DEVELOPMENT AND COMPARISON OF AUTOPILOT AND GUIDANCE ALGORITHMS FOR MISSILES

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ÇAĞDAŞ EVCİMEN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

AUGUST 2007

Approval of the thesis:

DEVELOPMENT AND COMPARISON OF AUTOPILOT AND GUIDANCE ALGORITHMS FOR MISSILES

submitted by **ÇAĞDAŞ EVCİMEN** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen _____
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. İsmet Erkmen _____
Head of Department, **Electrical and Electronics Engineering**

Prof. Dr. Kemal Leblebicioğlu _____
Supervisor, **Electrical and Electronics Engineering Dept., METU**

Examining Committee Members:

Prof. Dr. Mübeccel Demirekler _____
Electrical and Electronics Engineering Dept., METU

Prof. Dr. Kemal Leblebicioğlu _____
Electrical and Electronics Engineering Dept., METU

Prof. Dr. Kemal Özgören _____
Mechanical Engineering Dept., METU

Prof. Dr. Mustafa Kuzuoğlu _____
Electrical and Electronics Engineering Dept., METU

Dr. H. Burak Kaygısız _____
Chief of Guidance and Control Division, TÜBİTAK-SAGE

Date: _____

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: Çağdaş EVCİMEN
Signature :

ABSTRACT

DEVELOPMENT AND COMPARISON OF AUTOPILOT AND GUIDANCE ALGORITHMS FOR MISSILES

EVCİMEN, Çağdaş
M.S., Department of Electrical and Electronics Engineering
Supervisor: Prof. Dr. Kemal Leblebicioğlu

August 2007, 127 Pages

In order to have an interception with a target, a missile should be guided with a successful guidance algorithm accompanied with a suitable autopilot structure. In this study, different autopilot and guidance designs for a canard-controlled missile are developed. As a first step, nonlinear missile mathematical model is derived by using the equations of motion with aerodynamic coefficients found by Missile DATCOM program.

Autopilot design starts by the linearization of the nonlinear missile model around equilibrium flight conditions. Controllers based on the concepts of optimal control theory results and sliding mode control are designed. In all of the designs, angle of attack command and roll angle command type autopilot structures are used. During the design process, variations in angle of attack, Mach number and altitude can lead to significant performance degradation. This problem is typically solved by applying gain-scheduling methodology according to these parameters.

There are different types of guidance methods in the literature. Throughout this study, proportional navigation guidance and its modified forms are selected as a base algorithm in the guidance system design. Other robust forms of guidance methods, such as an optimal guidance approach and sliding mode guidance, are also formed for performance comparison with traditional proportional navigation guidance approach. Finally, a new guidance method, optimal proportional-integral guidance, whose performance is the best among all of the methods included in the thesis against highly maneuvering targets, is introduced.

Keywords: Missile, Autopilot, Guidance, Sliding Mode Control, Optimal Proportional-Integral Guidance

ÖZ

FÜZELER İÇİN OTOPİLOT VE GÜDÜM ALGORİTMALARI GELİŞTİRİLMESİ VE KARŞILAŞTIRILMASI

EVCİMEN, Çağdaş
Yüksek Lisans., Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi : Prof. Dr. Kemal Leblebicioğlu

Ağustos 2007, 127 Sayfa

Füzenin bir hedefi vurabilmesi için, başarılı bir güdüm algoritmasına eşlik eden uygun bir otopilot yapısıyla güdümünün yapılması gerekmektedir. Bu çalışmada, kanard kontrollü bir füze için çeşitli otopilot ve güdüm tasarımları geliştirilmiştir. İlk adım olarak, hareket denklemleri ile Missile DATCOM programı yardımıyla bulunan aerodinamik katsayılar kullanılarak füzenin doğrusal olmayan matematiksel modeli çıkarılmıştır.

Otopilot tasarımı, doğrusal olmayan füze modelinin denge uçuş noktaları etrafında doğrusallaştırılması ile başlar. Optimal kontrol teori sonuçları ve kayan kipli denetim genel kavramlarına dayanan denetimciler tasarlanmıştır. Bütün tasarımlarda, hücum ve yuvarlanma açılarını izleyen otopilot yapıları kullanılmıştır. Tasarım süreci boyunca, hücum açısı, Mach sayısı ve yükseklik değerlerindeki değişimler, tasarım sonuçlarında önemli miktarda performans kötüleşmesine neden olabilirler. Bu problem, genel olarak bu parametrelere bağlı olarak uygulanan kazanç tablolama yöntemiyle çözülmüştür.

Literatürde kullanılan birçok güdüm yöntemi mevcuttur. Bu çalışma boyunca, oransal navigasyon güdümü ve onun değişik formları güdüm sistemi tasarımda temel algoritma olarak seçilmiştir. Optimal bir güdüm yaklaşımı ve kayan kipli bir güdüm yöntemi gibi diğer gürbüz güdüm yöntemleri oransal navigasyon güdüm yaklaşımıyla performans karşılaştırması yapmak için oluşturulmuştur. Son olarak, tezin içersinde kullanılan tüm güdüm yöntemlerine göre yüksek manevra yapan hedeflere karşı daha başarılı bir güdüm yöntemi olan optimal oransal-tümlevsel güdüm yöntemi önerilmiştir.

Anahtar Kelimeler: Füze, Otopilot, Güdüm, Kayan Kipli Denetim, Optimal Oransal-Tümlevsel Güdüm

To My Fiancée,

To My Father

ACKNOWLEDGMENTS

I would like to express my special thanks and appreciation to my supervisor Prof. Dr. Kemal Leblebicioğlu for his valuable comments, and expertly guidance throughout all stages of this study.

I wish to thank Dr. A. Pınar Koyaz, Dr. H. Burak Kaygısız, Kadriye Tiryaki and all other colleagues at SAGE-GSG for their discussions and advices on the subject that guided me through. I would like to extend my thanks to our advisor Assist. Prof. Dr. Yakup Özkazanç at SAGE-GSG for his advices and comments during the thesis.

Finally, I would like to thank to my parents for their life-long love, support and unattainable understanding. I wish to express my deepest special thanks to my endless love, Nurdan Gürkan for her love, support, help and motivation.

This study is carried out at Guidance and Control Laboratories of TÜBİTAK-SAGE. The support provided by TÜBİTAK-SAGE is greatly acknowledged.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

X, Y, Z	axes of earth fixed reference frame
x, y, z	axes of body fixed reference frame
$C^{(a,b)}$	orthogonal transformation matrix that represents a transformation from frame F_b to frame F_a.
$\vec{u}_i^{(a)}$	i[th] unit basis vector of reference frame F_a
$C^{(e,b)}$	direction cosine matrix
ϕ, θ, ψ	euler angles (in roll, pitch and yaw planes)
V_x, V_y, V_z	velocity components in earth fixed reference frame
u, v, w	velocity components in body fixed reference frame
p, q, r	components of angular velocity in the body fixed reference frame with respect to the earth fixed reference frame
$\vec{F}$	sum of all externally applied forces to the missile
$\vec{V}$	total missile velocity
m	missile mass
F_x, F_y, F_z	components of total force acting on the missile expressed in the body frame
$\vec{M}$	sum of all externally applied torques to the missile
$\vec{H}$	angular momentum of the missile
$\hat{J}$	inertia dyadic
$\hat{J}$	inertia matrix
I_{xy}, I_{xz}, I_{yz}	cross inertia terms
I_x, I_x, I_y	elements of inertia matrix

M_x, M_y, M_z	components of moment acting on the missile expressed in the body frame
F_{ax}, F_{ay}, F_{az}	aerodynamic force components
M_{ax}, M_{ay}, M_{az}	aerodynamic moment components
g_x, g_y, g_z	body frame components of the gravitational acceleration
Q	dynamic pressure
A	cross sectional area of the missile
d	missile diameter
ρ	air density
ρ_0	air density at sea level
h	altitude
C_i $(i=x, y, z, l, m, n)$	aerodynamic coefficients
$\delta_a, \delta_e, \delta_r$	control surface deflections (aileron, elevator, rudder)
α	angle of attack
β	sideslip angle
M	mach number
C	speed of sound
γ	specific heat ratio of air
R	universal air gas constant
T	ambient temperature
T_0	temperature at sea level
C_x	axial force coefficient
C_y	side force coefficient
C_z	normal force coefficient
C_l	rolling moment coefficient
C_m	pitching moment coefficient
C_n	yawing moment coefficient
C_D	drag force coefficient
C_L	lift force coefficient

$\delta_1, \delta_2, \delta_3, \delta_4$	actual four control surface deflections
ω_n	natural frequency
ζ	damping ratio
X_m, Y_m, Z_m	axes of Missile DATCOM reference frame
n_c	normal acceleration command
N	effective navigation ratio
V_c	missile to target closing velocity
λ	line-of-sight angle
Y_r, Z_r	relative position errors between missile and target
R_{TM}	target to missile range
$a_H{}^c, a_V{}^c$	horizontal and vertical frame acceleration commands
$a_y{}^T, a_z{}^T$	target acceleration components in nonrotating frame
t_{go}	time-to-go until intercept
t_f	flight time
$M_y{}^{ZEM}, M_z{}^{ZEM}$	zero effort miss
K_P, K_I	gains of optimal proportional-integral guidance method
$\phi_e{}^c$	error in roll angle command
ϕ_c	roll angle command
α_c	angle of attack command

CHAPTER 1

INTRODUCTION

1.1 General Information

The objective in the intercept scenario of a missile and a target is to minimize the relative distance at closest approach (miss distance), to maximize the likelihood of destroying the target. To accomplish this aim, the missile should be designed for maximum range, maneuverability, and lethality [1].

The main role of the missile system is thus tracking the target and destroying the enemy weapon systems or designated targets. This fact leads to the problem of accuracy, stability and robustness. In other words, performance criteria of missile systems require obtaining powerful control theory results. However, due to characteristic properties of missile systems, applying control theory principles to intercept problem does not seem easy at all. Firstly, the speed of missile is high and the system model is six degrees of freedom nonlinear equations with complex system dynamic couplings. Moreover, it operates in an extremely wide range of operational environments [2]. These kinds of facts make missile control system design a very difficult challenging problem.

Among the sections of missile systems, Navigation Guidance Control (NGC) module is the most important part due to its direct effect on the performance of the missile. Navigation system recognizes current missile position with respect to position of target all the time. Navigation computation is done with the help of the signals coming from accelerometers and gyroscopes. Guidance system decides the optimal path to the target considering missile systems' capability with missile control constraints. The control system stabilizes the missile with respect to the guidance commands generated throughout the flight. The flight control system

generates its own commands for control mechanisms in order to properly execute the guidance commands.

A typical missile guidance and control system can be seen in Figure 1 below.

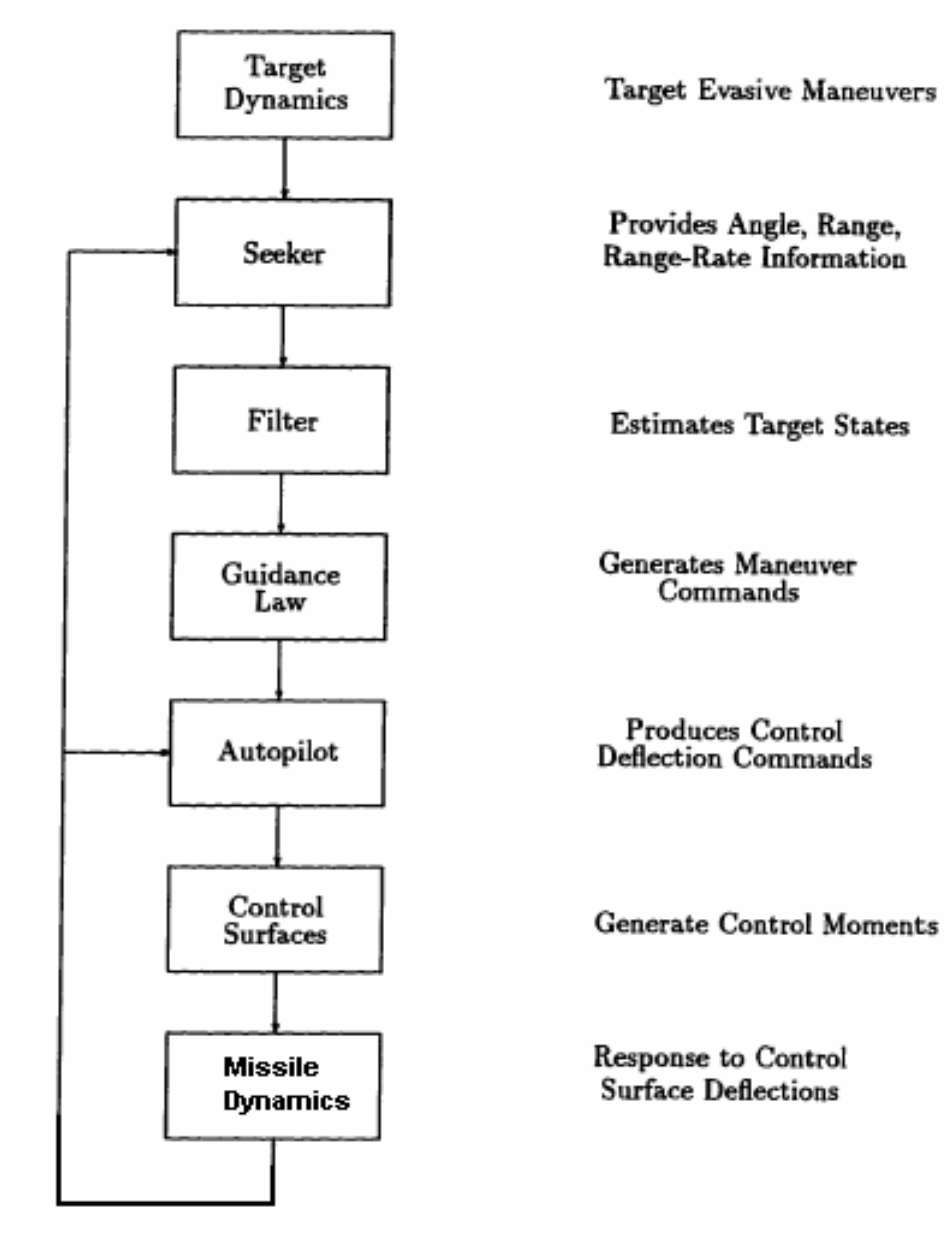

Figure 1 Missile guidance and control system

The seeker provides necessary angle measurements of possibly the range and range rate measurements of target. The filter uses the information given by the

seeker with a target acceleration model to obtain estimates of the relative position, velocity and target acceleration. In this study, no seeker model is designed, instead all the information that can be supplied by a seeker is supposed to be available. Furthermore, in the thesis a Kalman filter is used during the process of target state estimation. The guidance law then produces maneuver commands, which are sent to the autopilot. Next, autopilot converts commanded reference trajectory to a set of control surface deflections while maintaining stability. These control surface deflections provide the required control moments.

The focus of this study is on developing different autopilot and guidance algorithms. In order to make a proper design, nonlinear missile mathematical model is derived as the first step. Autopilot design is accomplished by linearization of the nonlinear missile model around various equilibrium flight conditions. Some trivial assumptions are made during the application of this procedure. Linear quadratic tracker based autopilot is selected as a reference controller to sliding mode autopilot design. In all of the designs, angle of attack command and roll angle command type autopilot structures are formed. Variations in angle of attack, Mach number and altitude can lead to significant performance degradation. This problem is typically solved by gain scheduling for both of the methods. Robustness analysis of the controllers is done using 50 percent aerodynamic derivative changes and 10 percent uniform disturbance added on feedback, separately. There are different types of guidance in the literature. Throughout this study, proportional navigation guidance and its modified forms are selected as a base algorithm in the guidance system design. Other robust forms of guidance methods are also formed and finally a new method is suggested that results in best performance outputs against highly maneuvering targets among all of the methods included.

1.2 About the Thesis

The missile considered here has no thrust vector subsystem on itself. The only control mechanisms are the control surfaces located on wings of the missile (canard-controlled). There are constraints on the control surfaces and states during the flight. The most important constraint is on the angle of attack capability

of the missile. Maximum angle of attack that the missile can perform is 8°. In addition, fin deflections and their rates are limited for $\pm 20^\circ$ and $\pm 200^\circ / s$, respectively. The saturation properties of attack angle and control surfaces add nonlinearity to overall system addition to nonlinear model of the missile itself.

The thesis is organized as follows;

Chapter 1: This chapter deals with general information about the interception problem and the key issues on guidance and autopilot system design of the missile.

Chapter 2: In this chapter, mathematical model of the missile model is constructed. Reference coordinate frames that are used during the thesis are defined as a first step. The equations of motion for a rigid missile are derived. These are composed of six kinematical and six dynamical first order nonlinear-coupled differential equations. Modeling of aerodynamic forces and moments acting on the missile are made. Linearization procedure is applied on both of aerodynamic model and equations of motion to get simplified, linear models to be used in the autopilot design. Control surface deflection convention and nonlinear actuator model details are also included in this chapter.

Chapter 3: This chapter deals with the autopilot design problem. Two different types of autopilot structures are formed based on linear quadratic tracker and sliding mode control approaches. Linear pitch and roll-yaw plane models derived in the previous chapter and their augmented state representations are used for both of the methods. Nonlinear simulation environment is formed without a guidance module to compare the methods as regards the tracking performance. Several reference command signals are formed and applied to the autopilot included nonlinear simulation constructed. As a final issue, robustness analysis of the controllers from the view of aerodynamic parameter variation and uniform disturbance rejection is performed.

Chapter 4: Different guidance algorithms are developed in this chapter. Reference guidance method is selected as the traditional proportional navigation

guidance. Modified forms of proportional navigation are also included here. After the development of the state space model of the interception problem, an optimal guidance approach and sliding mode guidance are formed based on this interception model. A new form of guidance method is developed as a technical contribution to the literature at this point. Namely, “Optimal Proportional-Integral Guidance” (which is shown to be superior to other methods) is suggested and derived. Acceleration commands, which are outputs of the guidance unit, are converted to angle of attack and roll angle commands by applying simple conversion logic. Finally, target state estimation with a Kalman filter is included for completeness.

Chapter 5: This chapter includes point mass simulation for the guidance methods proposed in this study. Point mass simulation is run for static, slow and fast maneuvering target models. Moreover, 6-DOF simulation environment is constructed to test system integration power. Here, only the new introduced guidance method combined with sliding mode autopilot structure is tested.

Chapter 6: This chapter summarizes the results presented in the rest of the thesis. It also states what future work can be done to improve the consequences obtained during the study of this work.

Appendix: In this part, Missile DATCOM (a program used in estimating missile aerodynamic coefficients) user manual is presented. It includes how and what a user enters as an input file (for005.dat) to the program such as missile physical properties, flight conditions, etc. It also represents which aerodynamic derivatives are obtained in the output file (for006.dat). Chapter closes with the DATCOM program file that includes all the physical data of the missile used in the simulations during the thesis.

CHAPTER 2

MISSILE MODELING

The equations that will be used to establish the motion of a missile are composed of twelve nonlinear first order differential equations. Six of them are the kinematical equations and other six are the dynamic equations. The kinematical equations of the missile can be established by investigating the trajectory of the motion and they are the consequences of transformation matrix applications, which build the bridge between the reference axis systems using Euler angles. The dynamic equations can be derived using the Newton's second law of motion and they relate the summation of the external forces and moments to the linear and angular accelerations of the body.

In this chapter, coordinate frames that will be used to describe the motion of the missile will be presented. Nonlinear mathematical model of the missile will be derived by using the equations of motion of the missile by combining aerodynamic model details in the derivations [3], [4], [5]. Linearization will be done around the equilibrium (trim) points by using some logical assumptions in order to obtain the linear models that will be used in the autopilot designs. The motions of the missile are controlled through the deflections of small fins that are governed by the autopilot models constructed. The control actuation system and some properties of aerodynamic derivatives are also included for completeness.

2.1 Reference Coordinate Frames

Two reference frames can be used to describe the motion of a missile, namely earth fixed coordinate frame and missile body coordinate frame. The earth fixed reference frame can be assumed inertial because the range of the missile is short

compared to the radius of the earth and motion of the missile is much faster compared to earth motion [6]. The axes of this inertial reference frame are represented by X, Y and Z. Here X axis points towards north, Z axis points downwards to earth's center, and Y axis is the complementing orthogonal axis found by the right hand rule. Body fixed reference frame has its origin at the missile's center of mass and its axes are x, y and z. Earth and body reference coordinate frames are represented as in Figure 2 [7].

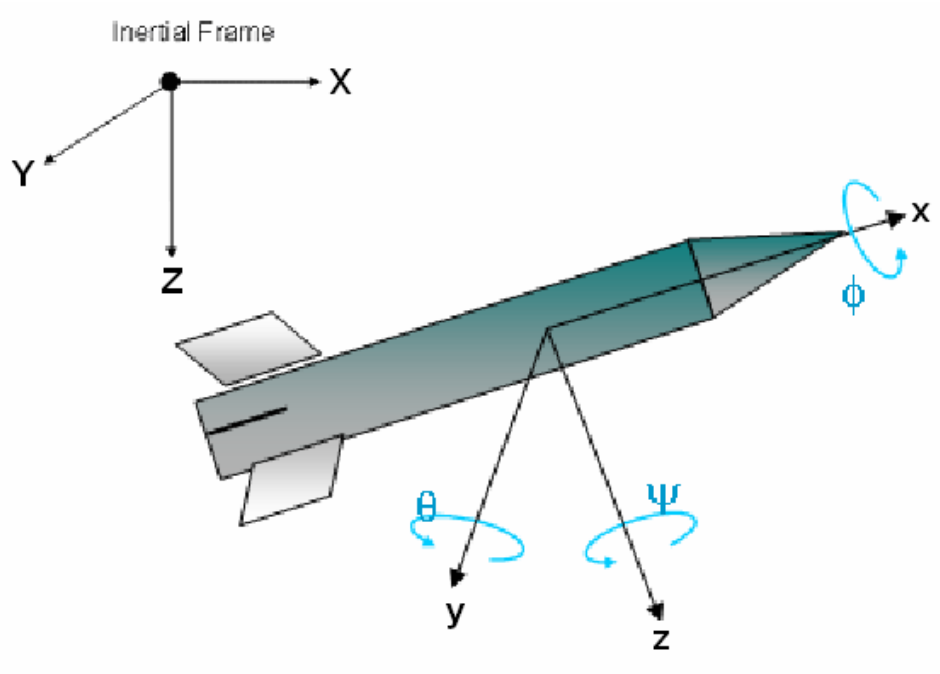

Figure 2 Earth and Body Axes

Any vector can be expressed in different coordinate frames as different column vectors. These column vectors can be related to each other by using linear transformations (rotations) between the specified coordinate frames. Such a transformation can be represented as

$$\vec{r}^{(a)} = C^{(a,b)} \vec{r}^{(b)} \tag{2.1}$$

$C^{(a,b)}$ is an orthogonal transformation matrix, which represents a transformation from frame F_b to frame F_a. The following property exists for this orthonormal matrix

$$C^{(a,b)} = C^{(b,a)^{-1}} = C^{(b,a)^T} \tag{2.2}$$

Let $\vec{u}_i^{(a)}$ and $\vec{u}_j^{(b)}$ be the ith unit basis vector of reference frame F_a and j[th] unit basis vector of reference frame F_b. Then the element at the ith row and the jth column of matrix $C^{(a,b)}$ can be expressed as

$$C_{i,j}^{(a,b)} = \cos(\theta_{i,j}) \tag{2.3}$$

where

$$\theta_{i,j} = \angle\left(\vec{u}_i^{(a)} \rightarrow \vec{u}_i^{(b)}\right) \tag{2.4}$$

Using the above equalities, the transformation between the body and the earth axis can be written as

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = C^{(e,b)} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{2.5}$$

$C^{(e,b)}$ is called the direction cosine matrix (representing the transformation from body frame to earth frame) and can be expressed uniquely using a set of Euler Angles as

$$C^{(e,b)} = \begin{bmatrix} \cos\theta\cos\psi & \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi \\ \cos\theta\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{bmatrix} \quad (2.6)$$

Definition of Euler Angles is done in missile coordinate system as in Figure 3 below.

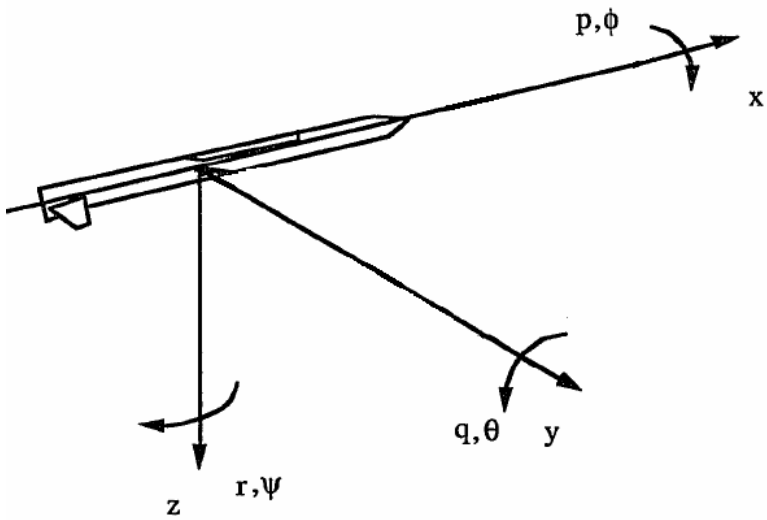

Figure 3 Missile coordinate system including system variables

2.2 Equations of Motion

2.2.1 Kinematical Equations of Motion

The position and the attitude of the missile with respect to the earth frame are found by solving the kinematical equations in order to describe the trajectory of motion. They are based on the basic coordinate transformations.

2.2.1.1 Translational Kinematics

Using the body to earth transformation logic given the details in equations (2.5) and (2.6), the velocity components in the earth reference coordinate frame and in the body reference coordinate frame are related as

$$\begin{bmatrix} V_x \\ V_y \\ V_z \end{bmatrix} = C^{(e,b)} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{2.7}$$

where V_x, V_y, V_z are the velocities of the missile in the earth reference frame and u, v, w correspond to the components of the missile's velocity in the body reference frame. $C^{(e,b)}$ is the transformation matrix from body frame to earth frame given in the equation (2.6). Expanding the relation in (2.7), the translational kinematical equations can be stated as follows

$$V_x = (\cos\theta\cos\psi)u + (\sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi)v + (\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi)w \tag{2.8}$$

$$V_y = (\cos\theta\sin\psi)u + (\sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi)v + (\cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi)w \tag{2.9}$$

$$V_z = (-\sin\theta)u + (\sin\phi\cos\theta)v + (\cos\phi\cos\theta)w \tag{2.10}$$

2.2.1.2 Rotational Kinematics

The rotational kinematical equations relate the time rate of change of the Euler Angles (orientation of the missile according to the reference coordinate frame) to the angular rates of the missile body [6], [8], [9]. Using the identity,

$$C^{(e,b)^T} C^{(e,b)} = I \tag{2.11}$$

it is seen that,

$$\frac{d}{dt}\left(C^{(e,b)^T} C^{(e,b)}\right) = \frac{d}{dt}(I) = 0 \tag{2.12}$$

$$C^{(e,b)^T}\,\dot{C}^{(e,b)} = -\dot{C}^{(e,b)^T}\,C^{(e,b)} \tag{2.13}$$

$$C^{(e,b)^T}\,\dot{C}^{(e,b)} = -\left(C^{(e,b)^T}\,\dot{C}^{(e,b)}\right)^T \tag{2.14}$$

The above equation indicates that $C^{(e,b)^T}\,\dot{C}^{(e,b)}$ is a skew symmetric matrix and so there exists a skew symmetric matrix $\tilde{\omega}$ generated from a column $\vec{w}$ such that $\tilde{\omega} = C^{(e,b)^T}\,\dot{C}^{(e,b)}$. $\vec{w}_{ang}$ is composed of the components of the angular velocity of the body frame with respect to the earth frame represented at the body frame. More explicitly,

$$\vec{w}_{ang} = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2.15}$$

thus the corresponding skew symmetric matrix is

$$\tilde{\omega} = \begin{bmatrix} 0 & -\mathrm{r} & \mathrm{q} \\ \mathrm{r} & 0 & -\mathrm{p} \\ -\mathrm{q} & \mathrm{p} & 0 \end{bmatrix} \tag{2.16}$$

Solving $C^{(e,b)^T}\,\dot{C}^{(e,b)} = \tilde{\omega}$ for the Euler angle rates, the rotational kinematical equations are found as

$$\dot{\phi} = p + \left(q\sin\phi + r\cos\phi\right)\tan\theta \tag{2.17}$$

$$\dot{\theta} = q\cos\phi - r\sin\phi \tag{2.18}$$

$$\dot{\psi} = \frac{\left(q\sin\phi + r\cos\phi\right)}{\cos\theta} \tag{2.19}$$

2.2.2 Dynamic Equations of Motion

The dynamic equations of motion can be found from Newton's second law of motion for rigid bodies, which states that time rate of change of the momentum is equal to the net force applied on the body and time rate of change of the angular momentum is equal to the net moment applied on the body. During the derivation of translational and rotational dynamic equations of motion, mass and inertia terms will be taken as constant, because mass and moment of inertia changes more slowly when compared with the translational and rotational velocities.

2.2.2.1 Translational Dynamics

For the derivation of translational dynamic equations, following equality will be used.

$$\vec{F} = \frac{d}{dt}(m\,\vec{V})\Big|_E \tag{2.20}$$

Here, $\vec{F}$ represents the sum of all externally applied forces to the missile. $\vec{V}$ is the total missile velocity and m is the missile mass. $\Big|_E$ indicates that the differentiations of the related vectors are done in the inertial frame. In the body frame equation (2.20) can be written again as

$$\vec{F} = m\left\{\frac{d}{dt}(\vec{V})\Big|_B + \vec{w}_{ang} \times \vec{V}\right\} \tag{2.21}$$

where $\Big|_B$ indicates that the differentiation is done in the body frame. In the body frame, the following column vectors will represent the relevant vectors

$$\vec{F} = \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} \tag{2.22}$$

$$\vec{V} = \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{2.23}$$

$$\vec{w}_{ang} = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2.24}$$

Substituting vectors in (2.22)-(2.24) into equation (2.21) and making the necessary manipulations, the translational dynamic equations are found as

$$\dot{u} = \frac{F_x}{m} - q\,w + r\,v \tag{2.25}$$

$$\dot{v} = \frac{F_y}{m} - r\,u + p\,w \tag{2.26}$$

$$\dot{w} = \frac{F_z}{m} - p\,v + q\,u \tag{2.27}$$

where F_x, F_y, F_z are components of the total force acting on the missile expressed in the body frame, including aerodynamic, thrust (propulsive) and gravitational forces. As expected, to be used in autopilot design procedure, force components should be clarified. It will be explained in later sections in a detailed way.

2.2.2.2 Rotational Dynamics

For the derivation of rotational dynamic equations, following equality will be used.

$$\vec{M} = \frac{d}{dt}(\vec{H})\Big|_E \tag{2.28}$$

Here, $\vec{M}$ represents the sum of all externally applied torques to the missile. $\vec{H} = \hat{J} \bullet \vec{w}_{ang}$ is the angular momentum of the missile, where $\hat{J}$ is the inertia dyadic. In the body frame equation (2.28) can be written again as

$$\vec{M} = \hat{J}\left\{ \frac{d}{dt}(\vec{w}_{ang})\Big|_B \right\} + \vec{w}_{ang} \times \hat{J} \bullet \vec{w}_{ang} \tag{2.29}$$

The inertia dyadic can be expressed in the body frame by the following matrix

$$\hat{J} = \begin{bmatrix} I_x & -I_{xy} & -I_{xz} \\ -I_{xy} & I_y & -I_{yz} \\ -I_{xz} & -I_{yz} & I_z \end{bmatrix} \tag{2.30}$$

The body axis of the missile is taken to be coincident with the principle axis of inertia. Hence product of cross inertia terms (I_{xy}, I_{xz} and I_{yz}) vanish [8]. Due to the symmetric structure of the missile used in this study, $I_y = I_z$ equality will be used. Thus the inertia matrix can be simplified for a symmetric missile as

$$\hat{J} = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_y \end{bmatrix} \tag{2.31}$$

By using the indications in equalities (2.24), (2.29) and (2.31) with the following column vector representation of total moment

$$\vec{M} = \begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} \tag{2.32}$$

The rotational dynamic equations can be found as

$$\dot{p} = \frac{M_x}{I_x} \tag{2.33}$$

$$\dot{q} = \frac{\left(M_y + r\,p\,(I_y - I_x)\right)}{I_y} \tag{2.34}$$

$$\dot{r} = \frac{\left(M_z + p\,q\,(I_x - I_y)\right)}{I_y} \tag{2.35}$$

where M_x, M_y, M_z are the components of the total moment acting on the body about its mass centre expressed in the body frame, including aerodynamic and thrust components. As expected, to be used in autopilot design procedure, moment components should be clarified.

2.3 Aerodynamic Model

In translational and rotational dynamic equations, force and moment components are not expressed clearly; instead, it is just emphasized that they have aerodynamic, thrust and gravitational components. Because the missile used in this study does not have a thrust model, thrust components can be dropped in the force and moment equations. After that, force and moment equalities that will be used in dynamic equations are stated as follows

$$\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = \begin{bmatrix} F_{ax} + m\,g_x \\ F_{ay} + m\,g_y \\ F_{az} + m\,g_z \end{bmatrix} \tag{2.36}$$

$$\begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} M_{ax} \\ M_{ay} \\ M_{az} \end{bmatrix} \tag{2.37}$$

In the above equations $F_{ax}, F_{ay}, F_{az}, M_{ax}, M_{ay}, M_{az}$ are forces and moments, which only occur due to aerodynamic effects on the missile during flight and g_x, g_y, g_z are body frame components of the gravitational acceleration, which is assumed to have a constant magnitude and to be oriented directly towards the earth center. Body frame components of the gravitational acceleration can be expressed as follows

$$\begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix} = \begin{bmatrix} -g.\sin\theta \\ g\sin\phi\cos\theta \\ g\cos\phi\cos\theta \end{bmatrix} \tag{2.38}$$

Unless otherwise stated, gravity compensation will be included as a part of the guidance design process. Thus, during the derivation of linear autopilot models, gravity terms can be neglected.

After the assumptions stated above, next comes the clarification of force and moment aerodynamic components. They can be calculated by using the dynamic pressure, some physical properties of the missile (i.e., cross sectional area and diameter) and aerodynamic coefficients found by using the well-known DATCOM program [10].

$$\begin{bmatrix} F_{ax} \\ F_{ay} \\ F_{az} \end{bmatrix} = Q\,A \begin{bmatrix} C_x \\ C_y \\ C_z \end{bmatrix} \tag{2.39}$$

$$\begin{bmatrix} M_{ax} \\ M_{ay} \\ M_{az} \end{bmatrix} = Q\,A\,d \begin{bmatrix} C_l \\ C_m \\ C_n \end{bmatrix} \tag{2.40}$$

Dynamic pressure in the equations (2.39) and (2.40) is expressed as

$$Q = \frac{1}{2}\rho V^2 \tag{2.41}$$

ρ is the air density and it changes with the altitude, as

$$\rho = \begin{cases} \rho_0 \left(1 - 0.00002256\, h\right)^{4.256} & for\ h \le 10000m \\ 0.412\, e^{-0.000151\,(h-10000)} & for\ h > 10000m \end{cases}$$

Here ρ_0 is the air density at sea level (1.223 kg/m^3). These formulas are obtained by curve fitting to the density variation in the ICAO standard atmosphere. Aerodynamic coefficients (C_i's ($i = x, y, z, l, m, n$)) usually depend on the speed of missile configuration and time history. But, according to the experimental results they are found to be functions of α, β, mach, body rates (p, q and r), $\dot{\alpha}, \dot{\beta}$, aerodynamic control surface deflections ($\delta_a, \delta_e, \delta_r$), centre-of-gravity changes and whether the main propulsion system is on or off (plume effects). In addition, the aerodynamic forces may depend upon whether reaction jets are on or off (jet interaction effects) [11].

Mach number, M, which is a dimensionless number defined as the ratio of the flight vehicle's velocity to the speed of sound at the altitude at which the vehicle is at flight. It is expressed as

$$M = \frac{V}{C} \tag{2.42}$$

where C is the speed of sound, i.e.

$$C = \sqrt{(\gamma R T)} \tag{2.43}$$

In the above equations γ is the specific heat ratio of the air, which is taken to be equal to 1.4, and R is the universal air gas constant, which is taken to be equal to 287 J/kgK. T is the ambient temperature, which also changes with altitude, and can be expressed as

$$T = \begin{Bmatrix} T_0\left(1 - 0.00002256\,h\right) & \text{for } h \le 10000\,m \\ 0.7744\,T_0 & \text{for } h > 10000\,m \end{Bmatrix}$$

Here T_0 is the temperature at sea level and it is equal to 293 K. These formulas are also obtained by curve fitting to the temperature variation in the ICAO standard atmosphere.

Throughout the flight, some flight angles are introduced to describe the motion of the missile. These angles are angle of attack (α) and sideslip angle (β). The definition of these angles can be seen in Figure 4.

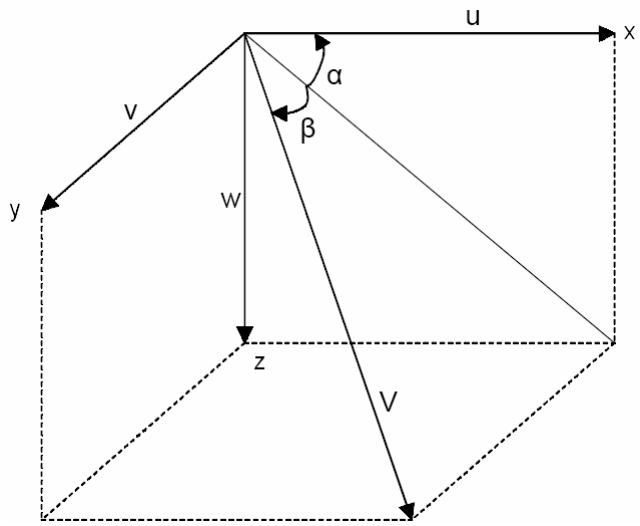

Figure 4 Definitions of flight angles (α and β)

Using the total velocity of the missile, α and β can be expressed as

$$\alpha = \arctan\left[\frac{w}{u}\right] \tag{2.44}$$

$$\beta = \arcsin\left[\frac{v}{V}\right] \tag{2.45}$$

From the above facts, aerodynamic coefficients can be stated to be a function of several flight variables

$$C_i = C_i(M, \alpha, \beta, \delta_a, \delta_e, \delta_r, p, q, r, \dot{\alpha}, \dot{\beta})$$

The subscripts a, e, r of the δ terms denote the aileron, elevator and rudder deflections, respectively. The aerodynamics effects due to the plunge effects (from $\dot{\alpha}, \dot{\beta}$) are assumed zero. The dimensionless C_i terms are designated as

C_x : Axial force coefficient

C_y : Side force coefficient

C_z : Normal force coefficient

C_l : Rolling moment coefficient

C_m : Pitching moment coefficient

C_n : Yawing moment coefficient

In order to better understand aerodynamic capabilities of the missile, 3D plots of the coefficients according to variations of angle of attack and Mach number can be analyzed. In fact, aerodynamic force acting on the missile may be resolved into two components, namely lift and drag force components. In resolving the aerodynamic force into the lift and drag components, lift is always taken normal to $\vec{V}$ and drag along $\vec{V}$ (but in the opposite direction). Lift and drag normally increase as angle of attack is increased. They are supposed to be acting on the centre of pressure that is not located at the centre of gravity, hence producing a moment [4], [12]. In order to better understand the relation between force coefficients, Figure 5 below gives the relational idea.

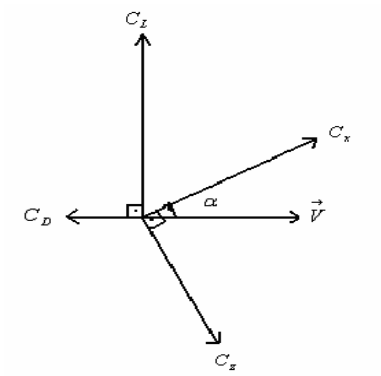

Figure 5 Relation between aerodynamic force coefficients

According to the representation in Figure 5, equalities between the components can be derived as follows

$$C_D = -C_x \cos\alpha - C_z \sin\alpha \tag{2.46}$$

$$C_L = C_x \sin\alpha - C_z \cos\alpha \tag{2.47}$$

Here, for the completeness of analysis, variation of axial, normal, drag and lift coefficients and pitching moment coefficient with Mach number and angle of attack will be given.

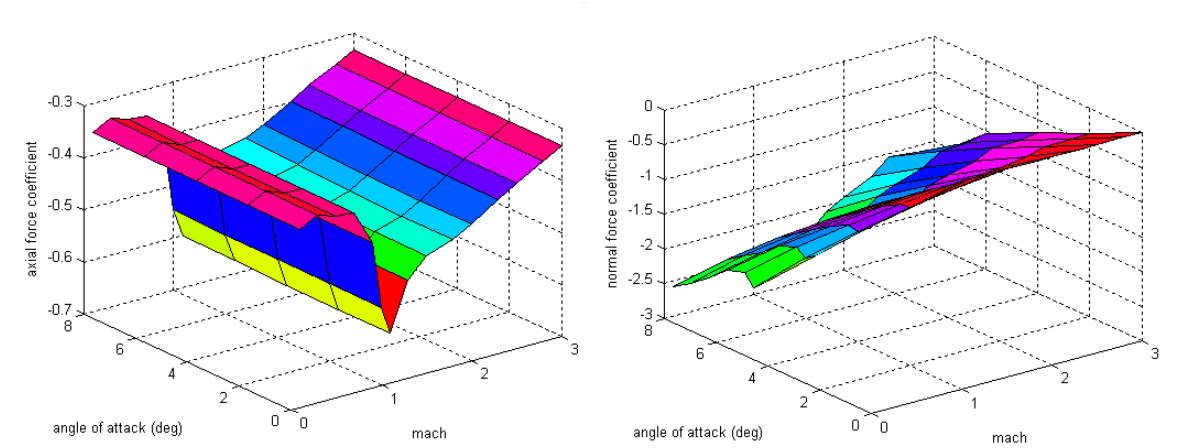

Figure 6 Axial and normal force coefficients

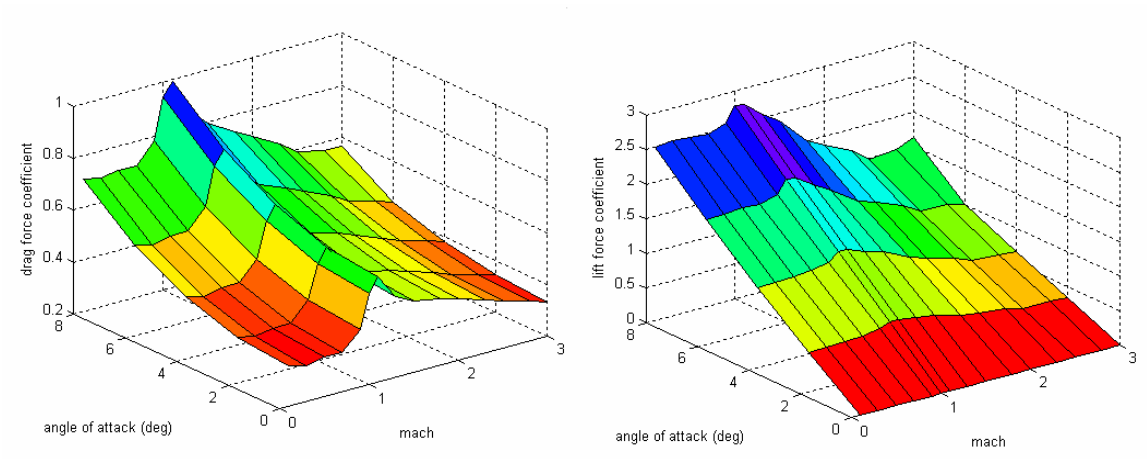

Figure 7 Drag and lift force coefficients

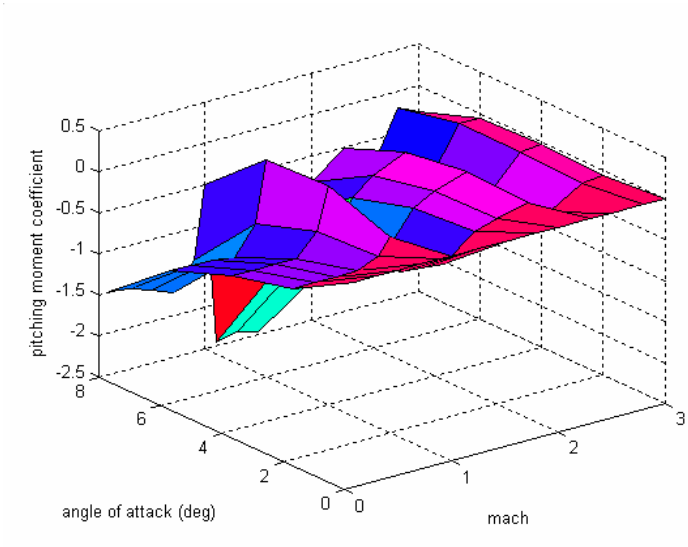

Figure 8 Pitching moment coefficient

As expected, drag and lift force coefficients increase as angle of attack increases. For one portion of Mach number variation (towards Mach number nearly equals 1.5), drag force coefficient increases with an increase in Mach number and lift does not change very much. Moreover, for supersonic speeds, drag force decreases as Mach number increases.

2.4 Linearization

2.4.1 Linearizing Nonlinear Differential Equations

Linearization refers to finding the linear approximation to a function at a given point. In the study of dynamical systems, linearization is a method for assessing the local stability of an equilibrium point of a system of nonlinear differential equations. Suppose that a nonlinear system model can be represented as in the following most general notation

$$\dot{x} = f(x,u) \qquad x \in \Re^n \text{ and } u \in \Re^m \tag{2.48}$$

Let $u_e = [u_{1e}\, u_{2e} \ldots u_{me}]^T$ be a constant input that forces the system in (2.48) to settle into a constant equilibrium state $x_e = [x_{1e}\, x_{2e} \ldots x_{ne}]^T$; that is to say, u_e and x_e satisfy

$$f(x_e, u_e) = 0 \tag{2.49}$$

Let us now perturb the equilibrium input vector by allowing

$$u = u_e + \Delta u \Rightarrow x = x_e + \Delta x \tag{2.50}$$

Taylor's expansion yields

$$\begin{aligned} \dot{x} &= f(x_e + \Delta x, u_e + \Delta u) = \underbrace{f(x_e, u_e)}_{0} + \underbrace{\frac{\partial f}{\partial x}(x_e, u_e)}_{A}\Delta x + \underbrace{\frac{\partial f}{\partial u}(x_e, u_e)}_{B}\Delta u + \ldots \\ &\ldots + \underbrace{HOT}_{neglect} = A\,\Delta x + B\,\Delta u \end{aligned} \tag{2.51}$$

In equation (2.51), Jacobian matrices of f with respect to x and u, evaluated at (x_e, u_e) (for this study at trim flight conditions) are equal to the following matrices

$$\frac{\partial f}{\partial x}(x_e, u_e) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} \frac{\partial f_1}{\partial x_2} \cdots \frac{\partial f_1}{\partial x_n} \\ . \\ . \\ \frac{\partial f_n}{\partial x_1} \frac{\partial f_n}{\partial x_2} \cdots \frac{\partial f_n}{\partial x_n} \end{bmatrix}_{\substack{x=x_e \\ u=u_e}} \tag{2.52}$$

$$\frac{\partial f}{\partial u}(x_e, u_e) = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} \frac{\partial f_1}{\partial u_2} \cdots \frac{\partial f_1}{\partial u_m} \\ . \\ . \\ \frac{\partial f_n}{\partial u_1} \frac{\partial f_n}{\partial u_2} \cdots \frac{\partial f_n}{\partial u_m} \end{bmatrix}_{\substack{x=x_e \\ u=u_e}} \tag{2.53}$$

Similarly, if the outputs of the nonlinear system model are of the form $y = h(x,u)$, its linear form can be derived same as the rules explained above and resultant model is

$$\begin{aligned} y &= y_e + \Delta y \\ \Delta y &= C\,\Delta x + D\,\Delta u \\ C &= \frac{\partial h}{\partial x}(x_e, u_e) \\ D &= \frac{\partial h}{\partial u}(x_e, u_e) \end{aligned} \tag{2.54}$$

2.4.2 Linearization of Aerodynamic Model

For the missile model to be used in simulations, the aerodynamic coefficients are taken to be nonlinear functions of flight parameters. However, in order to use in the autopilot studies, they can be linearized by using the Taylor series expansion around the trim values of the flight parameters as

$$C_i(M,\alpha,\beta,\delta_a,\delta_e,\delta_r,p,q,r) = C_{i0}(M) + C_{i\alpha}(M)\alpha + C_{i\beta}(M)\beta + C_{i\delta_a}(M)\delta_a + C_{i\delta_e}(M)\delta_e + C_{i\delta_r}(M)\delta_r + C_{ip}(M)\,p\frac{d}{2V} + C_{iq}(M)\,q\frac{d}{2V} + C_{ir}(M)\,r\frac{d}{2V} + H.O.T.$$

$$i = x, y, z, l, m, n \qquad (2.55)$$

In the above equation, H.O.T. stands for the higher order terms in the Taylor series expansion and will be neglected. $\frac{d}{2V}$ term is included as a multiplier to the dynamic aerodynamic derivatives to make the final product dimensionless. Following notation can be used for aerodynamic derivatives

$$C_{ij}(M) = \left.\frac{\partial C_i}{\partial j}\right|_{j=j_0} \qquad j = \alpha, \beta, \delta_a, \delta_e, \delta_r, p, q, r \qquad (2.56)$$

The aerodynamic derivatives are taken at the trim values of the flight parameters and they are kept only as a nonlinear function of the Mach number. Thus, in order to use in the autopilot design studies, the dimensionless aerodynamic coefficients for a missile can be expressed linearly according to DATCOM outputs as

$$C_x = C_{x0}(M) \qquad (2.57)$$

$$C_y = C_{y_\beta}(M)\beta + C_{y_{\delta_a}}(M)\delta_a + C_{y_{\delta_r}}(M)\delta_r + C_{y_p}(M)\,p\frac{d}{2V} + C_{y_r}(M)\,r\frac{d}{2V} \qquad (2.58)$$

$$C_z = C_{z_\alpha}(M)\alpha + C_{z_{\delta_e}}(M)\delta_e + C_{z_q}(M)\,q\frac{d}{2V} \qquad (2.59)$$

$$C_l = C_{l_\beta}(M)\beta + C_{l_{\delta_a}}(M)\delta_a + C_{l_{\delta_r}}(M)\delta_r + C_{l_p}(M)\,p\frac{d}{2V} + C_{l_r}(M)\,r\frac{d}{2V} \qquad (2.60)$$

$$C_m = C_{m_\alpha}(M)\alpha + C_{m_{\delta_e}}(M)\delta_e + C_{m_q}(M)\,q\frac{d}{2V} \qquad (2.61)$$

$$C_n = C_{n_\beta}(M)\beta + C_{n_{\delta_a}}(M)\delta_a + C_{n_{\delta_r}}(M)\delta_r + C_{n_p}(M)\,p\frac{d}{2V} + C_{n_r}(M)\,r\frac{d}{2V} \qquad (2.62)$$

2.4.3 Linearization of Equations of Motion

In the derivation of autopilot model that will be used within the application of control system design of the missile, in addition to the linearization of aerodynamic model as in the previous section, equations of motion should also be linearized around equilibrium flight (trim) points. In this study, steady state flight condition of the missile is taken as the equilibrium point where the linearization will be performed. Moreover, some logical assumptions should be made in order to simplify the nonlinear equations during the linearization process.

First, aiming to control roll angle and keeping it at a constant steady state value, and decoupling pitch and roll-yaw plane state equations result in taking roll rate approximately zero. By using this fact, nonlinear rotational dynamic equations (i.e., equations (2.34) and (2.35)) can be reduced to the simplified linear equivalents below

$$\dot{q} = \frac{M_y}{I_y} \tag{2.63}$$

$$\dot{r} = \frac{M_z}{I_y} \tag{2.64}$$

Together with equation (2.33), and using the linearized forms of aerodynamic coefficients given as in the previous section, rotational linear dynamic equations are obtained as follows

$$\dot{p} = \frac{Q\,A\,d}{I_x}\left(C_{l_\beta}\,\beta + C_{l_p}\,\frac{d}{2V}\,p + C_{l_r}\,\frac{d}{2V}\,r + C_{l_{\delta_a}}\,\delta_a + C_{l_{\delta_r}}\,\delta_r \right) \tag{2.65}$$

$$\dot{q} = \frac{Q\,A\,d}{I_y}\left(C_{m_\alpha}\,\alpha + C_{m_q}\,\frac{d}{2V}\,q + C_{m_{\delta_e}}\,\delta_e \right) \tag{2.66}$$

$$\dot{r} = \frac{Q\,A\,d}{I_y}\left(C_{n_\beta}\,\beta + C_{n_p}\,\frac{d}{2V}\,p + C_{n_r}\,\frac{d}{2V}\,r + C_{n_{\delta_a}}\,\delta_a + C_{n_{\delta_r}}\,\delta_r \right) \tag{2.67}$$

As a simplification in the demonstration, equations (2.65)-(2.67) can be stated as

$$\dot{p} = L_\beta \beta + L_p p + L_r r + L_{\delta_a} \delta_a + L_{\delta_r} \delta_r \tag{2.68}$$

$$\dot{q} = M_\alpha \alpha + M_q q + M_{\delta_e} \delta_e \tag{2.69}$$

$$\dot{r} = N_\beta \beta + N_p p + N_r r + N_{\delta_a} \delta_a + N_{\delta_r} \delta_r \tag{2.70}$$

In equations (2.68)-(2.70), the stability derivatives used are

$$L_\beta = \frac{Q A d}{I_x} C_{l_\beta} \quad L_p = \frac{Q A d}{I_x} \frac{d}{2V} C_{l_p} \quad L_r = \frac{Q A d}{I_x} \frac{d}{2V} C_{l_r} \quad L_{\delta_a} = \frac{Q A d}{I_x} C_{l_{\delta_a}}$$

$$L_{\delta_r} = \frac{Q A d}{I_x} C_{l_{\delta_r}} \quad M_\alpha = \frac{Q A d}{I_y} C_{m_\alpha} \quad M_q = \frac{Q A d}{I_y} \frac{d}{2V} C_{m_q} \quad M_{\delta_e} = \frac{Q A d}{I_y} C_{m_{\delta_e}}$$

$$N_\beta = \frac{Q A d}{I_y} C_{n_\beta} \quad N_p = \frac{Q A d}{I_y} \frac{d}{2V} C_{n_p} \quad N_r = \frac{Q A d}{I_y} \frac{d}{2V} C_{n_r} \quad N_{\delta_a} = \frac{Q A d}{I_y} C_{n_{\delta_a}}$$

$$N_{\delta_r} = \frac{Q A d}{I_y} C_{n_{\delta_r}}$$

Instead of linearizing translational dynamic equations, taking flight angles (α and β) as state variables will make things easier during autopilot design steps. In the derivations, small angle approximation is taken into account as

$$\alpha = \arctan\left[\frac{w}{u}\right] \cong \frac{w}{u} \tag{2.71}$$

$$\beta = \arcsin\left[\frac{v}{V}\right] \cong \frac{v}{V} \tag{2.72}$$

Furthermore, velocity component along the x-axis of missile's body frame is taken approximately constant (at every linearization point) and to be much larger than the other components [9] as

$$V \cong u >> v, w$$

By using the above approximations and taking derivatives of equations (2.71) and (2.72)

$$\dot{\alpha} \cong \frac{\dot{w}}{V} \cong \frac{\frac{F_z}{m} - p\,v + q\,u}{V} \tag{2.73}$$

$$\dot{\beta} \cong \frac{\dot{v}}{V} \cong \frac{\frac{F_y}{m} - r\,u + p\,w}{V} \tag{2.74}$$

Taking roll rate approximately equal to zero as before and using linearized forms of aerodynamic coefficients

$$\dot{\alpha} \cong \frac{Q\,A}{mV}\left(C_{z_\alpha}\alpha + C_{z_q}\frac{d}{2V}q + C_{z_{\delta_e}}\,\delta_e \right) + q \tag{2.75}$$

$$\dot{\beta} \cong \frac{Q\,A}{mV}\left(C_{y_\beta}\beta + C_{y_p}\frac{d}{2V}p + C_{y_r}\frac{d}{2V}r + C_{y_{\delta_a}}\,\delta_a + C_{y_{\delta_r}}\,\delta_r \right) - r \tag{2.76}$$

As a simplification in the demonstration, equations (2.75) and (2.76) can be stated as

$$\dot{\alpha} \cong Z_\alpha\alpha + \left(Z_q + 1\right)q + Z_{\delta_e}\,\delta_e \tag{2.77}$$

$$\dot{\beta} \cong Y_\beta\beta + Y_p\;p + \left(Y_r - 1\right)r + Y_{\delta_a}\,\delta_a + Y_{\delta_r}\,\delta_r \tag{2.78}$$

In equations (2.77) and (2.78), the stability derivatives used are

$$Z_\alpha = \frac{Q\,A}{mV}C_{Z_\alpha} \quad Z_q = \frac{Q\,A}{mV}\frac{d}{2V}C_{Z_q} \quad Z_{\delta_e} = \frac{Q\,A}{mV}C_{Z_{\delta_e}} \quad Y_\beta = \frac{Q\,A}{mV}C_{y_\beta}$$

$$Y_p = \frac{Q\,A}{mV}\frac{d}{2V}C_{y_p} \quad Y_r = \frac{Q\,A}{mV}\frac{d}{2V}C_{y_r} \quad Y_{\delta_a} = \frac{Q\,A}{mV}C_{y_{\delta_a}} \quad Y_{\delta_r} = \frac{Q\,A}{mV}C_{y_{\delta_r}}$$

Omitting linearization process of translational kinematical equations here is logical since they will only be used in the transformation of body axes velocities to earth fixed inertial frame equivalences or vice versa. Thus, they are not included in the linear state models. As the final step, linearization of rotational kinematical equations will be explicitly made by using steady state flight angle conditions, i.e., small roll and pitch angles (can be taken here equal to zero in the steady state for linearization purpose)

$$\dot{\phi} = p \tag{2.79}$$

$$\dot{\theta} = q \tag{2.80}$$

$$\dot{\psi} = r \tag{2.81}$$

2.5 Control Surface Deflection Angles

In this thesis, the missile under consideration is controlled by fins located near the wings (canard-controlled). There are four control surfaces that are 90° apart from each other. Control surface deflection angles are taken to be positive in the clockwise direction as in Figure 9, to create positive roll moment when looked behind.

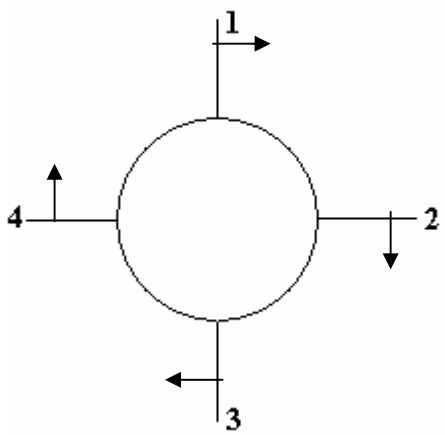

Figure 9 Control surface deflection angles (rear view)

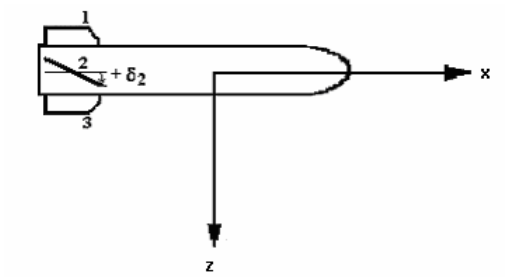

Figure 10 Control surface deflection angles (side view)

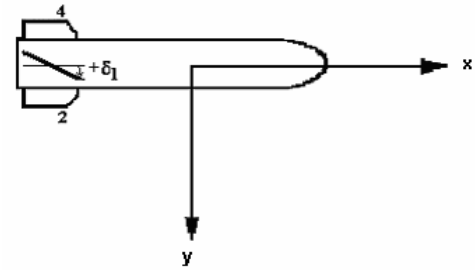

Figure 11 Control surface deflection angles (top view)

$\delta_1, \delta_2, \delta_3, \delta_4$ angles represent first, second, third and fourth control surface deflection angles, respectively. In the dynamical equations of motion, control surface deflections are categorized in three planes, namely roll, pitch and yaw planes as

δ_a : aileron surface (roll plane) deflections

δ_e : elevator surface (pitch plane) deflections

δ_r : rudder surface (yaw plane) deflections

These deflections are calculated by the help of the angles $\delta_1, \delta_2, \delta_3, \delta_4$ as below

$$\delta_a = \frac{\delta_1 + \delta_2 + \delta_3 + \delta_4}{4} \tag{2.82}$$

$$\delta_e = \frac{\delta_2 - \delta_4}{2} \tag{2.83}$$

$$\delta_r = \frac{\delta_1 - \delta_3}{2} \tag{2.84}$$

2.6 Control Actuation System Model

The dynamics of the control actuation system is also included in the overall simulation model. The control actuation system itself is a closed loop control system, which takes the fin deflection values produced by the autopilot as a command [6]. Its dynamics can also be expressed as a second order system as shown below

$$\frac{\delta(s)}{\delta_c(s)} = \frac{\omega_n^{\ 2}}{s^2 + 2\zeta\,\omega_n\, s + \omega_n^{\ 2}} \tag{2.85}$$

Natural frequency (ω_n) and damping ratio (ζ) of CAS can be taken equal to 75 rad/s and 0.707, respectively. The fin deflection in this missile is limited to $\pm 20^\circ$. Exceeding this value, the aerodynamic flow separation on the control surfaces cause a decrease in their lifting force performance, i.e., their control effectiveness. In addition, because of the actuator constraints, the rate of the fin deflection is also limited. This brings a nonlinear modeling of control actuation system as shown in Figure 12. The limiting values for the rate of fin deflections are taken to be $\pm 200^\circ / \sec$ in the six degrees of freedom simulations.

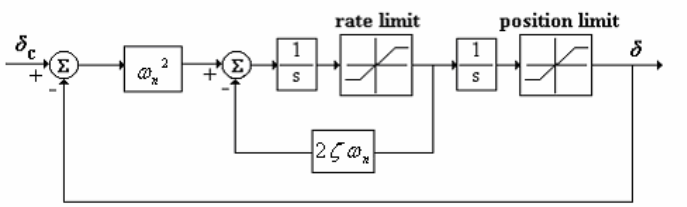

Figure 12 Nonlinear control actuation system model

2.7 Signs of the Aerodynamic Coefficients

The aerodynamic derivatives change sign when transferred from Missile Datcom to the body axes. If body axes representation is used, the coefficients should be corrected according to Table 1.

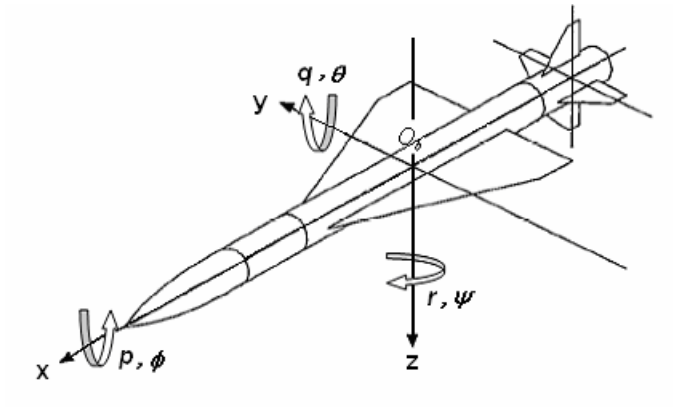

Figure 13 Missile body reference coordinate frame (with attitudes and rates in the positive direction)

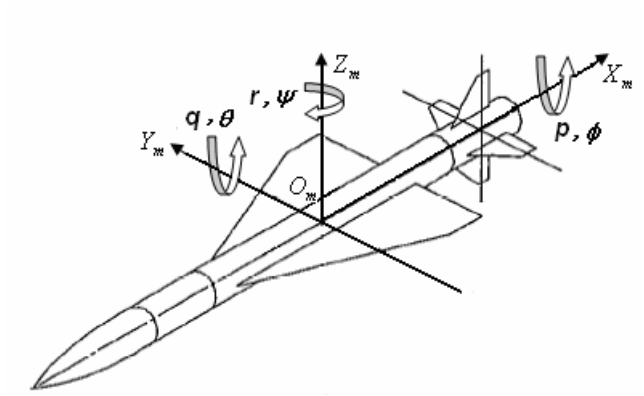

Figure 14 Missile DATCOM reference coordinate frame (with attitudes and rates in the positive direction)

Table 1 Sign changes during transfer of aerodynamic coefficients from Missile DATCOM to body axes

MISSILE DATCOM AERODYNAMIC COEFFICIENTS	**SIGN CHANGES**
C_X	$-C_X$
C_Y	C_Y
C_Z	$-C_Z$
C_{Z_α}	$-C_{Z_\alpha}$
C_{Z_q}	$-C_{Z_q}$
$C_{Z_{\delta_e}}$	$C_{Z_{\delta_e}}$
C_{m_α}	C_{m_α}
C_{m_q}	C_{m_q}
$C_{m_{\delta_e}}$	$-C_{m_{\delta_e}}$
C_{Y_β}	C_{Y_β}
C_{Y_r}	C_{Y_r}
$C_{Y_{\delta_r}}$	$-C_{Y_{\delta_r}}$
C_{n_β}	$-C_{n_\beta}$
C_{n_r}	$-C_{n_r}$
$C_{n_{\delta_r}}$	$C_{n_{\delta_r}}$
C_{l_β}	$-C_{l_\beta}$

Table 1 (cont'd)

MISSILE DATCOM AERODYNAMIC COEFFICIENTS	SIGN CHANGES
C_{l_r}	$-C_{l_r}$
C_{y_p}	$-C_{y_p}$
C_{n_p}	$-C_{n_p}$
C_{l_p}	$-C_{l_p}$
$C_{l_{\delta_a}}$	$C_{l_{\delta_a}}$
$C_{Y_{\delta_a}}$	$-C_{Y_{\delta_a}}$
$C_{n_{\delta_a}}$	$C_{n_{\delta_a}}$
$C_{l_{\delta_r}}$	$C_{l_{\delta_r}}$

2.8 Linear Missile Model

After the linearization procedure described step by step above and clarification of aerodynamic coefficients, equations (2.68)-(2.70) and (2.77)-(2.81) are ready to be used in the autopilot design. Moreover, it is possible to classify the linear equations into two different groups, namely pitch and roll-yaw plane equations. As can be recognized easily that roll-yaw autopilot that will be designed in the next chapter is a coupled one, having a multivariable nature. Two separated equation sets formed here can be rearranged or modified to be used for several different purposes such as regulation, tracking etc. In modeling, state space representation is used in order to apply the modern control theory principles to design the controllers.

2.8.1 Pitch Plane State Equations

To form pitch plane model in autopilot design, states of interest are selected as angle of attack, pitch rate and attitude, i.e., α, q and θ. By using the equations (2.77), (2.69) and (2.80), state space representation for pitch plane model is obtained as

$$\begin{bmatrix} \dot{\alpha} \\ \dot{q} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} Z_\alpha & Z_q + 1 & 0 \\ M_\alpha & M_q & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ q \\ \theta \end{bmatrix} + \begin{bmatrix} Z_{\delta_e} \\ M_{\delta_e} \\ 0 \end{bmatrix} \delta_e \tag{2.86}$$

2.8.2 Roll-Yaw Plane State Equations

In designing roll-yaw plane model for autopilot design, states of interest are sideslip angle, roll and yaw rates and roll and yaw angles, i.e., β, p, r, ϕ and ψ. By using the equations (2.78), (2.68), (2.70), (2.79) and (2.81), state space representation for roll-yaw plane model is obtained as

$$\begin{bmatrix} \dot{\beta} \\ \dot{p} \\ \dot{r} \\ \dot{\phi} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} Y_\beta & Y_p & Y_r - 1 & 0 & 0 \\ L_\beta & L_p & L_r & 0 & 0 \\ N_\beta & N_p & N_r & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \beta \\ p \\ r \\ \phi \\ \psi \end{bmatrix} + \begin{bmatrix} Y_{\delta_a} & Y_{\delta_r} \\ L_{\delta_a} & L_{\delta_r} \\ N_{\delta_a} & N_{\delta_r} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_r \end{bmatrix} \tag{2.80}$$

CHAPTER 3

AUTOPILOT DESIGN

Linear missile models derived in the previous chapter can be used to design autopilots that will create deflection commands to control missile flight. The autopilot-working loop formed in this thesis can be presented in block diagram representation as

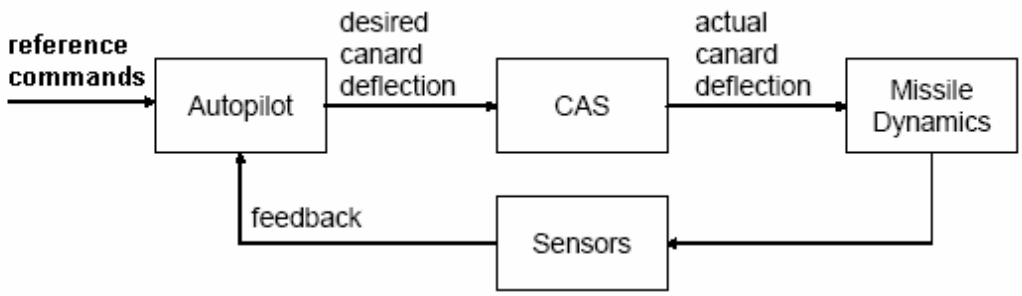

Figure 15 Autopilot loop

There are different types of autopilot structures in the literature. In this thesis, as derived in the previous chapter, two separated autopilot models will be designed to control pitch and roll-yaw planes of motion. Autopilots considered here should track reference commands decided by control engineer according to the system needs. Outputs of autopilot module will be surely the desired canard deflection commands that will pass through nonlinear control actuation system (CAS) to constitute actual canard deflection commands. These commands will be the input to the missile dynamics block to form states needed for feedback to the controller. One of the main problems in designing autopilots is the lack of full state information. In this study, full state feedback is assumed available to be used in design process. According to this fact, full state feedback controllers can be

implemented in the autopilot design. Here, linear quadratic and sliding mode controller structures are selected and tested by applying various reference signals to the autopilots.

In this chapter, linear quadratic tracker autopilot is formed as a reference controller. For a suitable trim point, controller's design details are presented. Next, sliding mode control principles are presented and a sliding mode autopilot is designed for comparison purposes. Effective gain scheduling method is used in order to make linear autopilots ready for nonlinear simulation. As a final step, 6DOF simulation program is constructed to compare the effectiveness of two methods. Robustness capabilities of the controllers are also included in this chapter.

3.1 Gain Scheduling

In control theory, gain scheduling is an approach to control of nonlinear systems that uses a family of linear controllers, each of which provides satisfactory control for a different operating point of the system.

One or more observable variables, called the scheduling variables, are used to determine what operating region the system is currently in and to enable the appropriate linear controller. In this study, angle of attack, Mach number and altitude are selected as the scheduling variables.

3.2 Linear Quadratic Controller (LQC) Design

3.2.1 Linear Quadratic Regulator (LQR)

The theory of optimal control is concerned with operating a dynamic system at minimum cost. The case where the system dynamics are described by a set of linear differential equations and cost is described by a quadratic functional is called the linear quadratic problem. One of the main results in the theory is LQR, a feedback controller that provides the solution to this problem.

LQR is thus an optimal control problem where the state equation (that will be used in control system design) is linear, the cost function is quadratic and the entire state is measured. The quadratic cost function can be used in a wide range of applications by appropriately selecting the weights on the state and control input [13].

For a continuous linear time-invariant system described by

$$\dot{x} = A\,x + B\,u \tag{3.1}$$

a control law is determined so that the performance index

$$J(t_0) = \frac{1}{2}\int_0^{\infty}\left(x^T Q\,x + u^T R\,u\right)dt \tag{3.2}$$

can be minimized, where $x \in R^n$ is a state vector and $u \in R^r$ is a control vector. Here, $A \in R^{n\times n}$ and $B \in R^{n\times r}$ are constant matrices, $Q \in R^{n\times n}$ is a positive semi-definite matrix, and $R \in R^{r\times r}$ is a positive definite one. Since the integration interval is infinite, this is called infinite horizon performance index; the performance objectives are referred to an infinite control interval $[0,\infty)$. Because the control horizon is infinite, Riccati equation may reach a steady-state (limiting) solution [4].

$$u = -K_{LQR}\,x \tag{3.3}$$

$$\dot{x} = (A - BK_{LQR})\,x = A_C\,x \tag{3.4}$$

In order to guarantee stability of the closed-loop system in equation (3.4) and a unique positive-definite limiting solution to Riccati equation, observability of $\left(\sqrt{Q}, A\right)$ pair (all plant modes should be weighted in the performance index) and controllability of (A, B) pair are needed. Thus, as long as the system and

performance index satisfy certain basic controllability and observability requirements, steady state LQR will yield gains that stabilize the system.

The performance index (3.2) can be interpreted as energy functional, so that making it small keeps the total energy of the closed-loop system small. Note that both the state $x(t)$ and the control inputs $u(t)$ are weighted; so that if J is small, then neither $x(t)$ nor $u(t)$ can be too large. Note that if J is minimized, then it is certainly finite, and since it is an infinite integral of $x(t)$, this implies that $x(t)$ goes to zero as t goes to infinity. This, in turn guarantees that the closed-loop system will be stable.

3.2.1.1 Quadratic Weight Selection

Two weighting matrices Q and R are selected by control design engineer. Depending on how these design parameters are selected, closed-loop system will exhibit a different response. Selecting Q large means that state $x(t)$ must be smaller to keep J small. On the other hand selecting R large means that control input $u(t)$ must be smaller to keep J small. This means that larger values of Q generally result in the poles of the closed-loop system matrix being further left in the s-plane so that the state decays faster to zero. On the other hand, larger R means that less control effort is used, so that the poles generally move to the right in the s-plane, resulting in larger values of the state $x(t)$.

In a general point of view, the selection of the state weighting matrix Q and the control-weighting matrix R are normally based on an iterative procedure using experience and physical understanding of the problem involved. To find suitable Q and R that provide a desired balance between the state variable responses and the control efforts while satisfying performance requirements and constraints, the transient response of a closed-loop system is typically examined. Because of indirect and nonlinear mapping between the weighting matrices and closed-loop eigenvalues, it is difficult to find suitable Q and R. The pole assignment methods can provide certain types of connection between closed-loop poles and feedback

gains, and it tends to result in more accurate transient responses. By using pole assignment only without consideration of an optimal control cost function, however, it is difficult to balance state and control variables and to account for control effectiveness [14]. Due to these reasons, here trial and error methodology is used in determining weighting matrices.

3.2.2 Linear Quadratic Tracker (LQT)

3.2.2.1 Optimal Linear Quadratic Tracker (OLQT)

OLQT can be derived using the well-known Hamiltonian approach. Instead of the performance index used in LQR, in this approach cost function contains the tracking error, not the state, because objective is to keep error small for tracking aim [15]. The performance index that is used for tracking purpose is expressed as

$$J(t_0) = \frac{1}{2}\int_0^{\infty}\left(e^T Q e + u^T R\, u\right)dt \tag{3.5}$$

Tracking error and performance output are

$$e = r - y \tag{3.6}$$

$$y = C x \tag{3.7}$$

where r refers to the constant reference signal. Though the reference input is assumed constant here for design purposes, this is to allow good closed-loop rise time and overshoot qualities. In practical applications as an interesting point of view, the designed controller works for any reference input $r(t)$, even though it is time-varying [15].

Thus, Hamiltonian matrix becomes

$$\mathrm{H} = \frac{1}{2}\left(e^T Q e + u^T R u\right) + p^T\left(A x + B u\right) \tag{3.8}$$

Next comes the use of equations (3.6) and (3.7) in (3.8) and making the necessary manipulations. As a final point Hamiltonian matrix are expended as

$$\mathrm{H} = \frac{1}{2}\left(r^T Q\, r - 2x^T C^T Q\, r + x^T C^T Q\, C\, x + u^T R\, u\right) + p^T \left(A\, x + B\, u\right) \tag{3.9}$$

The co-state equation is surely

$$\dot{p} = -\frac{\partial \mathrm{H}}{\partial x} = C^T Q\, r - C^T Q\, C\, x - A^T p \tag{3.10}$$

and the stationary condition is

$$0 = \frac{\partial \mathrm{H}}{\partial u} = R\, u + B^T p \tag{3.11}$$

Let us assume here that co-state is dependent on the state vector by the equation

$$p = K\, x - S \tag{3.12}$$

Then the control input to be determined using the equations (3.11) and (3.12) is

$$u = -R^{-1} B^T K\, x + R^{-1} B^T S \tag{3.13}$$

Putting this control input expression into the state equation (with the assumption of constant feed forward part by taking constant reference input during the design), and the co-sate assumption into the co-state equation, the following expressions are obtained.

$$\dot{x} = A\, x - B\, R^{-1} B^T K\, x + B\, R^{-1} B^T S = \left(A - B\, R^{-1} B^T K\right) x + B\, R^{-1} B^T S \tag{3.14}$$

$$\dot{p} = K\, \dot{x} = C^T Q\, r - C^T Q\, C\, x - A^T K\, x + A^T S \tag{3.15}$$

Next comes the combination of equations (3.14) and (3.15) as

$$\left(KA + A^T K - K\,B\,R^{-1}B^T K + C^T Q\,C\right)x = \left(A^T - K\,B\,R^{-1}B^T\right)S + C^T Q\,r \qquad (3.16)$$

According to the equation (3.16), algorithm can be stated systematically by making the assumption that left and right hand sides are exactly equal to zero.

<u>Algorithm</u>

- Solve the algebraic Riccati equation first and find K .

$$KA + A^T K - K\,B\,R^{-1}B^T K + C^T Q\,C = 0 \qquad (3.17)$$

- Next, solve for S .

$$S = -\left(A^T - K\,B\,R^{-1}B^T\right)^{-1} C^T Q\,r \qquad (3.18)$$

- Take equation (3.18) and use it in equation (3.13).

$$u = -R^{-1}B^T K\,x - R^{-1}B^T\left(A^T - K\,B\,R^{-1}B^T\right)^{-1} C^T Q\,r \qquad (3.19)$$

Define the following equalities to simplify the equality above as

$$K_T = R^{-1}B^T K \qquad (3.20)$$

$$G_r = R^{-1}B^T\left(A^T - K\,B\,R^{-1}B^T\right)^{-1} C^T Q \qquad (3.21)$$

$$G_T = B\,G_r \qquad (3.22)$$

- Then the closed loop system state equation and the simplified block diagram can be obtained as

$$\dot{x} = \left(A - BK_T\right)x + G_T\,r \qquad (3.23)$$

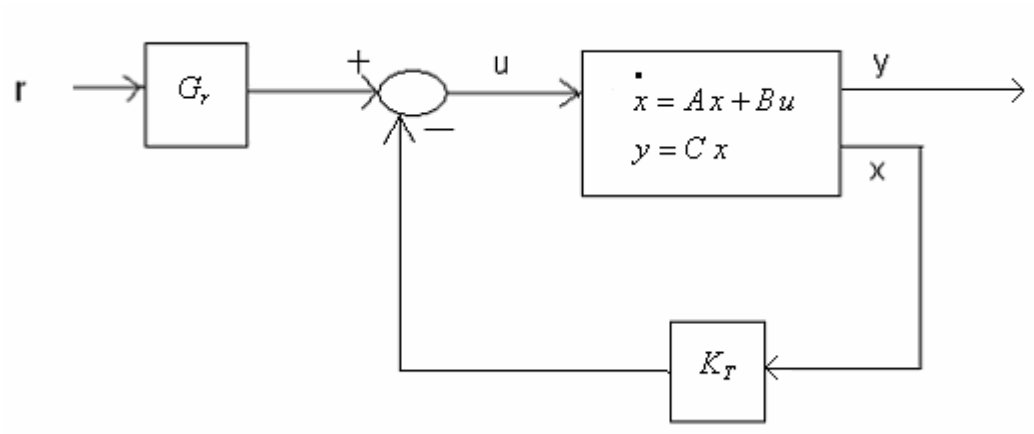

Figure 16 Block diagram representation of OLQT

3.2.2.2 Practical Suboptimal Tracker (PST)

This part shows how to design a suboptimal tracker that works well for practical applications and is robust to uncertainties and disturbances. Figure 17 shows the structure of the controller [16]. Here state-space system model in (3.1) is used with the assumption of complete state controllability.

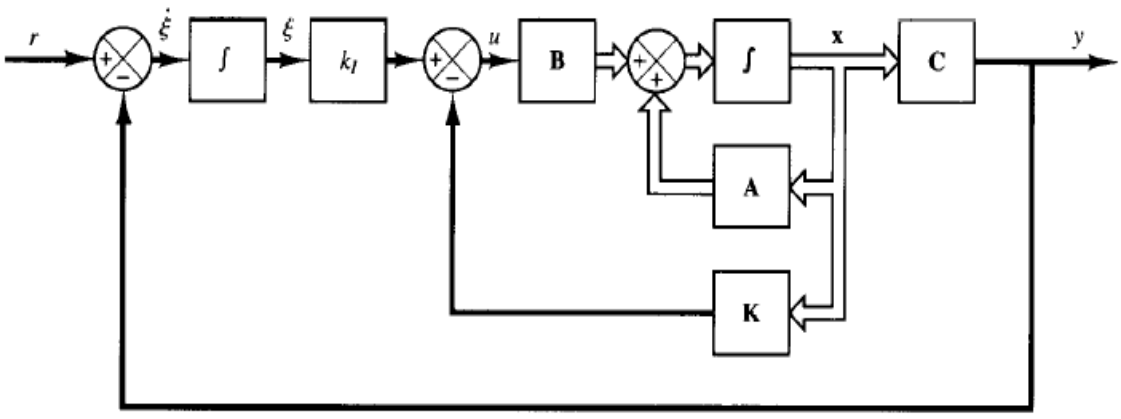

Figure 17 Block diagram representation of PST

Let us take ξ (integral of error) as a state variable and form the new representation as follows

$$\begin{bmatrix} \dot{x} \\ \dot{\xi} \end{bmatrix} = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix} \begin{bmatrix} x \\ \xi \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u + \begin{bmatrix} 0 \\ I \end{bmatrix} r \tag{3.24}$$

$$y = C\,x \tag{3.25}$$

$$u = -K\,x + k_I \xi\,,\ K_T = \begin{bmatrix} K & -k_I \end{bmatrix} \tag{3.26}$$

According to the new state variables formed ($\hat{x} = \begin{bmatrix} x & \xi \end{bmatrix}^T$), equation (3.24) and (3.26) can be represented as

$$\frac{d}{dt}\hat{x} = \bar{A}\,\hat{x} + \bar{B}\,u + \bar{G}\,r \tag{3.27}$$

$$u = -K_T\,\hat{x} \tag{3.28}$$

Here K_T can be found as in the classical LQR design and according to the nature of the method, new state variables will go to 0 in the steady state and error will approach to zero, thus the tracking purpose is achieved. Then the closed-loop system is

$$\frac{d}{dt}\hat{x} = \left(\bar{A} - \bar{B}\,K_T\right)\hat{x} + \bar{G}\,r \tag{3.29}$$

In addition, it has guaranteed stability and robustness properties of LQR (infinite gain margin and a guaranteed phase margin of at least 60°).

3.3 Linear Quadratic Autopilot Design Details

In this study, special purpose autopilots are designed to track reference commands supplied by guidance computer. As a reference command to the pitch

autopilot, angle of attack is selected to be the command signal and in addition, pitch rate must be limited in small values. For these aims, state representation in equation (2.86) is modified by dropping the pitch angle state. Like the pitch autopilot approach, for the roll-yaw autopilot, the roll angle is selected as the reference signal to the autopilot whereas the sideslip angle, roll and yaw rates are the variables to be minimized. Thus, the state representation in equation (2.87) is modified by dropping the yaw angle. Modified state representations will be given in the next sections for both of the autopilots. In this part of the thesis, autopilot design details for a sample design point selected (8000 m altitude, 1.5 mach speed and 2° angle of attack) will be presented. In order to have best performance outputs (tracking and regulation) in respect of above criteria, performance index parameters are decided by applying trial and error method. Gains found for every trim point (by using these penalties on the state variables) are scheduled to obtain the nonlinear controller for the simulations. In this study, according to missile aerodynamic capabilities during flight, trim points are selected as in the table below.

Table 2 Trim points used for autopilot design

Variables	Design Points
Angle of Attack (deg)	0°, 2°, 4°, 6°, 8°
Mach #	0.5, 1.0, 1.5
Altitude (m)	0, 4000, 8000

3.3.1 LQT Pitch Autopilot Design

OLQT autopilot is designed by using simplified linear pitch-plane missile model given below

$$\begin{bmatrix} \dot{\alpha} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} Z_\alpha & Z_q + 1 \\ M_\alpha & M_q \end{bmatrix} \begin{bmatrix} \alpha \\ q \end{bmatrix} + \begin{bmatrix} Z_{\delta_e} \\ M_{\delta_e} \end{bmatrix} \delta_e \tag{3.30}$$

On the other hand, PST model should involve integral of angle of attack error as an additional state

$$\begin{bmatrix} \dot{\alpha} \\ \dot{q} \\ \dot{\xi}_\alpha \end{bmatrix} = \begin{bmatrix} Z_\alpha & Z_q + 1 & 0 \\ M_\alpha & M_q & 0 \\ -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ q \\ \xi_\alpha \end{bmatrix} + \begin{bmatrix} Z_{\delta_e} \\ M_{\delta_e} \\ 0 \end{bmatrix} \delta_e + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \alpha_c \tag{3.31}$$

$$\xi_\alpha = \int (\alpha_c - \alpha) dt$$

Performance index parameters to be selected for this specific flight condition that meet the desired performance specifications for OLQT autopilot is selected as

$$Q = \begin{bmatrix} 30 & 0 \\ 0 & 0 \end{bmatrix}, \qquad R = 1 \tag{3.32}$$

On the other hand, for PST autopilot these parameters are chosen to be

$$Q = \begin{bmatrix} 0.625 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 10 \end{bmatrix}, \quad R = 1 \tag{3.33}$$

Detailed performance results for two different methods are given in Table 3 and Table 4. Furthermore, step responses of angle of attack and pitch rate are given in the same figure below for performance comparison.

Table 3 Performance results for OLQT (pitch autopilot)

Design Specs	OLQT
Eigenvalues	$-1.9012 \pm i2.0639$
Damping	0.678
Natural Frequency (rad/s)	2.81
Gain Margin	Inf
Phase Margin	66.3112
Settling Time (sec)	1.61
Rise Time (sec)	0.6344
K_T	$[-5.0241 \quad -2.6276]$
G_r	-5.4588

Table 4 Performance results for PST (pitch autopilot)

Design Specs	PST
Eigenvalues	$-0.8254 \pm 1.5313i$ -1.4972
Damping	0.474 1
Natural Frequency (rad/s)	1.74 1.5
Gain Margin	Inf
Phase Margin	61.7446
Settling Time (sec)	3.44
Rise Time (sec)	1.6283
K_T	$[-3.3652 \quad -2.173 \quad 3.1623]$

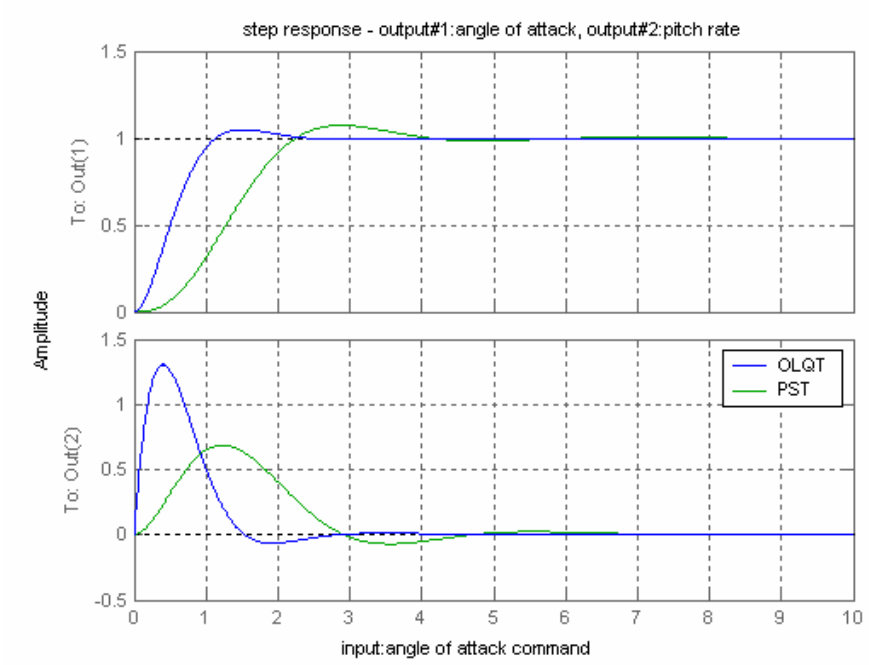

Figure 18 Step responses for OLQT and PST (pitch autopilot)

According to performance results obtained, two different approaches for autopilot synthesis seem to be successful for pitch plane. In addition, OLQT autopilot seems to be better than PST. Eigenvalues obtained with this design give improved damping and natural frequency. Phase margin is greater, thus further than the 60° phase margin limit. Furthermore, settling time and rise time control objectives are greatly improved and decreased compared with PST. However, noncausal nature of OLQT means that it cannot be implemented in practice when reference signal is not predetermined. For this reason, PST autopilot is selected as reference autopilot in nonlinear simulations.

3.3.2 LQT Roll-Yaw Autopilot Design

OLQT autopilot is designed by using the simplified linear roll-yaw plane missile model given below

$$\begin{bmatrix} \dot{\beta} \\ \dot{p} \\ \dot{r} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} Y_\beta & Y_p & Y_r - 1 & 0 \\ L_\beta & L_p & L_r & 0 \\ N_\beta & N_p & N_r & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \beta \\ p \\ r \\ \phi \end{bmatrix} + \begin{bmatrix} Y_{\delta_a} & Y_{\delta_r} \\ L_{\delta_a} & L_{\delta_r} \\ N_{\delta_a} & N_{\delta_r} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_r \end{bmatrix} \tag{3.34}$$

On the other hand, PST model should involve integral of roll angle error as an additional state

$$\begin{bmatrix} \dot{\beta} \\ \dot{p} \\ \dot{r} \\ \dot{\phi} \\ \dot{\xi}_\phi \end{bmatrix} = \begin{bmatrix} Y_\beta & Y_p & Y_r - 1 & 0 & 0 \\ L_\beta & L_p & L_r & 0 & 0 \\ N_\beta & N_p & N_r & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} \beta \\ p \\ r \\ \phi \\ \xi_\phi \end{bmatrix} + \begin{bmatrix} Y_{\delta_a} & Y_{\delta_r} \\ L_{\delta_a} & L_{\delta_r} \\ N_{\delta_a} & N_{\delta_r} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_r \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \phi_c \tag{3.35}$$

$$\xi_\phi = \int (\phi_c - \phi) dt$$

Performance matrix to be selected for this specific flight condition that meets the desired performance specifications for OLQT autopilot is given below

$$Q = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 30 \end{bmatrix}, \quad R = 1 \tag{3.36}$$

On the other hand, for PST autopilot this matrix is selected as

$$Q = \begin{bmatrix} 0.625 & 0 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 & 0 \\ 0 & 0 & 0.1 & 0 & 0 \\ 0 & 0 & 0 & 0.625 & 0 \\ 0 & 0 & 0 & 0 & 10 \end{bmatrix}, \qquad R = 1 \tag{3.37}$$

Detailed performance results for two different methods are given in Table 5 and Table 6.

Table 5 Performance results for OLQT (roll-yaw autopilot)

Design Specs	OQLT
Eigenvalues	$-0.0768 \pm i1.1813$ $-4.1366 \pm i4.0974$
Damping	0.0649 0.71
Natural Frequency (rad/s)	1.18 5.82
K_T	$\begin{bmatrix} -0.1947 & 0.9329 & -0.1510 & 3.7450 \\ 0.2358 & -0.9566 & -0.1288 & -3.9968 \end{bmatrix}$
G_r	$\begin{bmatrix} 3.7450 & -3.9968 \end{bmatrix}$

Table 6 Performance results for PST (roll-yaw autopilot)

Design Specs	PST
Eigenvalues	$-0.2383 \pm i1.1919$ $-1.6599 \pm i\,2.0785$ -2.7777
Damping	0.196 0.624 1
Natural Frequency (rad/s)	1.22 2.66 2.78
K_T	$\begin{bmatrix} -0.0910 & 0.7800 & -0.5872 & 1.8814 & -2.1486 \\ 0.3145 & -0.6404 & -0.4189 & -1.8521 & 2.3202 \end{bmatrix}$

Furthermore, step responses for sideslip angle and roll angle are given in the same figure below for performance comparison

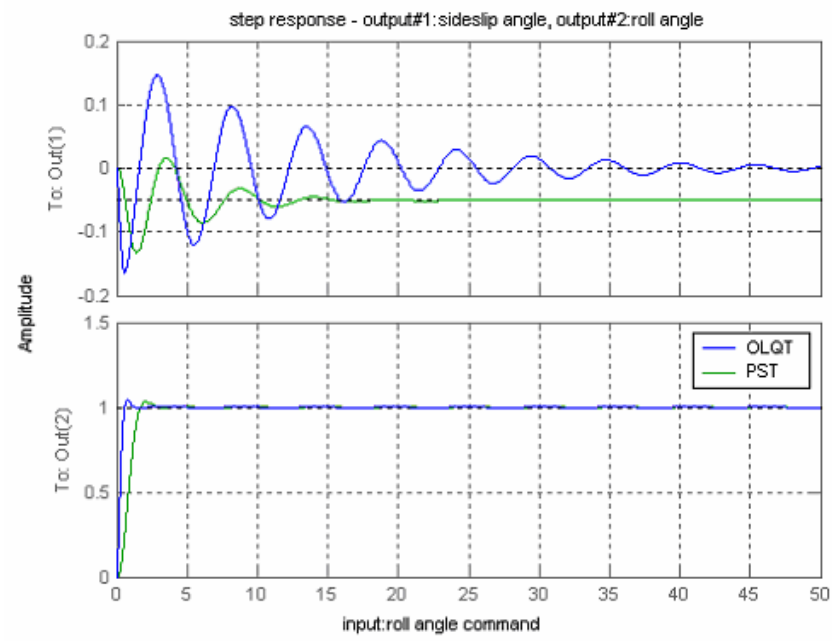

Figure 19 Step responses for OLQT and PST (roll-yaw autopilot)

According to performance results obtained, two different approaches for autopilot synthesis seem to be successful for roll-yaw plane. Since these autopilots are multivariable in nature, implementing methods according to gain and phase margin values are impossible. By comparing two different approaches due to rise and settling times of roll angle responses, OLQT seems a little better than PST approach. In addition, with the performance parameters selected as in equation (3.37), as can be seen in Figure 19, there is a little steady state error in sideslip angle performance. This problem can be solved by increasing the first diagonal element of the Q matrix in equation (3.37). Improved sideslip angle performance is obtained as in Figure 20 (sideslip angle is kept in a very small range of angle). On the other hand, oscillations in sideslip angle are a negative side of OLQT autopilot. Furthermore, noncausal nature of OLQT means that it cannot be implemented in practice when reference signal is not predetermined. At this point, PST autopilot is selected to be the reference roll-yaw autopilot in nonlinear simulations.

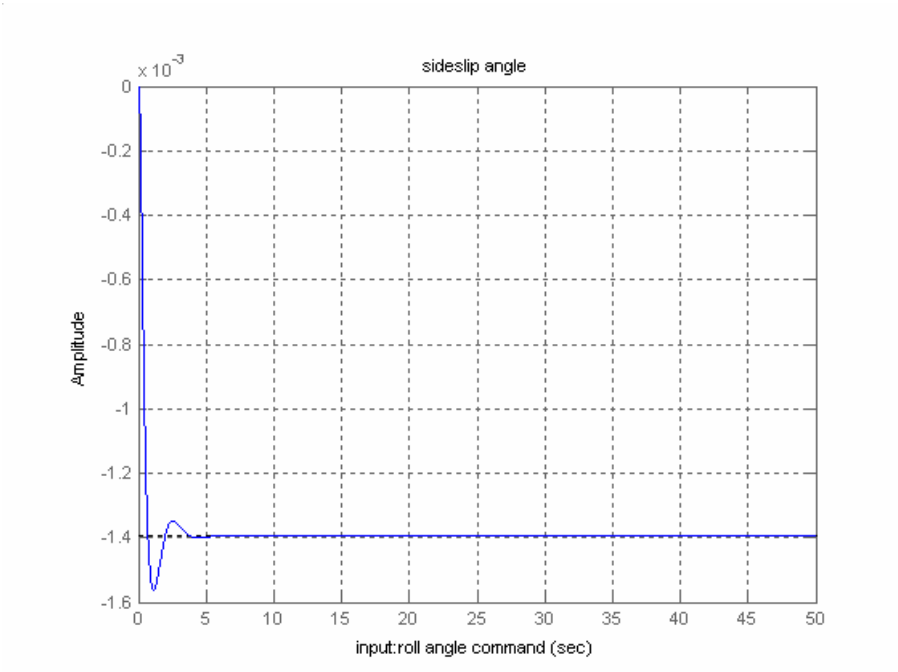

Figure 20 Improved sideslip angle performance for PST roll-yaw autopilot

3.4 Sliding Mode Controller (SMC) Design

3.4.1 Problem Statement

SMC is a viable high-speed switching feedback control. This variable structure control (VSC) law provides an effective and robust means of controlling nonlinear plants. The most distinguished feature of VSC is its ability to result in very robust control systems; in many cases invariant control systems. Loosely speaking, the term invariant means that the system is completely insensitive to parametric uncertainty and external disturbances [17]. If the controller is invariant to certain system parameter changes, there is no need to perform system failure detection or parameter identification. Since parameter identification is the main problem in control reconfiguration, employing a controller that does not require it provides a large advantage. Furthermore, an ideal sliding mode controller with no parasitic dynamics or actuator limits easily handles noise, parameter changes, and unmodeled nonlinearities with absolutely no degradation in tracking performance.

However, since real systems always have parasitic dynamics and actuator limits, the actual implementation of an SMC design becomes challenging.

In variable structure systems, the control is allowed to change its structure, that is, to switch at any instant from one to another member of a set of possible continuous functions of the state. The variable structure design problem is then to select the parameters of each of the structures and to define the switching logic [18].

Essentially, VSC utilizes a high-speed switching control law to drive the nonlinear plant's state trajectory onto a specified and user-chosen surface in the state space, and to maintain the plant's state trajectory on this surface for all subsequent time. This surface is called the switching surface because if the state trajectory of the plant is above the surface a control path has one gain and a different gain if the trajectory drops below the surface. As illustrated in the figure below, the period where the states are moving toward the sliding surface is known as the reaching phase, and the phase where the states follow the sliding surface is called the ideal sliding mode.

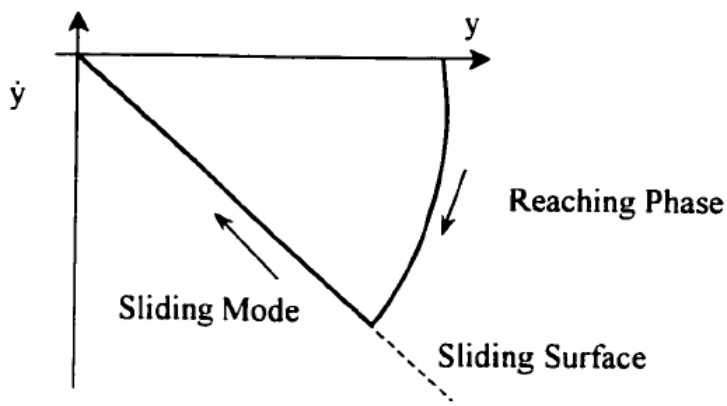

Figure 21 Phases of SMC

The plant dynamics restricted to this surface represent the controlled system's behavior. By proper design of the sliding surface, VSC attains the conventional goals of control such as stabilization, tracking, regulation, etc., [19].

In order to maintain the motion on the sliding mode, the control effort requires infinite frequency switching because the control law is undefined on the surface. If this infinite frequency switching were possible, the motion would be constrained to the sliding mode, and the dynamics of the closed-loop system would be that of switching surface equal to zero [20].

Consider the general systems linear with respect to control

$$\dot{x}(t) = f(x,t) + B(x,t)\,u(x,t) \tag{3.38}$$

where $x \in R^n$ and $u \in R^m$. The objective is to define;

- m switching functions, represented in vector form as $s(x) = [s_1(x), ..., s_m(x)]^T$ with the desired state trajectories.
- a variable structure control $u(x,t)$ such that any state outside the switching surface is driven to the surface in finite time and remains on this surface for all subsequent time.

3.4.2 Sliding Surface Design

Although general nonlinear switching surfaces are possible, linear ones are more prevalent in design. Moreover, for a large class of systems, design of linear switching surfaces proves amenable to classical linear controller techniques. Thus for clarity, convenience, and simplicity of exposition, here focus will be on linear switching surfaces of the form

$$s(x) = S\,x(t) \tag{3.39}$$

where S is a $m \times n$ matrix. A formal technique named equivalent control method will be used for finding equations of ideal sliding modes. Loosely speaking, the equivalent control is the continuous control action needed to maintain the ideal sliding motion. In this technique, the time derivative of the vector $s(x)$ along the system trajectory (3.38) is set equal to zero and resulting algebraic system is solved for control vector. This equivalent control (if it exists) is substituted into the original system. The resulting equations are the equations of ideal sliding mode. From a geometric point of view, the above method means finding a continuous control that directs the velocity vector along the intersection of the discontinuity surfaces [18]. Thus, "A sliding mode will exist for a system if in the vicinity of the switching surface; the state velocity vector (the derivative of the state vector) is directed towards the surface". An ideal sliding mode exists only when the state trajectory $x(t)$ of the controlled plant satisfies $s(x(t)) = 0$ at every $\mathrm{t} \geq \mathrm{t}_0$ for some t_0. This requires infinitely fast switching. In actual systems, all facilities responsible for the switching control function have imperfections such as delay, hysteresis, etc., which force switching to occur at a finite frequency. The representative point then oscillates within a neighborhood of the switching surface. This oscillation is called chattering. If the frequency of the switching is very high compared with the dynamic response of the system, the imperfections and the finite switching frequencies are often but not always negligible [19].

Differentiating equation (3.39) and using together with equation (3.38)

$$\frac{\partial s}{\partial x}\dot{x} = S\left[f(x,t) + B(x,t)u\right] = 0 \tag{3.40}$$

Solving equation (3.40) for u results in

$$u_{eq} = -\left(S\,B(x,t)\right)^{-1} S\,f(x,t) \tag{3.41}$$

Thus, the equivalent control method results in the sliding mode equations for the system (3.38)

$$\dot{x} = \left\{ I - B(x,t)\left(S\,B(x,t)\right)^{-1} S \right\} f(x,t) \tag{3.42}$$

where $m \times n$-dimensional matrix S and $\det\left(S\,B(x,t)\right)$ are assumed to be different from zero. The equivalent control is not actually applied in practice-at least, alone. If the plant model is correct (i.e., there are no unmodeled dynamics or disturbances), u_{eq} will maintain the sliding mode assuming the switching manifold has been reached.

After giving sliding mode equivalence dynamics of the system, problem reduces finding suitable S matrix to give required performance in the sliding mode. However, there is no systematic approach in the literature for this purpose by direct using nonlinear model. Instead, linear models (by modifying to have tracking aim, i.e., PST autopilot models) can be used to find switching surface matrix, systematically as explained below. For each flight trim point, linear switching surfaces are designed and then scheduled to be used in nonlinear simulations.

Consider the nominal linearized model of an uncertain system for tracking purpose

$$\dot{x}(t) = A\,x(t) + B\,u(t) + G\,r_c(t) \tag{3.43}$$

where state vector $x(t) \in \Re^n$, control input $u(t) \in \Re^m$, reference input $r_c(t) \in \Re^p$, $rank(B) = m$ and (A, B) is a controllable pair. Using a switching surface of the form (3.39), equivalent control can be determined as

$$u_{eq}(t) = -\left(S\,B\right)^{-1} S\,A\,x(t) - \left(S\,B\right)^{-1} S\,G\,r_c(t) \tag{3.44}$$

and the sliding mode dynamics can be found as

$$\dot{x}(t) = \left\{ I - B\left(S\,B\right)^{-1} S \right\} A\,x(t) + \left\{ I - B\left(S\,B\right)^{-1} S \right\} G\,r_c(t) \tag{3.45}$$

Defining $A_{eq} = \left\{ I - B\left(S\,B\right)^{-1} S \right\} A$ and $G_{eq} = \left\{ I - B\left(S\,B\right)^{-1} S \right\} G$, then the sliding mode dynamics equation in a more representative form will become

$$\dot{x}(t) = A_{eq}\, x(t) + G_{eq}\; r_c(t) \tag{3.46}$$

The system (3.43) can be transformed into regular form via a change of coordinates defined by an orthogonal matrix T_r such that

$$z(t) = T_r x(t) \tag{3.47}$$

where T_r is found by a QR decomposition of the input distribution matrix, that is [21]

$$T_r B = \begin{bmatrix} 0 \\ B_2 \end{bmatrix} \tag{3.48}$$

Then, defining

$$T_r\, A T_r^{\,T} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix},\; T_r\, G = \begin{bmatrix} G_1 \\ 0 \end{bmatrix} \tag{3.49}$$

and

$$S\, T_r^{\,T} = \begin{bmatrix} S_1 & S_2 \end{bmatrix} \tag{3.50}$$

The system can be expressed in the well-known regular form as

$$\begin{aligned} \dot{z}_1(t) &= A_{11}\, z_1(t) + A_{12}\, z_2(t) + G_1 r_c(t) \\ \dot{z}_2(t) &= A_{21}\, z_1(t) + A_{22}\, z_2(t) + B_2\, u(t) \end{aligned} \tag{3.51}$$

and

$$s(t) = S_1 z_1(t) + S_2 z_2(t) \tag{3.52}$$

During the sliding motion, the switching function must be identically zero, so

$$S_1 z_1(t) + S_2 z_2(t) = 0 \tag{3.53}$$

It can be shown that S_2 is non-singular, so z_2 can be solved for on the sliding mode

$$z_2(t) = -S_2^{-1} S_1 \, z_1(t) = -M \, z_1(t) \tag{3.54}$$

where $M \in \Re^{m \times (n-m)}$. The sliding mode is then governed by

$$\begin{aligned} \dot{z}_1(t) &= A_{11} \, z_1(t) + A_{12} \, z_2(t) + G_1 r_c(t) \\ z_2(t) &= -M \, z_1(t) \end{aligned} \tag{3.55}$$

This is an $(n-m)^{th}$ order system in which z_2 acts in the role of a linear full-state feedback control signal. Closing this loop gives the free motion of the system

$$\dot{z}_1(t) = \left(A_{11} - A_{12} M\right) z_1(t) + G_1 r_c(t) \tag{3.56}$$

By using quadratic minimization approach and performance index weightings on the states (similar weightings are used with PST in order to make logical comparison), the hyper plane matrix S will be determined from M by letting $S_2 = I_m$, giving

$$S T_r^T = \begin{bmatrix} M & I_m \end{bmatrix} \tag{3.57}$$

This approach minimizes the preceding calculations from M to S and so reduces the possibility of numerical errors.

3.4.3 The Control Law

3.4.3.1 Reaching Law Approach

Remembering that, sliding mode control procedure consists of two different steps, after designing sliding surfaces for every flight trim point; designer will face with the problem of forcing the states to the surfaces designed. This problem known as controller design in the reaching phase can be solved by using several different approaches.

The main requirement in the design is that the control should satisfy the reaching condition, i.e., $\dot{V}_{lya} < 0$ where $V_{lya} = \frac{1}{2} s^T s$. Additional needs are fast reaching and low chattering. In this study, one of the recent methods to design the reaching mode controller, namely "The Reaching Law Approach" is selected as the control law [22]. This method directly specifies the dynamics of the switching function. Moreover, by changing the parameters in switching function, dynamic quality of the system can be controlled in the reaching mode. The reaching mode is less sensitive to system perturbations and external disturbances if the reaching law method is used in designing the VSC system. A practical general form of reaching law is

$$\dot{s} = -Q\,\mathrm{sgn}(s) - K\,g(s) \tag{3.58}$$

where gains Q and K are diagonal matrices with positive elements (q_i and k_i s) that control the convergence speed of s and

$$\mathrm{sgn}(s) = \left[\mathrm{sgn}(s_1)\ldots\mathrm{sgn}(s_m)\right]^T$$
$$g(s) = \left[g_1(s)\ldots g_m(s)\right]^T$$

The scalar functions g_i satisfy the condition

$$s_i g_i(s_i) > 0 \qquad \text{when} \quad s_i \neq 0 \qquad \mathrm{i} = 1 \text{ to } \mathrm{m} \tag{3.59}$$

Using the reaching condition based on the lyapunov function stated beforehand, it can be proved that reaching law approach guarantees reaching to the sliding surface as follows

$$\dot{V}_{lya} = s\,\dot{s} = s\left(-Q\,\mathrm{sgn}(s) - K\,g(s)\right) = -Q\,s\,\mathrm{sgn}(s) - K\,s\,g(s) = -Q\underbrace{s\,\mathrm{sgn}(s)}_{+} - K\underbrace{s\,g(s)}_{+} < 0$$

Chattering can be reduced by tuning parameters q_i and k_is in the reaching law approach. By choosing q_is small, the momentum of the motion will be reduced as the system trajectory approaches the switching surface. As a result, the amplitude of the chatter will be reduced. However, q_is cannot be chosen equal to zero because the reaching time would become infinite; the system fails to be a sliding mode control system. On the other hand, for a disturbance greater than the magnitude of signum term, the VSC system will loose its robustness feature, creating a steady-state error. In summary, q_is should be kept large enough to cover the expected maximum perturbation, but not too large to avoid unnecessarily large chattering. Larger values for k_is increase the reaching rate when the state is not near the switching surface.

From now on, the control is solved directly from the reaching law specification described by (3.58) with nonlinear tracking model

$$\dot{s} = S\left(f(x,t) + B(x,t)\,u + G\,r_c(t)\right) = -Q\,\mathrm{sgn}(s) - K\,g(s) \tag{3.60}$$

Thus, VSC is easily solved from (3.60) as

$$u(x,t) = -\left(S\,B(x,t)\right)^{-1}\left[S\,f(x,t) + S\,G\,r_c(t) + Q\,\mathrm{sgn}(s) + K\,g(s)\right] \tag{3.61}$$

However, in order to get a real time simulation environment, instead of using nonlinear tracking model, linear model in equation (3.43) can be used. Then gain

scheduling is applied for adapting the overall controller to nonlinear simulation environment. Then, VSC becomes

$$u(x,t) = -(S\,B)^{-1}\left[S\,A\,x(t) + S\,G\,r_c(t) + Q\,\mathrm{sgn}(s) + K\,g(s)\right] \tag{3.62}$$

One of the underlying assumptions in the design and analysis of VSC systems is that the control can be switched from one value to another at will, infinitely fast. In practical systems, however, it is impossible to achieve the high switching control that is necessary to most designs. There are several reasons for this. One cause is the presence of finite time delays for control computation. The second cause is the limitations of physical actuators. Since it is impossible to switch the control at infinite rate, chattering always occurs in the sliding and steady state modes of a VSC system. In the steady state, chattering appears as a high-frequency oscillation about the desired equilibrium point and may serve as a source to excite the unmodeled high frequency dynamics of the system. The basic idea to overcome the chattering problem is to replace the discontinuous signum function with an arbitrarily close approximation. In this study, instead of $\mathrm{sgn}(s)$, an approximate function $\frac{2}{\pi}\arctan(s)$ is used.

By satisfying the equality in (3.59), in order to have fast reaching capability, cubic reaching function, i.e., $g(s) = s^3$ can be selected. After the determination of the functions to be used in the reaching phase of sliding mode controller, final form of SMC by modifying equation (3.62) becomes as

$$u(x,t) = -(S\,B)^{-1}\left[S\,A\,x(t) + S\,G\,r_c(t) + Q\frac{2}{\pi}\arctan\big(S\,x(t)\big) + K\,\big(S\,x(t)\big)^3\right] \tag{3.63}$$

3.4.4 Robustness and Invariance

Robustness is one of the most distinguishing properties of VSC systems. For a plant represented by either a linear or nonlinear high-order differential equations, the differential equations of the sliding mode can be entirely independent of

effects due to modeling errors and external disturbances. Thus, the sliding mode is said to be invariant (better than just robust) to modeling errors and disturbances. The invariance property requires that certain matching conditions be satisfied.

For general nonlinear systems

$$\dot{x} = f(x,t) + \Delta f(x, pv, t) + \left[B(x,t) + \Delta B(x, pv, t)\right] u + v(x, pv, t) \tag{3.64}$$

where pv is an uncertain parameter vector, v is external disturbance and Δf and ΔB are system perturbations, invariance hold true if the following matching conditions are satisfied

$$\begin{aligned} \Delta f(x, pv, t) &= B(x,t)\Delta \tilde{f}(x, pv, t) \\ \Delta B(x, pv, t) &= B(x,t)\Delta \tilde{B}(x, pv, t) \\ v(x, pv, t) &= B(x,t)\Delta \tilde{v}(x, pv, t) \end{aligned} \tag{3.65}$$

for certain $\Delta \tilde{f}(x, pv, t)$, $\Delta \tilde{B}(x, pv, t)$ and $\Delta \tilde{v}(x, pv, t)$.

The property of invariance in reaching mode is guaranteed only if SMC system is designed via reaching law approach as it is done in this study [22].

3.5 Nonlinear Simulations

Nonlinear simulation environment that is used to test two autopilot performances include nonlinear autopilot module (formed by gain scheduling method using either PST or SMC), nonlinear missile model (nonlinear equations of motion and gain scheduled aerodynamic forces and moments) and nonlinear control actuation system (second order nonlinear actuator module ready to be used in Matlab environment). Autopilots are tested with step-like and ramp behavior reference command signals, separately.

3.5.1 Simulations Studied

In the first simulation, step-like commands are given to the autopilots. Pitch autopilot is commanded as below

$$\alpha_c = \begin{cases} 0 & when \quad t < 25\,s \\ 5^\circ & otherwise \end{cases}$$

Roll-yaw autopilot is commanded with roll angle trajectory as follows

$$\phi_c = \begin{cases} 10^\circ & when \quad t < 20\,s \\ 0 & otherwise \end{cases}$$

Second simulation will cover commands that have ramp as well as step parts. Here, angle of attack command is defined to be

$$\alpha_c = \begin{cases} 1/4t & when \quad t < 20\,s \\ 5^\circ & when \quad 20\,s \le t < 25\,s \\ 0 & otherwise \end{cases}$$

On the other hand, roll-yaw autopilot command will be like

$$\phi_c = \begin{cases} 0 & when \quad t < 20\,s \\ 10^\circ & when \quad 20\,s \le t < 25\,s \\ (45-t)/2 & otherwise \end{cases}$$

3.5.1.1 Simulation I

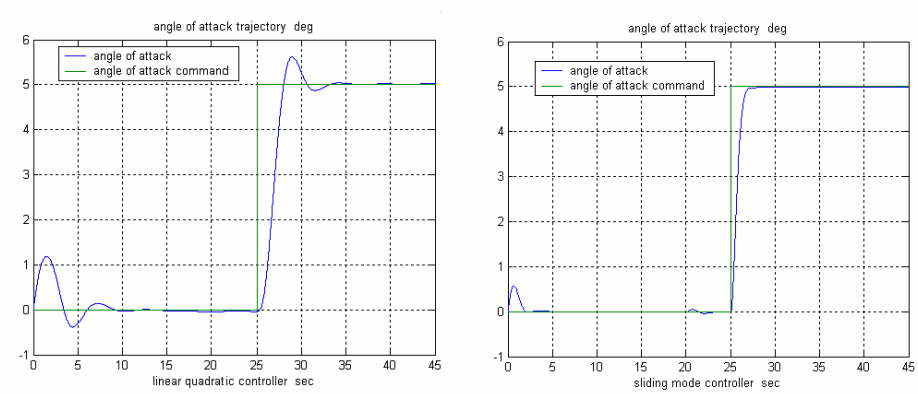

Figure 22 Angle of attack trajectories (Simulation I)

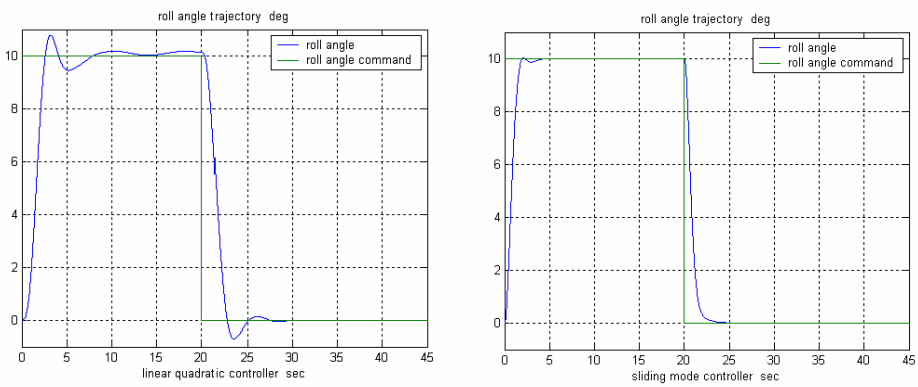

Figure 23 Roll angle trajectories (Simulation I)

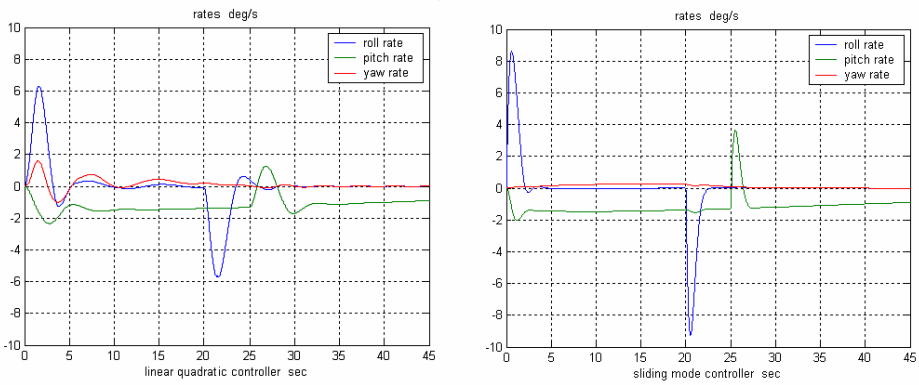

Figure 24 Angular rates comparison (Simulation I)

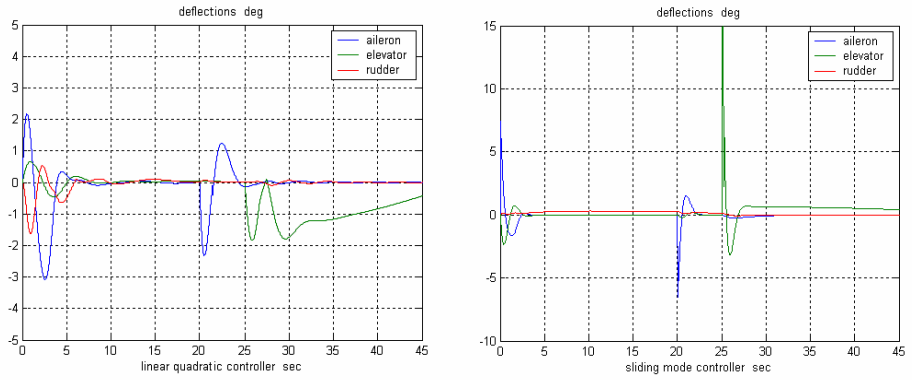

Figure 25 Actual canard deflection angles (Simulation I)

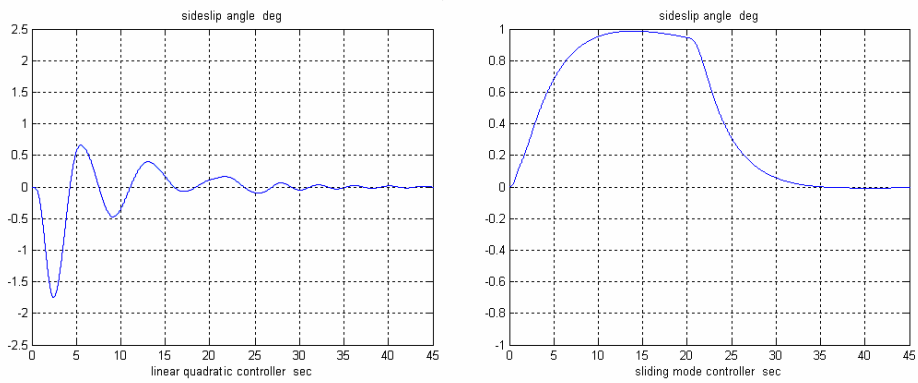

Figure 26 Sideslip angle (Simulation I)

3.5.1.2 Simulation II

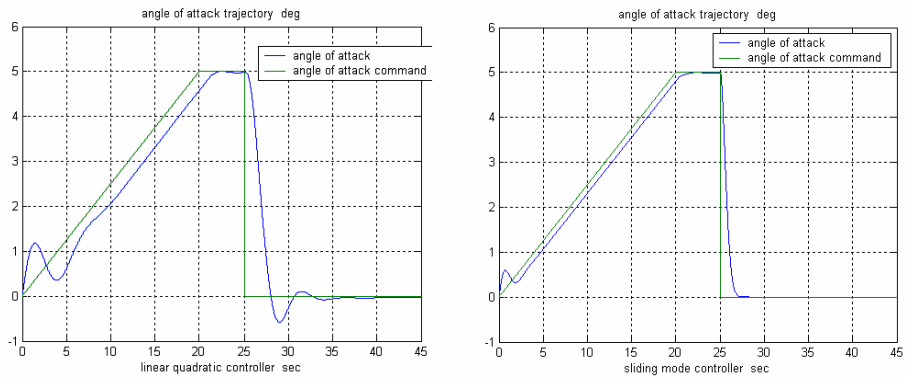

Figure 27 Angle of attack trajectories (Simulation II)

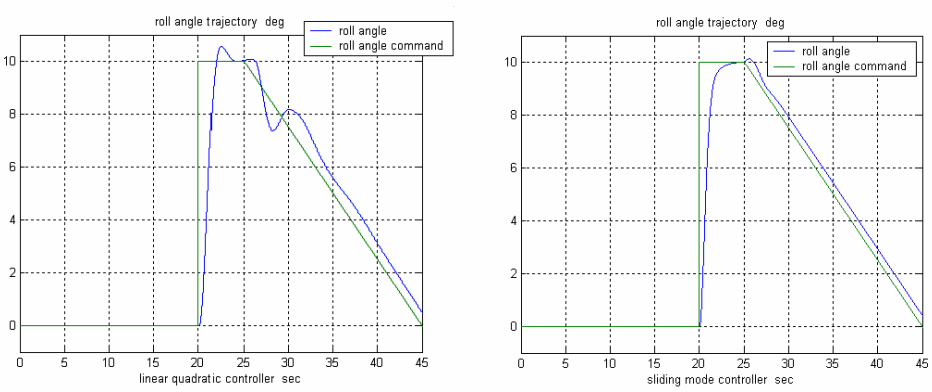

Figure 28 Roll angle trajectories (Simulation II)

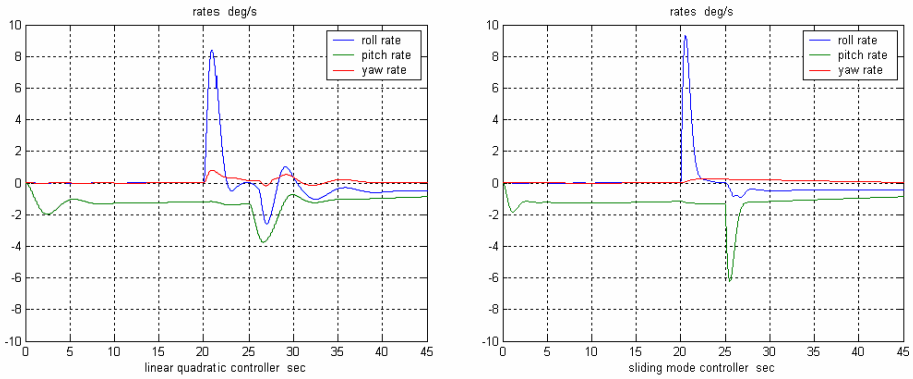

Figure 29 Angular rates comparison (Simulation II)

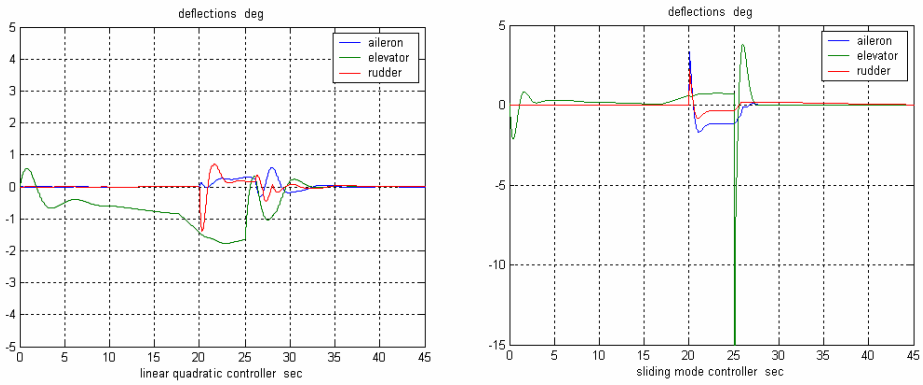

Figure 30 Actual canard deflection angles (Simulation II)

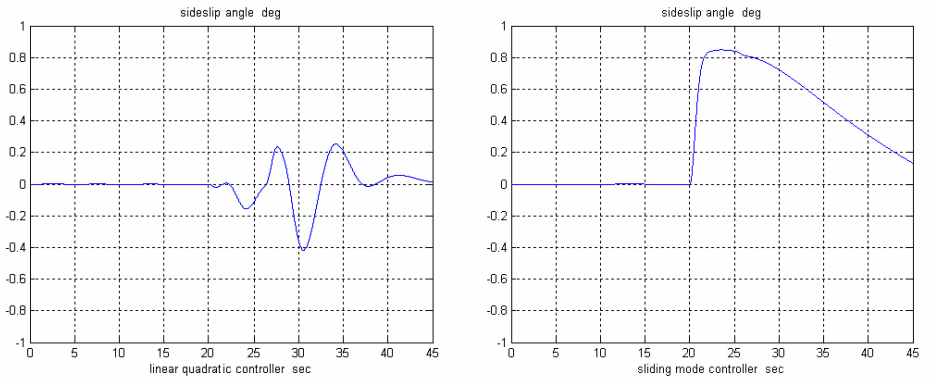

Figure 31 Sideslip angle (Simulation II)

According to the nonlinear simulation results, sliding mode autopilot seems better than linear quadratic tracker autopilot. Tracking performances are improved and oscillation magnitudes are greatly decreased. Smaller canard deflection angles, angular rates and sideslip angle assumptions are preserved for both of the methods.

3.5.2 Robustness Analysis

Nonlinear simulations done till here assume the ideal case, that is to say, no disturbance and aerodynamic parameter variations exist in the environment. However, in real world these effects play an important role on controllers overall performances. Thus, autopilot performances are tested against 50 percent aerodynamic derivative changes and 10 percent uniform disturbance added on feedback, separately. Here only tracking performance figures will be given only for simulation I, others are similar.

3.5.2.1 Aerodynamic Parameter Variation

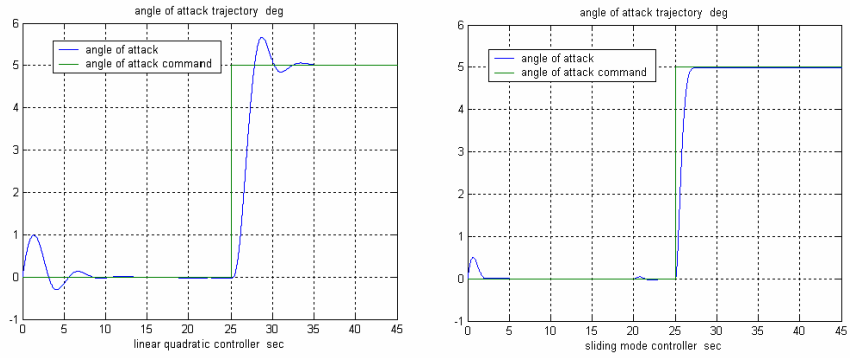

Figure 32 Pitch autopilot robustness against parameter variations

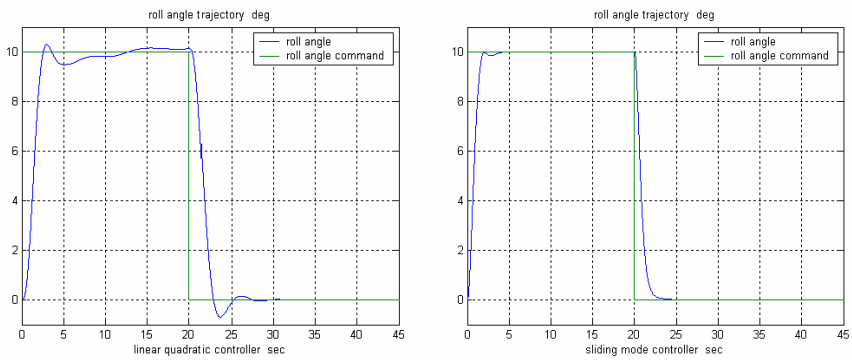

Figure 33 Roll-Yaw autopilot robustness against parameter variations

3.5.2.2 Uniform Disturbance Rejection

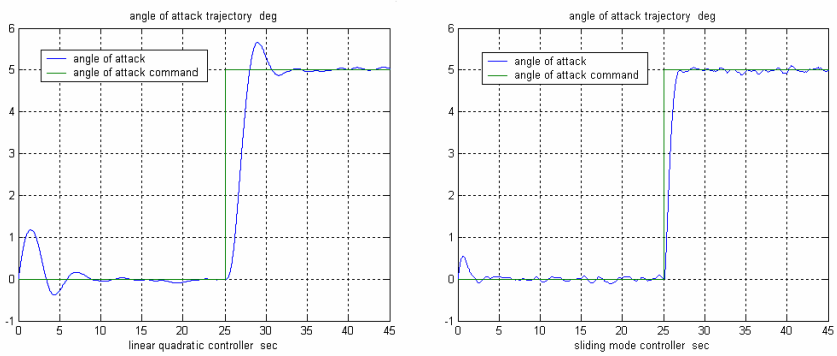

Figure 34 Pitch autopilot disturbance rejection performance

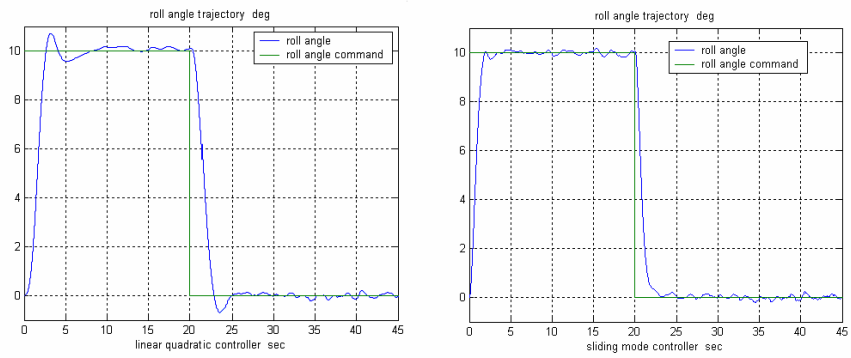

Figure 35 Roll-Yaw autopilot disturbance rejection performance

According to the results obtained in this part, controllers designed using both of the methods seem to be robust not only to disturbances but also to parameter changes as well.

CHAPTER 4

GUIDANCE DESIGN

Guidance can be defined as the strategy for how to steer the missile to intercept. It has normally been divided into;

- Target related guidance, where target-tracking data are provided in real time from a sensor, which can be on-board or off-board.
- Non-target related guidance, where the missile navigates to some predetermined point, which can be the target or the point where target related guidance can begin.

Type of guidance system used depends on the design of the missile and the accuracy requirements. In any case, the guidance command serves as the input to the control system. The command may be in the form of an attitude command, an angular rate command, a pitch or yaw acceleration command (for roll stabilized missiles), or a flight angle depending upon the type of guidance scheme used. Motion parameters of the missile can be obtained by the inertial navigation system (INS) or the global positioning system (GPS).

In this study as stated above, autopilots designed for pitch and roll-yaw planes are driven by angle of attack and roll angle reference commands, respectively. Thus, commands generated by guidance unit should be these two commands and fed to the autopilot block. Since outputs of all of the guidance methods analyzed here are acceleration commands in horizontal and vertical planes in the inertial frame, they should all be converted to angle of attack and roll angle reference commands in the body frame. Generalized guidance loop with all other system blocks can be shown as in the block diagram representation in Figure 36

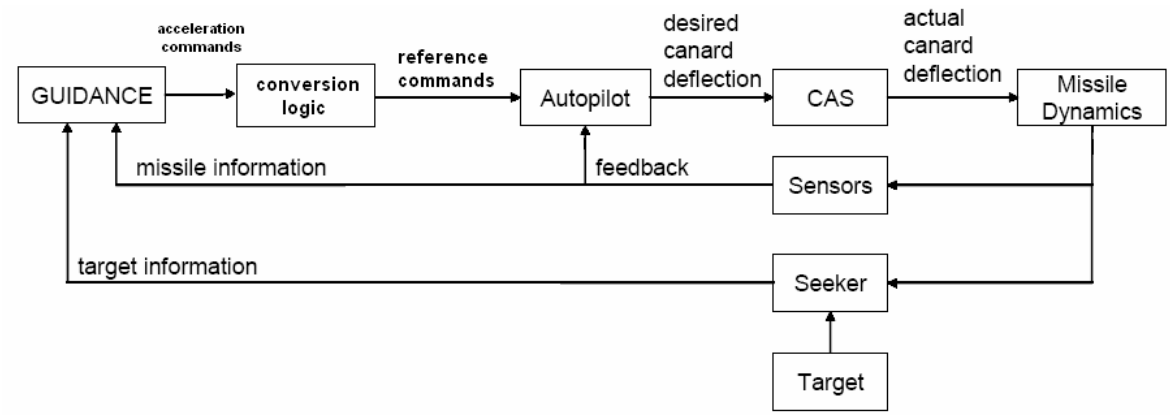

Figure 36 Guidance loop

Here, different guidance algorithms implemented in the thesis are introduced. As well as the traditional Proportional Navigation (PN) guidance methodology, Sliding Mode (SM) guidance (due to its robust nature) and an Optimal Guidance (OG) approach are constructed. A new guidance algorithm, namely "Optimal Proportional Integral (OPI) guidance" is developed and compared with the other methods. Conversion logic used in literature is adapted to the overall system and clearly explained here for convenience.

4.1 Proportional Navigation (PN) Guidance

Theoretically, PN guidance law issues acceleration command, perpendicular to the instantaneous missile-target line of sight, which is proportional to the line of sight (LOS) rate and closing velocity [23]. Mathematically, the guidance law can be stated as

$$n_c = N V_c \dot{\lambda} \tag{4.1}$$

where n_c is the normal acceleration command, N is a unit less, designer chosen gain (usually in the range of 3-5) known as the effective navigation ratio, V_c is the missile-target closing velocity, and λ is the LOS angle.

In tactical radar homing missiles using PN guidance, the seeker provides an effective measurement of the LOS rate, and a Doppler radar provides closing velocity information. In tactical IR missile applications of PN, LOS rate is measured; whereas the closing velocity, required by the guidance law is guesstimated. In tactical missiles, PN commands are usually implemented by moving fins or other control surfaces to obtain the required lift. On the other hand, strategic interceptors use thrust vector control, lateral divert engines, or squibs to achieve the desired acceleration levels.

To better understand how PN works, consider two-dimensional case, point mass missile-target engagement geometry as in Figure 37.

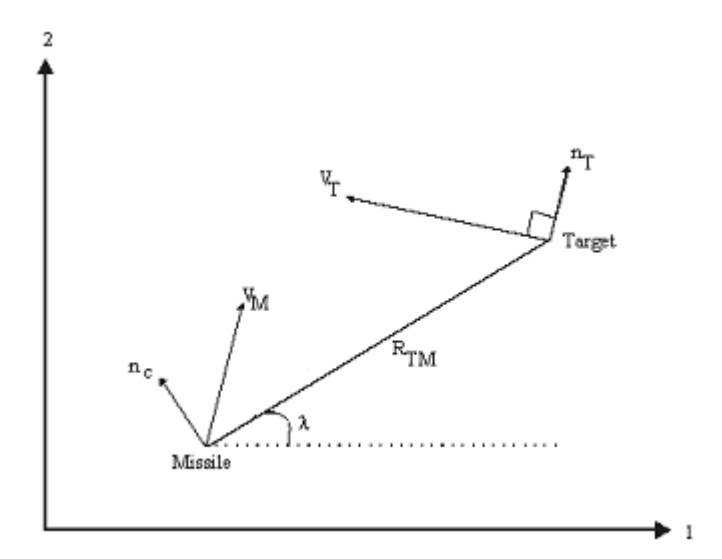

Figure 37 Two-dimensional missile-target engagement geometry

In the derivation of relative kinematics between missile and target in inertial reference frame, x-axis of the inertial frame is taken as aligned with the initial LOS direction [24]. Thus, inertial azimuth and elevation LOS angles are expressed respectively as

$$\lambda_{AZ} = \arcsin\left(\frac{Y_r}{R_{TM}}\right) \tag{4.2}$$

$$\lambda_{EL} = \arcsin\left(\frac{Z_r}{R_{TM}}\right) \tag{4.3}$$

where Y_r and Z_r denote the relative position errors and R_{TM} stands for target-to-missile range that can be expressed as

$$R_{TM} = \sqrt{X_r^{\ 2} + Y_r^{\ 2} + Z_r^{\ 2}} \tag{4.4}$$

In order to use equation (4.1) to compute the accelerations in a nonrotating fixed frame, V_c should be clarified as

$$V_c = -\dot{R}_{TM} \tag{4.5}$$

Once the equalities above are given, the guidance commands in the horizontal and vertical frames can be stated respectively as

$$a_H^{\ c} = N\,\dot{\lambda}_{AZ}\,V_c \tag{4.6}$$

$$a_V^{\ c} = N\,\dot{\lambda}_{EL}\,V_c \tag{4.7}$$

Effective navigation ratio N is selected to be equal to 4 during the simulations in this study.

4.1.1 Zero-Effort-Miss (ZEM) Concept

For the terminal engagement, PN guidance may be implemented using the concept of ZEM. The concept of ZEM is not only useful in explaining PN but it is also useful in deriving and understanding more advanced guidance laws. ZEM is defined to be the distance the missile would miss the target if the target continued along its present course and the missile made no further corrective maneuvers [23]. The ZEM can be expressed in terms of the previously defined relative quantities as

$$M_y^{ZEM} = Y_r + V_y t_{go} + 0.5 a_y^T t_{go}^2 \tag{4.8}$$

$$M_z^{ZEM} = Z_r + V_z t_{go} + 0.5 a_z^T t_{go}^2 \tag{4.9}$$

where V_y and V_z denote the relative velocity errors and a_y^T and a_z^T stands for the components of the acceleration of the target in the nonrotating frame. t_{go} is time to go until intercept and is assumed to be $t_{go} = t_f - t$ and t_f denotes the flight time. For the terminal phase, the flight time may be approximated by $t_f = R/V_c$ where R is initial range. Then, the guidance commands in the horizontal and vertical frames can be computed as given below instead of using the equations (4.6) and (4.7)

$$a_H^c = N\left(1/t_{go}^2\right) M_y^{ZEM} \tag{4.10}$$

$$a_V^c = N\left(1/t_{go}^2\right) M_z^{ZEM} \tag{4.11}$$

4.1.2 Other Forms of PN Guidance

4.1.2.1 Augmented PN (APN) Guidance

If the target maneuvers, the ZEM must be augmented by an additional term. The new guidance law, in the presence of target maneuver, would be

$$a_H^c = N\left(1/t_{go}^2\right) M_y^{ZEM} + 0.5 N a_y^T \tag{4.12}$$

$$a_V^c = N\left(1/t_{go}^2\right) M_z^{ZEM} + 0.5 N a_z^T \tag{4.13}$$

Due to the information concerning target maneuver, APN use up less acceleration capability than PN and in addition, it has monotonically decreasing acceleration history whereas PN does not [25].

4.1.2.2 Biased PN (BPN) Guidance

Irrespective of guidance laws, the controllability of the guided missile system is important in the guidance of missiles, because the loss of controllability, which occurs when the missile acceleration is near zero, will cause an unstable or abrupt change of roll angle. A method to avoid the discontinuous or unstable motion of roll angle is to use an additional bias term in the acceleration command [26]. BPN guidance was shown to have the optimality in the sense that it minimizes the integral of the square of the missile acceleration while producing zero miss distance and zero impact angle error. The new guidance acceleration commands will be

$$a_H{}^c = N\left(1/t_{go}{}^2\right)M_y{}^{ZEM} + \eta\left(1/t_{go}{}^2\right)Y_r \tag{4.14}$$

$$a_V{}^c = N\left(1/t_{go}{}^2\right)M_z{}^{ZEM} + \eta\left(1/t_{go}{}^2\right)Z_r \tag{4.15}$$

where the bias term η is a user-selective constant.

4.1.2.3 Modified PN (MPN) Guidance

Classical PN rule is a subject of the LOS rate. However, when the distance from target-to-missile is very big, or the target is in static mode and the distance from missile to target is not changing, derivative of the LOS angle is meaningless. Thus, LOS angle can be added to the PN rule as below [27]

$$a_H{}^c = N\,\dot{\lambda}_{AZ}\,V_c + K_M\,\lambda_{AZ} \tag{4.16}$$

$$a_V{}^c = N\,\dot{\lambda}_{EL}\,V_c + K_M\,\lambda_{EL} \tag{4.17}$$

where K_M is the LOS angle constant.

Although other representative forms of PN guidance are introduced here, only PN guidance algorithm is selected as the reference guidance method against the other guidance methods formed in the remaining parts of this chapter.

4.2 State-Space Formulation of Intercept Problem

In order to use modern control theory results in the application of missile guidance problem, missile-target interception state-space formulation should be constructed. Although interception problem reduces into two separate planes of motion, namely azimuth and elevation planes, only the results of the elevation plane equations will be given here because derivation in the azimuth plane is similar. Noting that inertial elevation LOS angle is expressed as in the equation (4.3) and differentiating once with respect to time produces the inertial elevation LOS rate ($\dot{\lambda}_{EL}$) as

$$\dot{\lambda}_{EL} = \frac{V_z R_{TM} + Z_r V_c}{R_{TM}\sqrt{R_{TM}^{\ 2} - Z_r^{\ 2}}} \tag{4.18}$$

where V_z represents relative velocity error in z-axis. Differentiating equation (4.18) once and using small angle approximation for equation (4.3) to formulate the intercept in state-space representation produces

$$\ddot{\lambda}_{EL} \cong \frac{1}{R_{TM}}\left(-\ddot{R}_{TM}\,\lambda_{EL} - 2\dot{R}_{TM}\,\dot{\lambda}_{EL} + a_z^{\ T} - a_V^{\ c} \right) \tag{4.19}$$

Taking λ_{EL} and $\dot{\lambda}_{EL}$ as state variables, missile-target engagement can be formulated in state space representation as [28]

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\dfrac{\ddot{R}_{TM}}{R_{TM}} & -\dfrac{2\dot{R}_{TM}}{R_{TM}} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ -\dfrac{1}{R_{TM}} \end{bmatrix} u(t) + \begin{bmatrix} 0 \\ \dfrac{1}{R_{TM}} \end{bmatrix} f(t) \tag{4.20}$$

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) - B(t)f(t)$$

where $x_1 = \lambda_{EL}$ and $x_2 = \dot{\lambda}_{EL}$ are the state variables, $u = a_V{}^c$ and $f = a_z{}^T$ are respectively, the control variable and the disturbance in system (4.20). After forming the above state space interception model, control theory results can be applied easily. However, due to the existence of disturbance term, i.e. unknown target acceleration, controllers that have robust properties inside should be preferred for missile guidance design.

4.3 Sliding Mode (SM) Guidance

The main disadvantage of the different approaches of the guidance problem, from the implementation point of view, is that they require the information of relative range, velocity and even target's acceleration. In fact, R_{TM} and $\dot{R}_{TM}$ cannot be measured or estimated accurately, so these are two perturbation parameters. In addition, if the target performs a maneuver normal to LOS, then its acceleration constitutes a disturbance. Thus, for intercepting a maneuvering target, an effective way is to design a robust guidance law, which is insensitive to parameter perturbations and external disturbances. A new form of guidance law based on SM control theory can be used to overcome robustness problems stated above. Remember the fact that, SM control systems are distinguished by their robustness properties against a class of bounded disturbances. Before going in deep to design formulation, assumptions made during the foundation of SM guidance should be stated here for convenience. Missile is supposed to have a speed advantage over the target. Furthermore, target acceleration, a_T is assumed to be bounded with a known bound on its magnitude. Under these assumptions, SM control theory is applicable to missile guidance design [29], [30], [28], [31].

To apply SM control to the guidance law design, a switching surface that represents the desired system dynamics and the aim in guidance logic is to be chosen. This selection is crucial because the structure of the guidance law and its robustness properties are very much dependent on it. The basic idea behind the choice of the switching surface is surely to null the LOS rate. Suppose that we are able to achieve $\dot{\lambda} = 0$ by a suitable choice of control. Then, together with the assumption of $v_M > v_T$, $\dot{R}_{TM} < 0$ can be achieved, thus guaranteeing interception. Note here that, nulling the LOS rate is also the objective of basic PN law and PN achieves it for nonmaneuvering targets and constant-speed missiles, by making the missile heading rate proportional to the LOS rate. Eventually, switching surface should be selected as

$$s = \dot{\lambda} \tag{4.21}$$

Having defined the switching surface, next step is to design a control law that will guarantee attractivity of the surface $s = 0$ and sliding along $s = 0$, i.e., reaching phase controller. In order to achieve this, Lyapunov stability method is used. The Lyapunov function is chosen as one half times the square of the LOS rate. The applicability of the law is determined by negative definiteness of derivative of the Lyapunov function. For the elevation plane, inequality below is obtained

$$\dot{V} = s\,\dot{s} = \frac{\dot{\lambda}_{EL}}{R_{TM}}\left\{ -\ddot{R}_{TM}\,\lambda_{EL} - 2\dot{R}_{TM}\,\dot{\lambda}_{EL} + a_z^{\,T} - a_V^{\,c} \right\} < 0 \tag{4.22}$$

By using the idea that the constant plus proportional rate reaching law approach to synthesize a sliding mode control system can achieve good dynamical performance, a choice of control that will ensure the equality (4.22) is suggested to be

$$a_V^{\,c} = -\ddot{R}_{TM}\,\lambda_{EL} - 2\dot{R}_{TM}\,\dot{\lambda}_{EL} + K\,\dot{\lambda}_{EL} + Q\,\mathrm{sgn}\,(\dot{\lambda}_{EL}) \tag{4.23}$$

where $Q \geq a_z^{T}{}_{\max} + \mu$ and $\mu, K > 0$. Substituting equation (4.23) into (4.22)

$$\dot{V} = \frac{\dot{\lambda}_{EL}}{R_{TM}} \left\{ -K\,\dot{\lambda}_{EL} - Q\,\mathrm{sgn}\,(\dot{\lambda}_{EL}) + a_z^{T} \right\} < \ldots$$

$$\ldots < \frac{\dot{\lambda}_{EL}}{R_{TM}} \left\{ -K\,\dot{\lambda}_{EL} - \mu\,\mathrm{sgn}\,(\dot{\lambda}_{EL}) \right\} = -K \underbrace{\frac{\left(\dot{\lambda}_{EL}\right)^2}{R_{TM}}}_{+} - \mu \underbrace{\frac{\dot{\lambda}_{EL}\ \mathrm{sgn}\,(\dot{\lambda}_{EL})}{R_{TM}}}_{+} < 0 \tag{4.24}$$

Thus the choice of $a_V^{\ c}$ given by equation (4.23) guarantees the attractiveness of the switching surface $s = 0$. A feasible choice for K and μ by adding to the guidance law time varying and/or adaptive nature can be as in the equations below

$$K = -K'\,\dot{R}_{TM} \tag{4.25}$$

$$\mu = -K'\,\rho\,\dot{R}_{TM} \tag{4.26}$$

In whole guidance period, the variation of $\dot{R}_{TM}$ is not large and can be approximated as a constant. Moreover, SM guidance is robust to not only target maneuvers, but also the system parameters' variations. By using this property, $\ddot{R}_{TM}$ term in equation (4.23) can be dropped. With the choice in equations (4.25) and (4.26), the guidance law can be rewritten as

$$a_V^{\ c} = -N\,\dot{R}_{TM}\,\dot{\lambda}_{EL} + Q\,\mathrm{sgn}\,(\dot{\lambda}_{EL}) \tag{4.27}$$

where $Q \geq a_z^{T}{}_{\max} - K'\,\rho\,\dot{R}_{TM}$ and $N = K' + 2$. It is assumed here that the initial conditions are such that the guidance law guarantees $\dot{R}_{TM} < 0$ during the reaching phase.

It is well known that disturbance rejection properties of guidance law are limited to the steady state value of the sliding phase and not the reaching phase. In order to guarantee robustness, system must be quickly attracted to the switching surface and maintained there. As the linear gain N is effective only during the reaching phase, it can be chosen such that it will not result in excessive missile acceleration. The switching gain ρ is important in the sense that it forces $\dot{\lambda}$ trajectories onto the switching surface and maintain the attractivity during the sliding mode. Theoretically, the larger the ρ, the more robust is the guidance law with respect to the unmodeled dynamics and the smaller is the reaching time. However, because of the presence of nonidealities, ideal sliding mode is not guaranteed, $\dot{\lambda}$ will stay within some boundary layer of the sliding surface. The presence of $\operatorname{sgn}(\dot{\lambda})$ term in the guidance law leads to chattering, whose amplitude is dependent to Q. In order to reduce the amplitude of the oscillations, it is necessary to choose as small value of ρ as possible in the sliding mode. Regarding all of these facts, the gain of the switching term, Q should be large enough to result in faster reaching without exceeding the limiting acceleration requirements while in the reaching phase and also small enough to not result in excessive chattering in the sliding mode [29].

The chattering can also be eliminated or reduced by replacing $\operatorname{sgn}(\dot{\lambda})$ with a continuous approximation, like the high gain saturation function (during guidance design process) [17]. The most simple saturation control is described by

$$u(s) = sat(s) = \begin{cases} +1 & when \quad s > L \\ \dfrac{s}{L} & when \quad |s| \le L \\ -1 & when \quad s < -L \end{cases} \tag{4.28}$$

where $L > 0$ and $\pm L$ defines the thresholds for entering the boundary layer. Outside the boundary layer, the control is identical to the ideal relay characteristic. Within the boundary layer, however, the control is a high gain, linear control.

Final form and important performance parameters of SM Guidance selected for vertical acceleration command to be used in simulations in the next chapter will be

$$a_c = (K'+2)V_c \dot{\lambda} + \left(a_z{}^T{}_{\max} + K'\rho V_c + \beta\right) sat(\dot{\lambda}) \tag{4.29}$$

$K' = 2$

$\rho = 0.01$

$\beta \geq 0$

Boundary layer threshold in the saturation function: $L = 0.1$

4.4 An Optimal Guidance (OG) Approach

Optimal linear quadratic tracker approach can be used in guidance applications. According to this method, the performance index that will be used for tracking design is stated as follows

$$\begin{aligned} J(t_0) &= \frac{1}{2}\left(y(T) - r(T)\right)^T H\left(y(T) - r(T)\right) + \ldots \\ &\ldots + \frac{1}{2}\int_{t_0}^{T}\left(\left(y(t) - r(t)\right)^T Q\left(y(t) - r(t)\right) + u(t)^T R\, u(t)\right)dt \end{aligned} \tag{4.30}$$

$t > t_0, \quad [t_0 \;\; T] \qquad H, Q \geq 0,\ R > 0$ are symmetric matrices.

Reference signal $r(t)$ will be taken as estimated positions of the target at the next decision time instant. Performance output $y(t)$ includes the position of the missile. All of the computations will be done in the inertial frame first, and then will be converted for the body axis equivalents [32].

First, missile model should be represented in the state space demonstration as

$$\dot{x}(t) = A\,x(t) + B\,u(t) \tag{4.31}$$

$$y(t) = C\,x(t) \tag{4.32}$$

where system matrices are taken as follows

$$A=\begin{bmatrix}0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&0&1\\0&0&0&0&0&0\\0&0&0&0&0&0\\0&0&0&0&0&0\end{bmatrix}\quad B=\begin{bmatrix}0&0&0\\0&0&0\\0&0&0\\1&0&0\\0&1&0\\0&0&1\end{bmatrix}\qquad C=\begin{bmatrix}1&0&0&0&0&0\\0&1&0&0&0&0\\0&0&1&0&0&0\end{bmatrix}$$

States composed of position and velocity components of the missile in inertial frame, whereas control input is surely the acceleration commands as

$$x=\begin{bmatrix}x_m\\y_m\\z_m\\V_{x_m}\\V_{y_m}\\V_{z_m}\end{bmatrix},\qquad u=\begin{bmatrix}a_{x_m}\\a_{y_m}\\a_{z_m}\end{bmatrix}$$

Instead of using the time varying tracker approach, infinite-horizon tracker problem is preferred by letting $H=0,\ t_0=0$ and $T\to\infty$. Then the algorithm is the same as in part 3.2.2.1and stated here once again

- Solve the algebraic Riccati equation firstly and find K.

$$KA+A^TK-K\,B\,R^{-1}B^TK+C^TQ\,C=0 \tag{4.33}$$

- Next, solve for S.

$$S=-\left(A^T-K\,B\,R^{-1}B^T\right)^{-1}C^TQ\,r \tag{4.34}$$

- Compute $u(t)$ and use it in equation (4.31).

$$u = -R^{-1}B^T K\, x + R^{-1}B^T S\, r \tag{4.35}$$

Steady state tracker can be devised for a finite control interval by using the steady state gains K and S.

Missile accelerations are computed using equation (4.35). Since the missile has no thrust, a_{x_m} cannot be used. Therefore, the element of weighting matrix R about this command is taken to be large with respect to the others in order to have small acceleration commands [32]. Moreover, a_{y_m} and a_{z_m} stands for $a_H{}^c$ and $a_V{}^c$ (horizontal and vertical guidance commands), respectively.

The main disadvantage of this method is that its miss distance performance is directly affected by a change even in one diagonal element of the weighting matrices. Because this method does not have robust properties against parameter changes of the weighting matrices, it should be improved to obtain a more powerful guidance method. In this study, performance results of this approach are not included in the simulations chapter. Moreover, suggested optimal guidance approach seems to be a challenging problem and a designer should spend much more effort on the details of this methodology while using in a practical application.

4.5 New Guidance Law Derivation

Most guidance schemes in the literature are based on the principle of PN, whose logic is to null LOS rate by making the missile heading rate proportional to LOS rate while closing in on range. Indeed, PN was shown to be optimal for linearized engagement equations in constant speed missile and static target intercept scenarios. However, its performance sharply degrades in the presence of rapidly maneuvering or fast moving targets [29].

In order to overcome miss distance problems in the presence of large target maneuvers, several modifications in the form of a bias added to PN for target acceleration compensation (or estimation) are suggested in the literature. One

form of modification is the well known APN given in part 4.1.2.1, where the commanded acceleration is a linear function of target acceleration as well [23]. The main disadvantage of this method is its direct dependence on target acceleration, which is generally an unknown. Suitable analytical estimation methods can be used in determining target acceleration for the implementation of this method. However, APN guidance using such estimates resulted in poor performance in some scenarios [29]. Another modification in PN guidance is derived by using the Lyapunov stability theory [33]. In addition to the methods developed using Lyapunov theory, sliding mode control theory can be applied to extend the results to a relatively robust guidance law against disturbances and parameter perturbations. However, chattering phenomenon due to switching in sliding mode guidance is the main reason behind the limited practical scope of the sliding mode guidance law.

A new form of guidance law based on the well-known classical proportional-integral-derivative (PID) controller is derived in this part to overcome miss distance problems faced in PN guidance for fast-accelerated target models. Moreover, it is almost as simple to implement as the PN guidance law due to its dependence on LOS angle and its rates only. As a consequence of the derivations, it is recognized that the derivative term can be dropped without significantly affecting overall performance. Tuning of constructed PI structure parameters is done by the help of the optimal control theory. Linear missile-target intercept model introduced in part 4.2 is used by employing a suitable performance index depending on LOS angle, LOS rate and acceleration command. The suggested OPI guidance method has an adaptive and/or time varying structure because of the fact that it employs relative range term and its rate of change [34], [35].

4.5.1 Proportional Integral (PI) Guidance Formulation

The well-known controller design method “PID controller”, can be surely applied from a different perspective to the missile guidance problem. Let us remember that a PID controller has a transfer function of the form

$$G_c(s) = K_1 + \frac{K_2}{s} + K_3 s \tag{4.36}$$

with the control input produced as

$$u(s) = G_c(s)\, e(s) \tag{4.37}$$

where $e(s)$ stands for the error to be nullified. By using the idea in equations (4.36) and (4.37) with the rule that LOS rate is the variable to be zeroed in guidance applications, a new approach to missile guidance problem can be stated below in the vertical inertial frame acceleration command equality as

$$a_V{}^c = K_1 \dot{\lambda}_{EL} + K_2 \int \dot{\lambda}_{EL}\, dt + K_3 \frac{d}{dt} \dot{\lambda}_{EL} = \left(K_1 + \frac{K_2}{s} + K_3\, s \right) \dot{\lambda}_{EL} \tag{4.38}$$

In equation stated above, controller input surely corresponds to steering command and error is the LOS rate in order to have a hit with the target. After forming equation (4.38), some manipulations on equation (4.38) can be done by inserting equation (4.19) into it as

$$a_V{}^c = K_1 \dot{\lambda}_{EL} + K_2 \lambda_{EL} + K_3 \frac{1}{R_{TM}} \left(-\ddot{R}_{TM}\, \lambda_{EL} - 2\dot{R}_{TM}\, \dot{\lambda}_{EL} + a_z{}^T - a_V{}^c \right) \tag{4.39}$$

Grouping the similar terms in equation (4.39) and using the main common assumption that a_z^T is unknown and viewed as a disturbance, PID guidance structure becomes a practical PI guidance law as

$$a_V{}^c = \frac{K_1 - 2K_3 \dot{R}_{TM}}{1 + K_3} \dot{\lambda}_{EL} + \frac{K_2 - K_3 \dfrac{\ddot{R}_{TM}}{R_{TM}}}{1 + K_3} \lambda_{EL} \tag{4.40}$$

By using the same assumptions as in SM guidance derivation (also to test robustness of new guidance method against parameters uncertainties), equation (4.40) can be simplified as follows

$$a_V{}^c = \underbrace{\frac{K_1 - 2K_3 \dot{R}_{TM}}{1+K_3}}_{K_P} \dot{\lambda}_{EL} + \underbrace{\frac{K_2}{1+K_3}}_{K_I} \lambda_{EL} \tag{4.41}$$

4.5.2 Optimal Proportional-Integral (OPI) Guidance

After the derivation of PI guidance law presented in equation (4.41), the parameters of PI structure (K_P, K_I) will be determined by using the optimal control theory approach (using state space formulation stated in part 4.2). By using linear quadratic regulator (LQR) control methodology explained in detail in part 3.2.1, states, i.e., LOS angle and its rate, and the control input, i.e., acceleration command, can be kept small enough to have smaller miss distances and limited accelerations. Then the control input, by minimizing the performance index in equation (3.2) will be stated in the form of

$$u = a_V{}^c = -K_{LQR}\, x = -\begin{bmatrix} -K_I & -K_P \end{bmatrix} \begin{bmatrix} \lambda_{EL} \\ \dot{\lambda}_{EL} \end{bmatrix} = K_P\, \dot{\lambda}_{EL} + K_I \lambda_{EL} \tag{4.42}$$

If equations (4.41) and (4.42) are interpreted from a different point of view, OPI guidance seems to be, in fact, an optimal, systematic approach for tuning time-varying PI guidance parameters. Again, for the computation cycle of LQR, $\ddot{R}_{TM}$ term in system (4.20) can be dropped because the suggested guidance method is expected to be a robust method against parameter variations and disturbances (due to the fact that selection of parameters are done by the help of linear quadratic regulator theory results!).

4.6 Conversion Logic

Up to here, guidance unit is designed such that its outputs will be acceleration commands in inertial frame. However, pitch and roll-yaw coupled controllers designed in this thesis accept angle of attack and roll angle commands as reference signals, respectively. Thus, suitable conversion logic is needed to form autopilot commands.

Roll angle command, which is the reference signal for roll-yaw autopilot designed, is found by using a method namely "polar converting logic (PCL)" [26], [24]. Frame transforming relation used in this method of conversion is given in Figure 38.

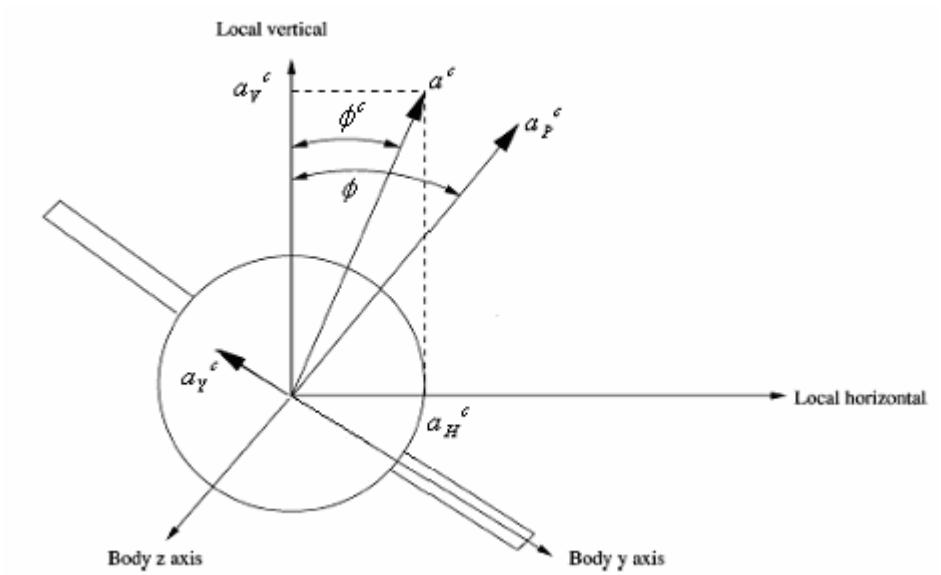

Figure 38 Frame-transforming relation

Referring to Figure 38, guidance commands in the rotating body frame can be recognized as

$$a_P^{\ c} = a_V^{\ c} \cos\phi + a_H^{\ c} \sin\phi \tag{4.43}$$

$$a_Y^{\ c} = a_V^{\ c} \sin\phi - a_H^{\ c} \cos\phi \tag{4.44}$$

The error in the roll angle command can be determined as follows

$$\phi_e^{\ c} = \tan^{-1}\left(-a_Y^{\ c} \Big/ a_P^{\ c} \right) \tag{4.45}$$

The inertial roll angle command for the roll-yaw channel is given by

$$\phi_c = \phi - \phi_e^{\ c} \tag{4.46}$$

Angle of attack command, which is the reference signal for pitch autopilot designed, can be approximated by the mathematical equality given below

$$\alpha_c = \int \left(\frac{a_P^{\ c}}{V} + q \right) \tag{4.47}$$

The integrator in the expression above adds a smoothing effect into the structure. This surely improves the performance and decreases the steady-state error. However, instead of using an open-loop expression for angle of attack command as given in equation (4.47), taking an outer acceleration loop in the controller structure (pitch autopilot) as in the figure below can be another but a closed-loop solution. However, it adds extra gain calculations (in $G_c(s)$) by using another control design methodology and the designer should perform more effort. Simulation results support the idea that open-loop conversion logic works well and thus, no need to use inner-outer loop controller structure here.

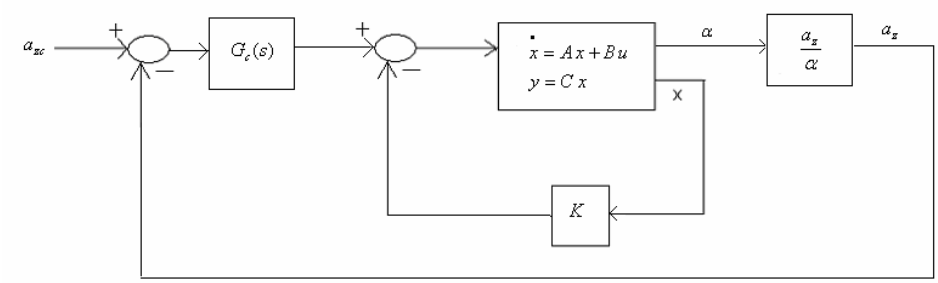

Figure 39 An alternative approach to conversion logic for pitch autopilot

Up to now, each block in Figure 36 except seeker model is analyzed one by one in this study. No seeker model is designed or used in this study. Instead, by using target position measurement data and running a Kalman filter algorithm, target states are estimated. Target's estimated position and velocity components are used in the guidance algorithms described above.

4.7 Target State Estimation

4.7.1 The Discrete Kalman Filter

The Kalman filter is a set of mathematical equations that provides an efficient computational (recursive) mean to estimate the state of a process, in a way that minimizes the mean of the squared error of state estimate. The filter is very powerful in several aspects; it supports estimations of past, present, and even future states, and it can do so even when the precise nature of the modeled system is unknown [37].

The key element in any recursive procedure is the use of the results of the previous step to aid in obtaining the desired result for the current step. This is one of the main features of Kalman filtering, and one that clearly distinguishes it from weighting function (Wiener) approach, which requires arithmetic operations on all past data [36].

Assume that target maneuver can be modeled in the form of state representation (by using simple constant velocity motion equations with accepting accelerations as disturbances to the model) as stated below

$$x_T(k+1) = A\,x_T(k) + B\,w(k) \tag{4.48}$$

$$y_T(k+1) = C\,x_T(k) + v(k) \tag{4.49}$$

where

$$A = \begin{bmatrix} 1 & 0 & 0 & T & 0 & 0 \\ 0 & 1 & 0 & 0 & T & 0 \\ 0 & 0 & 1 & 0 & 0 & T \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \qquad B = \begin{bmatrix} T^2 & 0 & 0 \\ 0 & T^2 & 0 \\ 0 & 0 & T^2 \\ T & 0 & 0 \\ 0 & T & 0 \\ 0 & 0 & T \end{bmatrix}$$

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

In the system matrices used in target motion model above, T represents sampling time, x_T stands for target states, i.e. Inertial position and velocity components and w and v represent the process noise (in this modeling selected to be unknown target acceleration) and measurement noise, respectively. Noises are assumed independent of each other, white, and with covariance data as below

$$E\left\{w(k)\,w(k)^T\right\} = Q = \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{bmatrix} \qquad E\left\{v(k)\,v(k)^T\right\} = R = \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{bmatrix}$$

By using, an initial estimate of the target states and their measurement data with the model constructed above, Kalman filter algorithm can be applied.

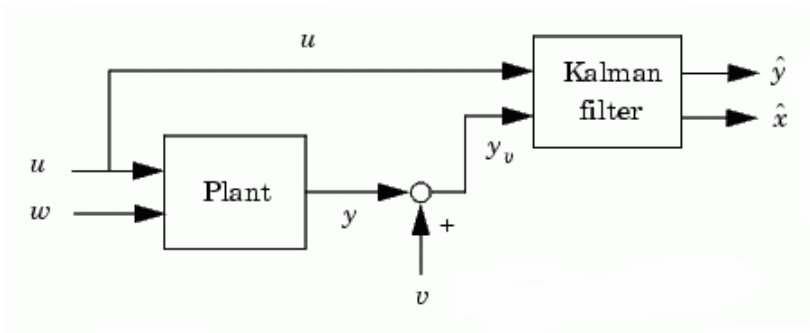

Figure 40 Kalman estimator model

In the actual implementation of the filter, the measurement noise covariance R is usually measured prior to operation of the filter. However, the determination of the process covariance Q is generally more difficult because it is impossible to directly observe the process estimated. In either case, tuning these covariance matrices offline will give better estimates [36]. In this study, both of the covariance matrices are taken equal, one represents noise on the process as unknown target acceleration whereas other is measurement noise on target position components.

CHAPTER 5

SIMULATION RESULTS

The simulations can be done in two different ways. To test and compare guidance methods presented in the previous chapter, point mass model is used for convenience. Point mass simulations will cover static, slow maneuvering and fast maneuvering target models in order to force the effectiveness of the methods and stress the accomplishment of the new method (OPI Guidance) introduced. Each of the scenarios is also tested for both unlimited and limited angle of attack ($|\alpha| \leq 8^\circ$) cases. Although unlimited case is unrealistic, it is also helpful to see how new proposed guidance method differs from traditional ones even for difficult scenarios. For testing overall system integration, OPI guidance is adapted to nonlinear environment with appropriate sliding mode autopilot structure. For all of the scenarios, PN, SM and OPI guidance techniques presented in the previous chapter, are built and compared for performance results; separately.

5.1 Simulations Created

5.1.1 Static Target Model

For this simulation, target is supposed to be a static one. Missile is fired from air and aimed to intercept with a target located on the surface. Initial properties of the missile and target is given in
Table 7 below.

Table 7 Initial positions and velocities for missile and target

	Missile	Target
Initial Position (m)	$[0 \quad 0 \quad -5000]$	$[8000 \quad 0 \quad 0]$
Initial Velocity (m/s)	$[400 \quad 0 \quad 0]$	Static

5.1.2 Slow Maneuvering Target Model

After processing the noisy position measurements of the slow maneuvering target modeled with a usual Kalman filter, actual and estimated trajectories for the target are obtained as in the Figure 41 and Figure 42 shown below.

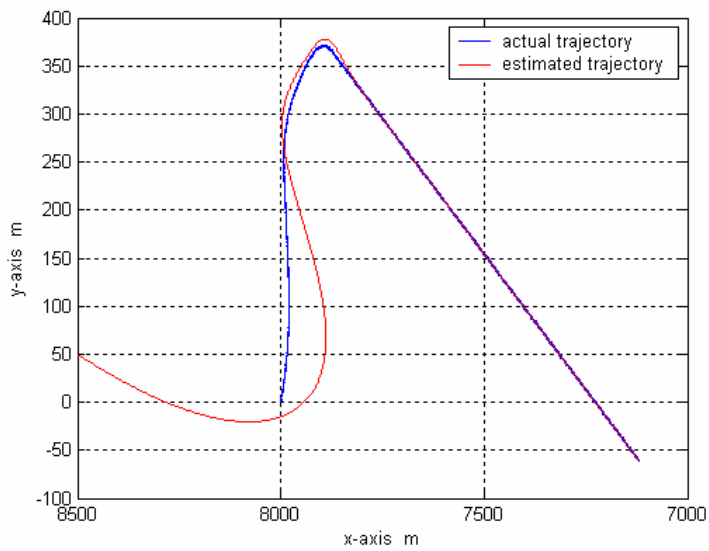

Figure 41 Inertial x-y trajectory for slow maneuvering target model

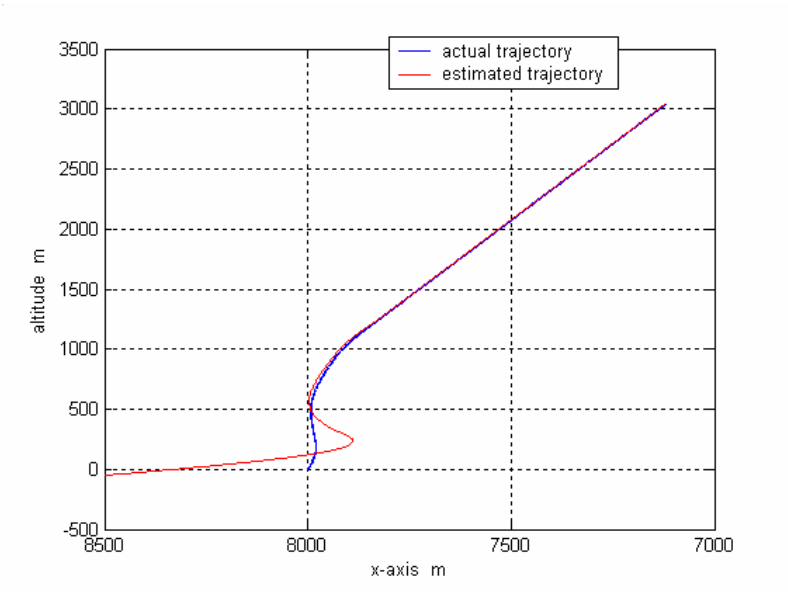

Figure 42 Inertial x-altitude trajectory for slow maneuvering target model

5.1.3 Fast Maneuvering Target Model

By running a Kalman filter algorithm once again for the fast maneuvering target model created, actual and estimated trajectories are obtained as in figures below.

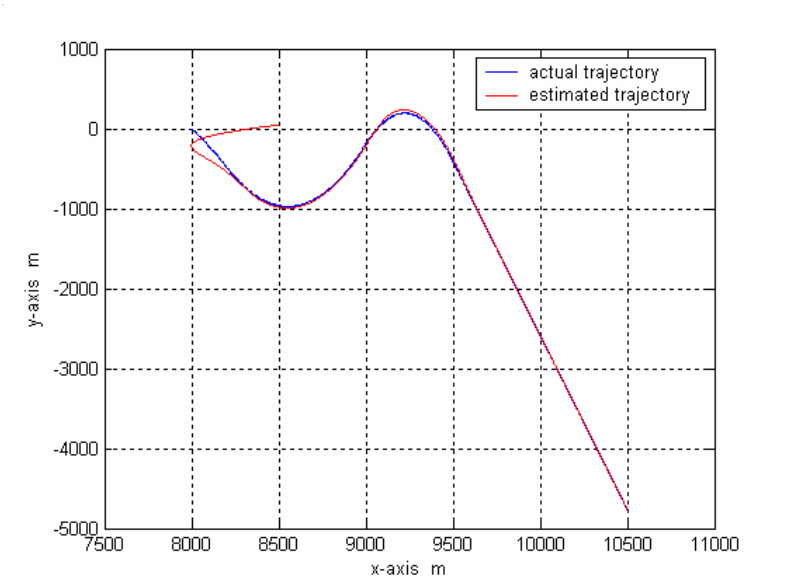

Figure 43 Inertial x-y trajectory for fast maneuvering target model

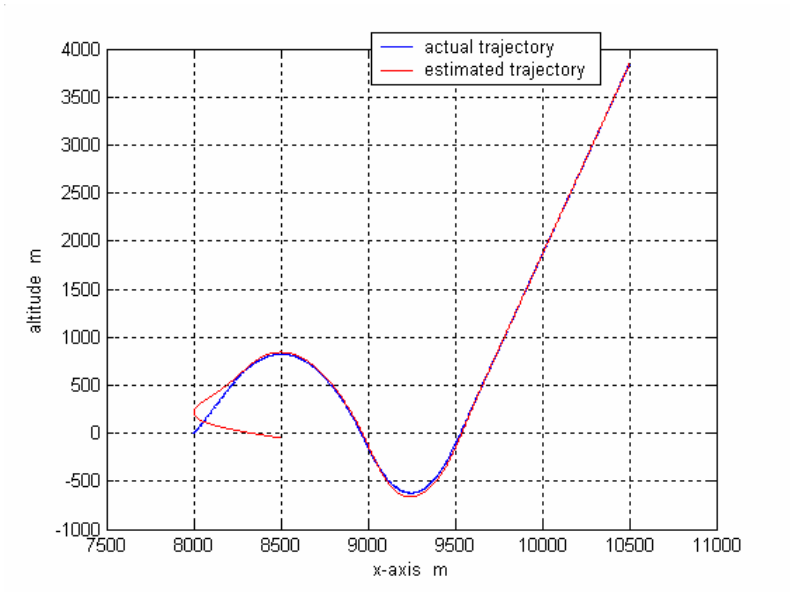

Figure 44 Inertial x-altitude trajectory for fast maneuvering target model

5.2 Point Mass Simulations

The aim of the point mass simulations is to test whether guidance algorithms result in a hit or a miss. Final miss distance values obtained are independent of autopilots performance efficiency criteria. By using point mass model for missile and target, static, slow and fast maneuvering scenarios can be simulated simultaneously.

In point mass simulations, it is assumed that controllers performances are ideal, that is to say, they track the reference signals as a perfect manner. Angle of attack is limited in an approximate way just using simple trapezoidal numerical integration technique. In addition, horizontal acceleration trajectory is supposed not to be affected from limiting angle of attack behavior.

5.2.1 Performance Results for a Point Mass Static Target

According to data given in the Table 7, following miss distance values are obtained as in the Table 8 for the static target model.

Table 8 Miss distance (m) comparison for static target model

Scenario		PN Guidance	SM Guidance	OPI Guidance
Static Target	Unlimited angle-of-attack	3.141e-008	1.195e-008	8.219e-010
	Limited angle-of-attack	19.517	9.326	19.139

Trajectories obtained for this simulation is given in the following figures (figures in the left correspond to unlimited case whereas figures in the right to limited one).

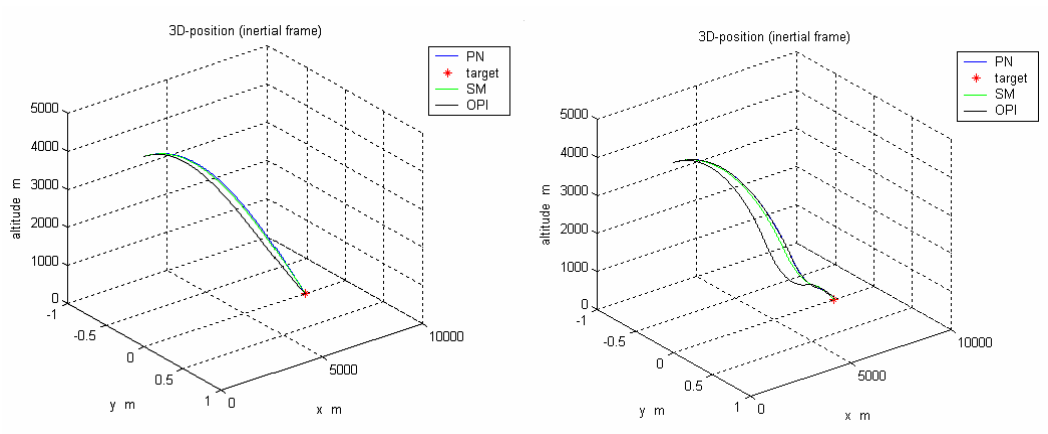

Figure 45 3D position trajectories (static)

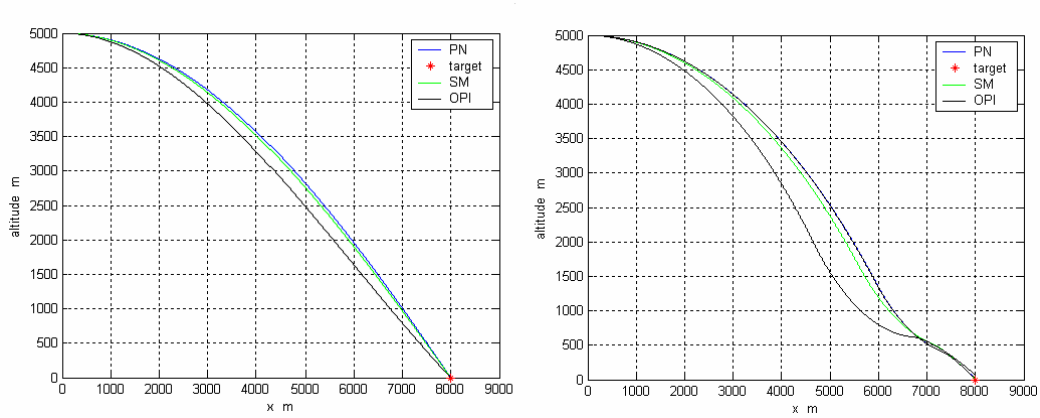

Figure 46 Inertial x-altitude trajectories (static)

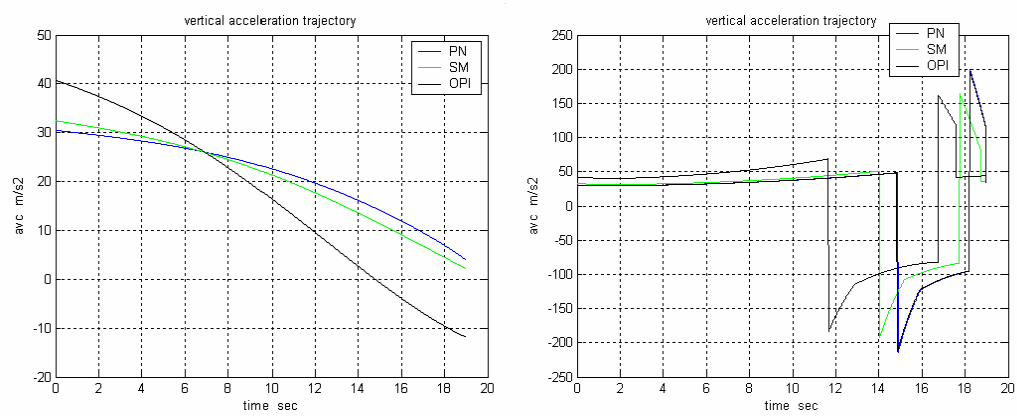

Figure 47 Vertical acceleration profiles (static)

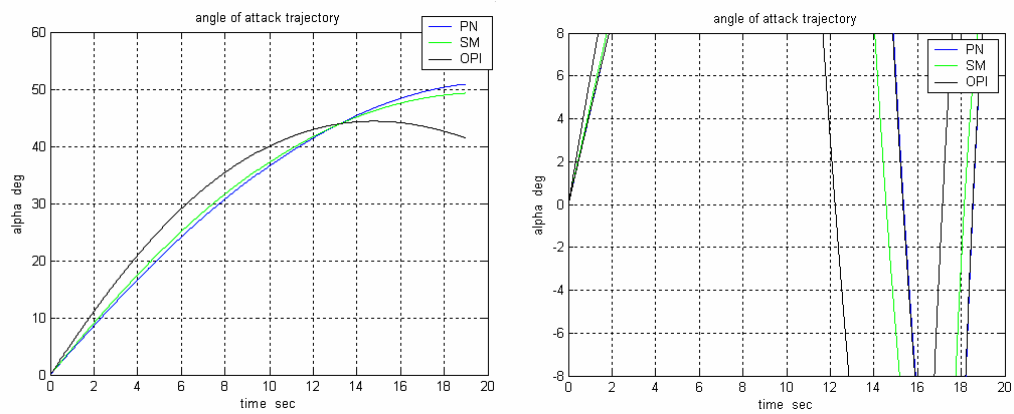

Figure 48 Angle of attack command (static)

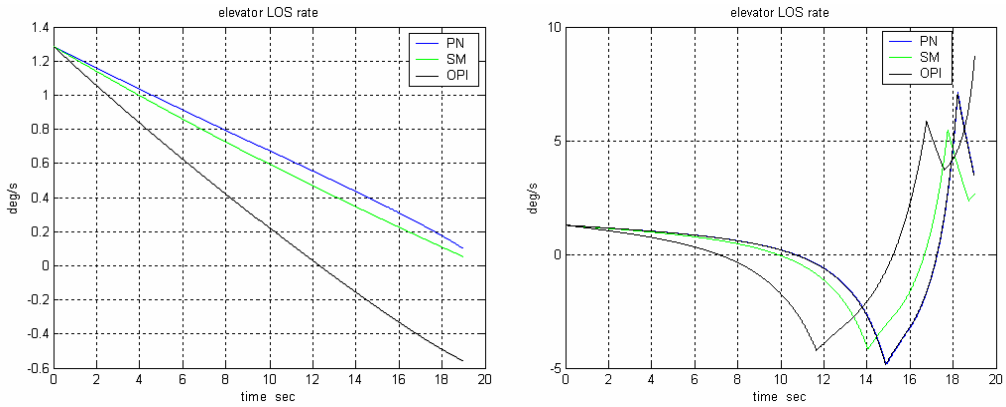

Figure 49 Elevator LOS rate profiles (static)

5.2.2 Performance Results for a Point Mass Slow Maneuvering Target

By using the estimated position components of the slow maneuvering target as can be seen in Figure 41 and Figure 42, point mass simulation for three different guidance methods can be run, again. As a result, following miss distance values are obtained.

Table 9 Miss distance (m) comparison for slow maneuvering target model

Scenario		PN Guidance	SM Guidance	OPI Guidance
Slow Maneuvering	Unlimited angle-of-attack	1.181	1.146	1.064
	Limited angle-of-attack	1.012	1.026	0.989

Trajectories obtained for this simulation is given in the following figures (figures in the left correspond to unlimited case whereas figures in the right to limited one).

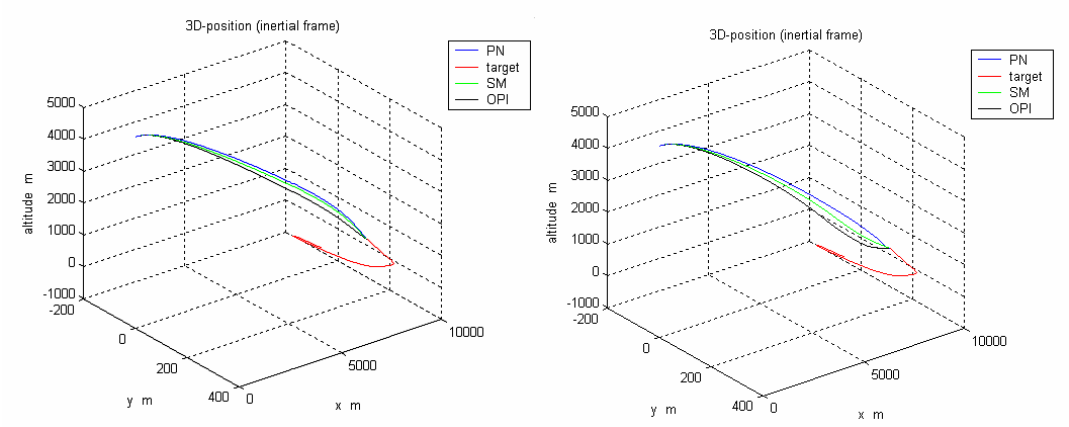

Figure 50 3D position trajectories (slow maneuvering)

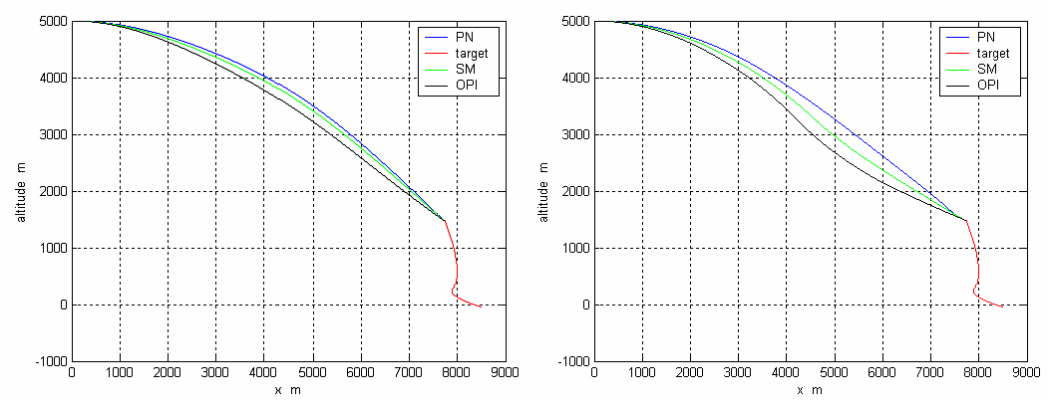

Figure 51 Inertial x-altitude trajectories (slow maneuvering)

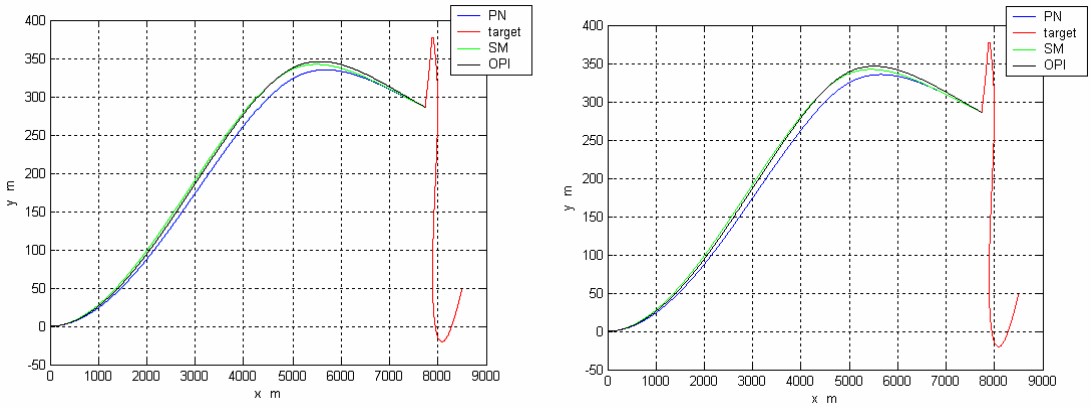

Figure 52 Inertial x-y trajectories (slow maneuvering)

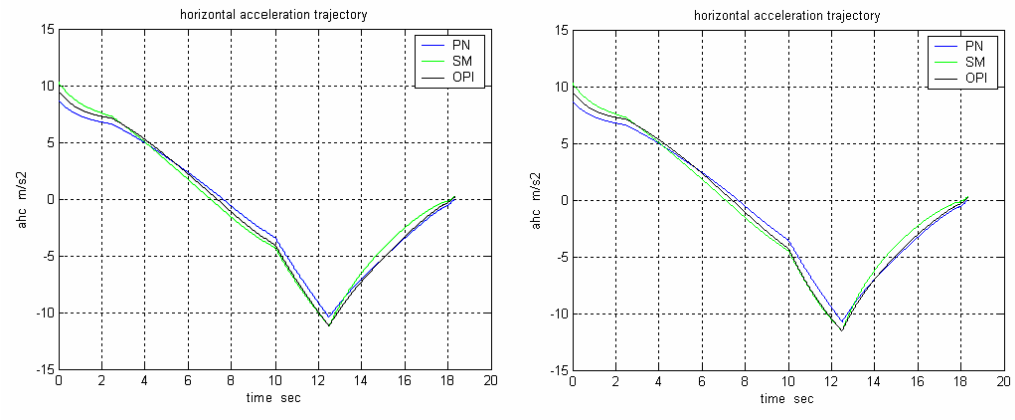

Figure 53 Horizontal acceleration profiles (slow maneuvering)

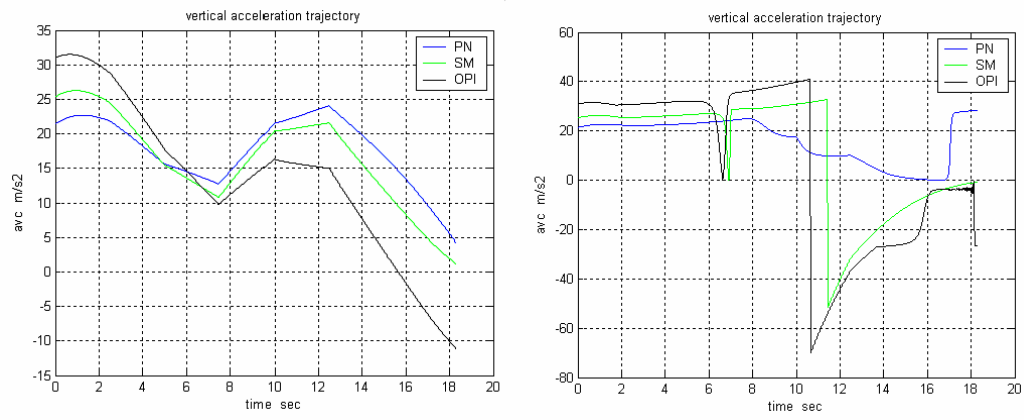

Figure 54 Vertical acceleration profiles (slow maneuvering)

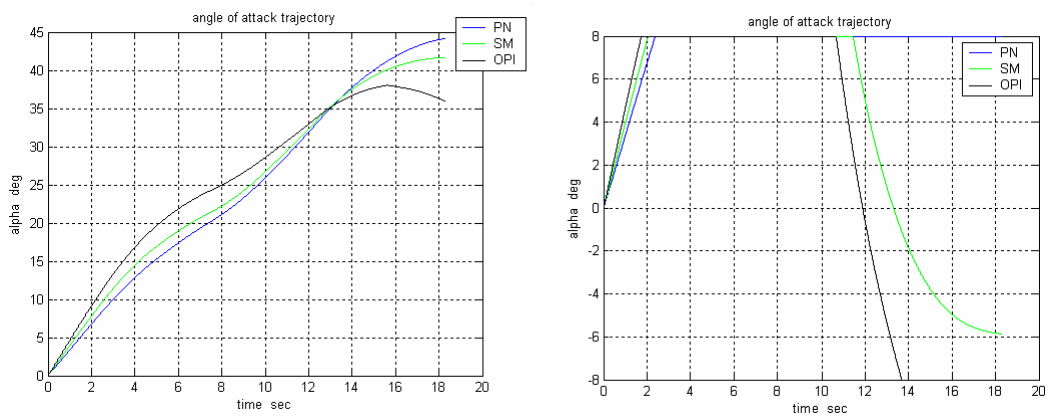

Figure 55 Angle of attack command (slow maneuvering)

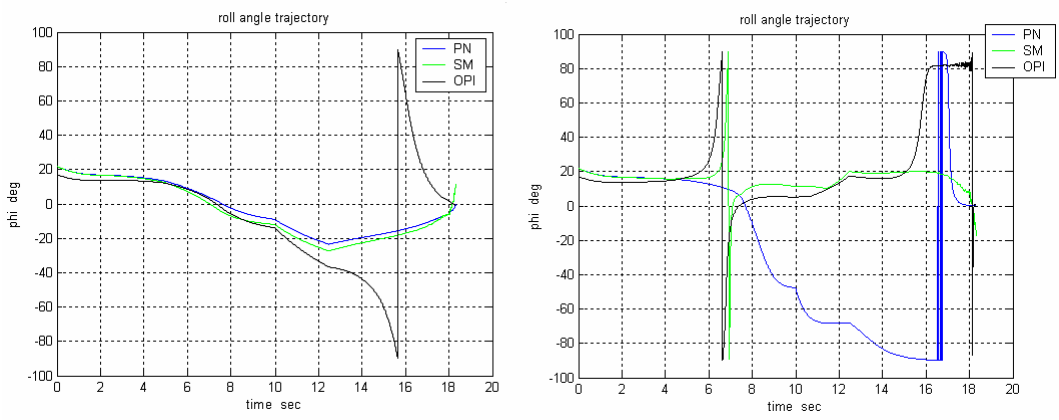

Figure 56 Roll angle command (slow maneuvering)

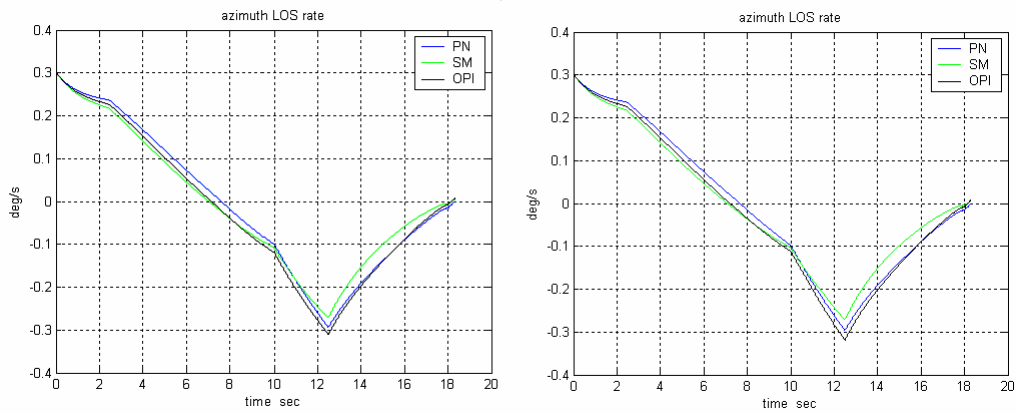

Figure 57 Azimuth LOS rate profiles (slow maneuvering)

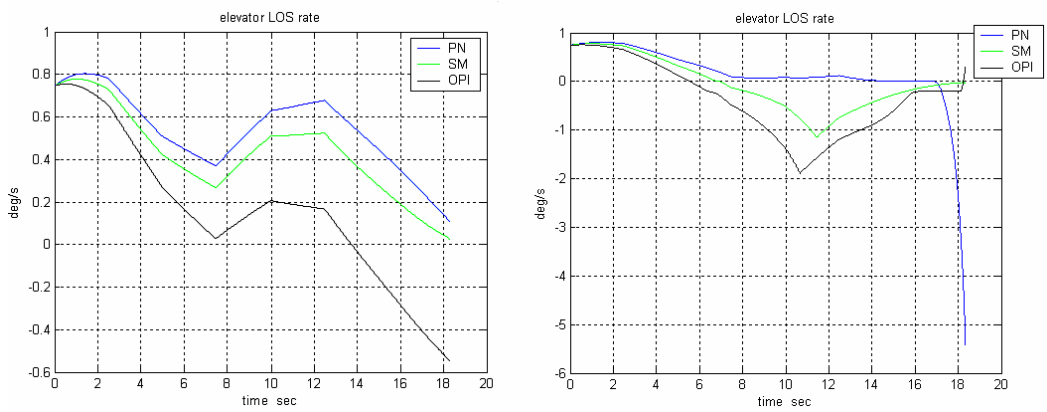

Figure 58 Elevator LOS rate profiles (slow maneuvering)

5.2.3 Performance Results for a Point Mass Fast Maneuvering Target

By using estimated position of the fast maneuvering target as can be seen in Figure 43 and Figure 44, point mass simulation for three different guidance methods can be run once again. As a result, following miss distance values and performance outputs are obtained.

Table 10 Miss distance (m) comparison for fast maneuvering target model

Scenario		PN Guidance	SM Guidance	OPI Guidance
Fast Maneuvering	Unlimited angle-of-attack	1.496	1.272	1.179
	Limited angle-of-attack	611.231 FAIL	50.847	4.096

Trajectories obtained for this simulation is given in the following figures (figures in the left correspond to unlimited case whereas figures in the right to limited one).

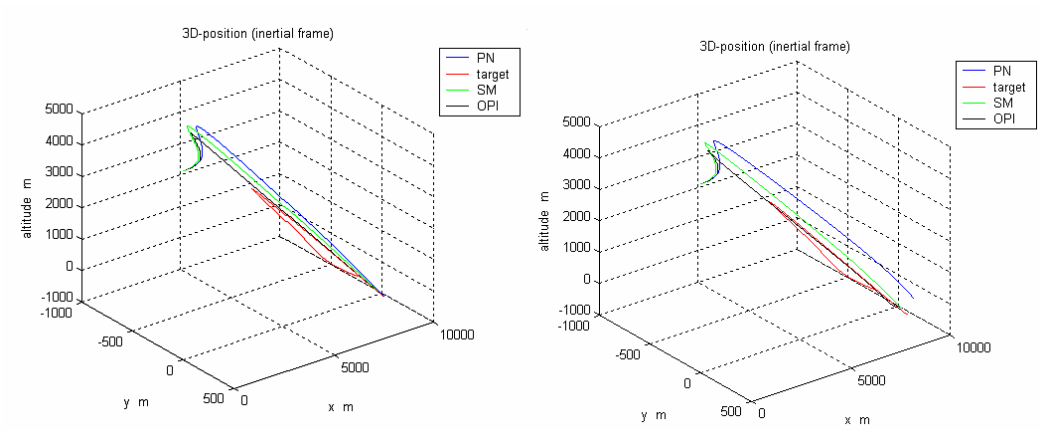

Figure 59 3D position trajectories (fast maneuvering)

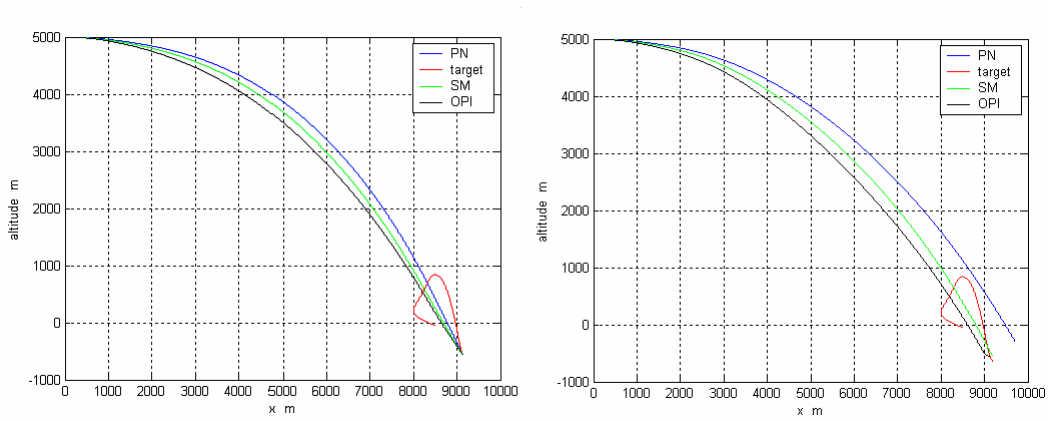

Figure 60 Inertial x-altitude trajectories (fast maneuvering)

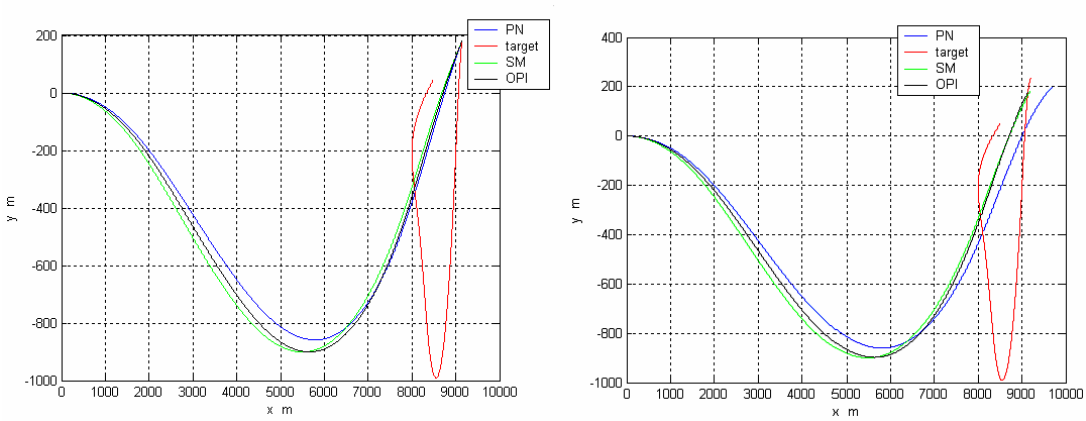

Figure 61 Inertial x-y trajectories (fast maneuvering)

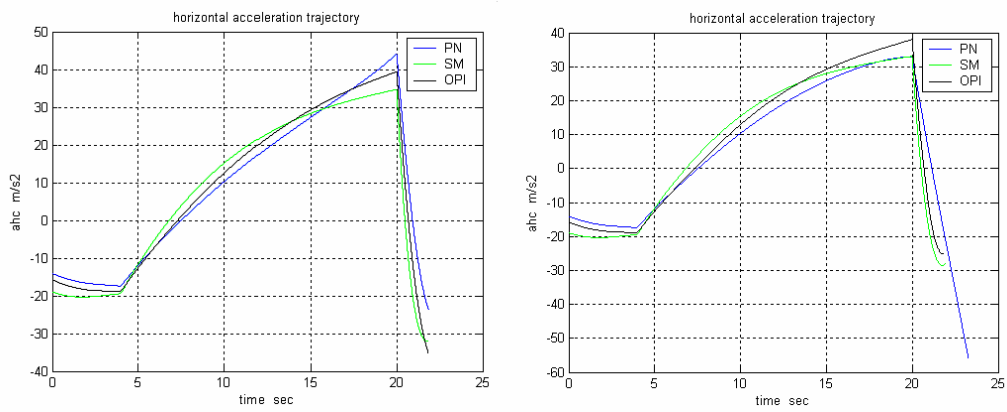

Figure 62 Horizontal acceleration profiles (fast maneuvering)

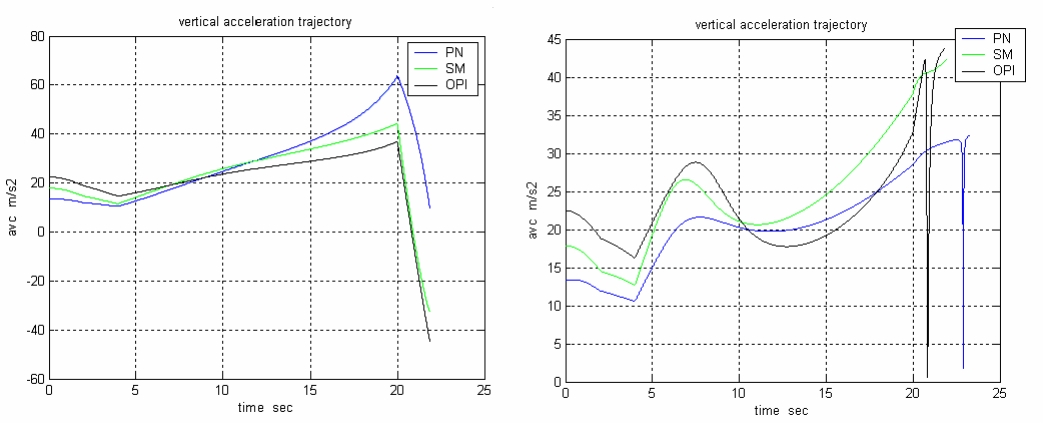

Figure 63 Vertical acceleration profiles (fast maneuvering)

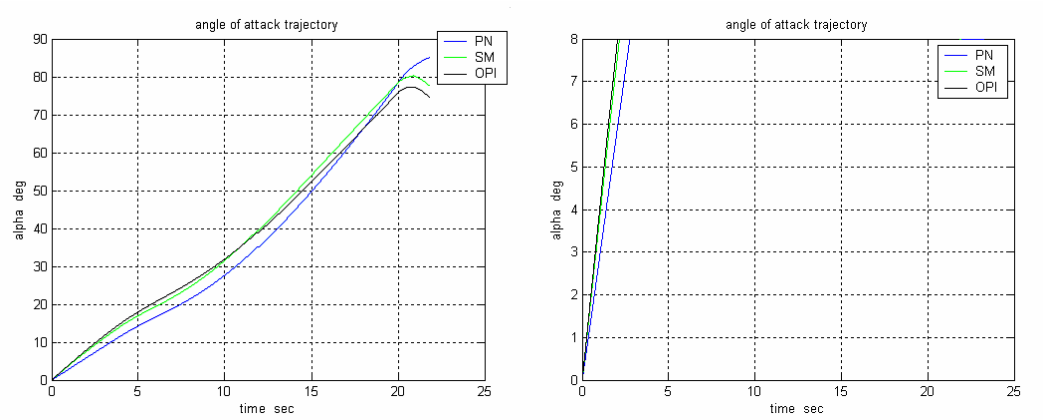

Figure 64 Angle of attack command (fast maneuvering)

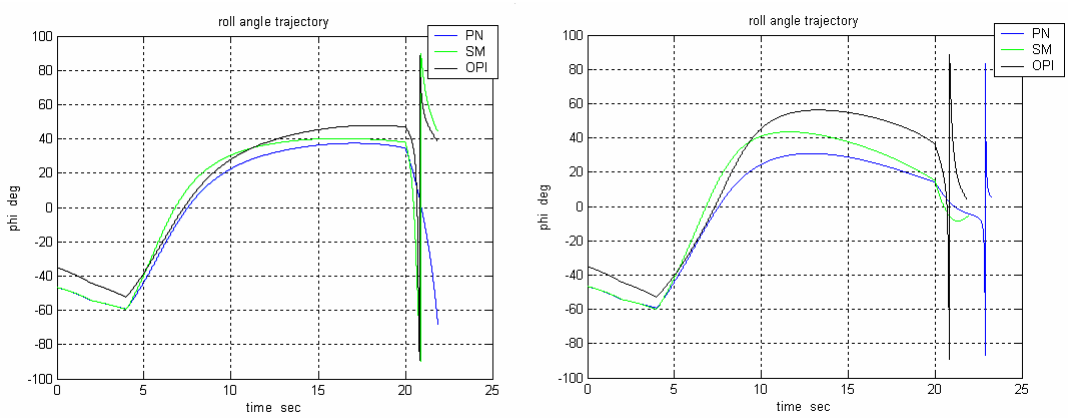

Figure 65 Roll angle command (fast maneuvering)

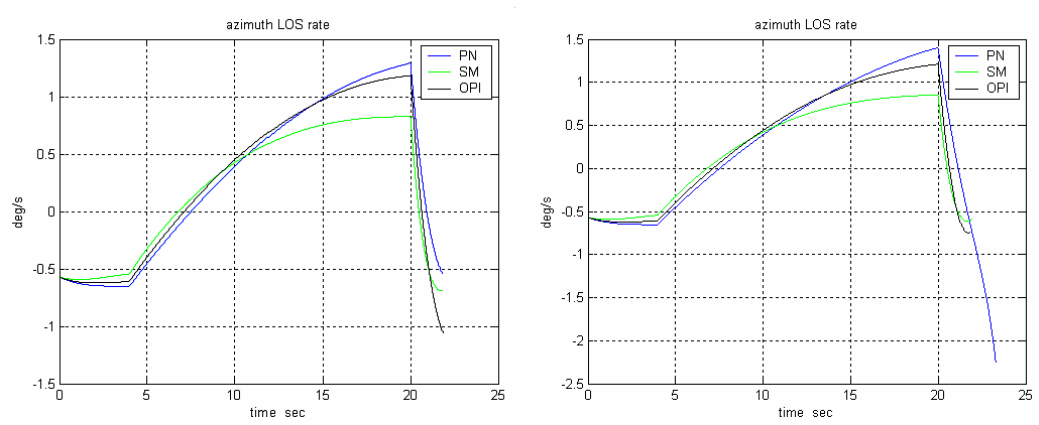

Figure 66 Azimuth LOS rate profiles (fast maneuvering)

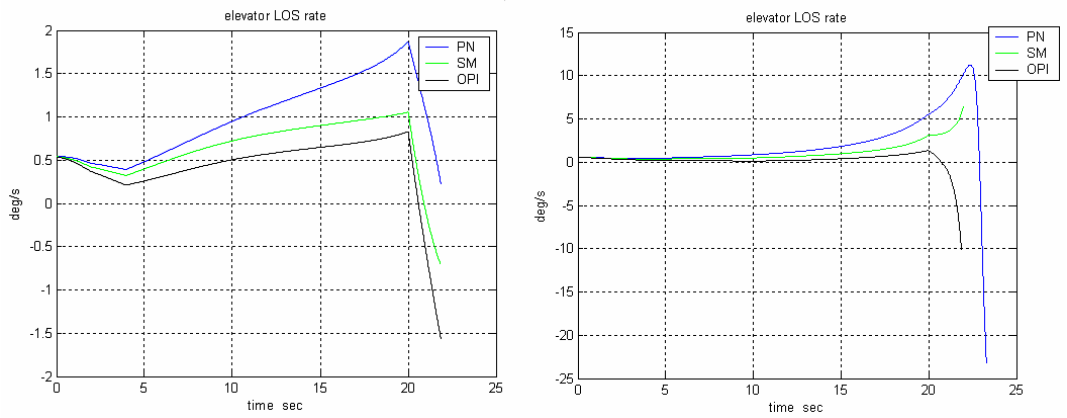

Figure 67 Elevator LOS rate profiles (fast maneuvering)

5.2.4 Comments on Point Mass Simulation Results

According to the performance results obtained from the simulations run until here, improvement in miss distance values can be easily recognized for the new OPI guidance methodology. Moreover, as can be seen from Table 10, traditional and mostly preferred guidance method “PN”, fails to hit the target for limited fast-accelerated scenario. On the other hand, missile guided with the OPI guidance strategy is able to hit the target even for this difficult scenario. Performance improvements can be because in OPI guidance method LOS rates are approximately nullified at the time of impact. Divergence of LOS rates at the time of impact for PN guidance method surely results in an increase in miss distance.

Furthermore, by adjusting parameters in the optimal cost of OPI guidance, miss distance outcome can be lowered, but this causes bigger acceleration capabilities. All of these results also support the idea that in order to have almost zero miss distance with a target, LOS rates should be minimized in the limiting range of angle of attack or acceleration.

5.3 Integrated Nonlinear Guidance-Autopilot Simulation

In order to test only the overall system integration, a nonlinear simulation setup can be formed with the new guidance method integrated with a suitable sliding mode autopilot structure. At this point, the important outcome of the simulation is whether overall system environment can work together without any integration problems. It helps also to see how much energy actuator fins will spend by performing the movements commanded by the autopilot. Nonlinear simulation environment includes guidance, autopilot, nonlinear missile model, aerodynamic model, control actuation system modules and gravitational effects. Since nonlinear simulation resembles more to real world characteristics, simulations performed here include all limiting behaviors due to the saturation of actuators and/or the physical capabilities of the missile to perform flight angles needed for interception with a static or a fast maneuvering target.

5.3.1 Performance Results for a Static Target Scenario

As the initial conditions for missile and target, data given in Table 7 are used. Sliding mode autopilot structure introduced and designed as in part 3.4 is operated together with OPI guidance derived in the previous chapter. Overall system performance results are obtained as follows.

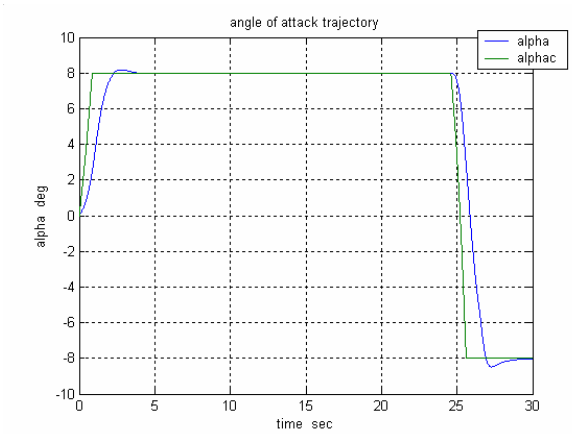

Figure 68 Angle of attack trajectory (static target, nonlinear simulation)

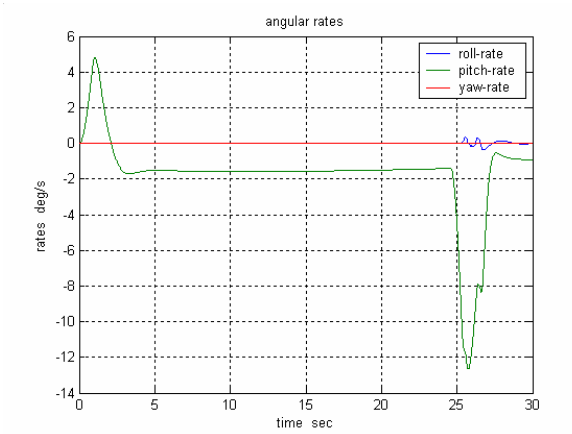

Figure 69 Angular rates (static target, nonlinear simulation)

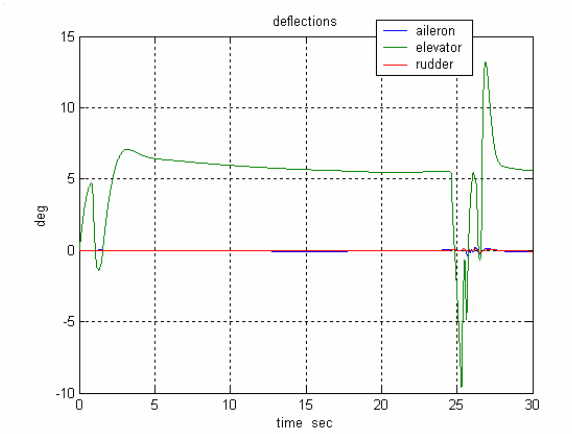

Figure 70 Control surface deflections (static target, nonlinear simulation)

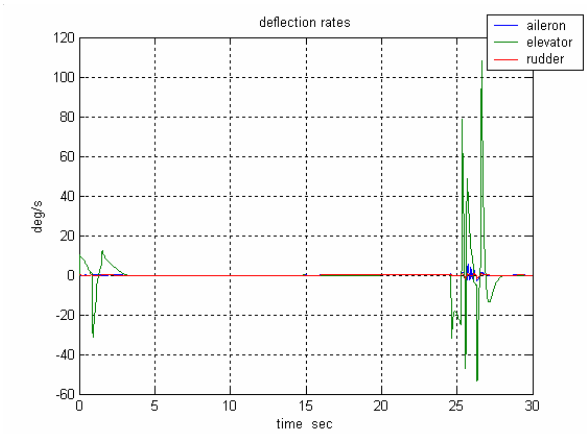

Figure 71 Deflection rates (static target, nonlinear simulation)

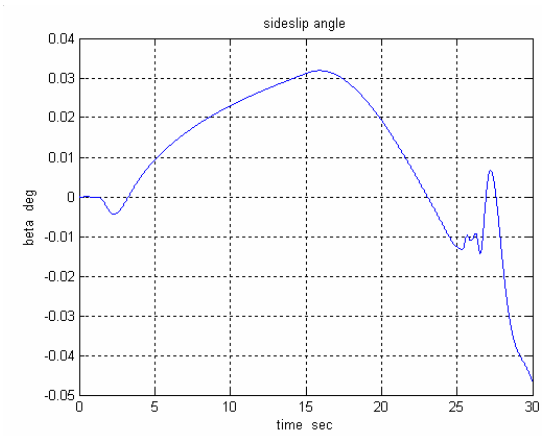

Figure 72 Sideslip angle (static target, nonlinear simulation)

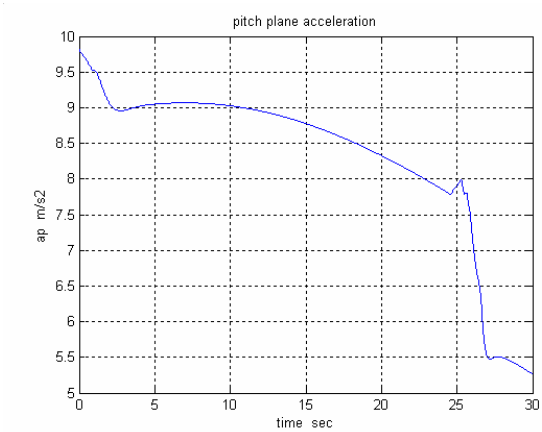

Figure 73 Acceleration in the pitch-plane (static target, nonlinear simulation)

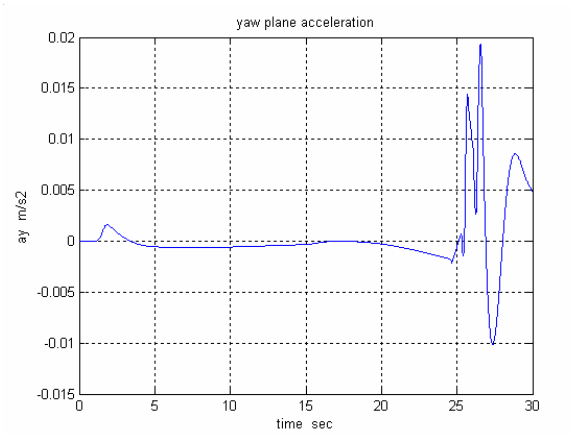

Figure 74 Acceleration in the yaw-plane (static target, nonlinear simulation)

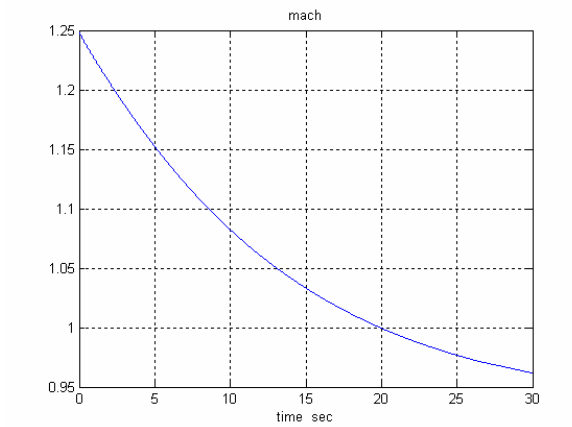

Figure 75 Mach number change (static target, nonlinear simulation)

According to the nonlinear simulation figures obtained above, following comments can be done. First, integration of overall system performs well as can be seen from the figures. Commanded angle of attack by guidance unit is tracked by the suitable autopilot structure designed. Moreover, by using small deflection energy, missile can be directed towards the target easily. As expected in this scenario, small sideslip and yaw acceleration profiles are obtained. Pitch plane acceleration is kept small which is indeed a constraint for nonlinear integrated simulations. Speed of missile is decreasing from fire to hit because of the drag force present. In order to compensate undesirable drag effects, a simple thrust model should be adapted to missile model.

5.3.2 Performance Results for a Maneuvering Target Scenario

Here fast maneuvering target scenario whose trajectories are given in part 5.1.3 is used. Missile is fired from the initial condition given as in Table 7.

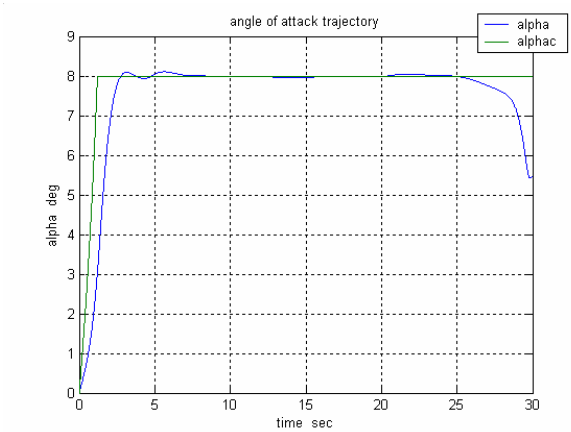

Figure 76 Angle of attack trajectory (maneuvering target, nonlinear simulation)

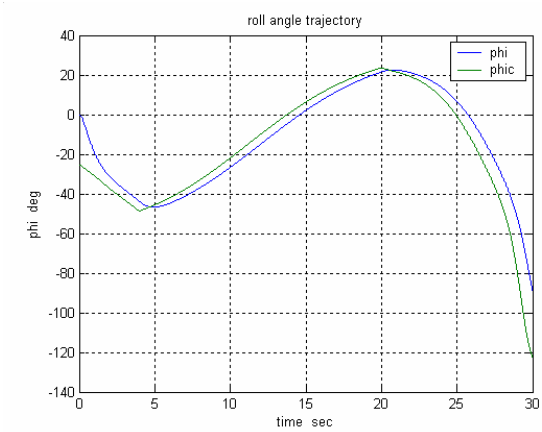

Figure 77 Roll angle trajectory (maneuvering target, nonlinear simulation)

As can be seen from Figure 76 and Figure 77, angle of attack command pitch autopilot and roll angle command roll-yaw autopilot performs good tracking features to the outputs of the guidance unit. Angle of attack is limited and saturated in 8° till the time of sudden drop in the roll angle trajectory from 20° to

-120°. At this time, angle of attack begins to fall as an expected result of the sudden change in roll behavior.

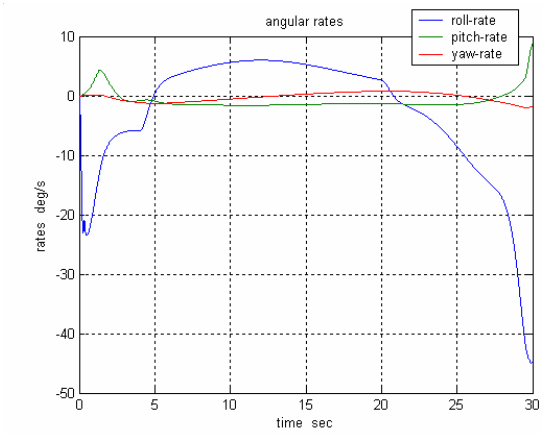

Figure 78 Angular rates (maneuvering target, nonlinear simulation)

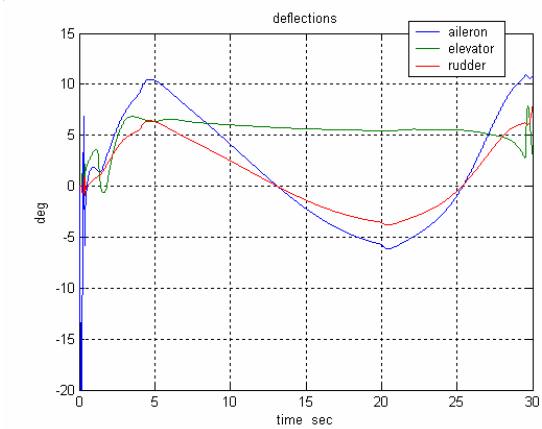

Figure 79 Control surface deflections (maneuvering target, nonlinear simulation)

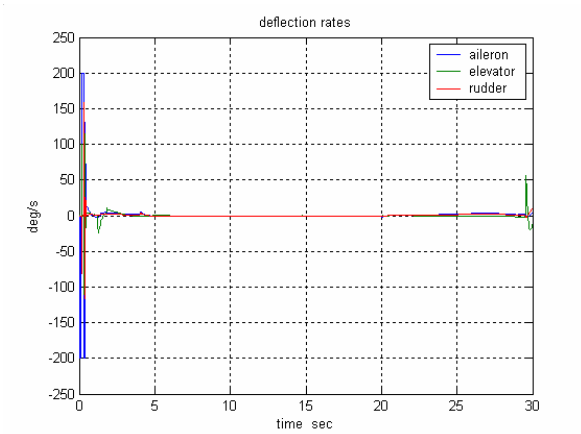

Figure 80 Deflection rates (maneuvering target, nonlinear simulation)

According to Figure 79, excessive energy spent on control surfaces can be recognized as a result of the compensation of fast target accelerations. Deflection rates are kept in the limits of $\pm 200^{\circ}/s$. However, even for that difficult interception case, deflection angles and rates stand in between the saturation limits.

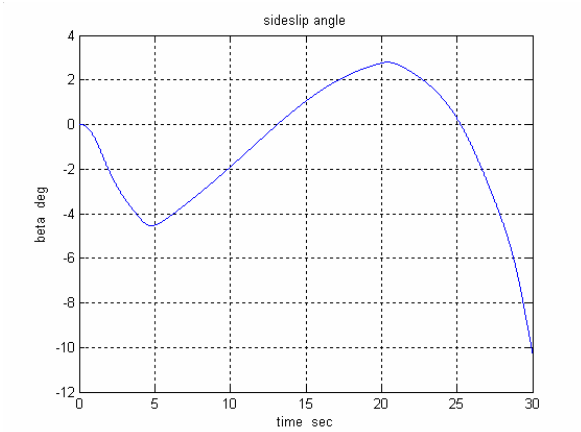

Figure 81 Sideslip angle (maneuvering target, nonlinear simulation)

Due to target x-y trajectory given as in Figure 43, sideslip angle cannot be kept in very small values as expected. In addition, sideslip angle trajectory resembles to y-range diagram resulting in keeping track of the target in the azimuth plane at every decision instant.

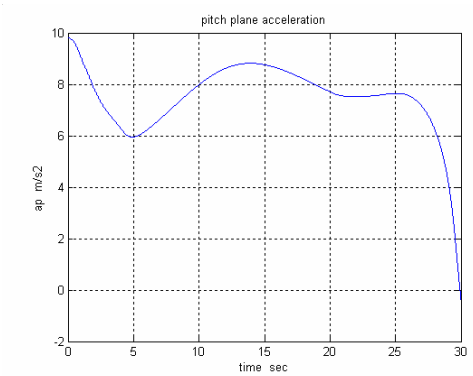

Figure 82 Acceleration in the pitch-plane (maneuvering target, nonlinear simulation)

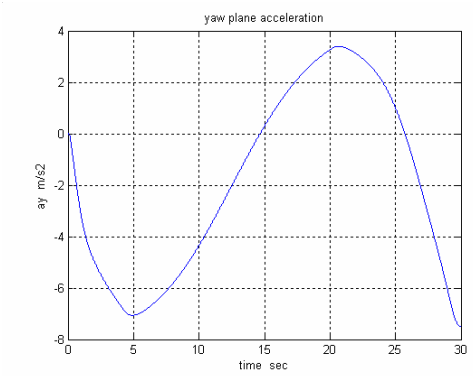

Figure 83 Acceleration in the yaw-plane (maneuvering target, nonlinear simulation)

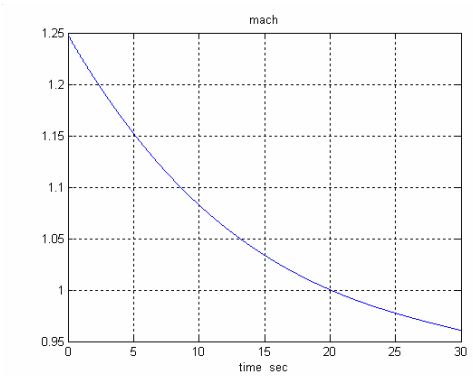

Figure 84 Mach number change (maneuvering target, nonlinear simulation)

CHAPTER 6

CONCLUSION

In this study, several autopilot and guidance algorithms are developed for a nonlinear, canard-controlled missile. First, nonlinear missile model is constructed by using generic equations of motion. These equations include aerodynamic properties inside. Aerodynamic force and moment derivatives are evaluated for different Mach number, altitude and angle of attack by using Missile DATCOM program.

Missile mathematical model constructed is decoupled into pitch and roll-yaw plane models separately by applying linearization and simplification procedures on 6-DOF equations. Optimal and robust control based methods are selected in designing the autopilot module. Linear quadratic tracker autopilot is taken as the reference controller and designed for each flight point. Gain scheduling methodology is used for joining linear controllers designed for each trim point. Nonlinear simulation environment is run with this nonlinear autopilot for different reference command signals. Among several robust control based methods, sliding mode control is selected for designing a comparable, nonlinear missile autopilot. The reasons behind this selection are its robust and invariant nature in direct controlling of nonlinear plants. Although as mentioned in literature, sliding mode control principles can be applied directly to nonlinear plants without linearization, there is no systematic rules in designing this type of a controller. Instead, sliding surfaces are designed systematically for linear models derived. Gain scheduling of sliding surfaces are built and interpolation based surfaces become the basis of a nonlinear autopilot. By using this autopilot in the nonlinear simulation program, comparison with linear quadratic tracker autopilot can be done. According to the nonlinear simulation results, sliding mode autopilot seems better than linear quadratic tracker autopilot. Tracking performances are improved and oscillation magnitudes are greatly decreased. Moreover, smaller canard

deflection angles, angular rates and sideslip angle assumptions are preserved for both of the methods. Furthermore, both of the methods are tested for robustness properties. Nonlinear simulation is performed under the effect of aerodynamic parameter variation and uniform disturbance in the feedback. According to the performance outputs, autopilots designed using both of the methods seem to be robust not only to disturbances but also to parameter changes as well.

Guidance unit is designed according to an intercept model of the missile with a target. Since the outputs of this module drive the autopilot block, they should be consistent with the reference command signals accepted by the pitch and roll-yaw controllers designed. Almost all the guidance methods in the literature give acceleration commands in the horizontal and vertical planes of the inertial reference frame. Thus, a converting logic should be attempted in order to convert inertial frame acceleration commands into angle of attack and roll angle reference signals ready to be used as the reference signals to the autopilot block.

In addition to traditional proportional navigation guidance method, optimal and sliding mode control based guidance algorithms are suggested for performance comparison. Most important of all the work performed in the guidance design, a new guidance method based on the principles of one of the most basic controller design methodology, proportional-integral-derivative controller, is derived as a technical contribution to the literature. Several target scenarios (static, slow and fast maneuvering) are formed and run for three methods (proportional navigation, sliding mode and optimal proportional-integral guidance algorithms) by using the point mass model. According to the performance results, improvement in the miss distance values can be easily recognized for the new guidance methodology. Moreover, as can be seen from Table 10, traditional and mostly preferable guidance method “proportional navigation”, fails to hit the target for limited fast-accelerated scenario. On the other hand, missile guided with optimal proportional-integral guidance strategy is able to hit the target even for this difficult scenario. Optimal proportional-integral guidance developed is also tested with sliding mode autopilot designed just for testing system integration ability. Full nonlinear simulation environment is built aiming to hit a static and/or a maneuvering target. Results support the idea that guidance and autopilot unit integrate each other well

and tracking the reference signals commanded by guidance unit is fulfilled in nearly a perfect manner. Moreover, by using up small deflection energy, missile can be directed towards the target easily. Pitch plane acceleration is also kept small which is indeed a constraint for nonlinear simulations.

During overall system nonlinear simulation, a decrease in the speed of the missile until intercept with the target is observed due to the effect of the drag force. In order to compensate the decrease in speed, in other words, to control the speed of the missile during a part of the flight, a thrust model should be integrated to the system constituted. As a future work, a little effort can be spent in adapting a thrust-vector controlled structure to the missile used here. In addition, in this study effects of wind or other similar disturbances during flight are not taken into consideration. Future studies will surely include the effects of these parameters on guidance and control units.

In summary, most important aim of this thesis is to control a missile's motion during full or a part of its flight envelope. In order to realize this objective, modeling of missile nonlinear model, which is based on the nonlinear first order differential equations and the aerodynamic force and moment equalities, should be clarified as a starting point. Decoupled linearized version of the model is used for designing the controllers that control some flight angles of the missile. Different guidance algorithms are developed to produce reference signals that drive these controllers, in aiming to hit a static or a moving target. By carrying out this work, guidance, control and dynamics of a missile are recognized and formed to use in an integrated structure.

Interested reader on the control and dynamics of a missile should refer to the references [38]-[41]. More detailed analysis about linear quadratic regulator theory and its results can be found in the references [42] and [43]. Furthermore, sliding mode control details and its guidance applications can be obtained from the references [44]-[50].

REFERENCES

[1] Schumacher, C. J., “Tactical Missile Autopilots: Gain-Scheduled Control and Dynamic Inversion”, A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy (Aerospace Engineering) in the University of Michigan, 1997.

[2] Song, K. S., “A Robust Adaptive Autopilot Design for Decomposed Bank to Turn Missiles”, A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy (Electrical Engineering) in Wayne State University, 2001.

[3] McLean, D., “Automatic Flight Control Systems”, Prentice Hall Inc., 1990.

[4] Stevens, B. L. and Lewis, F.L., “Aircraft Control and Simulation”, John Wiley & Sons, Inc., 1992.

[5] Siouris, G. M., “Missile Guidance and Control Systems”, Springer-Verlag New York, Inc., 2004.

[6] Ateşoğlu, Ö., “Different Autopilot Designs and Their Performance Comparison for Guided Missiles”, M. S. Thesis in METU, Department of Aeronautical Engineering, December 1996.

[7] Akkal, E., “Control Actuation Systems and Seeker Units of an Air-to-Surface Guided Munition”, M. S. Thesis in METU, Department of Electrical and Electronics Engineering, December 2003.

[8] Tiryaki, K., “Polynomial Guidance Laws and Dynamic Flight Simulation Studies”, M. S. Thesis in METU, Department of Aeronautical Engineering, June 2002.

[9] Doruk, R. Ö., “Missile autopilot design by projective control theory”, M. S. Thesis in METU, Department of Electrical and Electronics Engineering, August 2003.

[10] Burns, K., Deters, K., Stoy, S., Vukelich, S., “MISSILE DATCOM User’s Manual”, Revision 6-93, June 1993.

[11] Wise, K. A., Broy, D. J., “Agile Missile Dynamics and Control”, AIAA paper, July 1996.

[12] Roskam, J., “Airplane flight dynamics and automatic flight controls”, Lawrence, KS: DAR Corporation, 1995.

[13] Burl, J. B., “Linear optimal control: and methods”, Menlo Park, Calif.: Addison Wesley Longman, 1999.

[14] Luo, J., Lan, C. E., “Determination of Weighting Matrices of a Linear Quadratic Regulator”, J. Guidance, Vol. 18, No. 6, 1995.

[15] Levine, W. S., “The Control Handbook”, IEEE Press, CRC Press, 1996.

[16] Ogata, K., “Modern Control Engineering”, Prentice Hall Inc., 1997.

[17] Hung, J. Y., Gao, W. B. and Hung, J. C., “Variable structure control : A survey”, IEEE Trans. Industrial Electronics, vol. 40, no. 1 , February 1993.

[18] Utkin, V. I., “Variable structure systems with sliding modes”, IEEE Trans. Automat. Contr., Vol. AC-22, no. 2, pp. 212-222, 1977.

[19] DeCarlo, R. A., Zak, S. H. and Matthews, G. P., “Variable structure control of nonlinear multivariable systems: A tutorial”, Proc. IEEE, vol. 76, no. 3, pp. 212-232, 1988.

[20] Wells, S. R., “Application of Sliding Mode Methods to the Design of Reconfigurable Flight Control Systems”, A Dissertation submitted to the Graduate Faculty of University of California in Partial Fulfillment of the Degree of Doctor of Philosophy, 2002.

[21] Edwards, C., Spurgeon, S. K., “Sliding Mode Control: Theory and Applications” Copyright Taylor & Francis Ltd, 1998.

[22] Gao, W. B. and Hung, J. C., “Variable structure control of nonlinear systems: A new approach” IEEE Trans. Industrial Electronics, vol. 40, no. 1, February 1993.

[23] Zarchan, P., “Tactical and Strategic Missile Guidance”, American Institute of Aeronautics and Astronautics Inc., USA, 1994.

[24] No, T. S., Cochran, J. E. and Kim, E. G., “Bank-to-Turn Guidance Law Using Lyapunov Function and Nonzero Effort Miss”, Journal of Guidance, Control and Dynamics, Vol. 24, No. 2, 2001.

[25] Nesline, F. W. and Zarchan, P., “Classical vs. Modern Homing Missile Guidance”, 1979.

[26] Lee, J. G., Han, H. S. and Kim, Y. J., “Guidance Performance Analysis of Bank-to-Turn Missiles”, IEEE, 1999.

[27] Kılıç, Ç. K., “Defining Infrared Missiles Guidance and Control Algorithms and Developing Effective Counter-Measures for IR Guided Missiles”, M. S. Thesis in METU, Department of Electrical and Electronics Engineering, December 2006.

[28] Zhou, D., Mu, C. and Xu, W., “Adaptive sliding-mode guidance of a homing missile”, Journal of Guidance, Control, and Dynamics, vol. 22, no. 4, pp. 589-594, July-August 1999.

[29] Babu, K. R., Sarma, I. G. and Swamy, K. N., “Switched bias proportional navigation for homing guidance against highly maneuvering targets”, Journal of Guidance, Control, and Dynamics, vol. 17, no. 6, pp. 1357-1363, November-December 1994.

[30] Innocenti, M., Pellegrini, F. and Nasuti, F., “A vss guidance law for agile missiles”, AIAA-97-3473, pp. 179-188, 1997.

[31] Zhou, D., Mu, C., Ling, Q. and Xu, W., “Optimal sliding-mode guidance of a homing missile”, IEEE, pp. 5131-5136, 1999.

[32] Perkgöz, C., “A Guidance and Control Method for a Tail Controlled Missile”, M. S. Thesis in METU, Department of Electrical and Electronics Engineering, March 2002.

[33] Yanushevsky, R. T. and Boord, W. J., “New approach to guidance law design”, Journal of Guidance, Control, and Dynamics, vol. 28, no. 1, pp. 162-166, January-February 2005.

[34] Evcimen, Ç. and Leblebicioğlu, K., “An Optimal Proportional-Integral Guidance for Missiles”, 4. Ankara International Aerospace Conference, 10-12 September 2007.

[35] Evcimen, Ç. and Leblebicioğlu, K., “Füzeler için Optimal Oransal-Tümlevsel Güdüm Yaklaşımı”, TOK’07 Sabancı Üniversitesi İstanbul, 5-7 Eylül 2007.

[36] Welch, G. and Bishop, G., “An Introduction to the Kalman Filter “, 2004.

[37] Brown, R. G. and Hwang, P., YC., ”Introduction to Random Signals and Applied Kalman Filtering “, New York: J. Wiley, 1992.

[38] Wise, K. A. “Approximating A Linear Quadratic Missile Autopilot Design Using An Output Feedback Projective Control”, McDonnell Douglas Missile Systems Company St. Louis, Missouri, 63166 Farhad Deylami Electronic Data Systems Ypsilanti, Michigan 48198 AIAA-91-2613-CP.

[39] Eguchi, H., Obana, K. and Kiso, M., “Standard Airframe Model for Missile Autopilot Design and Evaluation”, AIAA Journal 1999.

[40] Arrow, A., “An Analysis of Aerodynamic Requirements for Coordinated Bank-to-Turn Autopilots”, NASA Contractor Report 3644.

[41] Wise, K. A., “Bank-to-Turn Missile Autopilot Design Using Loop Transfer Recovery”, J. Guidance, Vol. 13, No. 1, Jan.-Feb 1990.

[42] Soroka, E. and Shaked, U., “On the Robustness of LQ Regulators”, IEEE, 1984.

[43] Choi, J. W., Seo,Y. B., “LQR Design with Eigenstructure Assignment Capability“, IEEE, 1999.

[44] Thukral, A. and Innocenti, M., “Control design challenge: A variable structure approach“, Journal of Guidance, Control and Dynamics, Vol. 17, No.5, September-October 1994.

[45] Thukral, A., “Design of missile flight control systems using variable structure techniques“, A Dissertation submitted to the Graduate Faculty of Auburn University in Partial Fulfillment of the Degree of Doctor of Philosopy, June 7 1995.

[46] Lee, D. S., Kim, M. G., Kim, H. K. and Youn, M. J., “Controller design of multivariable variable structure systems with nonlinear switching surfaces“ IEEE Proceedings-D, Vol. 138, No. 5, September 1991.

[47] Gao, W. B., “On the Design of Variable Structure Controller“, Proc. IEEE, 1993

[48] Ghazi, S. E., “Analysis of VSC System Designed via the Reaching Law Approach”, Proc. IEEE, 1996.

[49] Innocenti, M., “Nonlinear guidance techniques for agile missiles”, Control Engineering Practice 9, pp. 1131-1144, 2001.

[50] Shkolnikov, I, Shtessel, T. and Lianos, D., “Integrated guidance-control system of a homing interceptor: sliding mode approach”, AIAA-2001-4218, pp. 1-11, 2001.

APPENDIX

MISSILE DATCOM USER MANUAL

Input File

Digital Datcom is a public domain software program that provides a systematic summary of methods for estimating stability and control characteristics for preliminary design applications. The program allows the user to specify a basic airframe geometry and flight condition, and returns an estimated set of aerodynamic force and moment coefficients. The software uses for005.dat file as an input file and creates other forxxx.dat files. Input file for005.dat can be modified by any text editor program. for006.dat is the main output file which includes all the force and moment coefficients [10].

Tables and figures below state the physical quantities of the missile run within Datcom, which is used during the studies of the thesis

Table 11 Reference quantity inputs

Parameter	Variable Name	Value	Unit
Reference Area	SREF	0.0404708	meter
Reference Length (Longitudinal)	LREF	0.227	meter
Reference Length (Lateral)	LATREF	0.227	meter
Longitudinal Position of Center of Gravity (+Aft)	XCG	1.66	meter
Boundary Layer Type	BLAYER	NATURAL	-
Roughness Height Rating	RHR	280	-

Table 12 Axisymmetric body geometry inputs

Parameter	Variable Name	Value	Unit
Nose Bluntness Radius	BNOSE	0.015	meter
Nose Section Base Diameter	DNOSE	0.227	meter
Nose Length	LNOSE	0.86	meter
Center body Length	LCENTR	2.837	meter
Diameter of Nozzle Exit	DEXIT	0	meter

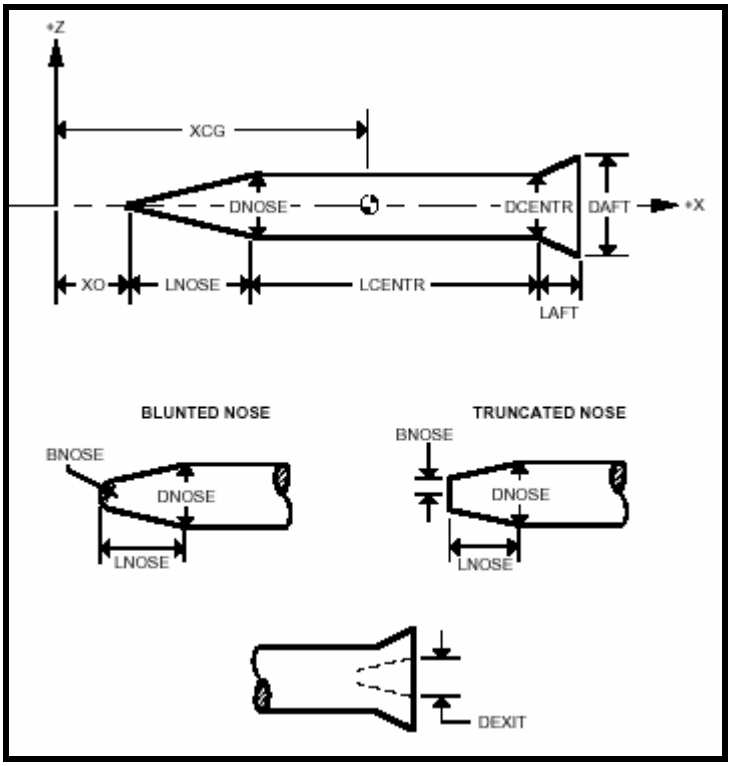

Figure 85 Body geometry inputs

Table 13 Fin geometry inputs

Parameter	Variable Name	Wing	Tail	Unit
Type of Section to be defined	SECTYP	HEX	HEX	-
# Panels in Set (1-8)	NPANEL	4	4	-
Roll Angle of Each Fin Measured Clockwise from Top Vertical Center Looking Forward	PHIF	0, 90, 180, 270	45, 135, 225, 315	degree
Distance from Missile Nose to Chord Leading Edge at Each Span Location	XLE	0.84325	3.377	meter
Sweepback Angle at Each Span Station	SWEEP	60	20	degree.
Chord Station used in Measuring Sweep : STA=0 Leading Edge STA=1 Trailing Edge	STA	0	0	-
Semi-span Locations	SSPAN	0, 0.1963	0, 0.159	meter
Panel Chord at Each Semi-span Location	CHORD	0.34, 0	0.25, 0.192	meter
Maximum Thickness to Chord Ratio of Upper Surface	ZUPPER	0.01, 0.01	0.01, 0.0130617	-
Fraction of Chord from Section Leading Edge to Maximum Thickness of Upper Surface	LMAXU	0.35, 0.35	0.05672, 0.07409	-
Fraction of Chord of Constant Thickness Section of Upper Surface	LFLATU	0.3, 0.3	0.88656, 0.851829	-

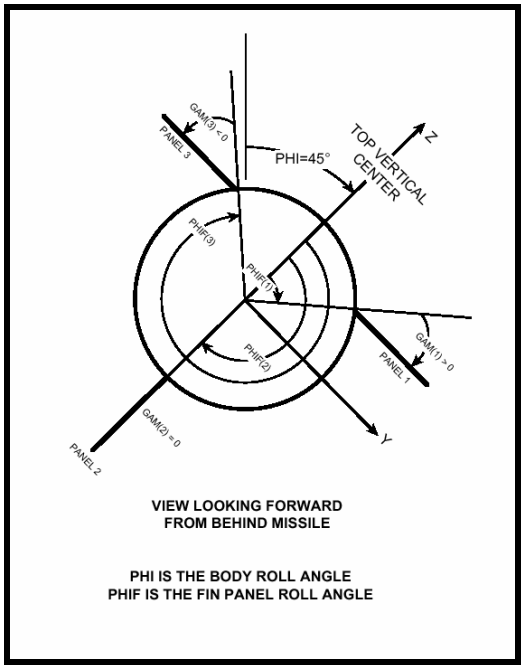

Figure 86 Roll attitude vs. fin orientation

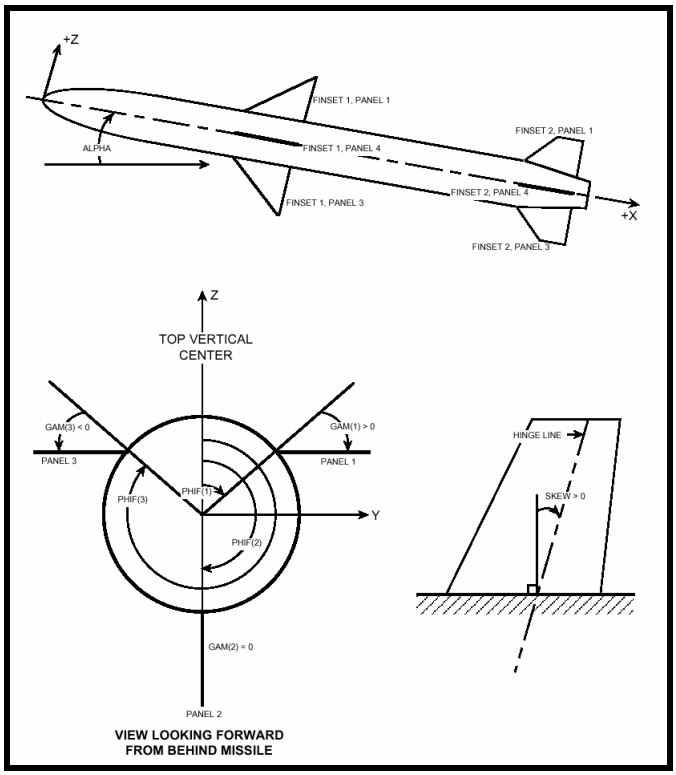

Figure 87 Fin numbering orientation

Table 14 Flight conditions

Parameter	Variable Name	Value	Unit
# Mach No.	NMACH	3	-
# Angles of Attack	NALPHA	5	-
Mach No.	MACH	0.5, 1.0, 1.5	-
Angle of Attack or Total Angle of Attack	ALPHA	0°, 2°, 4°, 6°, 8°	degree
Altitudes	ALT	0, 4000, 8000	meter
Sideslip Angle	BETA	0	degree

Output File

Missile DATCOM provides aerodynamic coefficients for the missile characterized by the input file according to attack angles and mach numbers. If the input includes only one angle of attack and only one Mach number, output file will contain only one “run” and of course only one output for all coefficients. Its output file gives the following set of aerodynamic force and moment coefficients in summary

CN	Normal force coefficient
CM	Pitching moment coefficient
CA	Axial force coefficient
CY	Side force coefficient
CLN	Yawing moment coefficient
CLL	Rolling moment coefficient
CNA	Normal force coefficient derivative with alpha
CMA	Pitching moment coefficient derivative with alpha
CYB	Side force coefficient derivative with beta
CLNB	Yawing moment coefficient derivative with beta
CLLB	Rolling moment coefficient derivative with beta
CL	Lift coefficient
CD	Drag coefficient

CL/CD	Lift to drag ratio
XCP	Center of pressure position from the moment reference center divided by reference length, positive values for c.p. forward of the moment reference point.
CNQ	Normal force coefficient due to pitch rate
CMQ	Pitching moment coefficient due to pitch rate
CAQ	Axial force coefficient due to pitch rate
CNAD	Normal force coefficient due to rate of change of angle of attack
CMAD	Pitching moment coefficient due to rate of change of angle of attack
CYR	Side force coefficient due to yaw rate
CLNR	Yawing moment coefficient due to yaw rate
CLLR	Rolling moment coefficient due to yaw rate
CYP	Side force coefficient due to roll rate
CLNP	Yawing moment coefficient due to roll rate
CLLP	Rolling moment coefficient due to roll rate

Some coefficients like “yaw moment aileron derivative coefficient” can not be found directly in the output file. But, this coefficient can be found by changing the “fin deflection angles”, running the Missile DATCOM again, substituting the two outputs for the “Yawing moment coefficient” [27].

Datcom Program

This chapter closes with the Datcom program which includes all of the physical information of the missile above and also the deflection angles given to the program to implement the control surface deflection aerodynamic coefficients.

```
--------- DATCOM INPUT PROGRAM ---------
CASEID MISSILE
DIM M
DERIV DEG
SOSE
$REFQ
 SREF=.0404708,
```

```
 LREF=.227,
 LATREF=.227,
 XCG=1.66,
 BLAYER=NATURAL,
 RHR=280.,
$END
$AXIBOD
 BNOSE=.015,
 DNOSE=.227,
 LNOSE=.86,
 LCENTR=2.837,
 DEXIT=0.,
$END
$FINSET1
 SECTYP=HEX,
 NPANEL=4.,
 PHIF=0.,90.,180.,270.,
 XLE=0.84325,
 SWEEP=60.,
 STA=0.,
 SSPAN=0.,.1963,
 CHORD= .34,0.,
 ZUPPER=.01,.01,
 LMAXU=.35,.35,
 LFLATU=.3,.3,
$END
$FINSET2
 SECTYP=HEX,
 NPANEL=4.,
 PHIF=45.,135.,225.,315.,
 XLE=3.377,
 SWEEP=20.,
 STA=0.,
 SSPAN=0.,.159,
```

```
 CHORD=.25,.192,
 ZUPPER=.01,.0130617,
 LMAXU=.05672,.07409,
 LFLATU=.88656,.851829,
$END
$FLTCON
 NMACH=3.,NALPHA=5.,
 MACH(1)=.5,1.,1.5,
 ALPHA(1)=0.,2.,4.,6.,8.,
 ALT(1)=0., 0., 0.,
 BETA=0.,
$END
$DEFLCT
 DELTA1=0.,0.,0.,0.,
$END
SPIN
DAMP
SAVE
NEXT CASE
$DEFLCT
 DELTA1=0.,-1.,0.,1.,
$END
SAVE
NEXT CASE
$DEFLCT
 DELTA1=1.,1.,1.,1.,
$END
SAVE
NEXT CASE
$DEFLCT
 DELTA1=-1.,0.,1.,0.,
$END
SAVE
NEXT CASE
```