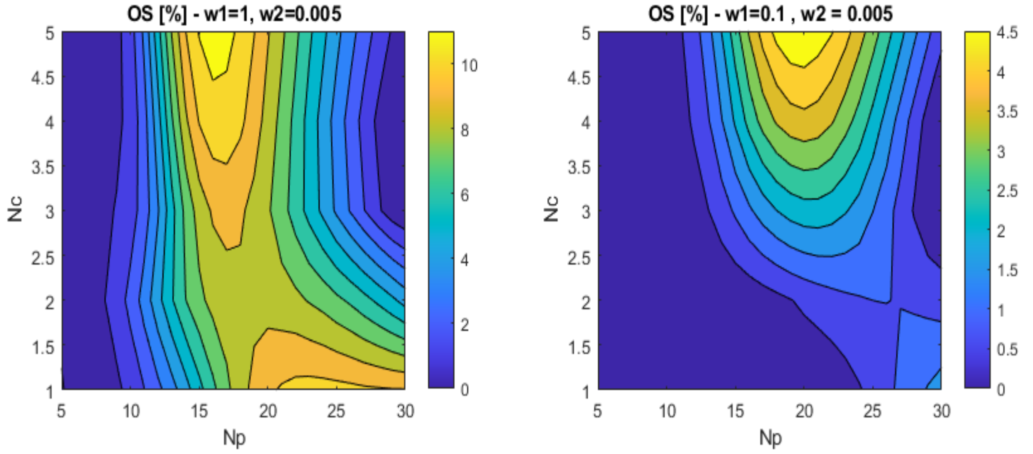


Figure 6.3 OS Contour Comprasion for two different actuator weight wrt N_p, N_c

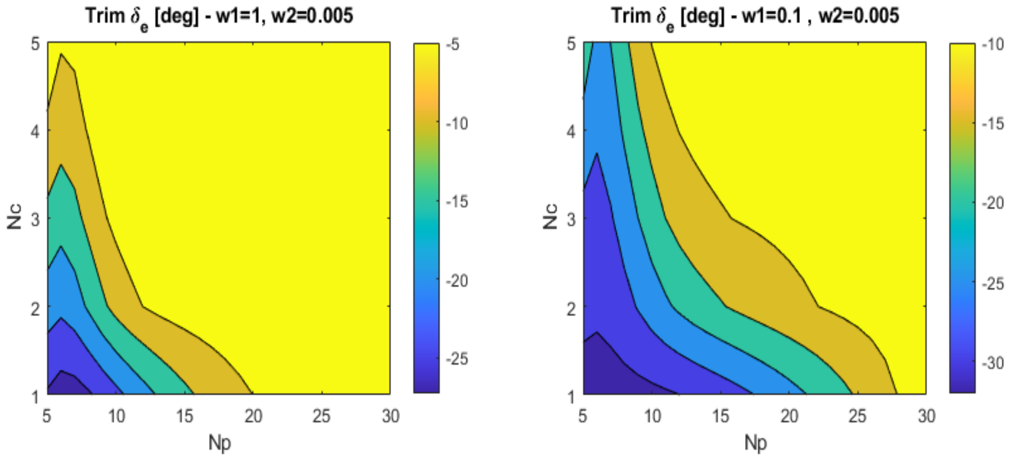


Figure 6.4 Trim δ_e under 5g acceleration command Contour Comprasion for two different actuator weight wrt N_p, Nc

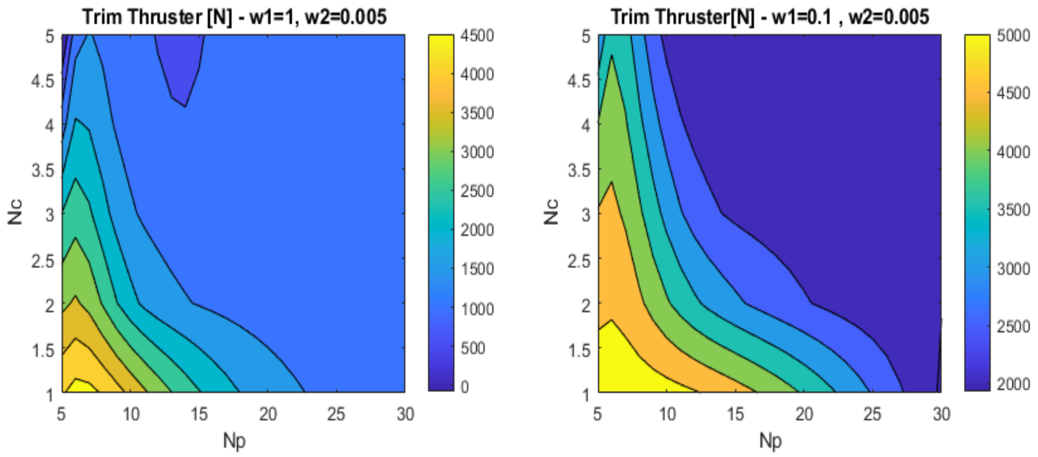


Figure 6.5 Trim Thruster Thrust under 5g acceleration command Contour Comprasion for two different actuator weight wrt N_p, N_c

Considering both the processing load and agility, a solution is being sought where the values of horizons are low. For $N_p = 10$, $N_c = 2$, system performance is evaluated as good enough for two different weights of actuators. Step response comparisons against a 5g acceleration command response are shown in Figure 6.6-6.8. When analyzing the figures, the missile operates with a higher angle of attack and requires less actuator power to reach a trim state when the fin weight is high. This is clearly shown in Figure 6.8. When considering the thrusters near the nose and the fins at the tail, the directions of the forces from the actuators are the same, but their moments are opposite. Increasing the fin weight relative to the lateral thruster weight allows the missile to generate more moment. Although this may result in some performance loss, it is critical for more effective actuator usage. Since performance is the main criterion of this study, a value of $w_1/w_2 = 20$ has been taken as the starting point for subsequent analyses.

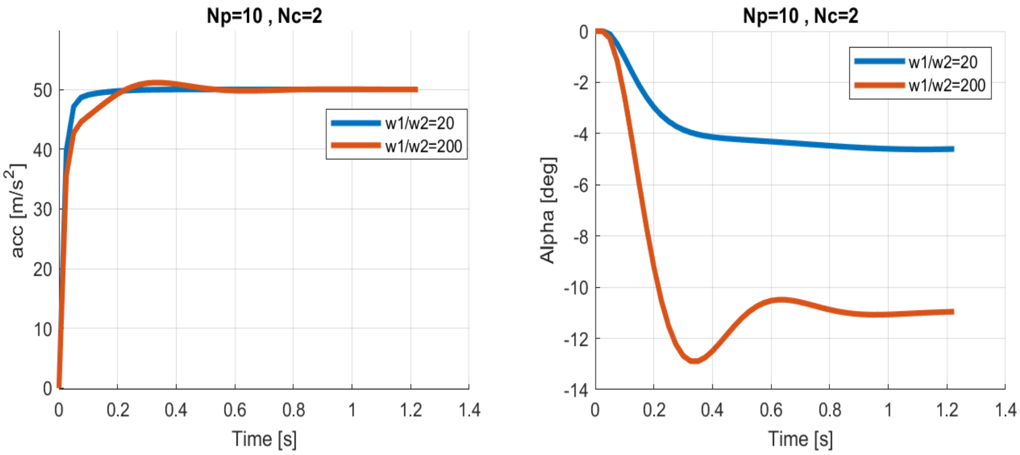


Figure 6.6 Acceleration and Angle of Attack Comparison under 5g acceleration command for two different actuator weight

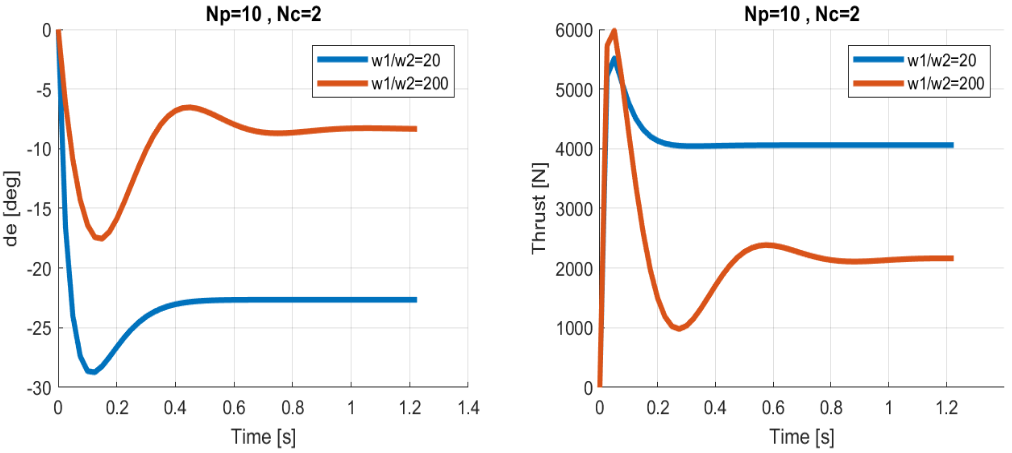


Figure 6.7 δ_e and Thruster Comparison under 5g acceleration command for two different actuator weight

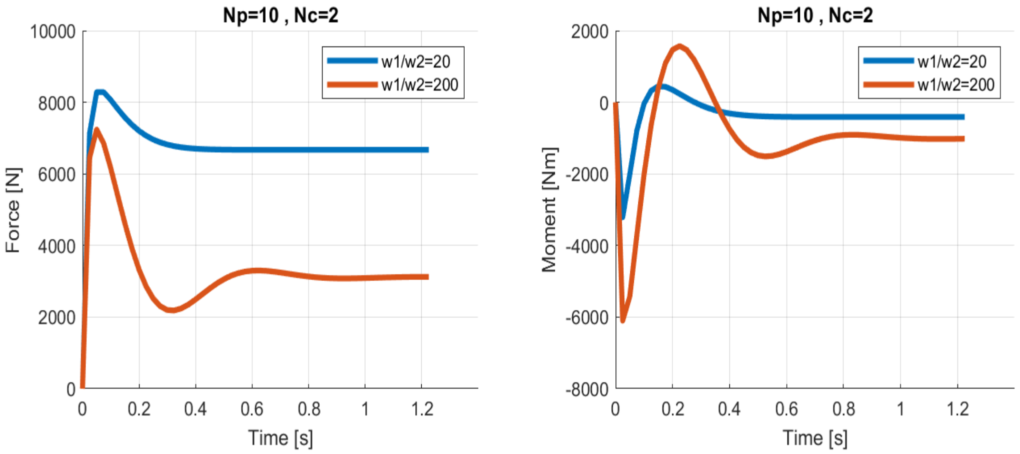


Figure 6.8 Force and Moment under 5g acceleration command for two different actuator weight

6.4 Lateral Thruster Constraint Solution with Mixed Integer Quadratic Programming

In this section, the optimization problem will be examined under the constraint by taking into account the lateral-thruster equality constraint. As mentioned in previous sections, there are multiple thrust options available in the system, and the most optimal one is desired to be selected as the solution. The MIQP solution has been considered in two stages: first stage is called generating decision tree which means a system constraint is defined for all possible firings in the thrust sequence, and in the second stage is called search algorithm which means all solutions are examined and the smallest value that minimizes the defined penalty function will be operated as the result. Recall the cost function and solution of equality constraint optimization:

$$\bar{J} = \frac{1}{2}\Delta U^T E \Delta U + \Delta U^T F + \lambda(M\Delta U - \gamma) \tag{6.11}$$

$$\Delta U = -E^{-1}(M^T\lambda + F) \tag{6.12}$$

$$\lambda = -(ME^{-1}M^T)^{-1}(\gamma + ME^{-1}F) \tag{6.13}$$

E and F values are determined combining the system dynamics and tuning parameters. Constraint matrices M and γ should be generated for the first part of the solution. Although only first element of control input is applied to the system according to receeding horizon control, constraint should define all control sequence. General representation of constraint matrices are:

$$M\Delta u \leq \gamma \tag{6.14}$$

For multi input case, inputs are aerodynamic surface and lateral thruster which denoted as Δu_1 and Δu_2 respectively. Our control inputs are defined as difference of control state but it is desired to identify the constraints on control state. Thus, constraints equations might be written with respect to control state and rearrange them to difference of control state. Note that constraints are applied only lateral thruster in this section and all future control inputs , so there is no constraint on aerodynamic surfaces. Relationship between control state and its difference can be written as:

$$u_2(k) = \Delta u_2(k) + u_2(k-1) \tag{6.15}$$

$$u_2(k+1) = \Delta u_2(k+1) + u_2(k) \tag{6.16}$$

$$\vdots$$

$$u_2(k+N_c-1) = \Delta u_2(k+N_c-1) + u_2(k+N_c-2) \tag{6.17}$$

Constraint equation can be written as:

$$u_2(k) = \gamma_i \tag{6.18}$$

$$u_2(k+1) = \gamma_i \tag{6.19}$$

$$\vdots$$

$$u_2(k+N_c-1) = \gamma_i \tag{6.20}$$

Combining the equation 6.15-6.17 to 6.18-6.20

$$\Delta u_2(k) = \gamma_i - u_2(k-1) \tag{6.21}$$

$$\Delta u_2(k+1) = \gamma_i - u_2(k) \tag{6.22}$$

$$\Delta u_2(k+1) = \gamma_i - (\Delta u_2(k) + u_2(k-1)) \tag{6.23}$$

$$\Delta u_2(k+1) + \Delta u_2(k) = \gamma_i - u_2(k-1)) \tag{6.24}$$

$$\vdots$$

$$\Delta u_2(k+N_c-1)+\Delta u_2(k+N_c-2)+\cdots+\Delta u_2(k+1)+\Delta u_2(k)=\gamma_i-u_2(k-1) \tag{6.25}$$

Equations in matrix form can be written as:

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & \cdots \\ 0 & 1 & 0 & 1 & 0 & 0 & \cdots \\ 0 & 1 & 0 & 1 & 0 & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & 0 & 1 & 0 & \cdots & 1 \end{bmatrix}_{m\times 2N_c} \begin{bmatrix} \Delta u_1(k) \\ \Delta u_2(k) \\ \Delta u_1(k+1) \\ \Delta u_2(k+1) \\ \vdots \\ \Delta u_1(k+N_c-1) \\ \Delta u_2(k+N_c-1) \end{bmatrix}_{2Nc\times 1} = \begin{bmatrix} \gamma_i-u_2(k-1)) \\ \gamma_i-u_2(k-1)) \\ \vdots \\ \gamma_i-u_2(k-1)) \\ \gamma_i-u_2(k-1)) \end{bmatrix}_{m\times 1} \tag{6.26}$$

where m is number of constraints and equation 6.24 is appropriate the form of $M\Delta u=\gamma$. Decision tree is a function of available lateral thruster option and control horizon value N_c and it can be expressed as:

$$G_{N_c\times(\underbrace{a\times a\times\ldots\times a}_{N_c})}=C(\underbrace{F,F,\ldots,F}_{N_c}) \tag{6.27}$$

$$\min_{\Delta U,\lambda}\mathcal{L}(\Delta U,\lambda) \tag{6.28}$$

where a is number of lateral thruster option (length of F array), C is combination operator which generates all combination of the vectors. Decision three length is defined $(L=\underbrace{a\times a\times\ldots\times a}_{N_c})$. Algorithm to find a optimal solution from decision tree is explained in Algorithm 1.

According to defining constraint on lateral thruster, step response of the system are shown Figure 6.9-6.11. When the results are examined, it is seen that the rise time is approximately 100ms and there is no overshoot. It is observed that the elevator angle and the lateral thruster converges at some points that lead to the solution very quickly.

Algorithm 1 Searching Algorithm

```
Generate decision tree.
    $G = C(\underbrace{F, F, \ldots, F}_{N_c})$
for $i = 1, 2, 3, \ldots$,L-1,L do
    Each firing options in decision three assign as equality constraint and cost function value is recorded.
        $\gamma = G(:, i)$
        $\lambda = -(ME^{-1}M^T)^{-1}(\gamma + ME^{-1}F)$
        $\Delta U = -E^{-1}(M^T\lambda + F)$
        $\bar{J}(i) = \frac{1}{2}\Delta U^T E \Delta U + \Delta U^T F + \lambda(M\Delta U - \gamma)$
end for
Select the optimal solution which makes cost function minumum. (Optimal index $i^o$ )
    $i^o = index(min(J))$
    $\Delta U(i^o) = Optimal\ Solution$
First step of control trajectory is presented as a solution. =0
```

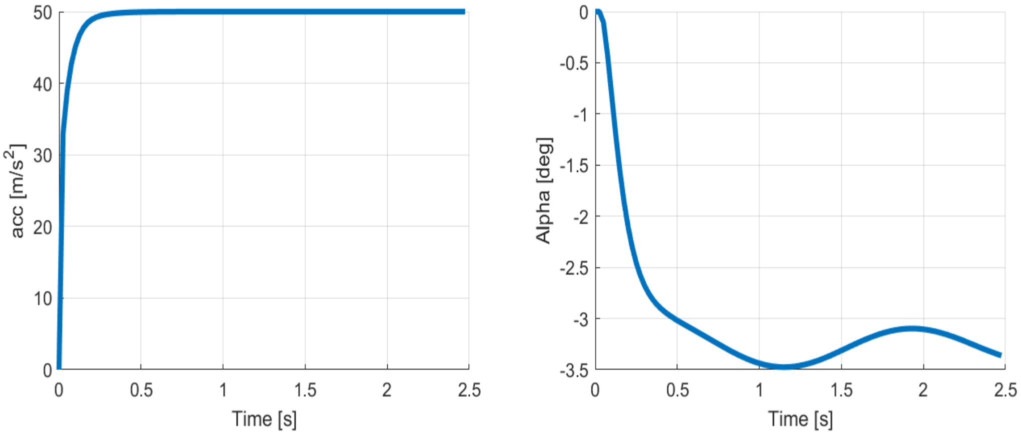


Figure 6.9 Acceleration and Angle of Attack Response under 5g acceleration command and lateral thruster constraint

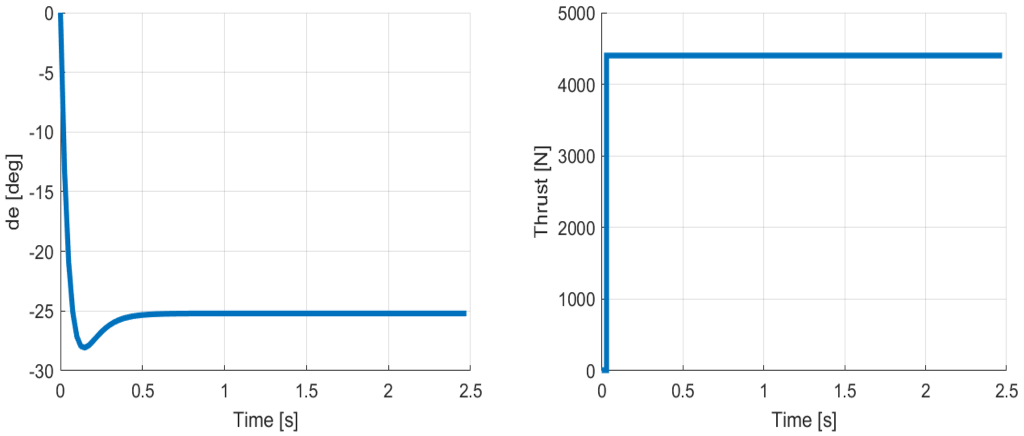


Figure 6.10 δ_e and Thruster Response under 5g acceleration command and lateral thruster constraint

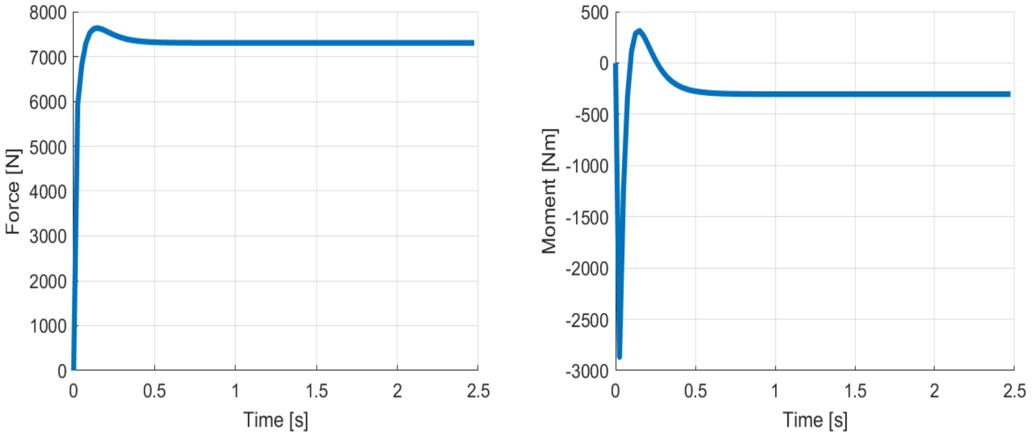


Figure 6.11 Force and Moment Response under 5g acceleration command and lateral thruster constraint

However, system request from lateral thruster to produce thrust continuously. As is known, there is a limited number of thrusters in the system according to the lateral thruster model defined in this thesis and real applications. For this reason, it is aimed to start the control as a blend and then switch the control entirely to the aerodynamic surfaces. To achieve this solution, the decision tree needs to be organized. By setting the final value of the control horizon to 0 for the lateral thruster, the system is forced to finish the solution without using the lateral thruster in the last control input. Now, new decision tree can be defined as:

$$G = C(\underbrace{F, F, \ldots, F}_{N_c - 1}) \tag{6.29}$$

$$G(N_c, :) = 0 \tag{6.30}$$

Algorithm 1 is repeated with new decision tree and same tuning parameters, and results are shown in Figure 6.12-6.14- As seen from the figures, thrust request is converged zero as time goes and elevator deflection do all the job that keep tracking the reference acceleration command. On the other hand, setting the last element of control sequence to zero reduced the length of the decision tree and results less computational effort. Rapid change of thrust level on lateral thruster prevents the smooth tracking performance and creates undershoots but it allows the reach the desired acceleration value very quickly. Since lateral thruster is required for terminal phase which is the phase missile needs agile maneuvers in short time, the overshoot at response might be tolerated.

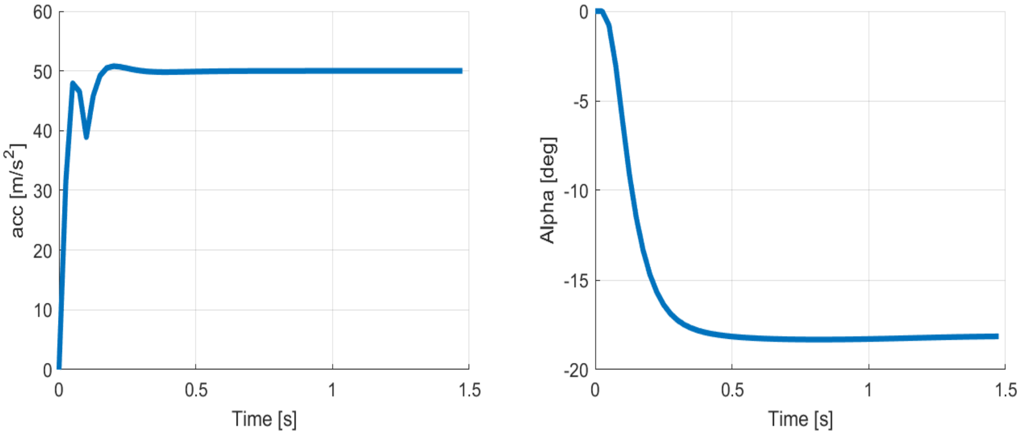


Figure 6.12 Acceleration and Angle of Attack Response under 5g acceleration command and lateral thruster constraint for limited thruster

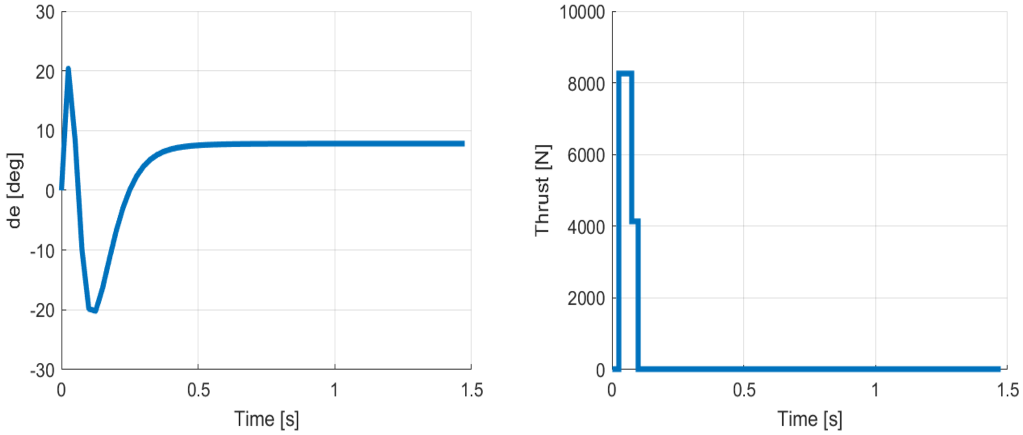


Figure 6.13 δ_e and Thruster Response under 5g acceleration command and lateral thruster constraint for limited thruster

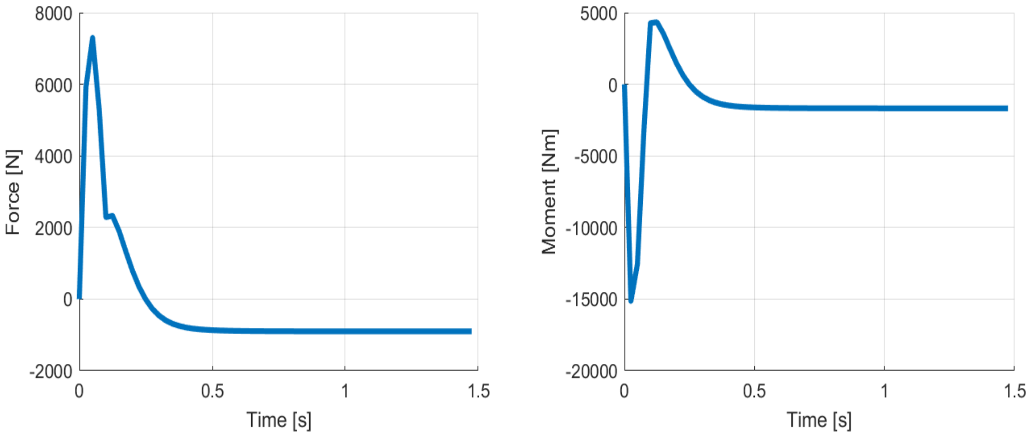


Figure 6.14 Force and Moment Response under 5g acceleration command and lateral thruster constraint for limited thruster

6.5 Lateral Thruster and Elevator Deflection Constraints Solution MPC based Weight Adaptation

In previous sections, the performance of the autopilot under thrust constraint was examined. In this section, the autopilot performance under both the thruster and elevator deflection constraint will be analyzed. Uniquely, the elevator deflection constraint is defined as an inequality constraint in the system. For the optimization solution under the inequality constraint, it should be determined whether the inequality constraint is an active constraint. Two different approaches are shown in this section. Firstly, if it is an active constraint, it should be added to the equality constraint equations, and the solution described in the previous section should be implemented. If not, equality constraint solution obtained in previous section can be applied directly. Elevator deflection limits can be defined as:

$$-25^o < \delta_e < 25^o \tag{6.31}$$

New constraint matrices can be defined as:

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & \cdots \\ 0 & 1 & 0 & 1 & 0 & 0 & \cdots \\ 0 & 1 & 0 & 1 & 0 & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & 0 & 1 & 0 & \cdots & 1 \end{bmatrix}_{m\times 2N_c} \begin{bmatrix} \Delta u_1(k) \\ \Delta u_2(k) \\ \Delta u_1(k+1) \\ \Delta u_2(k+1) \\ \vdots \\ \Delta u_1(k+N_c-1) \\ \Delta u_2(k+N_c-1) \end{bmatrix}_{2Nc\times 1} = \begin{bmatrix} \delta_{max} - u_1(k-1)) \\ \gamma_i - u_2(k-1)) \\ \gamma_i - u_2(k-1)) \\ \vdots \\ \gamma_i - u_2(k-1)) \\ \gamma_i - u_2(k-1)) \end{bmatrix}_{m\times 1} \tag{6.32}$$

Note that, only first element of elevator deflection in control sequence is constrained because receeding control use only first element of control sequence. Algoritmic flow is shown in Algorithm 2.

Algorithm 2 Elevator Constraint Addition Algorithm

```
Apply Algorithm 1
if $|\delta_e| > \delta_{max}$ then
    Generate constraint matrix including elevator constraint (equation 6.31)
    Apply Algorithm 1 with new matrices.
else
    continue Algorithm 1
end if=0
```

To test the algorithm, $-70m/s^2$ step acceleration command is applied to the system and results are shown in Figure 6.15-6.17 .

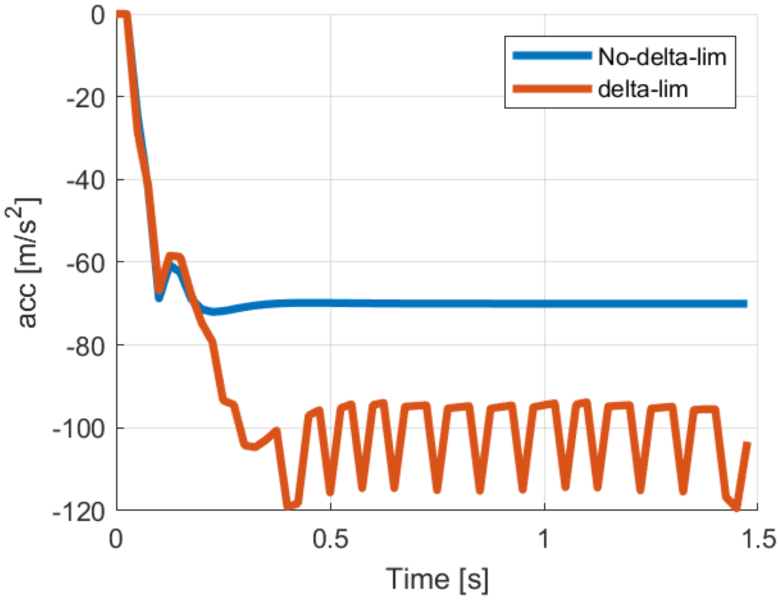


Figure 6.15 δ_e limit comparison, acc vs time

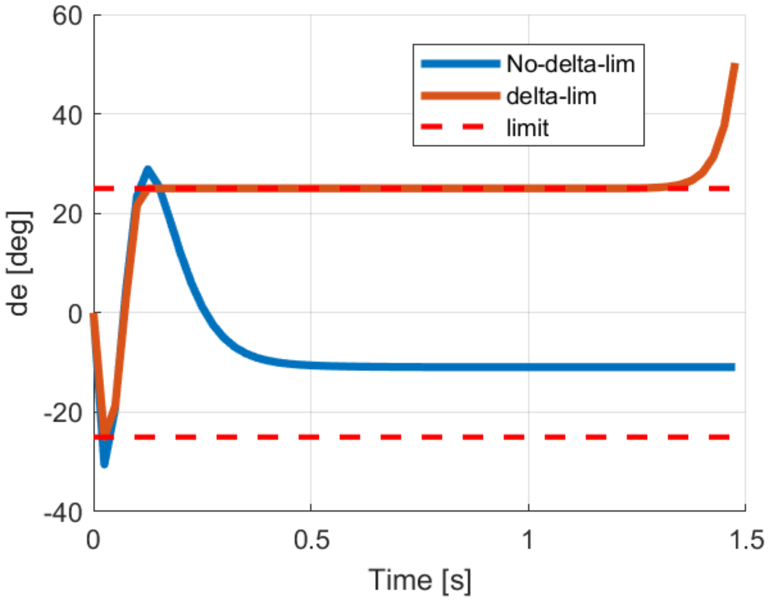


Figure 6.16 δ_e limit comparison, δ_e vs time

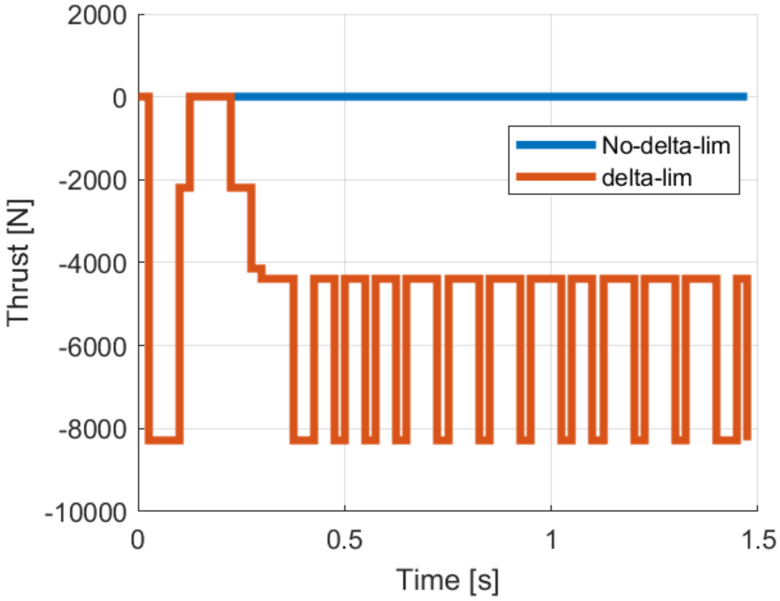


Figure 6.17 δ_e limit comparison, thr vs time

When the figures are examined, hitting the limit of the elevator angle make the system unstable because in case of removing the limit of elevation can track the acceleration command in desired way. In other words, defining an equality constraint for both actuators forces the system into a single solution which possibly distract the solution from optimal one. To overcome this issue, comprehensive approach taking into consideration elevator limit that is not exceed all flight condition is proposed ”Online MPC based Adaptive Weight Algorithm” method. Similar approach is studied in [4] but the approach was used to reject disturbance instead of preventing the actuators limit. The proposed solution at every step relies on examining the solution using MPC in narrow windows. If the elevator deflection limit is not exceeded in the this window, the solution proceeds. If there is an exceedance, the solution is repeated by updating the elevator deflection weight and flowchart of this algorithm is shown in Figure 6.18 and algorithm is presented in Algorithm 3. The aim here is to prevent hitting the limit over a broader range rather than just limiting the next step. The algorithm elevator can work in the opposite direction - to accelerate, just as it works to slow down. Thus, it may be possible to offer adaptive control in terms of performance during flight. This is because the decrease in the number of jets and their inability to produce the desired thrust will leave all the work to the elevator, allowing the updated weights to be reduced again for better utilization. However, since the system’s control aims to completely rely on the elevator in a steady state, it is intended that the determined elevator weights should have limit values under elevator control that will ensure closed-loop stability. This issue will not be examined in this study but may find a place in future work.

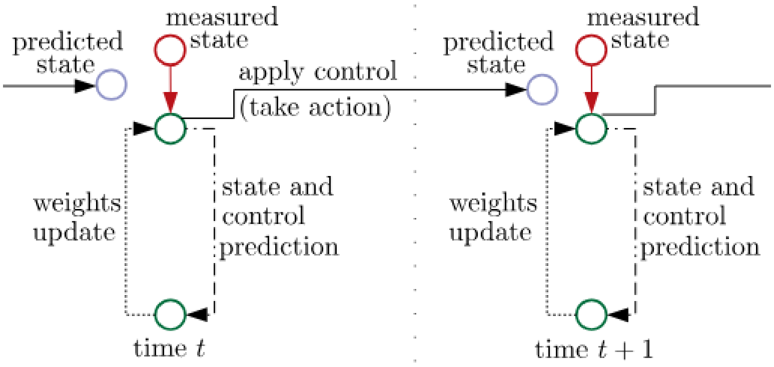


Figure 6.18 Online MPC based Adaptive Weight Flow Chart [4]

To test Algorithm 3 previous test case which is 7g acceleration command applied to the system and results are shown in Figure 6.19-6.22. When the results are examined, it is observed that when the step input command is received, Algorithm 3 detects a fin overshoot within a 10-step window and prevents it from reaching the limit by increasing the fin weight. When Figure 6.22 is examined, it is seen that the weight value increases from 0.1 to 0.1728 in a total of 3 iterations. Additionally, the instability resulting from Algorithm 2 has been eliminated, and the acceleration response command is followed with an approximate rise time of 0.75 ms.

Algorithm 3 Online MPC based Adaptive Weight Algorithm

```
$\delta_{check} = 0$
while $\delta_{check} = 0$ do
    for $j = 1, 2, 3, \ldots$,V-1,V do
        Algorithm 1 is called and the solution is progressed for V steps. (Assuming the reference input does not change)
        The fin values are recorded for V steps
    end for
    if $|{\delta_e}^{V_{max}}| > |\delta_{max}|$ then
        ${\delta_e}^{V_{max}}$, It represents the highest fin angle observed for V steps.
        The fin weight is increased by the adaptive coefficient ($a_k$).
            $w_1 = w_1 * a_k$
    else
        $\delta_{check} = 1$
        V adımlık kontrol yörüngesinin ilk adımı çözüm olarak çıkarılır.
    end if
end while=0
```

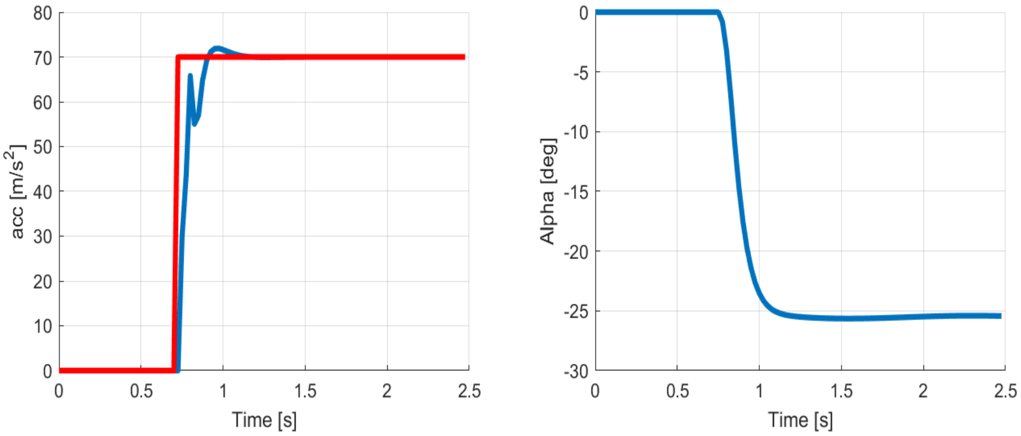


Figure 6.19 Acceleration and Angle of Attack Response under 7g acceleration command with adaptive weight algorithm

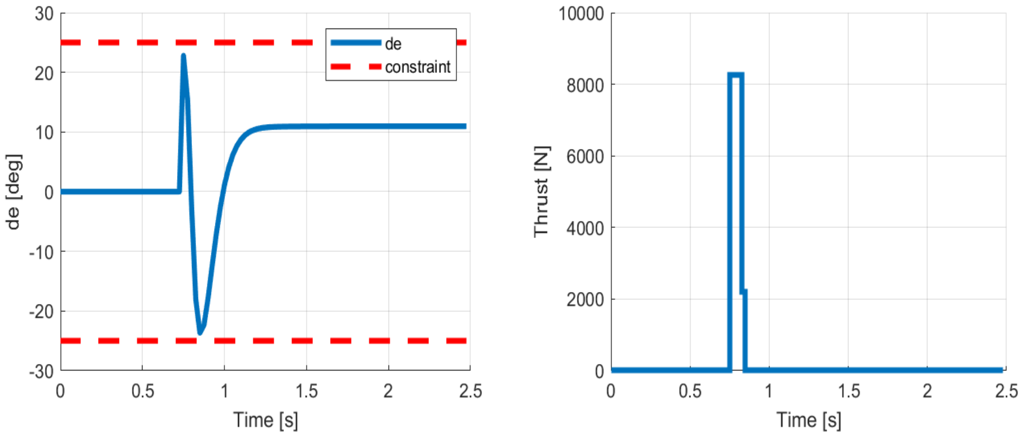


Figure 6.20 δ_e and Thruster Response under 7g acceleration command with adaptive weight algorithm

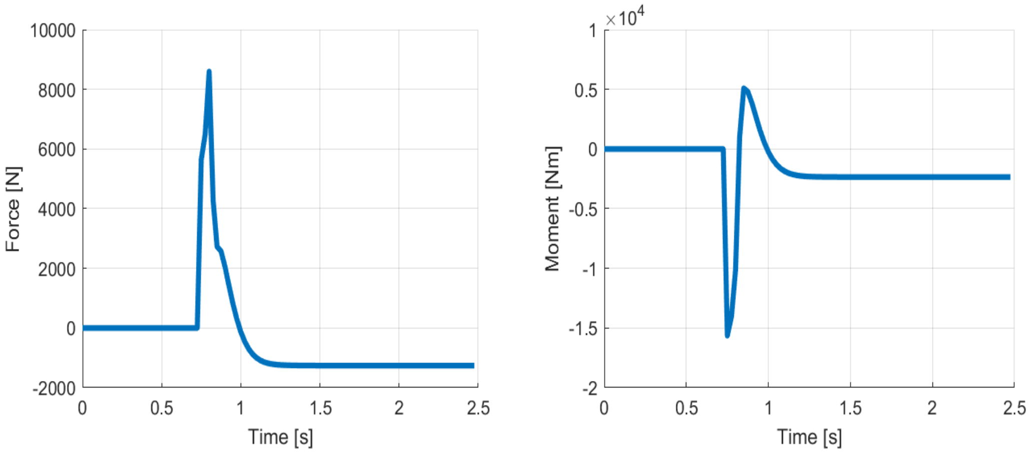


Figure 6.21 Force and Moment Response under 7g acceleration command with adaptive weight algorithm

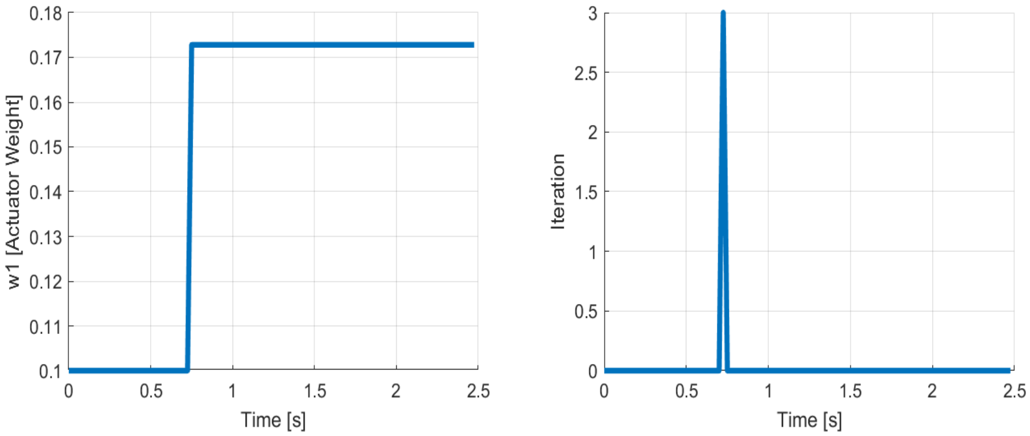


Figure 6.22 Weight and Iteration Response under 7g acceleration command with adaptive weight algorithm

So far, system performance tests have been conducted using step command responses for zero initial conditions. To create a more challenging scenario and measure algorithm performance, step inputs are desired to be generated for different initial condition states. The MPC-based adaptive weight method has performed well under different initial conditions and step input scenarios. When the Figure 6.23-6.26 is examined, it adapts to the solution under constraint by updating the elevator weight the moment a command exceeding the limit is given. Note that, weight value is protected after updating instead of reset the design value which means number of required iteration reduced after coming big update. This might make the system performance more slow than design. If computation time is met easily by missile computer, it can be reset for more efficient performance. Figure 6.26 shows that actuator weight is risen from 0.1 to 4.6 but after a big update at the 3rd second using 10 iterations, only one iteration is sufficient in rest of the solution. To better illustrate the algorithm's performance and the computational load of iterations, the MATLAB `cputime` function was utilized to measure the computational time. The test have been performed on AMD Ryzen 7 2700X 3.7GHz 16 Gb RAM but only one core of the CPU is used for this operation. CPU time is calculated each step and shown in Figure 6.27. Apparently seen that at the 3rd second cpu time jump to approximately 8 times the normal operating time.

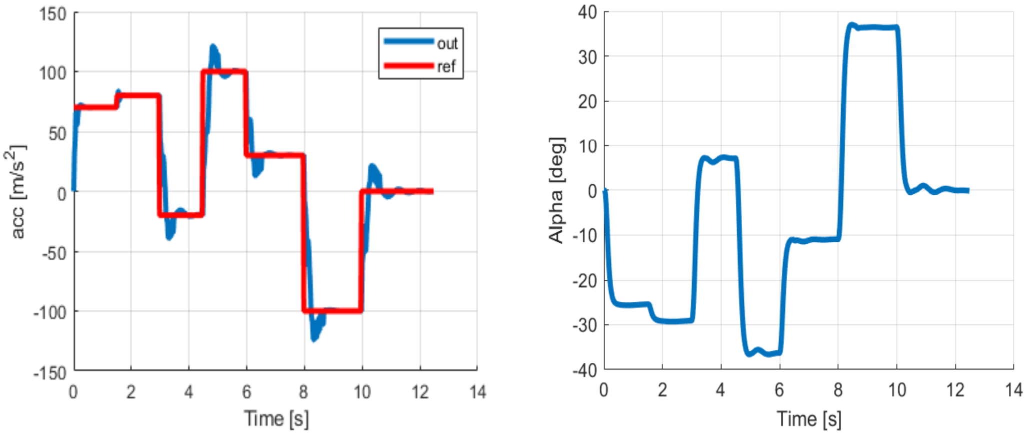


Figure 6.23 Acceleration and Angle of Attack Response under different IC's acceleration command scenario with adaptive weight algorithm

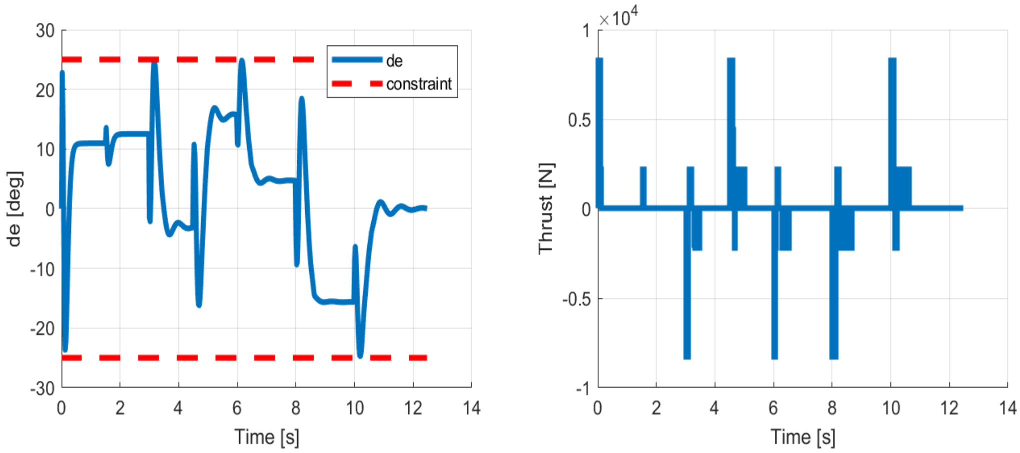


Figure 6.24 δ_e and Thruster Response under different IC's acceleration command scenario with adaptive weight algorithm

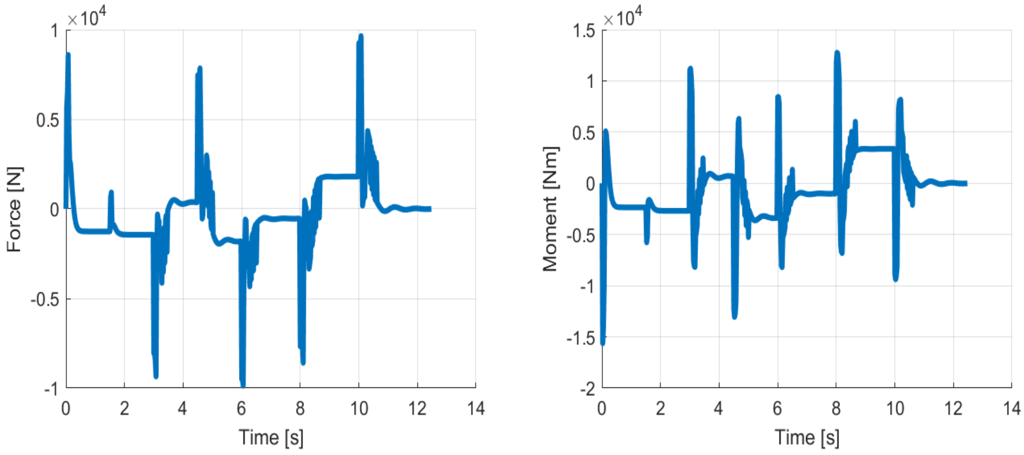


Figure 6.25 Force and Moment Response under different IC's acceleration command scenario with adaptive weight algorithm

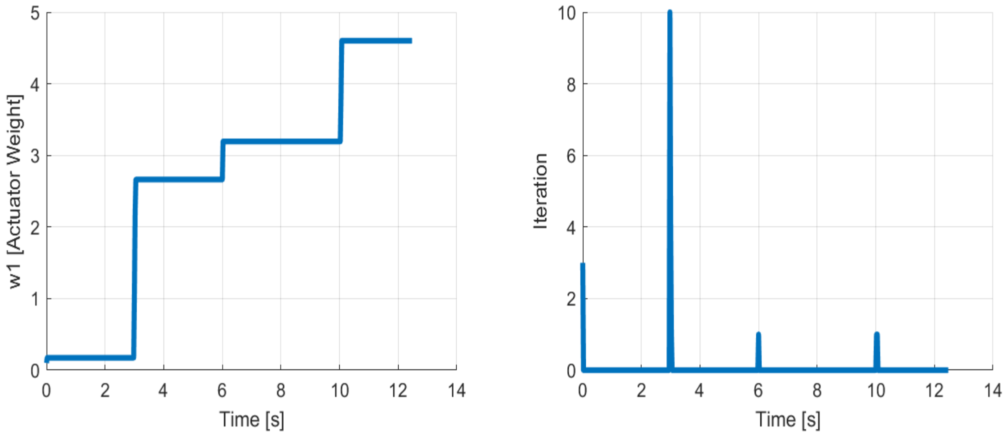


Figure 6.26 Weight and Iteration Response under different IC's acceleration command scenario with adaptive weight algorithm

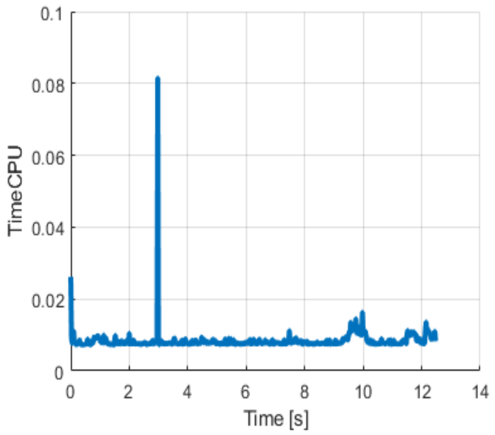


Figure 6.27 CPU Time Response under different IC's acceleration command scenario with adaptive weight algorithm

6.6 Robustness Analysis

In the previous sections, the process leading to an optimal solution within the defined linear space and specified constraint budgets was discussed. In this section, test scenarios will be created considering thrust uncertainty in lateral thrusters to assess the robustness of the designed autopilot when considering real-world applications, where the missile might exit the defined space. The interaction between lateral thrusters and the air stream in missile systems is a critical aspect of aerodynamic control. Lateral thrusters, often small solid propellant mounted on the sides of a missile, generate thrust perpendicular to the missile's main axis to adjust its trajectory. When activated, these thrusters expel high-speed gas jets, which interact with the surrounding air stream. This interaction creates complex aerodynamic forces and moments, which can alter the missile's flight path. The expelled gas jets disrupt the smooth flow of air around the missile, potentially causing turbulence and affecting the overall stability and control. It must carefully model and predict these interactions to ensure effective and precise maneuverability, maintaining the missile's stability and accuracy in reaching its target. When RCS thrusters are used, the jet plume interacts nonlinearly with the freestream flow, creating high and low-pressure areas around the nozzle shown in Figure 6.28. This interaction amplifies the thrust force, allowing for smaller reaction jets and reduced propellant use. However, if this distorter effect is not considered, it can potentially destabilizing the missile. Conversely, if the jet flow diffuse the free-stream, high-pressure area is diminished, reducing thrust and possibly decreasing loop gain, which also impacts stability. A critical factor in this interaction is the jet penetration height, which varies with flight conditions and affects whether the thrust is amplified or attenuated. These nonlinear effects will also cause an imbalance between the force and moment , complicating the control dynamics [9]. As mentioned in the article[3, 9], this interaction distorts the thrust more or less than expected depending on the missile's speed, flow direction, and orientation of the missile.

Since the identified effects are considered to play a significant role on control, they have been addressed in the robustness analysis. Disturbance is implemented to the system after generating control input by MPC and block diagram of this structure is shown Figure 6.29. Because the modeling of this effect is outside the scope of this thesis, a space will be defined that reflects the process, and the thrust-disturbance effect will be projected onto the system as a result of a statistical distribution. MATLAB ”randn” function is used to generate thrust amplification factor as a disturbance and the factor is multiplied to the control input of thrust produced by MPC.

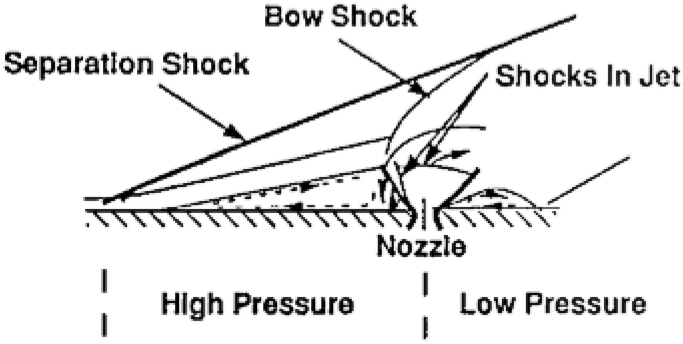


Figure 6.28 Jet interaction

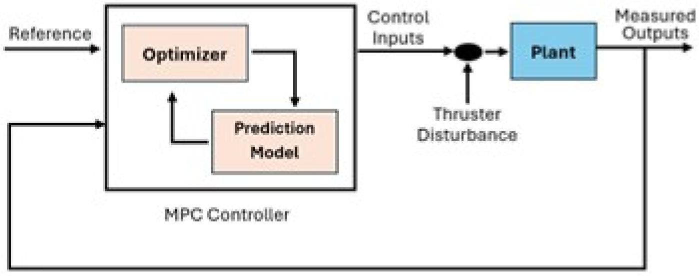


Figure 6.29 MPC with disturbance Blok Diagram

Initially, disturbance range is defined as 3 standart deviation(σ) is equal to % 30 of normal thrust and shown in Figure 6.30

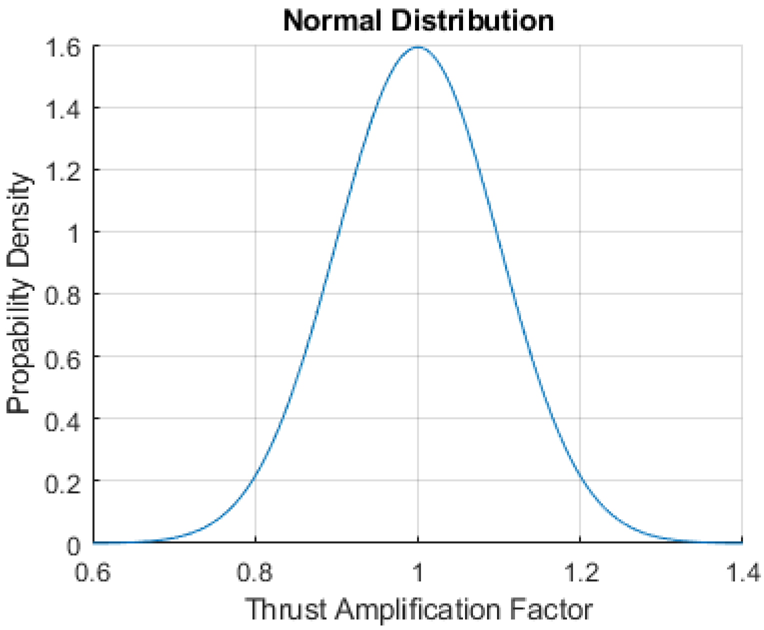


Figure 6.30 Normal Distribution of Thrust Amplification Factor

Performance comparison system with disturbance and disturbance-off is shown in Figure 6.31-6.33. Figure 6.33 shows that first firing of thruster is expected to -8269N but it come up with -8059N. There is an %2 reduction occurs for first ignition. For second ignition is expected -4135N but it take place as -3938N which approximately equals to %5 reduction. System can reach a solution using 2 ignition and, according to Figure 6.31, track the acceleration command a bit more oscillatory than disturbance-off case.

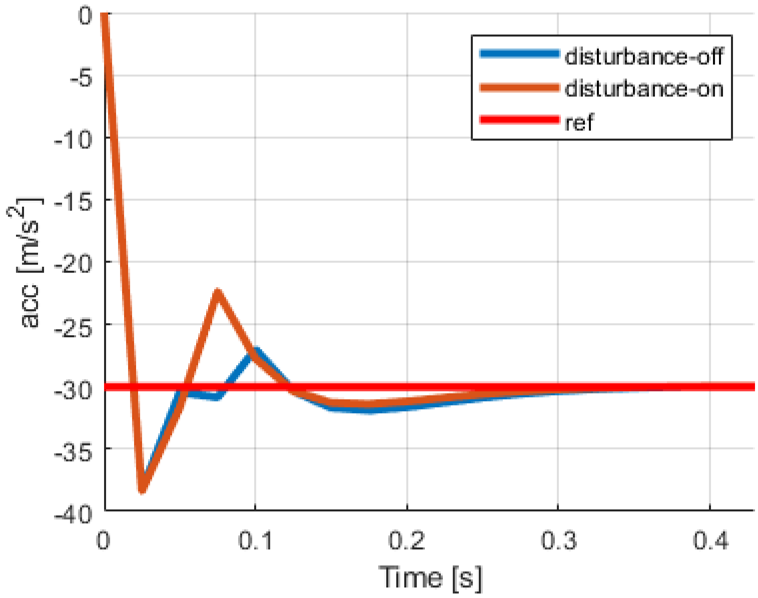


Figure 6.31 Effect of Thrust amplification factor to acceleration tracking performance

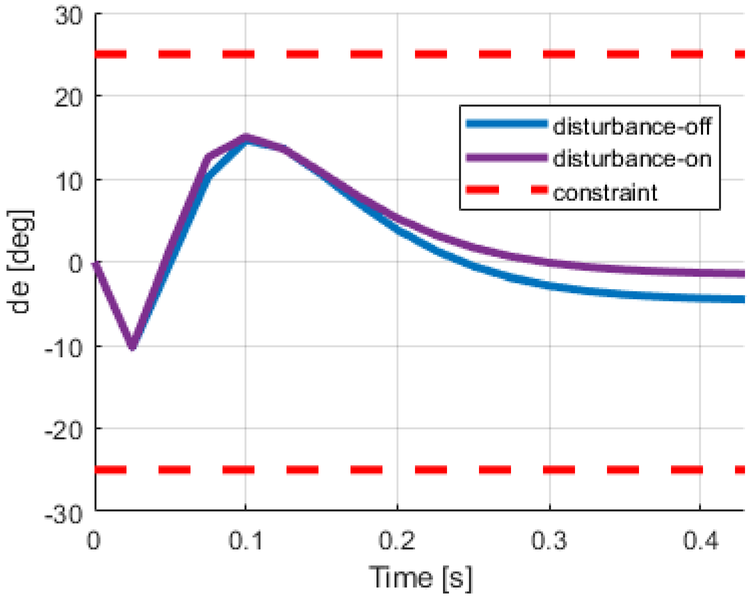


Figure 6.32 Effect of Thrust amplification factor to δ_e performance

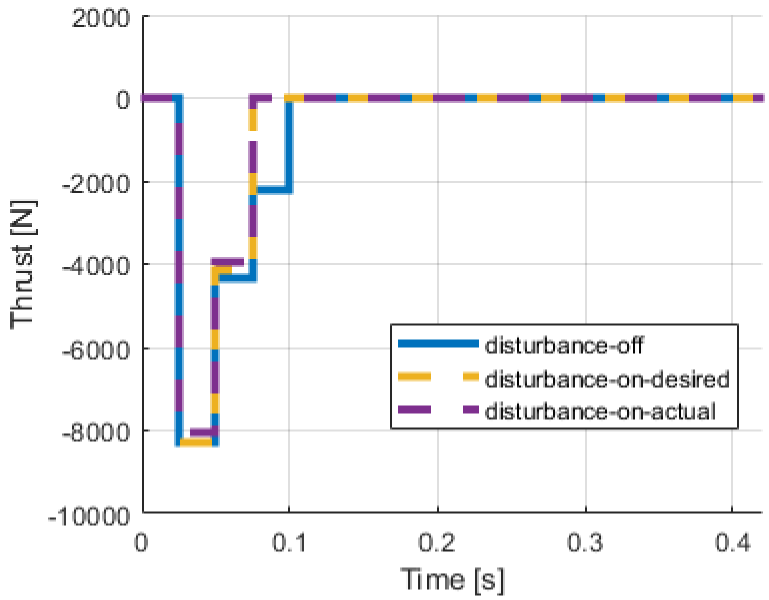


Figure 6.33 Effect of Thrust amplification factor to thruster performance

By increasing the number of runs, we can better analyze the behavior of the disturbance effect, so an analysis with 100 runs has been conducted and result is shown Figure 6.34.

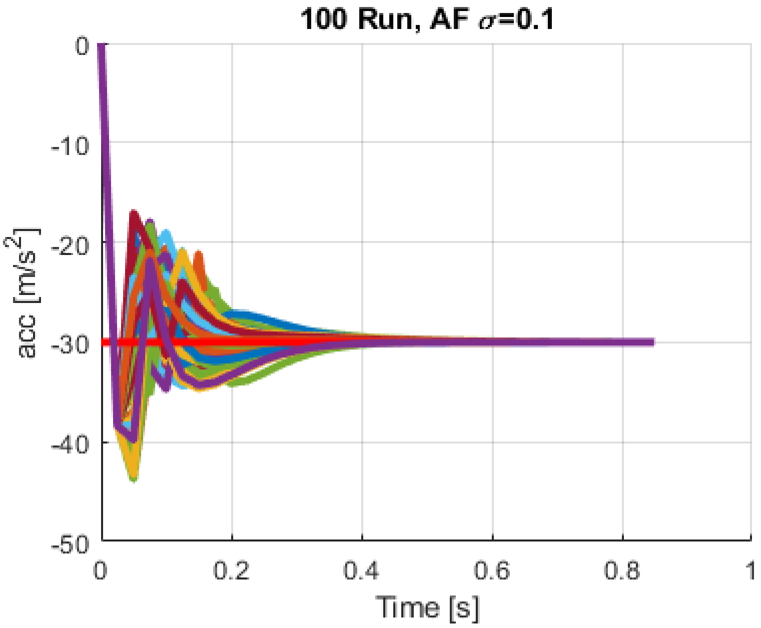


Figure 6.34 Multiple Run solution of Thrust Amplification effect

To see the error limit of the system, the analysis was repeated for different standard deviation values. The analysis was again carried out using the values generated from normal distribution curves, with 100 runs for each distribution, and the percentage performance was shown in Figure 6.35. For a run to be considered successful, it needs to set itself to the reference acceleration at steady state. When examining the results, it is seen that performance starts to decrease when the standard deviation is greater than 0.1 and that more than half of the results are unstable after 0.2. In future studies, the autopilot design can be improved by better defining the thrust uncertainty. When examining the Figure 6.34, it is seen that there are too many oscillations even if the system settles at the reference input in a stationary state. The stability criterion alone may not be sufficient considering operational conditions, but within the scope of this study, it is thought to be a good starting point for determining autopilot design and subsystem requirements.

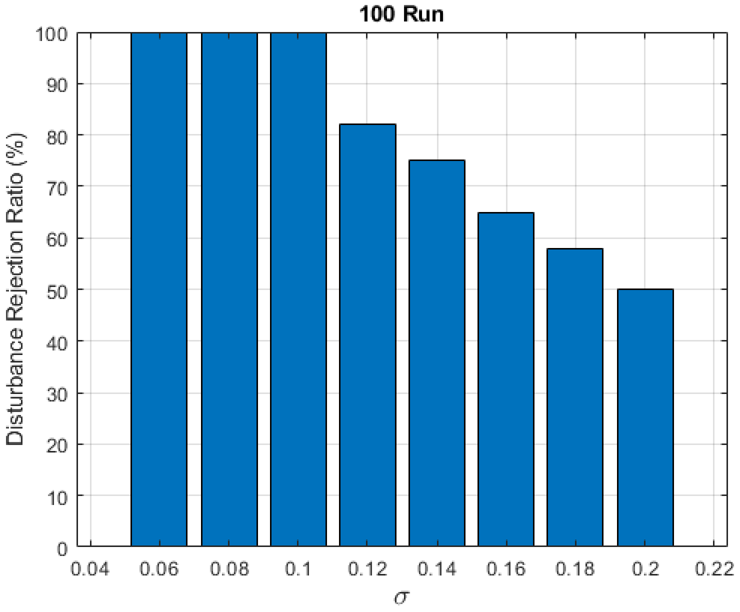


Figure 6.35 Multiple Run Disturbance Rejection characteristic of Thrust Amplification effect for different σ

6.7 Dual vs Aero Only Autopilot Comparison

After completing the ”dual control” autopilot analyses, comparing it with a traditional fin-controlled autopilot to understand the contribution of the lateral thrusters to system performance will provide a good output for evaluation, and this topic will be addressed in this section. The MPC method will be used again for a traditional fin-controlled controller, and no constraints will be defined. The MPC design method without constraints has been addressed in Section 6.3, and the same approach will be used here. As is known, the solutions up to this point have been handled as multi-input single-output(MISO). However, the solution here will be reduced to single-input single-output(SISO). A new system matrices can be reduced from equation 6.2.

$$\begin{bmatrix} \dot{a}_z \\ \dot{q} \\ \dot{\delta} \\ \ddot{\delta} \end{bmatrix} = \begin{bmatrix} -0.19 & -172.72 & 0 & -36.27 & -0.82 \\ 0.09 & -0.15 & -34.47 & 0 & -0.0045 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & -24674 & -188.49 & 0 \end{bmatrix} \begin{bmatrix} a_z \\ q \\ \delta \\ \dot{\delta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \omega_n^2 \end{bmatrix} \begin{bmatrix} \delta_{com} \end{bmatrix} \tag{6.33}$$

$$a_z = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} a_z \\ q \\ \delta \\ \dot{\delta} \end{bmatrix} \tag{6.34}$$

Similar scanning space in 6.3 can be defined here:

$$5 \leq N_p \leq 30$$

$$1 \leq N_c \leq 5$$

$$10 \leq w_1 \leq 0.0001$$

According to steps 6.3 design step solution of tunable parameters is obtained:

$$N_p = 10$$

$$N_c = 2$$

$$w_1 = 0.1$$

Comparison the performance of dual control and aero-only control autopilots in terms of acceleration, angle of attack, force, and moment responses over time are shown in Figure 6.36-6.37. Figure 6.36 shows that the dual control system reaches the reference acceleration 3 times faster solution and with less overshoot compared to the aero-only system, indicating a quicker and more stable response. The graph also indicates that the angle of attack for the dual control system stabilizes faster than the aero-only system, demonstrating better control authority. Figure 3.30 graph depicts the force response, where the dual control system initially produces a higher peak force but stabilizes quicker than the aero-only system, suggesting more effective force application and damping. The graph also shows the moment response, where the dual control system also exhibits a higher initial peak and faster stabilization compared to the aero-only system, indicating superior moment control. Overall, the dual control system outperforms the aero-only system in achieving quicker response times and better adherence to desired performance parameters across all metrics.

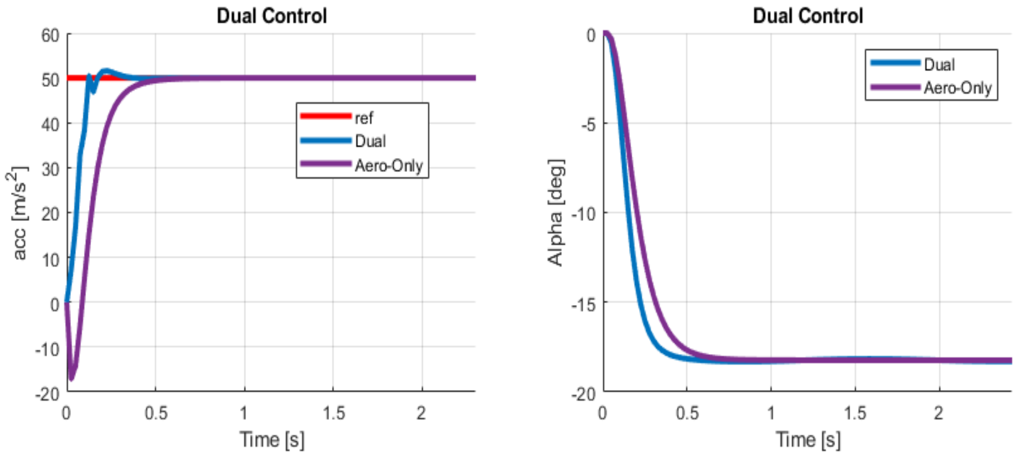


Figure 6.36 Acceleration and Angle of Attack Response Comparison for Dual and Aero-Only Autopilots

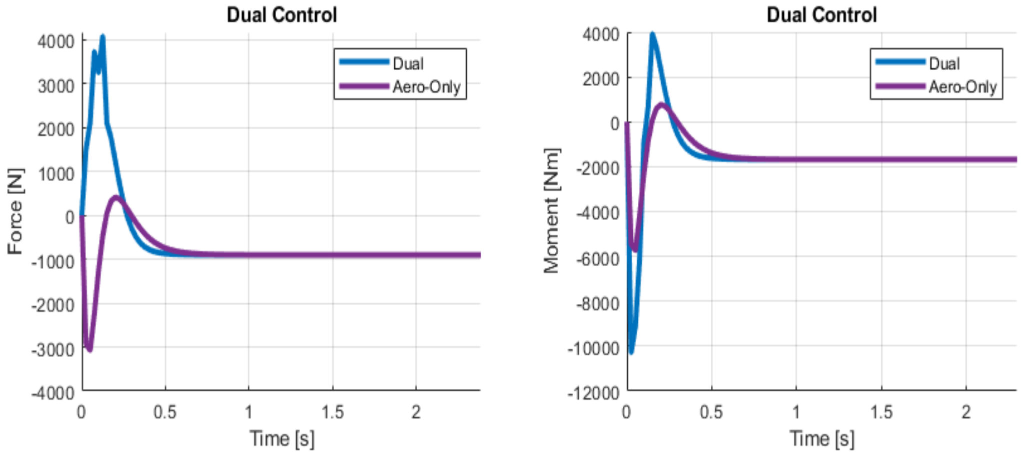


Figure 6.37 Total Force and Moment Response Comparison for Dual and Aero-Only Autopilots

6.8 Guidance Performance

To evaluate autopilot performance, including guidance in the loop and examining it will provide a good perspective. In this part, a performance comparison will also be made with a traditional aerofin-controlled missile and dual-controlled missile.

A ballistic missile has been selected as the target, and a scenario has been constructed to initiate the engagement from the terminal phase. It is assumed that predicted interception point (PIP) is provided in this scenario. Interception point guidance is crucial in anti-ballistic missile (ABM) systems, ensuring the accurate neutralization of incoming threats before they reach their targets. This guidance minimizes collateral damage by directing interceptor missiles to precise points for effective destruction of ballistic missiles. It optimizes interceptor trajectories, conserving fuel and energy, and enhances the ability to counter maneuverable warheads and decoys. Real-time adaptation and integration with broader defense systems improve overall effectiveness, allowing coordinated responses to high-speed threats. The guidance process involves target tracking and prediction, trajectory calculation, midcourse adjustments, and terminal guidance, each phase crucial for a successful interception. Despite challenges such as high-speed engagements, evasive tactics, and technological sophistication, interception point guidance remains essential for national and global security, protecting against the devastating potential of ballistic missile attacks [25, 26].

Since the autopilot design is done in a single channel, a 2D engagement kinematics has been created. Here, to create a realistic scenario, just before entering the terminal phase, the missile go through to a point of interception point provided radar measurement in mid-course phase, and at the terminal entry, the target position has been shifted by a certain amount as an mid-course guidance error. Pure Proportional Navigation(PPN) is selected as a guidance method. PPN is a fundamental missile guidance law that aims to ensure effective target interception by maintaining a proportional response to the rate of change in the line-of-sight (LOS) angle between the missile and the target. The method propose that generation of missile acceleration command that must be perpendicular to missile velocity vector, thereby simplifying control strategy without changing missile velocity. The lateral acceleration command in PPN is given by:

$$a_m = N \cdot V_m \cdot \dot{\lambda} \tag{6.35}$$

where:

- a_m is commanded acceleration.
- N is navigation constant, which dictates the responsiveness of the guidance system and typically ranges between 3 and 5.
- V_m is the velocity vector and assumed to align and fix missile body x axis.
- $\dot{\lambda}$ is the rate of change of the LOS angle, measured in radians per second.

This guidance law exploits the geometry of motion by ensuring that the acceleration vector, which is always orthogonal to the missile's velocity, modifies the missile's trajectory to constantly adjust the LOS rate. The missile thereby maneuvers directly toward an intercept course with the target based on continuous adjustment of its flight path relative to the motion of the target and shown in Figure 6.38. This strategy assumes linear target trajectories and does not account for potential evasive maneuvers by the target or measurement error which might come from radar or seeker. In this study, it is assumed that the predicted interception point(PIP) point where the missile will meet the target is known, and the missile is designed to guide itself towards this point. The analysis begins at the moment of transition from the mid-course guidance phase to the terminal phase. It is supposed that at the final part of the mid-course, the missile go through to the PIP_{mc} point. After the phase transition, it is assumed that the seeker head measurements are error-free, and a new PIP_t point is defined based on the mid-course measurement errors. The missile is then expected to guide itself to this new PIP_t point to successfully meet the target. Engagement scenario is shown in Figure 6.39 and target position has been shifted by mid-course error (r_{mce}) in terminal phase handover as mentioned.

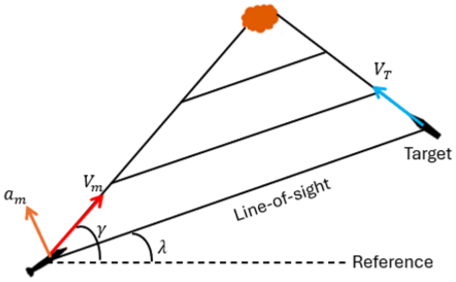


Figure 6.38 PPN Engagement Kinematics

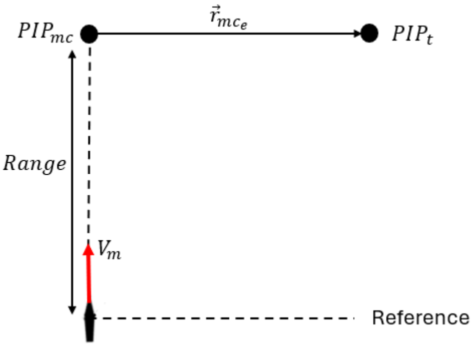


Figure 6.39 Midcourse to Terminal Phase Handover Engagement Kinematics

Missile and target engagement loop including guidance and autopilot is shown Figure 6.40. The diagram represents a missile guidance system, illustrating the flow of information between its key components. The target dynamics block outputs the target velocity vector, which, along with the missile velocity vector, is used by the geometry block to calculate the Line-of-Sight (LOS) to the target. The seeker detects and tracks the target, providing the LOS rate to the guidance computer. The guidance computer processes this information to generate guidance commands which is required acceleration, which are sent to the autopilot and missile dynamics block. This block translates the commands into control inputs, adjusting the missile's trajectory to stay on course towards the target. The missile velocity vector is continuously fed back into the system to update the LOS calculation, ensuring accurate and adaptive guidance.

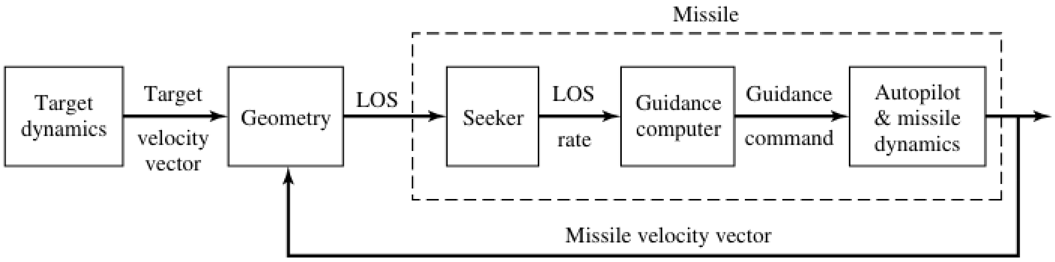


Figure 6.40 Guidance Loop Block Diagram

To describe the kinematics between missile and target, relative position and velocity between the missile and the target are calculated as:

$$\mathbf{r}_{\text{rel}} = \mathbf{r}_T - \mathbf{r}_M \tag{6.36}$$

$$\mathbf{v}_{\text{rel}} = \mathbf{v}_T - \mathbf{v}_M \tag{6.37}$$

LOS angle is defined as:

$$\lambda = \tan^{-2}\left(\frac{r_{\text{rel},y}}{r_{\text{rel},x}}\right) \tag{6.38}$$

where $r_{\text{rel},x}$, $r_{\text{rel}y}$ represent the x , y components of the relative position vector $\mathbf{r}_{\text{rel}}$.

The rate of change of the LOS angle, or LOS rate, is calculated as:

$$\dot{\lambda} = \frac{v_{\text{rel},y} \cdot r_{\text{rel},x} - v_{\text{rel},x} \cdot r_{\text{rel},y}}{r^2_{\text{rel},x} + r^2_{\text{rel},y}} \tag{6.39}$$

where V_m is the magnitude of the missile's velocity. The missile's velocity and position are updated using the acceleration command:

$$\mathbf{v}_M = \mathbf{v}_M + \mathbf{a}_M \cdot \Delta t \tag{6.40}$$

$$\mathbf{r}_M = \mathbf{r}_M + \mathbf{v}_M \cdot \Delta t \tag{6.41}$$

Note that the target's position and velocity are not required updating because of assuming it is a fixed point target.

In order to provide a better comparison between a traditionally only-fin-controlled missile and a 'dual control' missile, and to define the acceleration requirement throughout the guidance, the acceleration effort parameter is intended to be defined and is expressed as follows:

$$J = \int |\vec{a}_m|^2 \, dt \left[\left(\frac{m}{s^2} \right)^2 \cdot s \right] \tag{6.42}$$

Four different scenarios have been defined for the analysis and variable parameters are shown in the Table 6.3. Since we designed the guidance scenario to the PIP point based on a ballistic target, the seeker head lock-on range(LOR) should roughly be twice the range, considering that the target and missile speeds are the same. Notice that, mid-course error parameters defined as a vector because of 2D kinematics and only adapt x axis of the position vector of the target. The initial conditions and fixed parameters for the simulation are set as follows:

- Navigation constant: $N = 5$
- Time step: $\Delta t = 0.025$ seconds (Note:selected same as autopilot time step)
- Initial position of missile: $\mathbf{r}_M = [0 \;\; 0]$
- Initial velocity magnitude of missile: $|\mathbf{v}_M| = 885 \, m/s$ (Note:Autopilot design velocity at 20km 3 Mach)
- Initial flight path angles of missile: γ_M = $90 * \pi/180$
- Initial velocity of missile : $\mathbf{v}_M = |V_m| * [cos(\gamma_M) \;\; sin(\gamma_M)]$
- Initial position of target: $\mathbf{r}_T = [Range \;\; r_{mce}]$

Scenario	**Range(km)**	r_{mce}
1	5	[500 0]
2	5	[250 0]
3	2.5	[500 0]
4	2.5	[250 0]

Table 6.3 Scenario Tunable Parameters

The results have been visualized and shown in the figure for scenario 1 only and rest of scenario results are tabulated at Table 6.4 according to miss distance and control effort. In Figure 6.41, the missile's intersection point trajectory is shown for both autopilot systems, and it can be seen that in the blended autopilot structure, the missile directs itself towards the target earlier; in other words, the missile's velocity vector is turned towards the target faster. As seen in Figure 6.42, the acceleration command response profile is smoother in the ”dual” autopilot, and Figure 6.43 shows that the acceleration effort remains at a lower level for this autopilot. This indicates that the missile can reach the target with less energy consumption. This can be interpreted to mean that in a situation where the missile's speed change is modeled, the speed can be preserved more. For a correct comparison of the analysis, the guidance acceleration command in the traditional autopilot structure is constrained so as not to exceed the fin limit, as shown in Figure 6.44. When this figure is examined, it can be seen that both the dual and traditional autopilots are constrained by fin limits. Additionally, in the blended structure, the side thrusters are activated at the beginning of the guidance, and since the acceleration error value decreases afterwards, no further firing is needed for the rest of the trajectory. Overall, it is clear that the dual autopilot structure brings an improvement in guidance compared to a traditional structure.

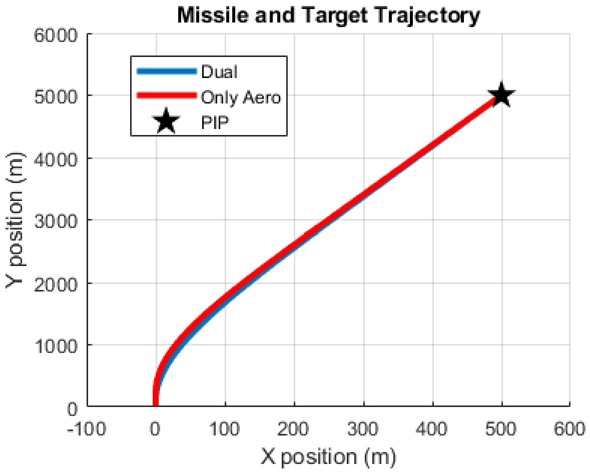


Figure 6.41 Missile Trajectory at Scenario 1

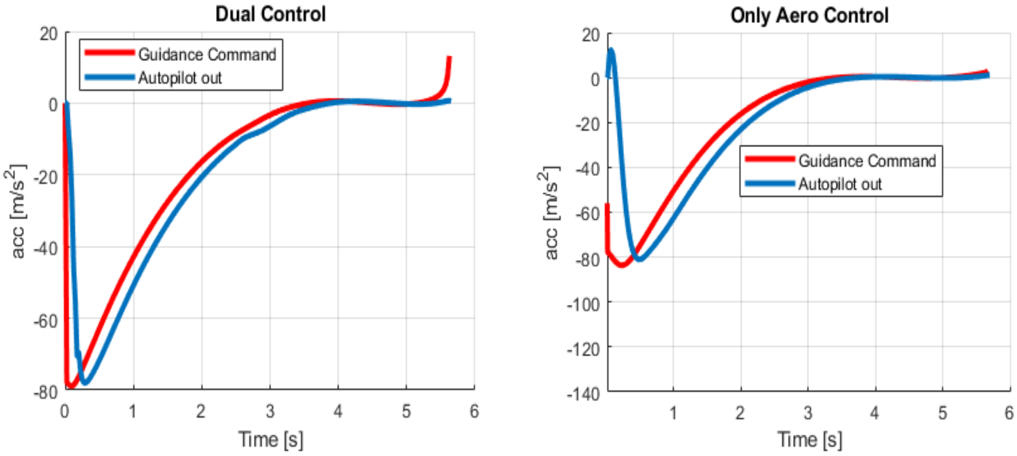


Figure 6.42 Guidance acceleration command and autopilot response at Scenario 1

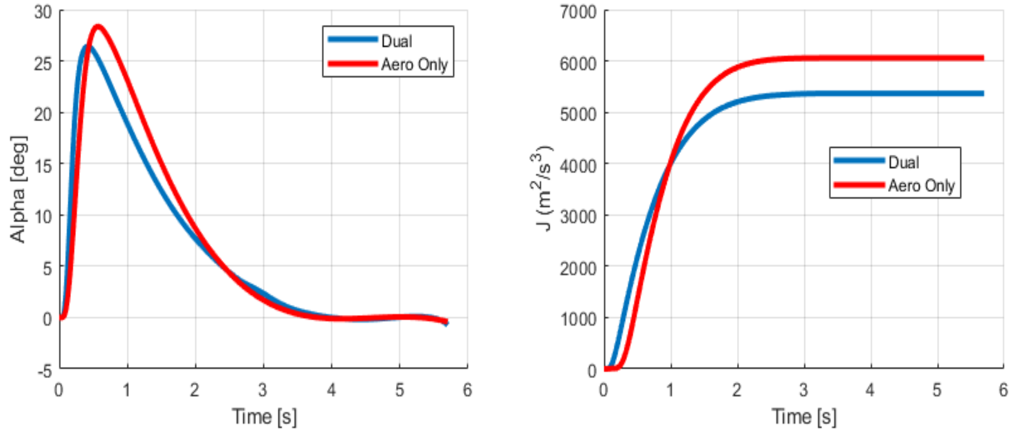


Figure 6.43 Angle of attack and Acceleration Effort at Scenario 1

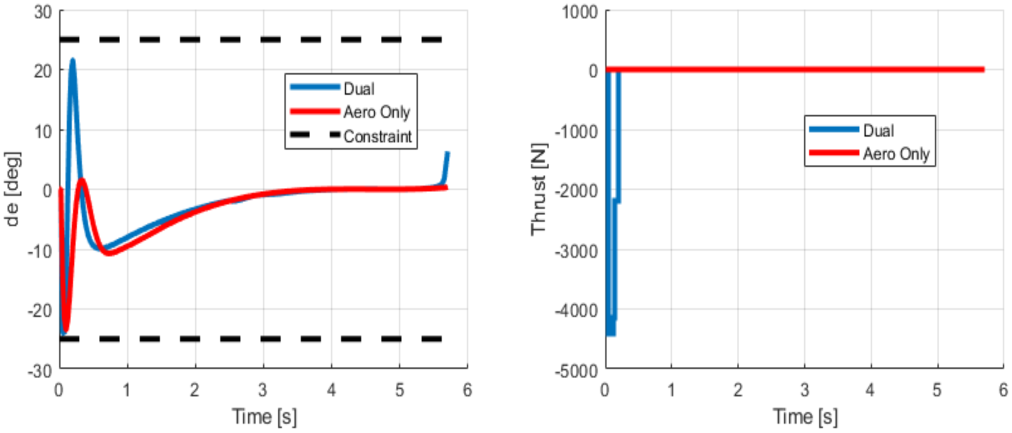


Figure 6.44 Actuators Performance at Scenario 1

Scenario	Otopilot	Miss Distance [m]	J [m^2/s^3]
1	Dual	5.11	5.3×10^3
1	Aero-Only	5.9	6.57×10^3
2	Dual	6.61	1.3×10^3
2	Aero-Only	7.2	1.5×10^3
3	Dual	10.5	4.7×10^4
3	Aero-Only	12.5	5.5×10^4
4	Dual	16.4	11.2×10^4
4	Aero-Only	17.2	17.9×10^4

Table 6.4 Dual vs Aero-Only Performance Comparison

7 CONCLUSION

This thesis has explored the design and implementation of a Mixed Integer Quadratic Programming (MIQP) autopilot based on Model Predictive Control (MPC) for air defense missiles. The research addressed the challenges of enhancing missile maneuverability and control in modern combat scenarios, especially against highly maneuverable and high-speed ballistic targets. By integrating secondary actuators such as lateral thrusters, the study demonstrated the improved performance of the missile under low dynamic pressure conditions, where traditional aerodynamic surfaces might be less effective. The fact that the actuators have different characteristics, that the side thrusters operate in an on-off manner, and that the fin can continuously produce input between predetermined position limits make autopilot design challenging compared to traditional methods. In addition to these, given the advanced target capabilities and some engagement scenarios, the missile is expected to be very agile. The study aimed to reach a conclusion by considering the factors mentioned in the thesis work. Moreover, thrust uncertainties resulting from the interactions of side thrusters with the flow over the missile have been examined. A ballistic target engagement scenario was created, and an evaluation was made based on the guidance performance with the autopilot in the loop. Additionally, this study examined how much lateral thrusters improved the system in terms of autopilot and guidance compared to traditional aerodynamically controlled missile within the scope of this study.

At the beginning of the study, the 6-DOF equations of motion were derived, and the coordinate frames required to define these equations were identified. The existing forces and moments on the system were defined to provide a general overview of the system. Since key contributions of this work include the development of a blended autopilot structure that effectively tracks reference acceleration commands in a linear domain, accommodating the on-off behavior of lateral thrusters and the angle constraints of aerodynamic surfaces. The inclusion of all actuator dynamics and constraints as state variables provided a comprehensive system model, which was tested under different scenarios. Modeling approach for the usage of side thrusters in the system is implemented and approximated to a first-order transfer function. The second-order transfer function approximation, commonly used in the literature for fin modeling, has also been applied in this study.

The design stages and the details of the MPC method were thoroughly discussed in this study. To understand the system and comprehend the method, the study initially addressed the situation where there were no constraints on the actuators. This stage, the impact of the adjustable parameters of MPC, including ”prediction horizon,” ”control horizon,” and ”sample time,” on system performance was examined. Firstly, ” sample time” was selected according to the ”rule of thumbs” established in the literature. Then, the effects of the N_p and N_c values were investigated. Shorter value of N_p and N_c enhances aggressive response through quicker predictions and enables faster computation due to reduced complexity. However, short N_c and N_p selection brings potential overshoot and oscillations. On the other hand, longer N_p and N_c considers long-term effects, improving prediction accuracy but leads to slower execution due to higher computational complexity.

Moreover, it offers better stability for systems requiring long-term foresight and minimizes overshoot and oscillations. Typically, it is recommended that N_p is chosen 10-30 and N_p is selected 10 in this study. N_c is often selected as a fraction of the prediction horizon, usually between $\%10$ and $\%50$ of N_p and $\%20$ percent of N_p is selected this study. These parameters should be tuned based on specific system characteristics and performance requirements. Simulation and real-world testing are crucial for finding the optimal balance. Subsequently, a solution was provided by incorporating lateral thruster constraints, which significantly limit the system's solution. The MIQP method was employed due to its suitability for handling the integer nature of the lateral thruster's on-off behavior, while MPC was utilized to predict future system behavior and generate optimal control inputs. In the side-thruster modeling, multiple firing options have been optionally prepared for use in the system. Therefore, at each simulation step, a decision tree is created from these options, and the control sequence that will minimize the defined cost function is selected. Increasing the N_c value or optional control options significantly increases the computational load at this stage. Depending on the processor speed, this value can be increased, or solutions that will narrow down the search space can be explored instead of scanning the entire space to reach the solution faster.

Another design stage is that fin constraint is included in autopilot design. It was observed that the test scenarios, where both actuator constraints were active, led to system instability. For this reason, the novel approach of incorporating an adaptive weight estimation algorithm within the MPC framework ensured that actuator constraints, particularly for aerodynamic fins, were dynamically adjusted to maintain system stability and performance. This method allowed for real-time adjustments to control inputs, enhancing the missile's ability to respond to rapid changes in flight dynamics. The method is a control mechanism that simply opens a narrow prediction window at each time step to monitor the fin command limit exceedance within this window. If there is a limit exceedance, it repeats the solution by reducing the fin weight in the MPC solution; if not, it continues with the previous weight. It should be noted that this window is derived from the main solution. Therefore, as N_c and the number of optional controls increase, this part will also slow down proportionally, necessitating extra caution.

Study evaluates the designed autopilot's robustness under real-world conditions by incorporating thrust uncertainty in lateral thrusters. The interaction between lateral thrusters and the air stream is critical, as the expelled gas jets can create complex aerodynamic forces, potentially affecting the missile's stability and control. To address this, the robustness analysis includes modeling these interactions and implementing disturbances in the control system using Model Predictive Control (MPC). A statistical approach is employed to simulate thrust amplification factors, which are introduced as disturbances. The analysis involves generating a disturbance range based on standard deviations of normal thrust and comparing system performance with and without these disturbances. Performance comparisons and multiple run simulations highlight how the system handles variability, ensuring reliable operation despite uncertainties. The robustness analysis underscores the importance of accurately modeling and mitigating the effects of jet interactions to maintain effective missile control in dynamic environments.

Finally, the evaluation focuses on comparing the guidance performance of a traditional aerofin-controlled missile with a dual-controlled missile. A ballistic missile is selected as the target, and an engagement scenario is constructed from the terminal phase. The scenario assumes that the predicted interception point (PIP) is provided, highlighting the importance of interception point guidance in anti-ballistic missile (ABM) systems. This guidance ensures the accurate neutralization of threats, minimizes collateral damage, and optimizes interceptor trajectories. The study involves a 2D engagement kinematics model, where the missile transitions from mid-course to terminal phase guidance using Pure Proportional Navigation (PPN). The guidance performance is analyzed by examining the missile's trajectory, acceleration command response, angle of attack, and actuator performance under different scenarios. The results demonstrate that the dual-controlled missile achieves better guidance performance with smoother acceleration profiles and lower energy consumption compared to the traditional aerofin-controlled missile, indicating improved efficiency and effectiveness in intercepting ballistic targets.

Simulation results validated the proposed MIQP-MPC approach, showing significant performance improvements over conventional aero-surface control methods. The robustness of the designed autopilot was also demonstrated through various scenarios, including the effects of thrust uncertainties and interactions. A comparative analysis with traditional fin-controlled autopilots further highlighted the advantages of the integrated lateral thruster system in a ballistic missile target guidance scenario.

In conclusion, this research has successfully developed a novel, efficient autopilot design that enhances missile performance and robustness. Future work could focus on extending the approach to more complex, multi-channel control systems and exploring the integration of adaptive control methods to further improve performance in dynamic and unpredictable environments.

REFERENCES

[1] Tayfun Çimen. A generic approach to missile autopilot design using state-dependent nonlinear control. *IFAC Proceedings Volumes*, 44(1):9587–9600, **2011**. ISSN 1474-6670. doi:https://doi.org/10.3182/20110828-6-IT-1002.03744. 18th IFAC World Congress.

[2] Ahmed Awad and Haoping Wang. Roll-pitch-yaw autopilot design for nonlinear time-varying missile using partial state observer based global fast terminal sliding mode control. *Chinese Journal of Aeronautics*, 29(5):1302–1312, **2016**. ISSN 1000-9361. doi:https://doi.org/10.1016/j.cja.2016.04.020.

[3] Baoqing Yang and Yuyu Zhao. Autopilot design method for the blended missile based on model predictive control. *International Journal of Aerospace Engineering*, 2015:1–13, **2015**. doi:10.1155/2015/718036.

[4] Dimche Kostadinov and Davide Scaramuzza. Online weight-adaptive nonlinear model predictive control. *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1180–1185, **2020**.

[5] Rui Hirokawa, Koichi Sato, and Shunji Manabe. Autopilot design for a missile with reaction-jet using coefficient diagram method. **2001**. doi:10.2514/6.2001-4162.

[6] R Chadwick, William. Augmentation of high-altitude maneuver performance of a tail-controlled missile using lateral thrust. Technical report, NAVAL SURFACE WARFARE CENTER DAHLGREN DIV VA, **1995 Jan 01**.

[7] Xing Lidan, Zhang Ke'nan, Chen Wanchun, and Yin Xingliang. Optimal control and output feedback considerations for missile with blended aero-fin and lateral impulsive thrust. *Chinese Journal of Aeronautics*, 23(4):401–408, **2010**. ISSN 1000-9361. doi:https://doi.org/10.1016/S1000-9361(09)60234-X.

[8] Liu Zhong and Jia Xiaohong. Novel backstepping design for blended aero and reaction-jet missile autopilot. *Journal of Systems Engineering and Electronics*, 19(1):148–153, **2008**. ISSN 1004-4132. doi:https://doi.org/10.1016/S1004-4132(08)60060-2.

[9] Kevin A. Wise and David J. B. roy. Agile missile dynamics and control. *Journal of Guidance, Control, and Dynamics*, 21(3):441–449, **1998**. doi:10.2514/2.4256.

[10] Yong Seok Choi, Ho Chul Lee, and Jae Weon Choi. Robust mixed control with aerosurface and side thruster using ltv control technique. *IFAC Proceedings Volumes*, 37(12):75–80, **2004**. ISSN

1474-6670. doi:https://doi.org/10.1016/S1474-6670(17)31446-5. IFAC Workshop on Adaptation and Learning in Control and Signal Processing (ALCOSP 04) and IFAC Workshop on Periodic Control Systems (PSYCO 04), Yokohama, Japan, 30 August - 1 September, 2004.

[11] Fleeman Eugene. *Tactical Missile Design*. American Institute of Aeronautics and Astronautics, Inc1801 Alexander Bell Drive, Reston, VA20191-4344,, **2001**.

[12] Raziye Tekin, Özgür Atesoglu, and Kemal Leblebicioglu. *Modeling and Vertical Launch Analysis of an Aero- and Thrust Vector Controlled Surface to Air Missile*. doi:10.2514/6.2010-7639.

[13] `https://www.lockheedmartin.com/en-us/products/thaad.html`.

[14] Yong Guo, Jin hua Guo, Xiang Liu, Ai jun Li, and Chang qing Wang. Finite-time blended control for air-to-air missile with lateral thrusters and aerodynamic surfaces. *Aerospace Science and Technology*, 97:105638, **2020**. ISSN 1270-9638. doi:https://doi.org/10.1016/j.ast.2019.105638.

[15] Manfred Morari Francesco Borrelli, Alberto Bemporad. *Predictive Control for Linear and Hybrid Systems*. Cambridge University Press, **2017**.

[16] Liuping Wang. *Model Predictive Control System Design and Implementation Using MATLAB*. Springer-Verlag London Limited, **2009**.

[17] D.Q. Mayne, J.B. Rawlings, C.V. Rao, and P.O.M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, **2000**. ISSN 0005-1098. doi:https://doi.org/10.1016/S0005-1098(99)00214-9.

[18] Yonghua Fan, Xin Li, Jun Yang, and Yuzhuo Zhang. Design of autopilot for aerodynamic/reaction-jet multiple control missile using variable structure control. In *2008 27th Chinese Control Conference*, pages 642–645. **2008**. doi:10.1109/CHICC.2008.4604993.

[19] Di Zhou and Chuntao Shao. Dynamics and autopilot design for endoatmospheric interceptors with dual control systems. *Aerospace Science and Technology - AEROSP SCI TECHNOL*, 13:291–300, **2009**. doi:10.1016/j.ast.2009.05.004.

[20] Michael B. McFarland and Anthony J. Calise. Neural networks and adaptive nonlinear control of agile antiair missiles. *Journal of Guidance, Control, and Dynamics*, 23(3):547–553, **2000**. doi:10.2514/2.4563.

[21] D. Ridgely, Yung Lee, and Todd Fanciullo. *Dual Aero/Propulsive Missile Control - Optimal Control and Control Allocation*. doi:10.2514/6.2006-6570.

[22] D. Ridgely, David Drake, Louis Triplett, and Cory Geise. *Dynamic Control Allocation of a Missile with Tails and Reaction Jets*. doi:10.2514/6.2007-6671.

[23] Liuping Wang. Model predictive control system design and implementation using matlab. **2009**.

[24] Alberto Bemporad, Manfred Morari, and N. Lawrence Ricker. Model predictive control toolbox™ getting started guide. **2015**.

[25] Sungjun Ann, Seokwon Lee, and Youdan Kim. Intercept point prediction for midcourse guidance of anti-ballistic missile. In *2018 26th Mediterranean Conference on Control and Automation (MED)*, pages 1–9. **2018**. doi:10.1109/MED.2018.8442989.

[26] Donglin Tan and Junfeng Wang. Recognition of warheads by range-profile matching with automatic threshold. *EURASIP Journal on Applied Signal Processing*, 2024(1):28, **2024**. doi:10.1186/s13634-024-01125-3.

CURRICULUM VITAE

Credentials

Name,Surname : MEHMET ALUÇ
Place of Birth : Ankara / TURKEY
Marital Status : Married
E-mail : aluccmehmet@gmail.com
Address : Department of Mechanical Engineering, Hacettepe University
Ankara / TURKEY

Education

B.Sc. : Mechanical Engineering, Hacettepe University, Ankara / TURKEY

Foreign Languages : English

Work Experience : Flight Mechanics Engineer, Roketsan

Areas of Experiences :

-

Projects and Budgets :

-